Modicon Quantum with Unity

Ethernet Network Modules
User Manual

10/2014

33002479.13

www.schneider-electric.com

All pertinent state, regional, and local safety regulations must be observed when installing and using this product. For reasons of safety and to help ensure compliance with documented system data, only the manufacturer should perform repairs to components.

When devices are used for applications with technical safety requirements, the relevant instructions must be followed.

Failure to use Schneider Electric software or approved software with our hardware products may result in injury, harm, or improper operating results.

Failure to observe this information can result in injury or equipment damage.

Table of Contents

Safety Information

Important Information

NOTICE

Read these instructions carefully, and look at the equipment to become familiar with the device before trying to install, operate, 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.

About the Book

At a Glance

Document Scope

This document is for users who want their Quantum PLC to communicate with devices over an Ethernet network. It assumes that you have some knowledge of PLC systems and a working knowledge of Modsoft, Concept, ProWORX, or Unity Pro programming tools. You also need to understand the use of Ethernet networks and TCP/IP.

This document describes the functionality of the Modicon Quantum Ethernet modules, the 104 NOE 771 xx and 140 NWM 100 00 modules, and the Ethernet port of the 140 CPU 651 x0 modules.

Model Number	Module Type
140 NOE 771 00	Ethernet option adapter
140 NOE 771 01	Ethernet option adapter
140 NOE 771 10	FactoryCast Ethernet option adapter
140 NOE 771 11	FactoryCast Ethernet option adapter
140 NWM 100 00	FactoryCast HMI option adapter
140 CPU 651 50	CPU with embedded Ethernet port
140 CPU 651 60	CPU with embedded Ethernet port

This document describes how to install, set up, and use Ethernet-enabled modules in a Modicon Quantum control system. These modules allow the Quantum system to communicate with devices over an Ethernet network. Topics addressed in this manual include:

- the hardware architecture of a Modicon Quantum Ethernet module, which is designed to fit into a single slot on the standard Modicon Quantum backplane
- capabilities and installation of 140 NOE 771 xx and 140 NWM 100 00 modules in a Modicon Quantum system
- capabilities of the Ethernet port on the 140 CPU 651 x0 CPU modules

This document describes the procedures for:

- setting up the modules to transfer data using either the Global Data modules (140 NOE 771 01, 140 NOE 771 11, or 140 CPU 651 x0 mobules) or the I/O scanner utility (in the 140 NOE 771 00/01/11 modules and the 140 CPU 651 50/60 modules)
- using an embedded Web server to access diagnostics and online configurations for the module and its associated controller
- using the FactoryCast Web server to customize your configuration via embedded Web pages (in 140 NOE 771 10/11 and 140 NWM 100 00 modules)
- using an NOE module in a Hot Standby solution for fault-tolerant remote I/O and communications

Nomenclature

The following table describes the naming rules for various groups of modules:

The Name ...	Refers to the ...
140 NOE 771 xx	140 NOE 771 00, 140 NOE 771 01, 140 NOE 771 10 and 140 NOE 771 11 modules
140 NOE 771 x0	140 NOE 771 00 and 140 NOE 771 10 modules
140 NOE 771 x1	140 NOE 771 01 and 140 NOE 771 11 modules
140 NOE 771 0x	140 NOE 771 00 and 140 NOE 771 01 modules
140 NOE 771 1x	140 NOE 771 10 and 140 NOE 771 11 modules
140 CPU 651 x0	140 CPU 651 50 and 140 CPU 651 60 modules

NOTE: Before adding a node to an existing corporate network, consult with your information technology (IT) department about any possible consequences.

Validity Note

This document is valid for Unity Pro 8.1 or later.

The technical characteristics of the devices described in this document also appear online. To access this information online:

Step	Action
1	Go to the Schneider Electric home page *www.schneider-electric.com*.
2	In the **Search** box type the reference of a product or the name of a product range. ● Do not include blank spaces in the model number/product range. ● To get information on grouping similar modules, use asterisks (*).
3	If you entered a reference, go to the **Product Datasheets** search results and click on the reference that interests you. If you entered the name of a product range, go to the **Product Ranges** search results and click on the product range that interests you.
4	If more than one reference appears in the **Products** search results, click on the reference that interests you.
5	Depending on the size of your screen, you may need to scroll down to see the data sheet.
6	To save or print a data sheet as a .pdf file, click **Download XXX product datasheet**.

The characteristics that are presented in this manual should be the same as those characteristics that appear online. In line with our policy of constant improvement, we may revise content over time to improve clarity and accuracy. If you see a difference between the manual and online information, use the online information as your reference.

Related Documents

Title of Documentation	Reference Number
Control Panel Technical Guide, How to Protect a Machine from Malfunctions Due to Electromagnetic Disturbance	CPTG003_EN (English), CPTG003_FR (French)
Grounding and Electromagnetic Compatibility of PLC Systems, Basic Principles and Measures, User Manual	33002439 (English), 33002440 (French), 33002441 (German), 33003702 (Italian), 33002442 (Spanish), 33003703 (Chinese)
Quantum with Unity Pro, TCP/IP Configuration, User Manual	33002467 (English), 33002468 (French), 33002469 (German), 31008078 (Italian), 33002470 (Spanish), 31007110 (Chinese)
Quantum with Unity Pro Hardware Reference Manual	35010529 (English), 35010530 (French), 35010531 (German), 35010532 (Spanish), 35013975 (Italian), 35012184 (Chinese)
Quantum with Unity Pro, Discrete and Analog I/O, Reference Manual	35010516 (English), 35010517 (French), 35010518 (German), 35013970 (Italian), 35010519 (Spanish), 35012185 (Chinese)
Quantum with Unity Pro, Experts and Communication, Reference Manual	35010574 (English), 35010575 (French), 35010576 (German), 35014012 (Italian), 35010577 (Spanish), 35012187 (Chinese)
Modicon M340, Premium, Atrium, and Quantum Using Unity Pro, Communication Services and Architectures, Reference Manual	35010500 (English), 35010501 (French), 35006176 (German), 35013966 (Italian), 35006177 (Spanish), 35012196 (Chinese)
Schneider Automation BOOTP Lite Ethernet IP Address Download Utility for Schneider Automation Ethernet Products Instruction Sheet	31002087

Title of Documentation	Reference Number
FactoryCast for Quantum, Premium and Micro User Guide	31001229
Modicon Quantum Hot Standby System User Manual	35010533 (English), 35010534 (French), 35010535 (German), 35013993 (Italian), 35010536 (Spanish), 35012188 (Chinese)
Modbus Protocol Reference Guide	PI-MBUS-300
Open Modbus Specification	http://www.modbus.org/
FactoryCast HMI 1.7, Setup Manual, Premium and Quantum HMI Modules	35007632 (English), 35007634 (French), 35007633 (German), 35007635 (Spanish), 35007636 (Italian)

You can download these technical publications and other technical information from our website at www.schneider-electric.com.

Part I
Modicon Quantum with Unity Ethernet Products

Purpose

This part introduces the Modicon Quantum with Unity products used for Ethernet communication.

What Is in This Part?

This part contains the following chapters:

Chapter 1
Product Description

Overview

This chapter includes product overviews for:

- the 140 NOE 771 xx modules
- the 140 NWM 100 00 module
- the built-in Ethernet port of the 140 CPU 651 x0 modules

When inserted in the backplane, these modules allow your Quantum PLC to communicate over Ethernet networks.

What Is in This Chapter?

This chapter contains the following sections:

Section 1.1
Module Description 140 CPU 651 x0

Overview

This is a discussion of the the Modicon Quantum High-End CPU (HE CPU) (140 CPU 651 *x*0).

What Is in This Section?

This section contains the following topics:

140 CPU 651 x0 Product Overview

General Description

The Quantum-140 CPU 651 x0 module *(see page 20)* is among the newest in a series of Quantum processor modules. It combines standard PLC functions with the diagnostic possibilities of a Web server. The 140 CPU 651 *x*0 communicates using an RJ-45 connection.

The 140 CPU 651 *x*0 module is in a double-width standard Quantum housing, which requires 2 sequential slots in a Quantum rack. Its server functions are diagnostic only, so use software to configure services.

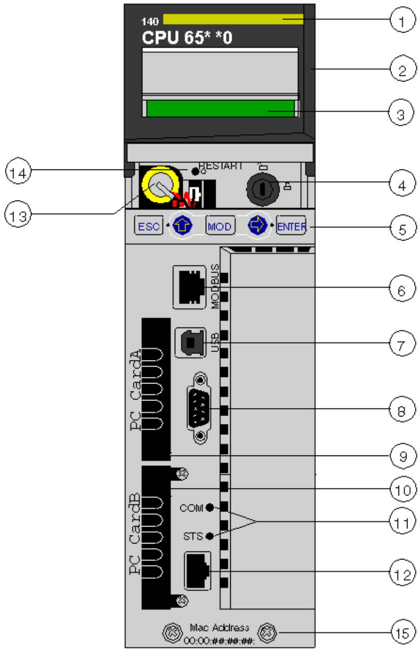
The table shows the key Ethernet services that are implemented:

Service	**140 CPU 651 x0**
HTTP Server *(see page 86)*	x
FTP Server *(see page 85)*	x
Flash File System *(see page 88)*	x
BOOTP Client *(see page 67)*	x
Address Server *(see page 74)*	x
SNMP V2 Agent (Network Management Service) *(see page 76)*	x
Modbus Messaging *(see page 77)*	x
I/O Scanner *(see page 79)*	x
Hot Standby	
Global Data (Publish/Subscribe) *(see page 81)*	x
Bandwidth Monitoring *(see page 84)*	x
Faulty Device Replacement (Server) *(see page 229)*	x
Enhanced Web Diagnosis *(see page 86)*	x
Schneider Private MIB *(see page 76)*	x
FactoryCast Application *(see page 88)*	
User-programmed Web pages	
RJ-45 Connection, see Modbus port in the **Quantum Platform manual**.	x
Fiber Optic Connection	
Time Synchronization Service *(see page 82)*	
Electronic Mail Notification Service *(see page 83)*	X

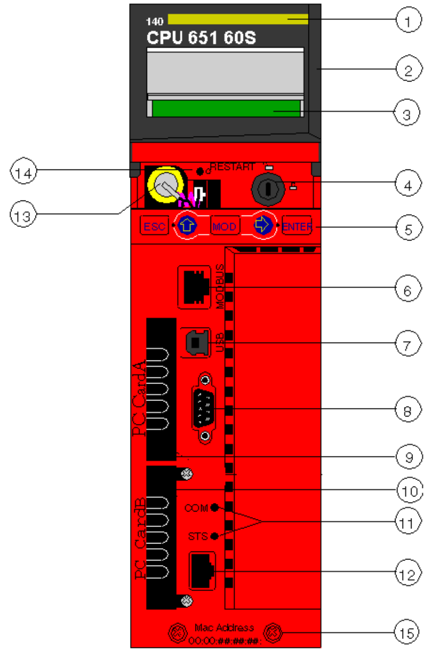
Physical Presentation and Mounting of Standard High End Modules

Illustration

The figure shows a standard High End module and its components.



standard processor



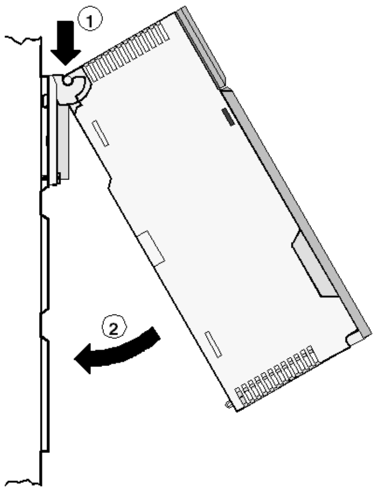
safety processor (red)

1 model number, module description, color code
2 lens cover (open)
3 LCD display (here covered by the lens cover)
4 key switch
5 keypad (with 2 red LED indicators)
6 modbus port (RS-232) (RS-485)
7 USB port
8 Modbus Plus port
9 PCMCIA slot A (depends on CPU reference)
10 PCMCIA slot B
11 LED indicators (yellow) for Ethernet communication
12 Ethernet port
13 battery (user installed)
14 reset button
15 2 screws

NOTE: Quantum High End processors are equipped with one PCMCIA slot (B) or two PCMCIA slots (A and B) in which to install Schneider PCMCIA cards (other cards are not accepted).

Mounting

Mounting the module onto the central back plane:



1 Hang the module.
2 Screw the module to the back plane.

CPU Controls and Displays

Lens Cover

The protective lens cover (2 on the CPU front panel) can be opened by sliding it upwards.

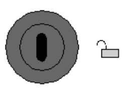
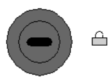
With the lens cover open you have access to the following items:
- key switch
- battery
- reset button

Key Switch

The key switch (4) is a security feature and a memory protection switch. The key switch has two positions: locked and unlocked. The key switch is only read and deciphered by the PLC OS portion of the firmware and not by the OS loader portion.

The Quantum processors have a set of system menus that enable the operator to:
- perform PLC operations (i.e., start PLC, stop PLC)
- display module parameters (i.e., communications parameters)
- switch to the maintenance mode (in Safety processors)

The key positions are shown in the table below:

Key Position	PLC Operation
unlocked: 	• System menu operations can be invoked and module parameters can be modified by the operator with the LCD and keypad. • Memory protection is OFF. • You can switch to Maintenance mode (in Safety processors).
locked: 	• No system menu operations can be invoked and module parameters are read-only. • Memory protection is ON. • Safe mode forced (in Safety processors).

Switching the key switch position from locked to unlocked or vice versa turns on the LCD's backlight.

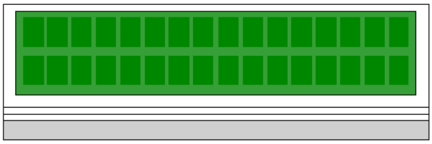
NOTE: For more explanations about Maintenance and Safe modes, refer to the *Quantum Safety PLC Safety Reference Manual*.

Reset Button

Pressing the reset button (12) forces a cold start of the PLC.

LCD Display

The liquid crystal display (LCD - 3) has 2 lines with 16 characters each with changeable backlight state and contrast:



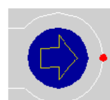
The backlight handling is entirely automated to save the life of the LCDs. The backlight turns on when one of the following occurs:

- a key is pressed
- the key switch state is changed
- an error message is displayed on the LCD

The backlight stays on for error messages as long as the error message is displayed otherwise, the backlight automatically turns off after five minutes.

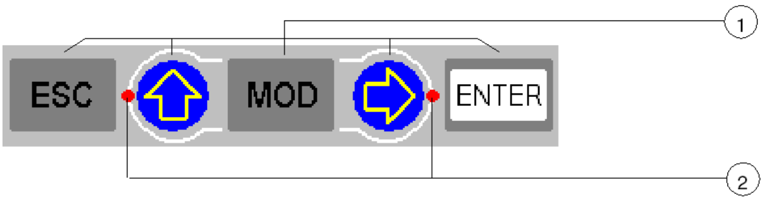
Adjusting the Contrast

The contrast is adjustable from the keypad when the default screen is displayed:

Step	Action	
1	Press the `MOD` key:	
2	To adjust the contrast darker press:	
3	To adjust the contrast lighter press:	
4	To confirm the setting press:	

Keypad

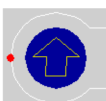
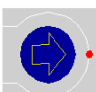
The keypad (5) has five keys that are mapped to hardware addresses. Each of the two arrow keys includes an LED:



1 5 keys
2 2 LEDs

Using the Keys

Keypad functions:

Key	Function	
	To cancel an entry, suspend or stop an action in progress To display the preceding screens successively (move up the menu tree)	
	To confirm a selection or an entry	
	To set a field on the display into the modify mode	
	LED: on	Key active: ● to scroll through menu options ● to scroll through modify mode field options
	LED: flashing	Key active: The modify mode field can be scrolled.
	LED: off	Key inactive: No menu options and no field options.
	LED on	Key active: ● to move around in a screen, field to field ● to go to the sub-menu
	LED flashing	Key active: Used to move digit to digit in a modify mode field.
	LED off	Key inactive, there is no: ● sub-menu for menu option ● scrolling around a screen ● scrolling around a field

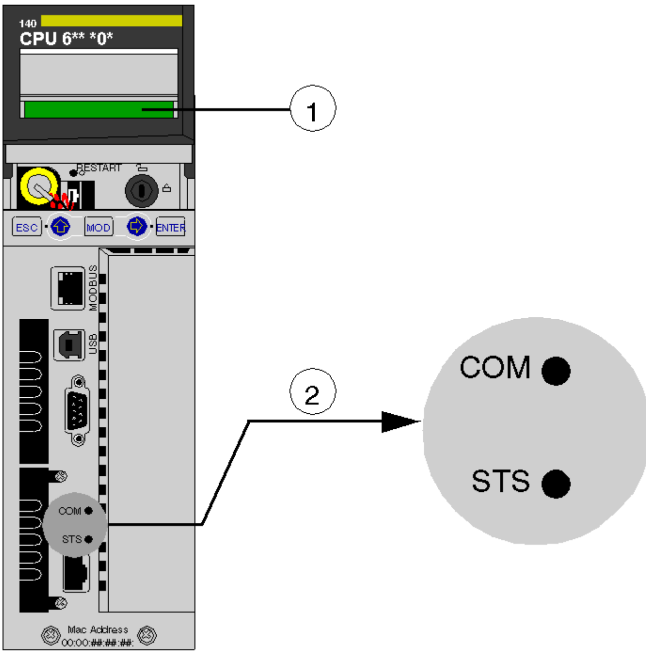
Indicators

Overview

The High End processors use two types of indicators:

1. LCD display: The Default Screen *(see page 28)* serves as a Controller status screen.
2. LED Indicators: The functionality of the LEDs is described in a table after the figure.

The following figure shows the two types of indicators.



High End processors

1 LCD Display (lens cover closed)
2 LED Indicators

LED Description

The following table shows the description for the LED indicators of the different High End CPU modules.

<table>
<tr><th rowspan="2">LEDs</th><th colspan="3">Indication</th></tr>
<tr><th colspan="2">Standard Processors
140 CPU 65• •0 / 140 CPU 651 60S</th><th>Hot StandBy Processors
140 CPU 67• 6• / 140 CPU 671 60S</th></tr>
<tr><td>COM (yellow)</td><td colspan="2">Controlled by the Coprocessor hardware
● Indicates Ethernet activity</td><td>Controlled by the Coprocessor hardware
● Indicates Primary or Standby activity</td></tr>
<tr><td rowspan="11">STS (yellow)</td><td colspan="2">Controlled by the Coprocessor software</td><td rowspan="11">Controlled by the Coprocessor firmware
● Blinking: system is redundant and data are exchanged from the Primary to Standby controller
● ON: system not redundant / Copro booting from power-on to end of self-tests
● OFF: Copro auto tests were not successfull</td></tr>
<tr><td>ON</td><td>Normal</td></tr>
<tr><td>OFF</td><td>Copro auto tests unsuccessful. Possible hardware problem.</td></tr>
<tr><td colspan="2">Flashing:</td></tr>
<tr><td>1 Flash</td><td>Configuration in progress.
Temporary situation.</td></tr>
<tr><td>2 Flashes</td><td>Invalid MAC address</td></tr>
<tr><td>3 Flashes</td><td>Link not connected</td></tr>
<tr><td>4 Flashes</td><td>Duplicate IP Address. Module is set to its default IP address.</td></tr>
<tr><td>5 Flashes</td><td>Waiting for IP address from address server</td></tr>
<tr><td>6 Flashes</td><td>Invalid IP address. Module is set to its default IP address.</td></tr>
<tr><td>7 Flashes</td><td>Firmware incompatibility between PLC OS and Copro firmware</td></tr>
</table>

Using the CPU LCD Display Screens

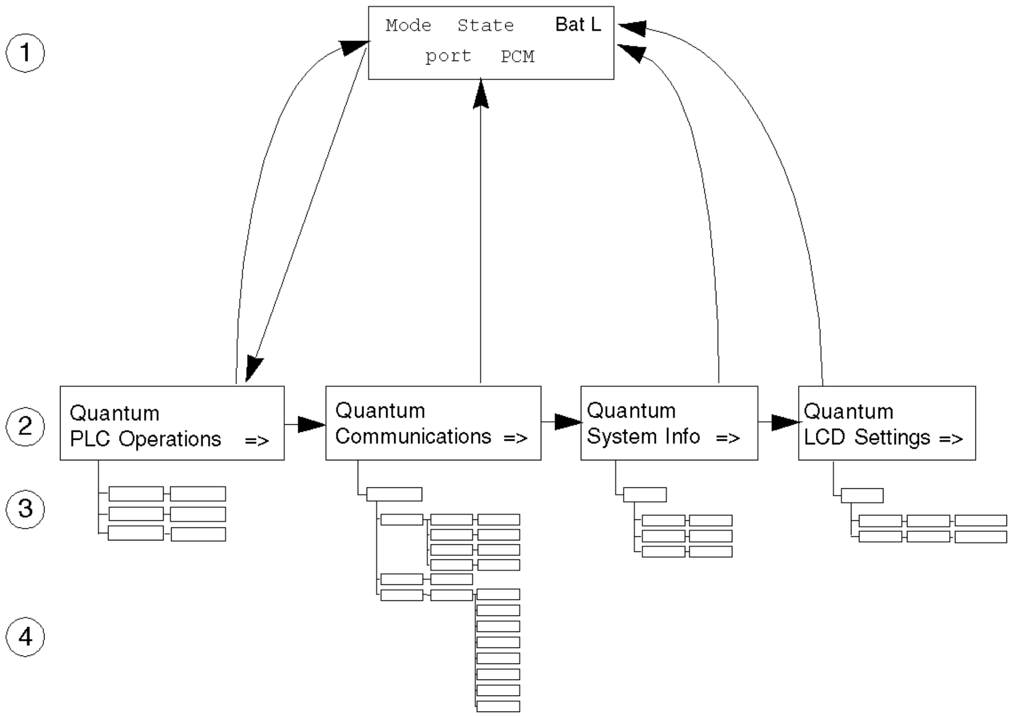
Overview

The controller LCD displays messages. These messages indicate the controller status. There are four levels of menus and submenus. Menus are accessed using the keypad *(see page 24)* on the front of the controller.

For detailed information about the menus and submenus see:

- PLC Operations Menus and Submenus *(see page 30)*
- Using the Communications Menus and Submenus *(see page 33)*
- Using the LCD Settings Menus and Submenus *(see page 35)*
- Using the System Info Menus and Submenus *(see page 36)*

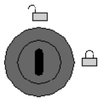
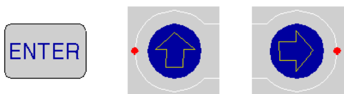
Structure: LCD display menus and submenus:



1 Default screen
2 System menus
3 Sub menus
4 Sub screens

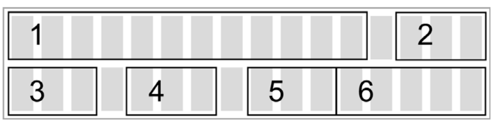
Accessing the Screens

Use the keys on the keypad to access the system menus and submenus:

Step	Action
1	To access the screens, ensure that the key switch is in the unlocked position. 
2	To step down to a lower menu, operate one of the following keys: 
3	To return to the previous menu, press: 

140 CPU 65• •• Default Screen

The default screen is read-only and contains the following fields:



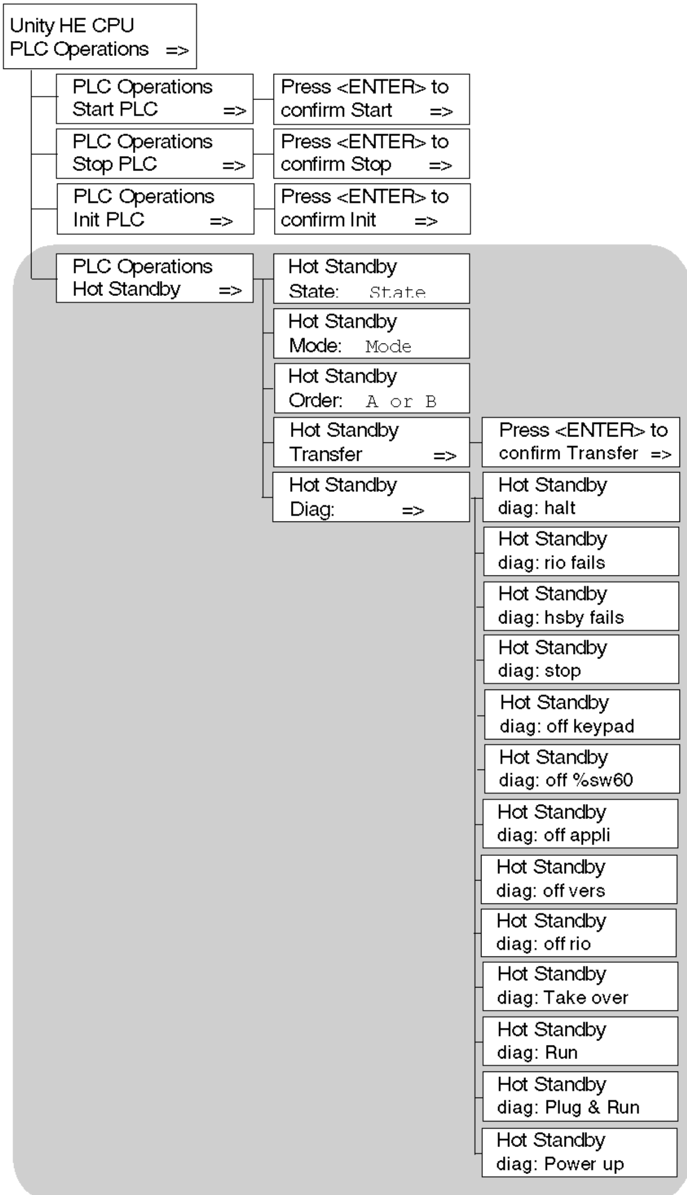
The default screen displays the following information:

Field		Display	Description
0	CPU Mode	M	Maintenance Mode (only on safety processors)
		S	Safe Mode (only on safety processors)
2	CPU State	RUN	Application program is running.
		STOP	Application program is NOT running
			STOP Offline
		No Conf	Processor has no application program
		Halt	Detected state error (in maintenance mode for safety modules)

<table>
<tr><th colspan="2">Field</th><th>Display</th><th>Description</th></tr>
<tr><td colspan="2">BatL</td><td></td><td>Indicates battery health:
● Steady = battery is low
● No message = battery is OK</td></tr>
<tr><td>Port</td><td>USB</td><td></td><td>Indicates that the port has activity</td></tr>
<tr><td rowspan="10"></td><td rowspan="5">Modbus Plus</td><td>MB+</td><td>Indicates Modbus Plus activity</td></tr>
<tr><td>mb+</td><td>No activity</td></tr>
<tr><td>Dup</td><td>Duplicate MB+ address</td></tr>
<tr><td>ERR</td><td>Detected Modbus communications error</td></tr>
<tr><td>INI</td><td>Initial Network Search</td></tr>
<tr><td rowspan="2">Modbus</td><td>232</td><td>Serial port activity for RS-232</td></tr>
<tr><td>485</td><td>Serial port activity for RS-485</td></tr>
<tr><td rowspan="3">PCM</td><td>1</td><td>Displayed status indicates battery health of the PCMCIA card in slot 1:
● Steady = battery is OK
● Flashing = battery is low (only for green PCMCIAs (version <04)) (1)</td></tr>
<tr><td>2</td><td>Displayed status indicates battery health of the PCMCIA card in slot 2:
● Steady = battery is OK
● Flashing = battery is low (only for green PCMCIAs (version < 04)) (1)</td></tr>
<tr><td colspan="2">(1) With blue PCMCIAs (version >= 04), when main battery is low there is no flash.</td></tr>
</table>

PLC Operations Menu

The structure of the PLC Operations menu and submenus are:



Submenu for PLC Operations: Start, Stop and Init:

Start, Stop, Init Screens Display	**Fields Available**	**Description**
Start PLC	Press <ENTER> to confirm Start	Pressing <ENTER> starts the controller
Stop PLC	Press <ENTER> to confirm Stop	Pressing <ENTER> stops the controller
Init PLC	Press <ENTER> to confirm Init	Pressing <ENTER> initializes the controller On safety processors, this command is only available in the maintenance mode.

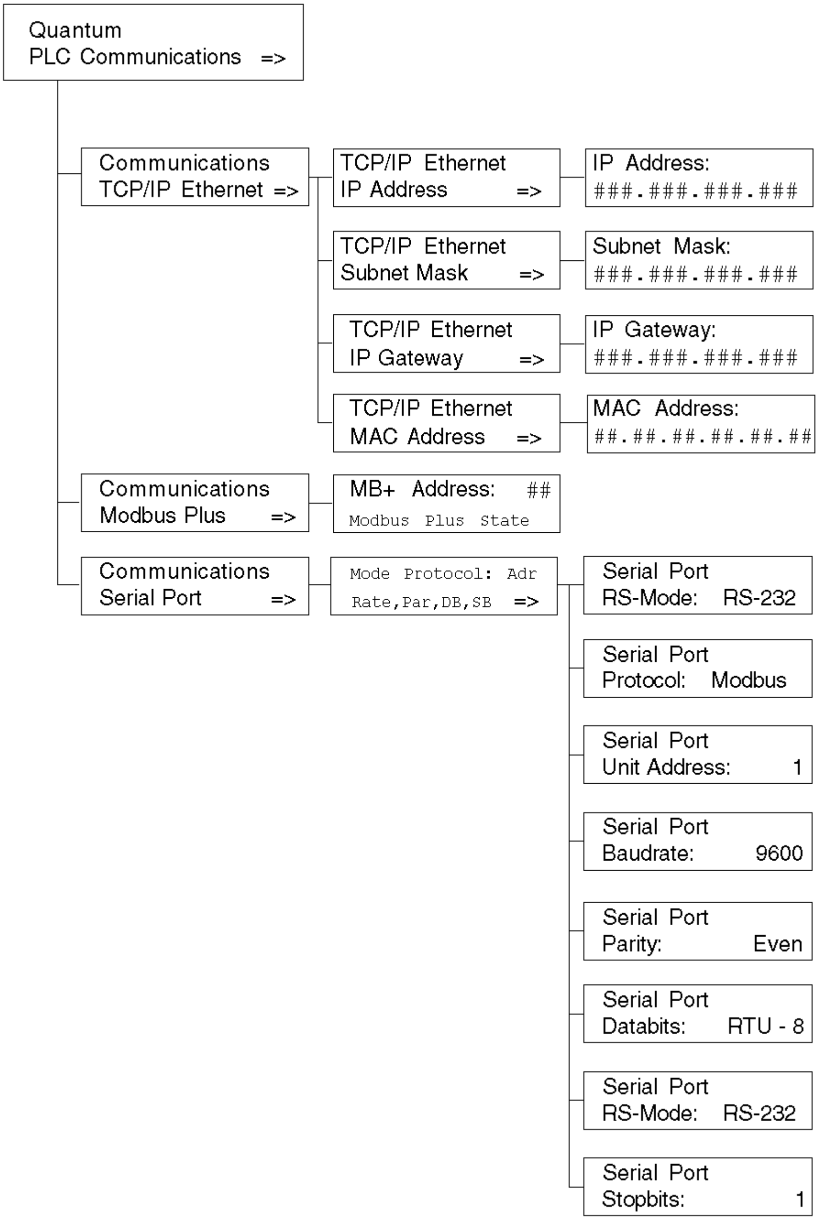
Submenu for PLC Operations Hot Standby CPU:

<table>
<tr><th>Screen</th><th>Field</th><th colspan="2">Option</th><th>Description</th></tr>
<tr><td rowspan="3">Hot Standby State:</td><td rowspan="3"><code>State</code>
read only</td><td colspan="2">PRIMARY CPU</td><td>Controller serves as Primary unit</td></tr>
<tr><td colspan="2">STANDBY CPU</td><td>Controller serves as Standby unit</td></tr>
<tr><td colspan="2">Offline</td><td>Controller is neither Primary nor Standby unit</td></tr>
<tr><td rowspan="4">Hot Standby Mode:</td><td rowspan="4"><code>Mode</code>
(modifiable only if the key switch is in the unlocked position)</td><td rowspan="2">RUN</td><td>STS steady</td><td>Controller is active and is either serving as Primary PLC or able to take over the Primary CPU role if needed</td></tr>
<tr><td>STS flashing</td><td>Controller is transferring/updating. When the transfer is done, RUN stays on steady</td></tr>
<tr><td rowspan="2">OFFLINE</td><td>STS steady</td><td>Controller is taken out of service without stopping it or disconnecting it from power.
If the controller is the Primary PLC when the mode is changed to Offline, control switches to the Standby PLC.
If the Standby PLC changes to Offline, the Primary CPU continues to operate without a backup.</td></tr>
<tr><td>STS flashing</td><td>Controller is transferring/updating.When the transfer is done, OFFLINE stays on steady.</td></tr>
<tr><td rowspan="2">Hot Standby Order:</td><td rowspan="2"><code>A or B</code>
(Modifiable only if the key switch is in the unlocked position.)</td><td colspan="2">FIRST</td><td rowspan="2">Hot Standby Power Order
NOTE: To change the A/B order the PLC must be in the STOP mode.</td></tr>
<tr><td colspan="2">SECOND</td></tr>
<tr><td>Hot Standby Transfer:</td><td colspan="3">-
(This menu option is only enabled, if the key switch is in the unlocked position)</td><td>Pressing the <ENTER> key confirms the Transfer. The transfer initiates the request of a program update from the Primary PLC. Pressing any other key cancels the Transfer initiation and returns to the Hot Standby Transfer menu option screen.</td></tr>
</table>

Screen	Field	Option	Description
Hot Standby Diag:	The order of diagnostic screen varies with the operation.		
	Halt		User task in halt mode
	RIO fails		Detected error reported by RIO head
	HSBY fails		Detected error reported by optical link
	Stop		Stop command sent
	Off keypad		Offline command entered on keypad
	Off %SW60		Offline command set in command register
	Off appli		Offline due to application mismatch
	Off vers		Offline due to PLC or Copro OS mismatch
	Off RIO		Offline due to Remote I/O error
	Take over		Standby CPU switched to Primary CPU mode
	Run		Run command sent
	Plug & Run		Sun-link operational and Standby CPU is started
	Power up		No message: PLC has just started

Communications Menu

Communications menu and submenus:



Submenu for TCP/IP Ethernet PLC Communications submenus:

TCP/IP Ethernet Screen Displays	Fields Available	Options Available	Description
TCP/IP Ethernet IP Address[1,2]	###.###.###.###	decimal numbers	Displays IP address
TCP/IP Ethernet Subnet Mask[1,2]	###.###.###.###	decimal numbers	Displays Subnetwork Mask address
TCP/IP Ethernet IP Gateway[1,2]	###.###.###.###	decimal numbers	Displays Ethernet IP Gateway address
TCP/IP Ethernet MAC Address	##.##.##.##.##.## (read only)	hexadecimal numbers	Displays MAC (Medium Access Control) address

1)Parameters can be modified only if no applications have been downloaded (in NO CONF state).

2)When a new PLC application has been downloaded, the Ethernet address on the screen is only updated after accessing the highest level of the menu structure.

Modbus Plus PLC Communications submenus:

Fields Available	Options Available	Description
## (Modifiable only if the key switch is in the unlocked position.)	1-64	Enter a valid Modbus Plus address
Modbus Plus State	Monitor Link	Modbus Plus State
	Normal Link	
	Sole Station	
	Duplicate address	
	No Token	

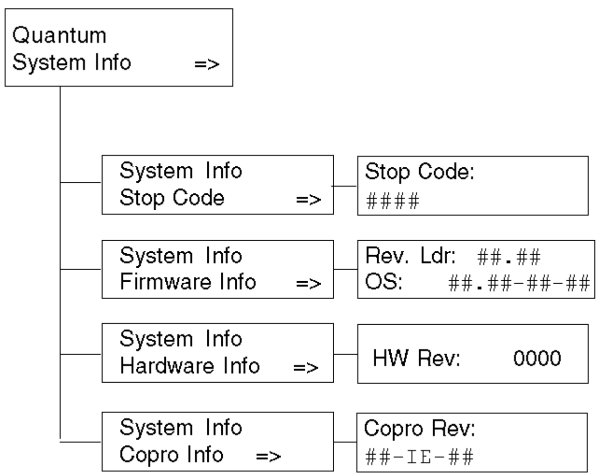
Serial PLC Communications submenus:

Fields Available*	Options Available	Description
Mode	232	RS mode
	485	
Protocol	ASCII	Protocols available
	RTU	
Adr	1 - 247	Unit address
	for Modbus Switchover Primary CPU 1-119 Standby CPU 129 - 247	

Fields Available*	Options Available	Description
Rate	50, 75, 110, 134.5, 150, 300, 600, 1200, 1800, 2400, 3600. 4800, 7200, 9600, 19200 bits/s	Baud rate
Par	NONE	Parity
	ODD	
	EVEN	
DB	7,8	Data bits, if Protocol is Modbus, then RTU-8 or ASCII-7.
SB	1,2	Stop bits
*If the key switch is in the unlocked position, fields are modifiable.		

System Info Menu

Structure of System Info menus and submenus:

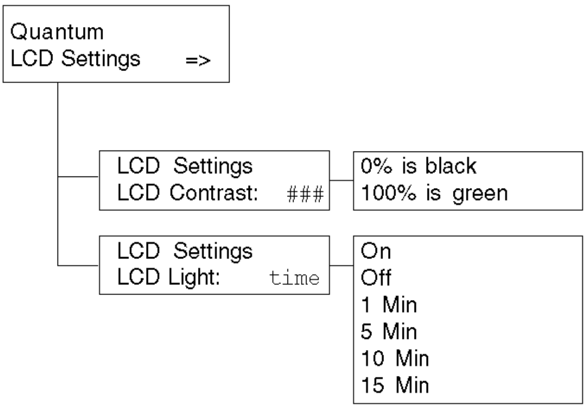


System Info, PLC Communications submenus:

System Info Screen Displays	Fields Available*	Option Available	Description
Stop Code	####		Machine stop code
	Description		Description of the machine stop code
Firmware Info	Rev.Ldr: ##.##		OS revision
	OS: ##.##-##-##		OSLoader revision
Hardware Info	HW Rev: ####		Hardware revision
Copro Info	##-IE-##		Copro revision
*Fields are read only.			

LCD Settings Menu

LCD Settings menus and submenus:



LCD Contrast settings submenu:

LCD Screen Contrast Screen Displays	Fields Available	Description
LCD Contrast:	####	Use the arrow keys to adjust the setting: ● Up arrow increases percent (brighter) ● Right arrow decreases percent (darker)

LCD Light setting submenus:

Screen Displays	Fields Available	Description
LCD Light:	On	LCD remains on permanently or until changed.
	Off	LCD remains off permanently or until changed.
	1 Min	LCD remains on for 1 minute.
	5 Min	LCD remains on for 5 minutes.
	10 Min	LCD remains on for 10 minutes.
	15 Min	LCD remains on for 15 minutes.

Section 1.2
Module Description 140 NOE 771 xx and 140 NWM 100 00

Overview

This section contains the module descriptions for 140 NOE 771 xx and 140 NWM 100 00.

What Is in This Section?

This section contains the following topics:

Modicon Quantum Ethernet Modules Overview

Overview

The following information provides overviews of all Modicon Quantum Ethernet modules.

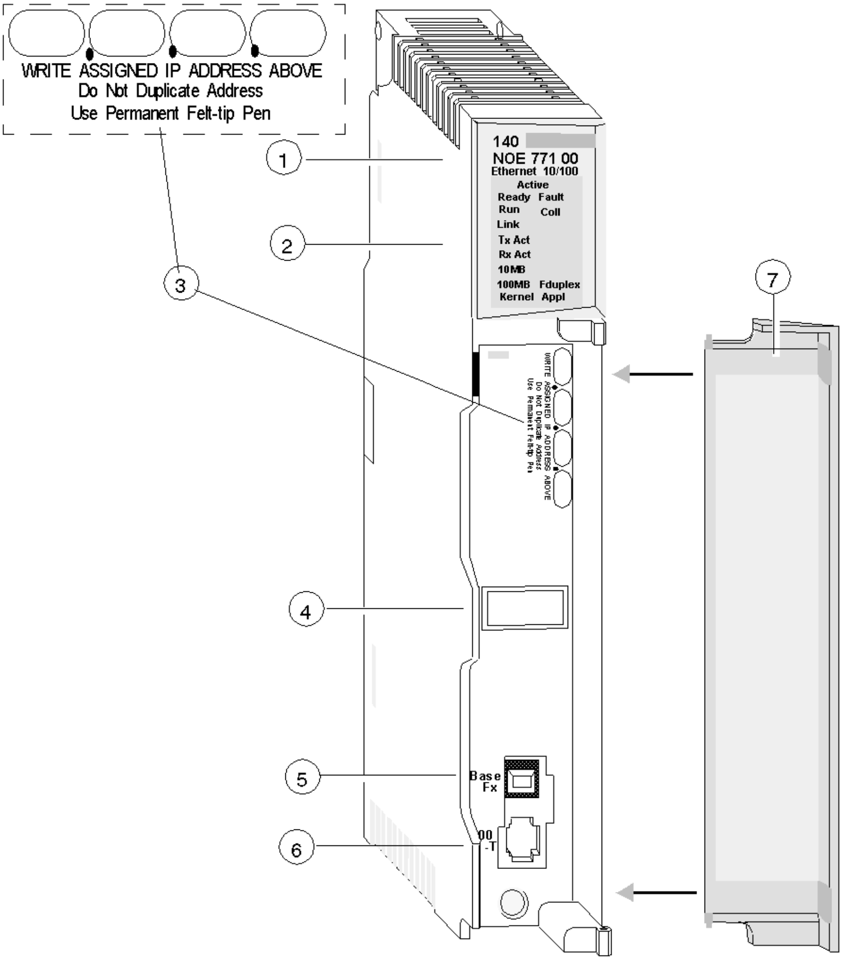
General Description

The Modicon Quantum Ethernet module, shown below, is one of the latest models in a line of Modicon Quantum Ethernet TCP/IP modules designed to make it possible for a Modicon Quantum PLC to communicate with devices over an Ethernet network. The electronics for the Ethernet modules are contained in a standard Modicon Quantum single-width case that takes up one slot in a Modicon Quantum backplane. The module, which is capable of being hot swapped, can be plugged into any available slot in the backplane.

The NOE 771 *x*0 and NOE 771 *x*1 modules provide real-time peer-to-peer communications and I/O scanning and a Modbus/TCP server. The included HTTP services provide maintenance and configuration utilities to the module.

Front View

The following figure shows the front of the NOE 771 00 Ethernet module as an example for all Ethernet modules.



1 model number, module description, color code
2 LED display
3 IP Address writable area
4 Global address label
5 100 BASE-FX MT-RJ cable connector
6 10/100 BASE-T RJ-45 cable connector
7 removable door

Key Ethernet Services

The key Ethernet services of the 140 NOE 771 (-00, -01, -10, -11) and 140 NWM 100 00 models are listed below:

Service	-00	-01	-10	-11	NWM
HTTP Server *(see page 86)*	X	X	X	X	X
FTP Server *(see page 85)*	X	X	X	X	X
Flash File System *(see page 88)*	X	X	X	X	X
BOOTP Client *(see page 67)*	X	X	X	X	X
Address Server *(see page 74)*	X	X	X	X	
SNMP V2 Agent (Network Management Service) *(see page 76)*	X	X	X	X	X
Modbus Messaging *(see page 77)*	X	X	X	X	X
I/O Scanner *(see page 79)*	X	X		X	
Hot Standby		X		X	
Global Data (Publish/Subscribe) *(see page 81)*		X		X	
Bandwidth Monitoring *(see page 84)*		X		X	
Faulty Device Replacement (Server) *(see page 229)*		X		X	
Enhanced Web Diagnosis *(see page 86)*		X		X	X
Schneider Private MIB *(see page 76)*		X		X	X
FactoryCast Application *(see page 88)*			X	X	X
User-programmed Web pages			X	X	X
JAVA Virtual Machine					X
Fiber optic connection	X	X	X	X	
RJ-45 connection	X	X	X	X	
Time Synchronization Service *(see page 82)*				X	
Electronic Mail Notification Service *(see page 83)*		X		X	

NOTE: In the detailed description of the key features, only modules in the NOE family are named. The features are also available for the 140 NWM 100 00 module, depending on the listed properties in the above table.

NOTE: In Unity Pro software, the 140 NWM 100 00 module is set in the TCP/IP Regular Network family, although it belongs to the TCP/IP FactoryCast network family. So, the services listed above (I/O scanning, Global Data, address server, Bandwidth monitoring) are not supported by the module. However, they can be selected in the TCP/IP regular network configuration in Unity Pro. (Even if they are configured, those services won't work with the module.)

Maximum Number of Networks per CPU

The following table summarizes the maximum number of networks per CPU, where "networks" means the sum of NOE, MODBUS+ and any other communication modules:

Modicon Quantum CPU Type	Supported Number of Networks
140 CPU 311 10	2
140 CPU 434 12A	6
140 CPU 534 14A	6
140 CPU 651 50	6
140 CPU 651 50	6
140 CPU 652 60	6
140 CPU 658 60	6
140 CPU 670 60	3
140 CPU 671 60	6
140 CPU 672 60	6
140 CPU 672 61	6
140 CPU 678 61	6

Front Panel Components

The front panel of the Ethernet modules contains identification markings, color codes, and LED displays. A writable area for an IP address, a global address label, and two Ethernet cable connectors is located behind the removable front panel door.

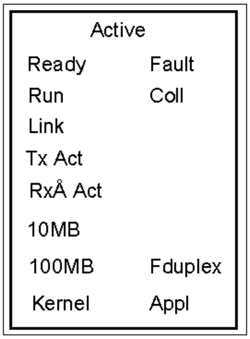
The following table provides a description of the front panel components that are shown in following figure:

Component	Description
LED Indicator Panel (*see page 43*)	Indicates the operating status of the module, and the fiber optic or Ethernet communications network to which it is connected.
IP Address Area	Provides a writable area to record the module's assigned IP address.
Global Address Label	Indicates the module's global Ethernet MAC address assigned at the factory.
100 BASE-FX Connector	Provides an MT-RJ socket for connection to a 100-megabit fiber-optic Ethernet cable.
10/100 BASE-T Connector	Provides an RJ-45 socket for connection to a shielded, twisted pair Ethernet cable.

Indicators for Ethernet Modules

Illustration

The following figure shows the NOE 771 00 LED indicators as a placeholder for all other Ethernet modules:



Description

The following table shows the LED descriptions:

LED	Color	Description
Active	Green	Indicates the backplane is configured.
Ready	Green	Indicates module is healthy.
Fault	Red	Flashes when the NOE is in crash state.
Run	Green	Flashes to indicate diagnostic code, as described below.
Coll.	Red	Flashes when Ethernet collisions occur.
Link	Green	On when Ethernet link is active.
Tx Act	Green	Flashes to indicate Ethernet transmission.
Rx Act	Green	Flashes to indicate Ethernet reception.
10MB	Green	On when the module is connected to a 10-Megabit network.
100MB	Green	On when the module is connected to a 100-Megabit network.
Fduplex		On when Ethernet is operating in the full duplex mode.
Kernel	Amber	On when in Kernel Mode. Flashing while in download mode.
Appl	Green	On when crash log entry exists.

Run LED Status

The following table lists each available state of the Run LED indicator and provides diagnostic information for that state in both the 140 NOE 771x1 module and the 140 NWM 100 module.

Indicator State	Status for 140NOE771x1	Status for 140NWM100
On (steady)	Normal operation: The NOE module is ready for network communication.	Normal operation: The NOE module is ready for network communication.
Number of flashes in sequence		
1	Not used	Not used
2	Not used	Not used
3	No Link: the network cable is not connected or is defective	No Link: the network cable is not connected or is defective
4	Duplicate IP address: The module will be set to its default IP address.	Duplicate IP address: The module will stay off-line.
5	No IP address: The module is attempting to obtain an IP address from a BOOTP server. Module is set to its default IP address.	No IP address: The module is attempting to obtain an IP address from a BOOTP server.
6	Invalid IP configuration. (Likely cause: Default gateway is not on the same subnet mask.) Module is set to its default IP address.	Using default IP address
7	No valid executive NOE present	No valid executive NOE present
8	Not used	Not used
9	-	Flash file system inoperative.

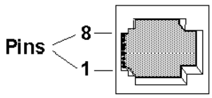
Connectors and Cabling

Overview

The following information describes the 10/100 BASE-T and 100 BASE-FX connectors.

10/100 BASE-T Twisted Pair Connector

The NOE 771 *xx*, NWM 100 00, and CPU 651 *x*0 modules' 10/100 BASE-T connector (shown below) is a standard RJ-45 twisted pair socket:



Schneider Electric recommends that you use Category 5 STP cabling, which is rated to 100 Mbps, with an RJ-45 connector.

The eight pins are arranged vertically and numbered in order from the bottom to the top. The RJ-45 pinout used by this module is:
- Receive Data (+) 3
- Receive Data (-) 6
- Transmit Data (+) 1
- Transmit Data (-) 2

100 BASE-FX

The 100 BASE-FX connector for the NOE 771 *xx*, NWM 100 00, and CPU 651 60 modules is an MT-RJ socket or a mating fiber optic cable connector.

For the NOE 771 *xx* and NWM 100 00, you may need an MT-RJ to SC (duplex) multimode fiber optic cable assembly 62.5/125mm. Schneider Electric recommends cable number 490NOC00005 to connect to fiber hubs/switches.

NOTE: The NOE 771 *xx* and NWM 100 00 are one-channel devices that can communicate over either 10/100 BASE-T or 100 BASE-FX Ethernet networks at any given time, <u>but not over both at the same time</u>.

Using the 140 NOE 771 11 Ethernet Module in a Quantum Safety Project

Overview

Version 4.2 and higher of the 140 NOE 771 11 Ethernet module can be included in one of the following:
- Quantum safety applications
- non-safety applications

A Quantum safety application is controlled by a Quantum safety PLC, consisting exclusively of safety modules that perform safety functions. A safety module is denoted by the letter *S* at the end of its module name.

When used in a Quantum safety application, the 140 NOE 771 11 is a *non-interfering module*–it does not negatively affect the execution of the PLC’s safety functions.

For more information about Quantum safety products, refer to the *Quantum Safety PLC Safety Manual*.

Restricted and Unrestricted Memory Areas

The memory addresses of a Quantum safety PLC can be configured as one of the following:
- safety-restricted memory areas
- unrestricted memory areas

Only safety modules can write data to a safety-restricted memory area. For more information about restricted versus unrestricted memory areas, refer to the topic *Safety PLC Write Protection* in the *Unity Pro XLS Operating Mode Manual*.

The 140 NOE 771 11 Ethernet module–as a non-interfering module–cannot write data to safety-restricted memory areas. Instead, the 140 NOE 771 11 Ethernet module can write data only to unrestricted memory areas.

The memory areas of a Quantum safety project are user-configurable. Using Unity XLS, you can designate address ranges as either safety-restricted or unrestricted. For instructions on how to configure memory areas, refer to the topic *Configuration of Quantum Processors with Unity Pro XLS* in the *Unity Pro XLS Operating Mode Manual*.

Health Bit Assignments

CAUTION

RISK OF DATA LOSS

The default address assignments for the following data blocks overlap:

- Global Data Health Bit block (%MW1)
- I/O Scanning Health Bit block (%MW1)
- I/O Scanning Device Control block (%MD1)

You must edit these address assignments so they do not overlap. If these address assignments overlap, the PLC will overwrite data and the overwritten data will be lost.

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

The 140 NOE 771 11 makes different address assignments–regarding both I/O Scanning and Global Data Health Bit Blocks–for safety and non-safety applications, as follows:

Health Bit Block	Non-Safety Application Addresses	Safety Application Addresses
I/O Scanning	%I %IW	%M[1] %MW[1]
Global Data	%I %IW %MW	%MW[1] (only)
[1] All I/O Scanning and Global Data Health Bit Block address assignments, for a Quantum safety application, must be made in unrestricted memory address areas.		

Section 1.3
Installing the Module

Introduction

This section contains installation and configuration information for the NOE 771 *xx* Ethernet modules.

What Is in This Section?

This section contains the following topics:

Before You Begin

Initial Checks

⚠CAUTION
UNINTENTIONAL OPERATION
If you do not enter the correct address pairs into the BOOTP server, you could communicate to the wrong device.
● Ensure that the MAC address matches the intended IP address in your BOOTP server.
Having two or more devices with the same IP address can cause unpredictable operation of your network.
● Ensure that your Ethernet module receives a unique IP address.
Failure to follow these instructions can result in injury or equipment damage.

Complete the following checks before installing the module.
- Determine how the module will be assigned its Ethernet address parameters. The default method is BOOTP.
- Verify that your Ethernet network is properly constructed.
- Verify that you are using the proper cabling scheme for your application Cabling Schemes *(see page 51)*.

Determine the Appropriate Ethernet Address Parameters

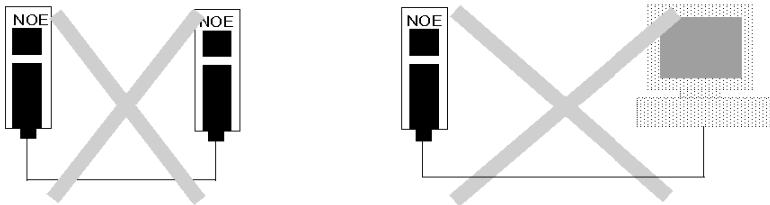
Consult your system administrator to determine whether:
- you must configure new IP, gateway, and subnet mask addresses
  or
- the module will obtain its Ethernet address parameters from a BOOTP server Assigning Ethernet Address Parameters

If the administrator assigns new address parameters, you will need to configure the module through the Unity Pro interface module configuration screen.

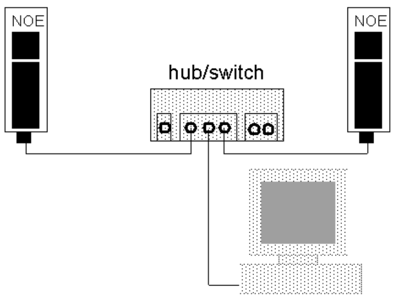
Verify the Network Topology

Because the module includes an Ethernet embedded web server, you must use a cross-link cable to connect it to another device. Do not use a standard cable. For the network to operate properly, you must route the cable for each device through an Ethernet hub or switch.

The following figure shows two incorrect network topologies.



The following figure shows a correct network topology.



Cable Schemes

Overview

The following information provides guidelines on proper cabling schemes for your Ethernet configuration.

Twisted Pair Length

In a standard Ethernet cabling scheme network nodes such as the Quantum Ethernet module connect via a cable to a port on a central hub or switch. The maximum cable length between nodes depends on whether they are connected through hubs or switches, as the following table describes:

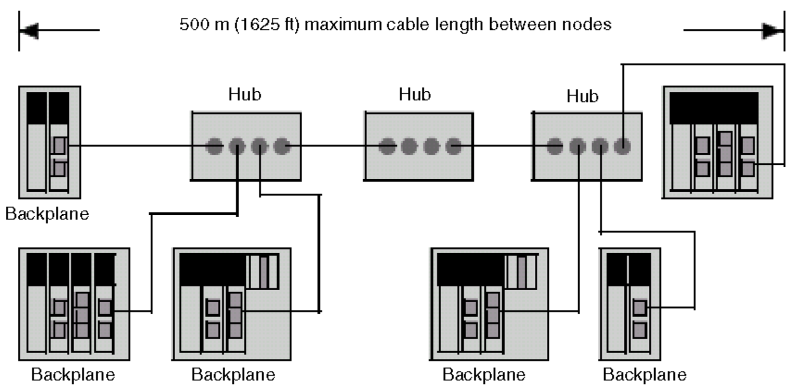
Device Type	Max. Cable Length from Node to Hub/Switch	Max. Number of Hubs/Switches Between Any Two Nodes	Max. Distance Between Network Nodes
hub	10 BASE-T: 100 m	3	500 m
switch	10/100 BASE-T: 100 m 100 BASE-FX: 2 km	unlimited	unlimited

Cabling with Conventional Hubs

The figures and tables that follow show the maximum number of hubs and the maximum cable length between nodes allowed if using hubs.

10BASE-T Distances with Hubs

The 10 BASE-T cabling allows for 3 hubs with a link maximum distance of 100 m (325 ft) and a total network diameter of 500 m (1625 ft).



10/100BASE-T Cable Distances with Switches

The 10/100BASE-T cabling has a link maximum distance of 100 m. The is no limit on the number of switches.

100BASE-FX

The maximum cable segment length for 100BASE-FX multimode cable (1300 nm) is 2 km. The number of switches you can implement over that distance is unlimited.

Security

Overview

The following information describes firewalls. A firewall is a gateway that controls access to your network. To restrict access to your Ethernet controller and I/O network, you may want to consider a firewall.

Types of Firewalls

There are network-level and application-level firewalls:

- **Network-level firewalls:** These firewalls are frequently installed between the Internet and a single point of entry to an internal, protected network.
- **Application-level firewalls:** An application-level firewall acts on behalf of an application. It intercepts all traffic destined for that application and decides whether to forward that traffic to the application. Application-level firewalls reside on individual host computers.

Port Numbers Used by NOE

The following table contains the port numbers used by NOE:

Protocol	Port Number
Modbus/TCP	TCP 502
HTTP	TCP 80
SNMP	UDP 61
FTP	TCP 21

You may need to provide the information in this table to your system administrator so that the firewall configuration will allow access to your PLC from outside of your facility.

Mounting the Module in the Quantum PLC Backplane

Before you Begin

Locate the backplane in which you will mount the module. Ensure that an open slot is available.

NOTE: The module can be installed only in a local backplane.

NOTE: Ensure when installing the module that it does not exceed the Quantum backplane power requirements as specified in the *Quantum with Unity Pro Hardware Reference Manual*.

Backplane Slot Replacement

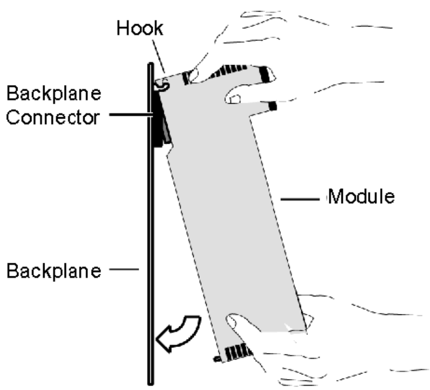
You may place the module in any slot on the backplane. You do not need to place it next to other modules.

Tools Required

You will need one medium-size, Phillips-head screw driver.

Mounting the Module in the Backplane

Perform the following steps to mount the module.

Step	Action
1	Holding the module at an angle, mount it on the two hooks located near the top of the backplane. The following figure shows the correct way to hold the module. 
2	Swing the module down so its connector engages the backplane connector.
3	Using a Phillips-head screw driver, tighten the screw at the bottom of the module between 2 and 4 in-lbs or between .22 and .45 Newton meters of torque.

Grounding the Module

The module is grounded upon installation in the Quantum PLC backplane. For instructions on grounding the backplane, see the *Quantum with Unity Pro Hardware Reference Manual*.

NOTE: If you connect the module to the Ethernet hub or switch using a shielded cable, the cable is also grounded. Elsewhere in this guide is a discussion of cabling schemes and recommendations Cabling Schemes *(see page 51)*, as well as instructions for connecting the module to an Ethernet network Connecting the Cable *(see page 56)*.

Connecting the Cable

Overview

The following information pertains to cabling.

NOTE: The 140 NOE 771 xx is capable of communicating over either a 10/100BASE-T or a 100BASE-FX Ethernet network at any given time, but not both at the same time.

Accessories

Schneider Electric sells the following switches:

Hub or Switch	Description
499NEH10410	hub with 4 ports 10 BASE-T
499NOH10510	hub with 3 ports 10 BASE-T and 2 ports 10 BASE-FL
499NTR10010	transceiver 10 BASE-T/10 BASE-FL
499NEH14100	hub with 4 ports 100 BASE-TX
499NTR10100	transceiver 100 BASE-TX
499NES18100	switch with 8 ports 10/100 BASE-TX
499NES17100	managed switch with 7 ports 10/100 BASE-TX
499NOS17100	managed switch with 5 ports 10/100 BASE-TX and 2 ports 100 BASE-FX

The following Schneider Electric cables support multicast filtering *(see page 186)*:

Cable	Description
490NTW000 02/05/12/40/80 U	StraightThru cable
490NTC000 05/15/40/80 U	Crossover cable

Fiber Optic

Remove the protective cap from the module’s MT-RJ connector port and the protective cap from the tip of the black connector on the MT-RJ fiber optic cable (as shown in the following figure). The plug only fits to the socket in one way. It should snap into place.

Assigning Ethernet Address Parameters

Overview

⚠CAUTION
UNINTENTIONAL OPERATION If you do not enter the correct address pairs into the BOOTP server, you could communicate to the wrong device. ● Ensure that the MAC address matches the intended IP address in your BOOTP server. Having two or more devices with the same IP address can cause unpredictable operation of your network. ● Ensure that your Ethernet module receives a unique IP address. **Failure to follow these instructions can result in injury or equipment damage.**

The following information describes how to assign IP address parameters.

As shipped from the factory, the 140 NOE 771 *xx* module does not contain an IP address. If you have not programmed the unit with an Ethernet configuration extension, the module does not contain an IP address. When the module starts up without an IP address, the module will attempt to obtain an IP address from the network's BOOTP server.

You can assign IP address parameters using the BOOTP Lite software utility.

NOTE: You can configure the IP address using Web pages on the modules 140 NOE 771 01, -11, and NWM 100 00.

Using a BOOTP Server

A BOOTP server is a program that manages the IP addresses assigned to devices on the network. Your system administrator can confirm whether a BOOTP server exists on your network and can help you use the server to maintain the adapter's IP address.

See *Using BOOTP Lite to Assign Address Parameters, page 67*.

How an Unconfigured Module Obtains an IP Address

On startup, an unconfigured NOE 771 *xx* module ("as shipped") will attempt to obtain an IP address by issuing BOOTP requests. When a response from a BOOTP server is obtained, the IP address in the response is used. If no BOOTP response is received within two minutes, the module uses the default IP address derived from its MAC address.

NOTE: The MAC address is assigned at the factory and is recorded on a label on the front panel, above the cable connector. This is a unique 48-bit global assigned address. It is set in PROM. The Ethernet address is recorded on the label in hexadecimal, in the form 00.00.54.*xx.xx.xx*.

Connecting to the Default IP Address

To connect to the default IP address with your PC, set up an active route from your PC. To do this with Windows 95/98/ME/NT/2000 or Windows XP, use the following procedure. You can use the routes for connecting Ethernet components with other address ranges.

<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>1</td><td>Be sure the NOE module is running.</td></tr>
<tr><td>2</td><td>Obtain the default IP address of the NOE derived from its MAC address (for example, 84.0.0.2).</td></tr>
<tr><td>3</td><td>Open an MS-DOS window.</td></tr>
<tr><td>4</td><td>Add an active route for the local NOE by typing:
C:\>ROUTE ADD <target> MASK <mask>   <gateway>
e.g.
C:\>ROUTE ADD 84.0.0.0 MASK 255.0.0.0   205.217.193.205
Use the default IP address of the NOE module as target address. Use a class A subnet mask for connecting to every 84.0.0.0 address. The gateway address is the IP of your PC. The result is that MS Windows will now talk to any address that starts with an 84 that is directly connected to a hub or switch directly accessible to your machine, or that can be seen by the specified route/gateway.</td></tr>
<tr><td>5</td><td>Confirm that there is a new entry in the active route table by typing:
C:\>route print:
The following figure confirms that the new entry was added to the active route table.
Active Routes:
<table>
<tr><th>Network Address</th><th>Netmask</th><th>Gateway Address</th><th>Interface</th><th>Metric</th></tr>
<tr><td>0.0.0.0</td><td>0.0.0.0</td><td>205.217.193.205</td><td>205.217.193.205</td><td>1</td></tr>
<tr><td>84.0.0.0</td><td>255.0.0.0</td><td>205.217.193.205</td><td>205.217.193.205</td><td>1</td></tr>
<tr><td>127.0.0.0</td><td>255.0.0.0</td><td>127.0.0.1</td><td>127.0.0.1</td><td>1</td></tr>
</table></td></tr>
<tr><td>6</td><td>Verify that a connection is made by typing:
C:\>ping 84.0.0.2
The following figure shows that the connection is verified.
Reply from 84.0.0.2: bytes=32 time<10ms TTL=32
Reply from 84.0.0.2: bytes=32 time<10ms TTL=32
Reply from 84.0.0.2: bytes=32 time<10ms TTL=32
Reply from 84.0.0.2: bytes=32 time<10ms TTL=32</td></tr>
</table>

Specifying Address Parameters

Consult your system administrator to obtain a valid IP address and an appropriate gateway and a subnet mask, if required. Then use your programming panel to make the appropriate changes.

BOOTP Server Responds

If the server responds with address parameters, the NOE 771 *xx* module will use those parameters as long as power remains applied to the module.

If the server does not respond, the module will retry its request for two minutes.

BOOTP Server Does Not Respond

If no BOOTP response is received, the NOE 771 *xx* module will use the default IP Address.

During this time the Run indicator displays a pattern of five flashes for a BOOTP attempt and six flashes for using the default IP.

NOE 771 xx Duplicate IP Address Test

In all cases, when the NOE 771 *xx* module receives an IP address, it will test for duplicate addresses by sending broadcast ARP requests three times at 5-second intervals.

If a Duplicate IP Address is found on the network, the NOE 771 *xx* will stay off-line to avoid a network disruption. It will display a pattern of four flashes to indicate a Duplicate IP Address detection.

Automatic ARP

If there are no replies to its requests, the NOE 771 *xx* automatically sends three ARP requests at 2-second intervals to announce its presence on the network.

Establishing the FTP Password

Establishing the FTP Password

The FTP password is established using the Embedded Web Server. This topic contains information on how to access the web server for purposes of changing the FTP and HTTP passwords. The first thing the system administrator should do upon accessing the web server is change the FTP password. Doing this restricts access to the web server functionality to the system administrator.

Web server pages and their functionality are discussed in deatil in the chapter Embedded Web Pages.

Introduction to Accessing the Web Server

Each Quantum NOE module contains an embedded Web server, which allows you to access diagnostics and online configurations for the module and its associated controller (PLC).

The web pages can only be viewed using Internet Explorer 4.0 or higher supporting JRE 1.4.2_04 or higher.

For information about the additional functionality provided by the FactoryCast system in the NOE modules, see the *FactoryCast Manual*.

How to Access the Web Server

Before you can access the module's home page, you must enter the full IP address in the Address or Location box in the browser window. For example: *http://hostname* (*hostname* is the full IP address or DNS host name).

After you do this, the Schneider Electric Web Utility home page displays.

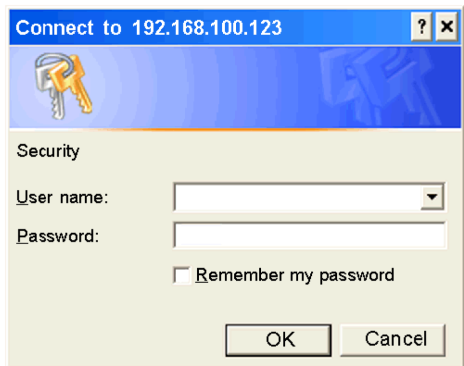
Schneider Web Utility Home Page

From the Quantum home page, you can access pages for:

- changing the FTP password *(see page 62)*
- changing the HTTP password *(see page 65)*
- diagnostic and configuration information (the embedded Web pages chapter provides additional information)

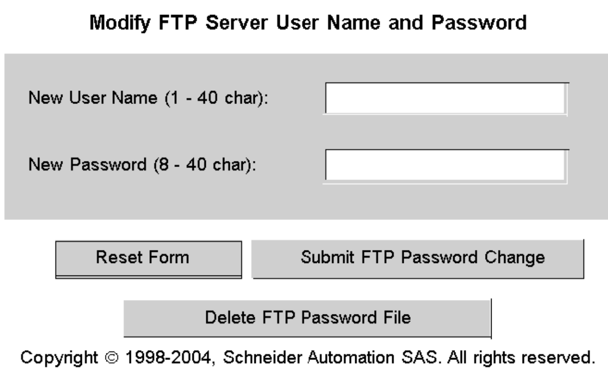
Modifying the FTP Server Password

The following steps detail how to link to the correct web page for modifying the FTP password

Step	Action
1	Enter the URL, for example, *http://hostname/secure/embedded/ftp_passwd_config.htm*
2	At that URL, supply a user name and password:  **Note**: The default User name is USER, and the default Password is USERUSER. Both should be changed by the system administrator during module installation.
3	Upon supplying the user name, password, and clicking the **OK** button, the **Modify FTP Server User Name and Password** page displays.

FTP Username and Password Modify Page Overview

The following figure shows the page used for modifying the FTP user name and password:

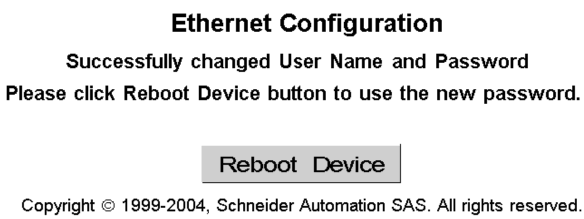


Change the Username and Password

At this point, the system administrator should change the Username and Password to restrict future access to the system. The following steps should be used.

Step	Action
1	Type in the new Username in the **New User Name** field.
2	Type in the new Password in the **New Password** field.
3	Click the **Submit FTP Password Change** button.

The following figure shows the message that is generated if you click on the Submit FTP Password Change button:



Clicking the **Reboot Device** button will reset the username and password for the NOE board.

NOTE: The reboot requires about 40 seconds. (With large applications, reboot may require up to one (1) minute.)

NOTE: During the reboot, all services provided by the NOE module are not available.

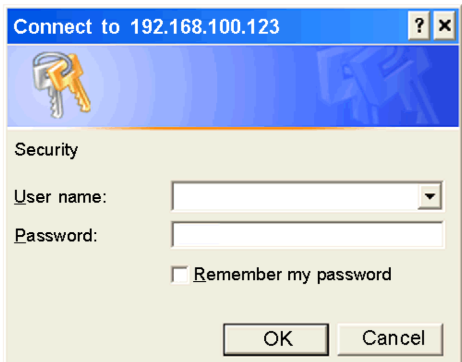
Establishing HTTP and Write Passwords for NOE

Overview

The following information describes how to set the passwords for HTTP and Data Monitor Write for the NOE.

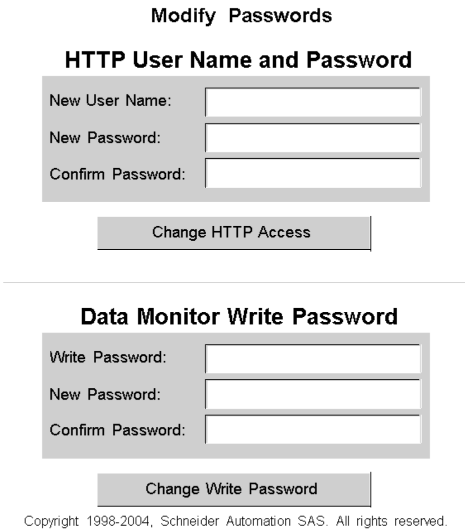
Connect to the Web Page

Before you can change the HTTP or Data Monitor Write passwords, connect to the correct web page:

Step	Action
1	Enter the URL, for example, http://hostname/secure/embedded/http_passwd_config.htm
2	At that URL, supply a user name and password in the dialog box. The size of the user name and password is a maximum of 16 characters in non-extended ASCII.  **Note:** The default User Name is USER, and the default Password is USER. Both should be changed by the system administrator during module installation.
3	Upon supplying the user name, password, and clicking the **OK** button, the **Modify Passwords** page displays.

Modify Passwords Page

A single web page is used to modify both the HTTP and Data Monitor Write passwords:



Changing Security Access Parameters

A system administrator should change the default username (USER) and password (USER) to restrict future access to the system. After you decide whether you are changing the HTTP or Write Password security parameters, follow the appropriate procedure below:

HTTP (Username and Password)

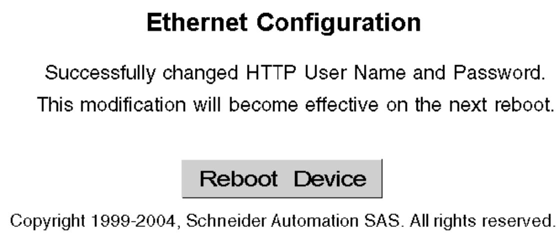
Step	Action
1	Type a new username in the **New User Name** field.
2	Type a new password in the **New Password** field.
3	Confirm the new password in the **Confirm Password** field.
4	Click the **Change HTTP Access** button.

Data Monitor Write (Password)

Step	Action
1	Type the old password in the **Write Password** field.
2	Type a new password in the **New Password** field.
3	Confirm the new password in the **Confirm Password** field.
4	Click the **Change Write Password** button.

Security Access Change Verification

The following figure shows the message that is generated if you click the **Change HTTP Access** or **Change Write Password** button:



Clicking the **Reboot Device** button will reset the username and password for the NOE board.

NOTE: The Reboot will take about 40 seconds. Larger applications can take up to one (1) minute.

NOTE: During the reboot all services provided by the NOE are not available.

Using BOOTP Lite to Assign Address Parameters

Overview

⚠CAUTION
UNINTENTIONAL OPERATION
Ensure that the MAC address matches the intended IP address in your BOOTP server. If you do not enter the correct address pairs into the BOOTP server, you could communicate to the wrong device.
Failure to follow these instructions can result in injury or equipment damage.

The following information describes how to use the BOOTP Lite utility software.

BOOTP Lite Utility

Instead of a BOOTP server, Schneider Electric's *BOOTP Lite Server Software* utility can be used to provide the IP address, subnet mask, and default gateway to the NOE 771 xx module.

Refer to the BOOTP Lite Server Software user documentation for instructions.

NOTE: BOOTP Lite Server Software and user documentation are available for download at www.modicon.com.

Section 1.4
Customer Support

Overview

This section tells you how to get customer support from Schneider Electric's documentation and regional help centers.

What Is in This Section?

This section contains the following topics:

Customer Support Documentation

Support Documentation

If you have any problems, please first consult the documentation listed above or the MS-Windows documentation.

For the most up-to-date NOE Ethernet controller information, please:

Step	Action
1	Access the Schneider Electric Web site.
2	Search technical information.
3	Select Quantum from the list of cross-product families.
4	Access either: ● resolutions for resolutions to product issues ● product manuals for the most recently published user documentation ● product announcements

Contact Information

Contact Information

Please find the nearest Schneider Electric sales office by visiting **http://www.schneider-electric.com**. In the **Select a country** list, click the country closest to you for customer support.

Chapter 2
Ethernet Communication Services

Introduction

This chapter describes the Ethernet communications services available on 140 NOE 771 *x*1 and 140 CPU 651 *x*0.

What Is in This Chapter?

This chapter contains the following topics:

Modicon Quantum with Unity Ethernet Services

Table of Ethernet Services

The Ethernet services of the 140 NOE 771 01, 140 NOE 771 11, and CPU 651 x0 add functionality to the system. Some services can be configured, others require no configuration. Some services provide diagnostic information. Access to the services is through either a Web page or the Unity Pro application.

		NOE 771 01	**NOE 771 11**	**CPU 651 x0**
Service				
Address Server *(see page 74)*	configure	Web Unity Pro	Web Unity Pro	Web
	diagnostic	NA	NA	NA
BOOTP Client *(see page 67)*	configure	Unity Pro	Unity Pro	Unity Pro
	diagnostic	NA	NA	NA
SNMP V2 Agent (Network Management Service) *(see page 76)*	configure	Web Unity Pro	Web Unity Pro	Web Unity Pro
	diagnostic	NA	NA	NA
Schneider Private MIB *(see page 76)*	configure	NA	NA	NA
	diagnostic	NA	NA	NA
Modbus Messaging *(see page 77)*	configure	Unity Pro	Unity Pro	Unity Pro
	diagnostic	Web Unity Pro	Web Unity Pro	Web Unity Pro
I/O Scanner *(see page 79)*	configure	Unity Pro	Unity Pro	Unity Pro
	diagnostic	Web Unity Pro	Web Unity Pro	Web Unity Pro
Global Data (Publish/Subscribe) *(see page 81)*	configure	Web Unity Pro	Web Unity Pro	Unity Pro
	diagnostic	Web Unity Pro	Web Unity Pro	Web Unity Pro
Time Synchronization Service *(see page 82)*	configure	NA	Web	NA
	diagnostic	NA	Web	NA
Electronic Mail Notification Service *(see page 83)*	configure	Web	Web	Web
	diagnostic	Web	Web	Web
Bandwidth Monitoring *(see page 84)*	configure	Unity Pro	Unity Pro	Unity Pro
	diagnostic	Web Unity Pro	Web Unity Pro	Web Unity Pro
FTP Server *(see page 85)*	configure	NA	NA	NA
	diagnostic	NA	NA	NA

		NOE 771 01	NOE 771 11	CPU 651 x0
Service				
HTTP Server *(see page 86)*	configure	NA	NA	NA
	diagnostic	NA	NA	NA
Faulty Device Replacement (Server) *(see page 229)*	configure	Web Unity Pro	Web Unity Pro	Unity Pro
	diagnostic	NA	NA	NA
Hot Standby	configure	NA	NA	NA
	diagnostic	NA	NA	NA
Flash File System *(see page 88)*	configure	NA	NA	NA
	diagnostic	NA	NA	NA
FactoryCast Application *(see page 88)*	configure	NA	Web	NA
	diagnostic	NA	Web	NA

Address Server

Overview

The following information describes the services provided by the address server.

- BOOTP server
- DHCP server

The DHCP server responds to both a DHCP and a BOOTP server.

BOOTP Server

NOTE: The BOOTP server is available on the 140 NOE 771 *xx* models.

The BOOTstrap Protocol (BOOTP) software, compliant with RFC 951, is used to assign IP addresses to nodes on an Ethernet network. Devices (hosts) on the network issue BOOTP requests during their initialization sequence. A BOOTP server that receives the request will extract the required IP address information from its database and place it in BOOTP response messages to the requesting devices. The devices will use the assigned IP addresses from the BOOTP server for all network communication.

NOTE: In order for BOOTP to work properly, you need to correctly configure the following:

- IP address
- subnetwork mask
- gateway address

NOE BOOTP Server

Your NOE 771 *xx* module comes supplied with a BOOTP server. This feature allows you to provide IP addresses to all the I/O devices being serviced by the NOE 771 *xx*. Providing a BOOTP server that is built into your NOE 771 *xx* module eliminates the need for an external BOOTP server.

NOTE: The NOE 771 xx's BOOTP server can not provide its own IP address.

You can configure your NOE 771 *xx*'s BOOTP server from the module's HTTP Web page. Configure the 140 CPU 651 *x*0 using the Unity Pro editors. Configuring allows you to add to, remove from, and edit devices in the BOOTP server's database, which is maintained in the module's non-volatile memory.

DHCP Server

NOTE: The DHCP server is available on the 140 NOE 771 *x*1 models.

The Dynamic Host Configuration Protocol (DHCP) is a superset of the BOOTP protocol. Your 140 NOE 771 *x*1 has a DHCP server. The DHCP server is compliant with RFC 1531. The DHCP server can be used to provide the IP configuration to devices using BOOTP or devices using DHCP.

The DHCP server has entries that use the MAC address to serve the IP configuration and entries in the server that use the role name to serve the IP configuration. Elsewhere in this book are details for the NOE address server configuration *(see page 229)*.

If you are migrating a BOOTP configuration from a 140 NOE 771 *x*0 module to the new 140 NOE 771 *x*1 module, see Address Server Configuration/Faulty Device Replacement *(see page 229)* for details on the automatic upgrade of your configuration for the new DHCP server.

NOTE: Before placing the NOE on a corporate network, Schneider Electric recommends that you discuss the installation with your MIS department. It is likely that your company's corporate network has at least one DHCP server running already. If the NOE's DHCP server is running on the same network, it may disturb the network.

NOTE: To avoid any possible problem related to the NOE's DHCP server on the corporate network, ensure that the DHCP server is not running in the NOE by not having address entries in the configuration. If there are no configured devices in the address server configuration page *(see page 229)*, then the NOE will not start the DHCP server.

Faulty Device Replacement

Faulty device replacement and the address server are discussed in detail at Address Server Configuration/Faulty Device Replacement *(see page 229)*.

SNMP and Schneider Private MIB Overview

Overview

Simple Network Management Protocol (SNMP) is configured on your NOE or 140 CPU 651 *x*0.

Introduction

Network management software allows a network manager to:
- monitor and control network components
- isolate problems and find their causes
- query devices, such as a host computer(s), routers, switches, and bridges, to determine their status
- obtain statistics about the networks to which they are attached

Simple Network Management Protocol

Your NOE module or 140 CPU 651 x0 controller supports the Simple Network Management Protocol (SNMP), which is the standard protocol used to manage a local area network (LAN). SNMP defines exactly how a *manager* communicates with an *agent*.

The SNMP defines the format of:
- requests that a manager sends to an agent
- replies that the agent returns to the manager

MIB

Each object to which SNMP has access must be defined and given a unique name. Manager and agent programs must both agree on the names and meanings of the fetch and store operations. The set of all objects SNMP can access is known as a *Management Information Base (MIB)*.

Private MIB

Schneider obtained a private MIB, Groupe_Schneider (3833). Under the Groupe Schneider private MIB is a Transparent Factory Ethernet (TFE) private MIB. The Transparent Factory SNMP embedded component controls the Schneider private MIB function.

Modbus Messaging

Overview

The following information describes the functionality of the Modbus/TCP server.

The Client

The node that initiates a data transaction is called a *client*. All Modicon Quantum Ethernet modules provide the user with the capability to transfer data to and from nodes on a TCP/IP network using a communication instruction. All PLCs that support networking communication capabilities over Ethernet can use either the MBP_MSTR Ladder Logic instruction to read or write controller information or IEC communication blocks.

The Server

The node that receives an inquiry is the *server*. Using the standard Modbus/TCP protocol, all Modicon Quantum Ethernet modules provide access to controller data. Any device, PC, HMI package, another PLC, or any Modbus/TCP compliant device, can access data from the PLC. The Modbus/TCP server also allows programming panels to log into the controller over Ethernet.

Modbus/TCP Connections and Commands

The Ethernet module supports up to 64 simultaneous Modbus/TCP server connections. To guarantee consistency of changes to the controller configuration, the Ethernet module allows only one programming panel to be logged in at a time.

The Ethernet modules supports these Modbus/TCP commands:
- Read Data
- Write Data
- Read/Write Data
- Get Remote Statistics
- Clear Remote Statistics
- Modbus 125 Commands (used by programming panels to download a new exec to the NOE)

The maximum register size for read requests, write requests, or read/write requests made using either a MBP_MSTR or Modbus command is as follows:

Command Request Type	Maximum size
Read	125 read registers
Write	123 read registers
Read/Write	125 read registers / 121 write registers

Performance

The following table shows the performance characteristics of the NOE Ethernet module's Modbus/TCP server.

Parameter	Value
Typical Response Time (ms)	0.6
Number of Modbus connections (client and server)	64 (NOE 771 x1, NWM 100 00, HE CPU 651 x0), 16 Client (NOE 771 x0), 32 Server (NOE 771 x0)
Number of simultaneous login channels	1

NOTE: Ethernet module's Modbus/TCP performance measurements are made with a Modicon Quantum 140 CPU 534 14A PLC.

I/O Scanner

Introduction

The functionality of your Ethernet module is further enhanced by the addition of an I/O Scanner, which you can configure with the Schneider Electric programming panels.

I/O Scanner Features

NOE version 3.5 and later include these enhancements:

Feature	Availability	Improvement	Description
Improved Timeout and Retry Transmission algorithm	Unity Concept ProWORX	Improves TCP connection management	2004 version uses a variable timeout. Older versions used a fixed timeout. Difference: Faster retransmission rates
Enable/Disable I/O Scanner entry	Unity Concept	Reduces network traffic volume	Use the 'Device Control Block' ● Set = 0 Enable channel for normal data exchange ● Set = 1 Disable channel Note: Concept Users Concept uses diagnostic words. Set = FF Disable channel
Send Modbus requests across a router	Unity Concept ProWORX	Allows routers to connect remote I/O devices to a controller	Automatic
Dynamic TCP port allocation	Unity Concept ProWORX	Improves connection/disconnection performance	Reserves TCP client ports 3072 (0xC00) through 4992 (0x1380) Allocates ports dynamically
Repetitive rate display	Unity	Check status using GUI	Status displays in I/O Scanning tab of module editor

I/O Scanner Parameters

Functionality of the I/O Scanner.

Parameter	Value
Max. No. of Devices	64: 140 NOE 771 00 (Version 2.2 or earlier) 128: 140 NOE 771 00 (Version 3.0 or later), 140 NOE 771 01, and 140 NOE 771 11 only 128: HE CPU 651 x0
Max. No. of Input Words	4000
Max. No. of Output Words	4000
Health Timeout Value	User configured: 1...65535 ms in 1 ms increments
Last Value (Input)	User configured (zero or Hold)

Parameter	Value
IP Address	User configured IP address of scanned device (Slave IP)
Local and Remote Register Reference	User configured
Repetitive Rate	User configured: 0...65535 in multiples of: ● 16 ms, for 140 NOE 771 *x*1 ● 10 ms, for 140 CPU 651 *x*0
Unit ID	User configured Configure ID only if using a bridge
Operation through a bridge	Modbus bridge: Supported
	Modbus Plus bridge: Supported
Gateway/Bridge Device	Select the check box if you are using a bridge or a gateway device.

The I/O Scanner Concepts topic *(see page 196)* explains how to configure the I/O scanner.

Performance

Performance data details are provided at I/O Scanner Response Times topic *(see page 225)*.

Global Data

Overview

The Global Data service is a real-time publisher/subscriber mechanism providing the most efficient data exchange for PLC application coordination.

Devices that support Global Data are arranged in a distribution group for the purpose of application variable exchange and synchronization. Each Global Data device can publish up to one network (application) variable and subscribe up to 64 network (application) variables.

The Quantum NOE module’s embedded Global Data Configuration Page provides a configuration screen to determine which and how many application variables are exchanged with this service. After configuration, the exchanges between all stations belonging to the same distribution group are done automatically.

The Global Data service uses %MW words (4x registers) or unlocated variables for Global Data exchanges.

Key Features of Global Data

The main features for Global Data are:

- One publisher and multiple subscribers
- A device can publish one network variable of up to 512 %MW words (4x registers) or unlocated variables
- A device can subscribe to several network variables of up to 2048 %MW words (4x registers) or unlocated variables
- A device subscribes to the complete network variable
- One distribution group per network IP address
- Application defined publication rate
- Up to 64 Global Data network variables (numbered from 1 to 64) can be part of the data distribution group
- The NOE module has only one multicast address; consequently, it can only publish and subscribe inside the group
- A device can participate in several distribution groups by using multiple NOE communication modules in the rack

Global Data’s publish/subscribe mechanism allows multiple subscribers to receive a single data transmission. This is an advantage over client/server services, which require multiple transmissions to specific destinations. There are two immediate benefits:

- reduces overall network traffic
- ensures tighter synchronization of multiple subscribers

Time Synchronization

General

The time service synchronizes computer clocks over the Internet. For example, the time of one client is synchronized either with another server or to a referenced time source like a radio or satellite receiver.

Typical time service configurations utilize multiple redundant servers and diverse network paths to achieve high accuracy and reliability. Time service accuracy can be within a millisecond on LANs and up to a few tenths of milliseconds on WANs.

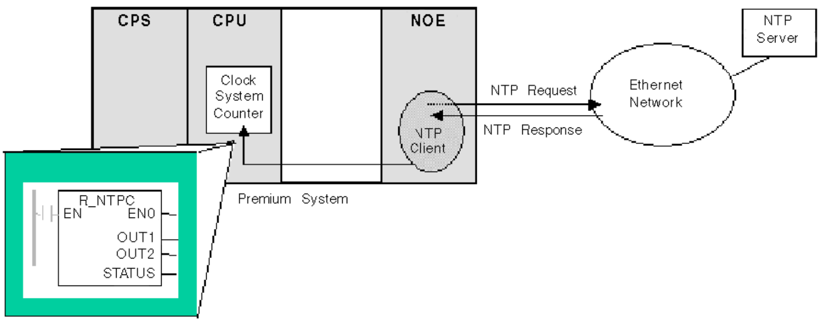
Use the time service for:
- event recording: sequence events
- event synchronization: trigger simultaneous events
- alarm and I/O synchronization: time stamp alarms

Time Synchronization and Time Stamps

On an Ethernet network, all controllers should be synchronized with the same clock time, which is provided by an NTP server. In each application program, events or application data (I/O values for example) can be time stamped using the application stacks.

The Ethernet interface module connects to an NTP time server and provides the source-time synchronization signal, which is sent to the CPU.

To read the clock, a function block is used in the controller's project (application program):



Electronic Mail Notification

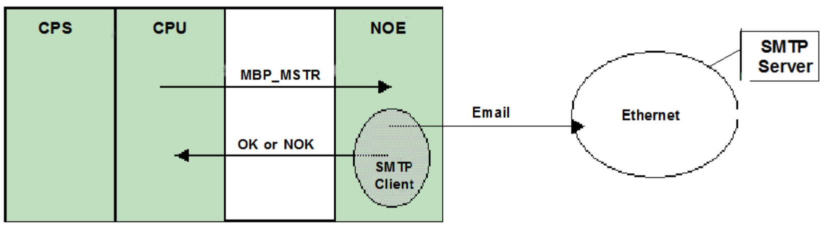
General

The electronic mail notification service allows controller-based projects to report alarms or events. The automation controller monitors the system and can automatically create an electronic mail message alert with data, alarms, and/or events. Mail recipients can be either local or remote.

- Based on predefined events or conditions, messages are created using the MBP_STR function block.
- The email message is constructed from predefined headers plus variables and text (a maximum of 238 bytes). This message is sent directly from the automation system to the local email server.
- Mail headers contain common predefined items—recipient list, sender name, and subject. These items can be updated by an authorized administrator.

Mail Service Client

The NOE communications module and 140 CPU 651 *x*0 modules include an SMTP client. When the module receives a specific request from the controller, the module sends an email message to the mail server.



Bandwidth Monitoring

Overview

Bandwidth Monitoring allows the user to access and monitor the NOE module's CPU allocation for each of the following services:
- Global Data *(see page 181)*
- I/O scanning *(see page 195)*
- Modbus messaging *(see page 77)*

The Bandwidth Monitoring service retrieves workload data and returns one of two pieces of information: whether the module has free resources or whether the module is working at capacity. Knowing the resource allocation helps you:
- assess resource allocation
- determine the number of NOEs needed in a system

NOTE: Users who want to use Bandwidth Monitoring do not need to develop a new set of access functions. The actual NOE CPU load is computed each second.

Bandwidth Monitoring Load Rates

The Bandwidth Monitoring service checks once a second and computes four (4) values in private data. These values are returned as the percentage of the NOE's CPU that is allocated to:
- Global Data *(see page 181)*
- I/O scanner *(see page 195)*
- Modbus messaging *(see page 77)*
- other services and idle

CPU time spent in other services is shown as "Other" or "Free." Bandwidth Monitoring uses the same functions as used by SNMP.

The three service rates, Global Data, I/O Scanner, and Messaging, are computed using the following formula:

```
(Current load * 100) / Maximum Load
```

The table shows the (dynamically computed) **Maximum Load Rate** for the NOE module:

Diagnostic Service	Workload Data Returned	Maximum Load
Global Data	Number of published variables per second	800
I/O Scanner	Number of transactions per second	4200
Messaging	Number of messages treated per second	410

NOTE: The loads depend on controller scan times. Each application has an expected scan time. Therefore, when evaluating loads, ensure that the controller scan time is set to the expected scan time for the modeled application.

FTP Server

Overview

The following information describes the services provided by the FTP Server. The FTP server is available on all:

- 140 NOE 771 ***xx*** communications modules (including the NOE module)
- CPU 651 *x*0 modules

FTP Server

The FTP server on the NOE module and CPU 651 *x*0s is available as soon as the module receives an IP address. Any FTP client can log on to the module, which requires the client use the correct user name and password.

The FTP Server can:

- update the NOE module’s firmware by downloading a new Exec
- provide error log visibility by uploading error log files
- upload/download BOOTP server and SNMP configuration files

NOTE: There is only one FTP client per module.

The FTP server is protected with a default user name and password.

Default user name	USER case-sensitive
Default password	USERUSER case-sensitive
See the FTP password *(see page 61)* topic to change the password or add/delete usernames on the FTP server.	

Embedded Web Pages (HTTP Server, Web Configuration and Diagnostics)

HTTP Server

The Modicon Quantum with Unity Ethernet modules' Hypertext Transport Protocol (HTTP) server is available as soon as the module receives an IP address.

Use the HTTP Server to:

1. view . . .

- the module's Ethernet statistics
- the controller and I/O information
- the server information (BOOTP/DHCP/FDR)
- the diagnostic information for some Ethernet services

2. configure the module's Ethernet services

Use Internet Explorer version 4.0 or later. For a complete list of services, see the Key Features table *(see page 41)*.

The HTTP server is protected with a default user name and password.

Default user name	USER case-sensitive
Default password	USER case-sensitive
The size of the user name and password is a maximum of 16 characters in non-extended ASCII.	

NOTE: The NOE 771 *xx* and CPU 651 *xx* support a maximum of 32 HTTP simultaneous connections. Browsers may open multiple connections so 32 HTTP connections do not indicate 32 simultaneous users.

NOTE: The NOE 771 00 and NOE 771 01 modules do not support user downloaded Web pages. You will need to purchase either the NOE 771 10, NOE 771 11, or the NWM 100 00 module to support user downloaded Web pages.

Web Diagnostics

The embedded Web server provides Web pages to diagnose the following Transparent Factory/Real Time services.

Diagnostic Service	Description
Global Data *(see page 81)*	• status of all Global Data services • status of all subscribed and published variables • publication/subscription rate
I/O Scanning *(see page 79)*	• status of all I/O Scanning services • status of individual scanned devices • actual I/O scanning rate

Diagnostic Service	Description
Messaging *(see page 77)*	• diagnostic information for Port 502 messaging
Time Synchronization *(see page 82)*	• status of client and link to the server • date and time • status of Daylight Saving Time (DST) option
Electronic Mail Notification *(see page 83)*	• status of SMTP server • track messages sent and received • track errors
Bandwidth Monitoring *(see page 84)*	• throughput measurement of NOE by service

NOTE: All these pages are protected by the general HTTP password.

Web Configuration

Elsewhere in this guide is a description of the web configuration pages *(see page 86)*.

Additional Ethernet Services

Hot Standby

The Ethernet Hot Standby system consists of two identical Modicon Quantum systems. Each Quantum system contains:

- a Modicon Quantum Hot Standby with Unity controller (140 CPU 671 *xx*0), (140 CPU 672 **xx**0) or (140 CPU 678 61)
- a remote I/O head
- NOE 771 *xx* modules (no more than six)
- a power supply

The Hot Standby modules are connected to each other via a fiber-optic cable. Both remote I/O heads are connected to the remote I/O network and to each other.

Schneider Electric recommends:

1. remote I/O networks use redundant cables (drops are not redundant)
2. switch connecting the NOEs to the network (available switches):
   - 499NES17100
   - 499NOS17100

FactoryCast and User Customizable Web Pages

FactoryCast is a software package that you use to customize a Web site on the embedded Web server module. The site can be accessed via a browser to view and modify data from a Modicon Quantum with Unity controller (PLC).

FactoryCast provides all the Web pages and Java applets you need to view run-time data from your controller. You can use the FactoryCast default Web site simply by configuring the module and accessing it with a browser over an intranet.

Modules that use FactoryCast to add customized web pages on a site are:

- NOE 771 10
- NOE 771 11
- 140 NWM 100 00

Flash File System

The NOE 771 *xx* modules are equipped with a Flash File System, which allows changing or updating the executive, kernel, and Web site files by an upload.

Part II
Modicon Quantum with Unity Ethernet Modules Services

Chapter 3
Start Communication with Unity Pro

Introduction

This chapter tells you how to begin Ethernet network configuration with Unity Pro.

What Is in This Chapter?

This chapter contains the following sections:

Section 3.1
How to Configure the Communication

Overview

This section describes how to configure the communication.

What Is in This Section?

This section contains the following topics:

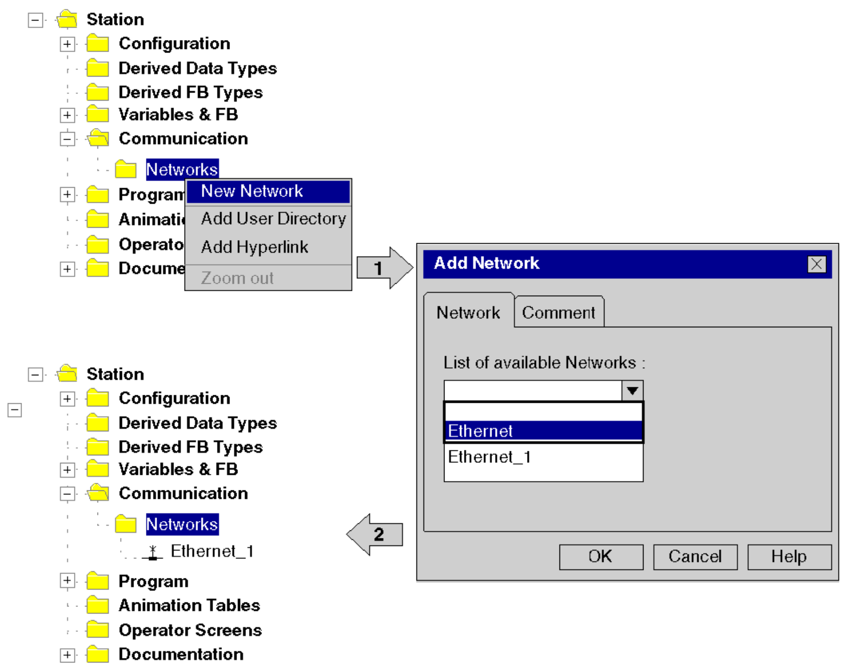
Add a New Network to the Communication Folder

Add a New Network to the Communication Folder

After starting a new application, the **Communication** folder under the **Station** tree branches the **Network** folder and the **Routing** table folder (Premium platforms only). These two folders are empty. Under the **Network** folder, the user can insert the networks by menu. A click on the right mouse-button above **Network** pops up a contextual menu. The user selects the type of network he wants to add. For easier use, a network name will be suggested with the prefix of the network type (**Ethernet_1** or **Modbus+_1**). By choosing a new network the next available number for the network is chosen automatically, for example, **Ethernet_1** then **Ethernet_2** and so on. At any moment, the user may rename any Netlink.

The user can also attach a comment that describes each configured network. The OK button adds the network as subfolder.

The names of network nodes are also called NetLink. These are the names of logical networks.

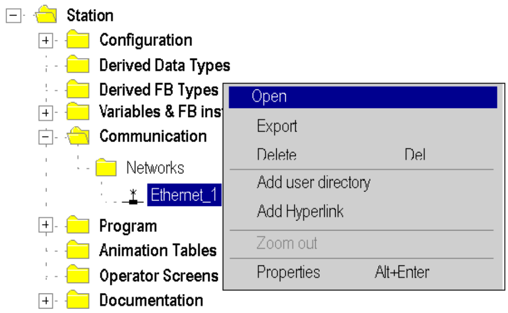


Configure Network

Configure Network

On the network folder, by a double-clicking action or by the Open item on contextual menu, the editor of the corresponding communication screen is opened in order to set the specific network services.

The figure shows the contextual menu to start network properties.

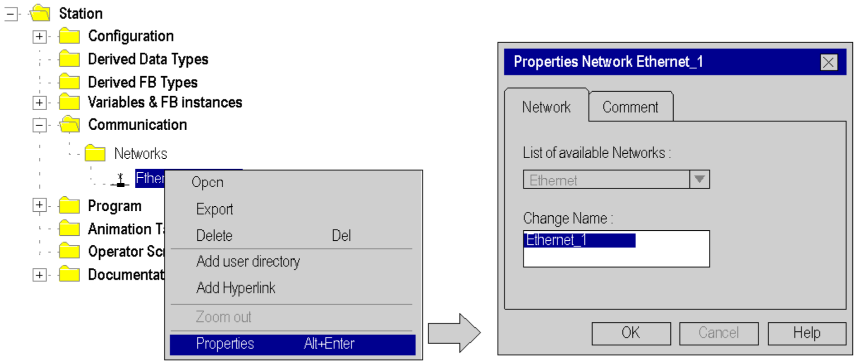


Properties of a Network

Properties of a Network

The contextual menu proposes the user to see again the properties of a configured network. Here, the user can change the NetLink name and the associated comment.

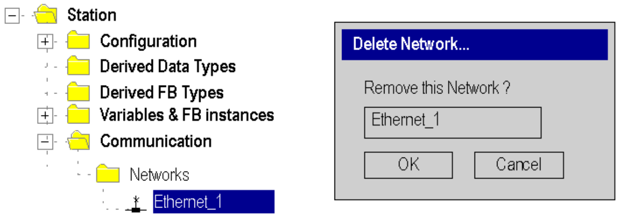
The figure shows the Ethernet property window:



Delete an Existing Network Folder

Delete an Existing Network Folder

With a right-mouse-click above the network folder, a contextual menu appears. Here the user is able to delete the network configuration. In this case, the subfolder of the network will also be removed in application browser.



NOTE: If this removed network was previously attached to a communication module, this module loses its link and it will work with its default parameters.

Section 3.2
Unity Soft Communication Links

Overview

This section presents the principle of communication implementation and describes the relationship between software configuration of networks and the hardware configuration of the network controllers.

What Is in This Section?

This section contains the following topics:

Communication Configuration Principle

Introduction

The configuration of communication links between different devices with Unity Soft includes three different configuration parts.

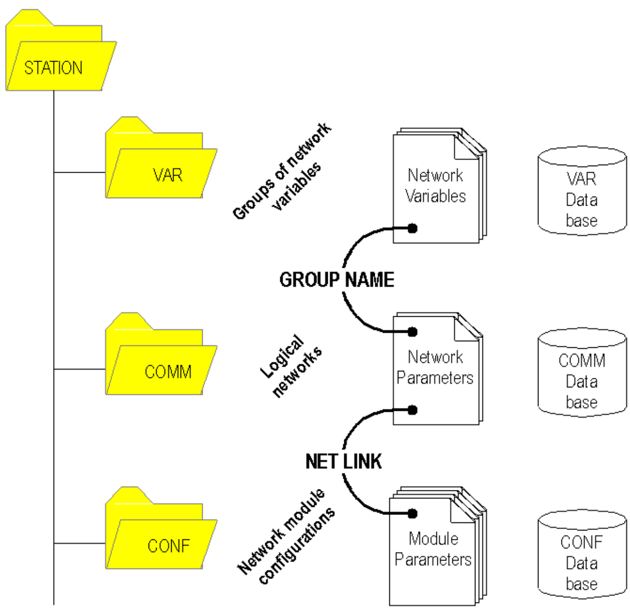
- configuration of the network controller
- configuration of the logical network
- configuration of network variables

Configuration

The communication configuration supports the *free mode* of Unity Soft. That means the user can first configure the module and then the communication or the user can configure the communication and then the module.

This will be provided through a NetLink that must be selected in the module configuration. The network variables including in the VAR folder are linked with a group name that defines an IP domain over Internet network.

The illustration shows the three parts involved in communication configuration:



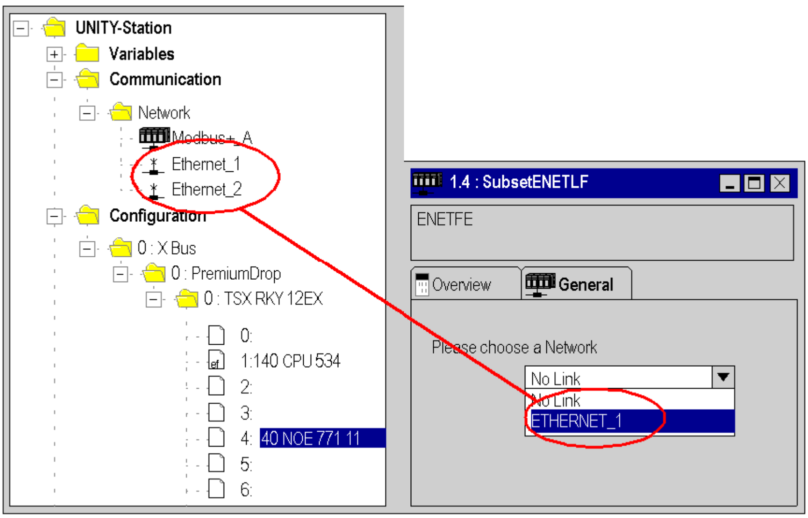
Link between Configuration and Communication

NetLinks

During Unity Pro application design, the NetLinks are created and inserted on sub-folder Communication under Network. These are the names of logical networks.

Under configuration folder, on the communication module node included in the current station, the list of existing NetLinks is proposed to select and attach one network to one module. Only the NetLink that can be managed by this module, are displayed in the list box on module configuration screen. No NetLink can be edited and created here (no edit box), but this list contains at least the No_Link field.

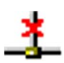
The following figure shows the window for the Ethernet link for the Quantum NOE module.



Attaching a NetLink to a Module

When a network is attached to a module, the icon of the corresponding node is changed and the network editor displays the address of the module in the rack .

The Icon in the Network folder indicates whether the link is attached to a module or not:

	Icon when no communication module is attached to the NetLink
	Icon when a communication module has been attached to the NetLink

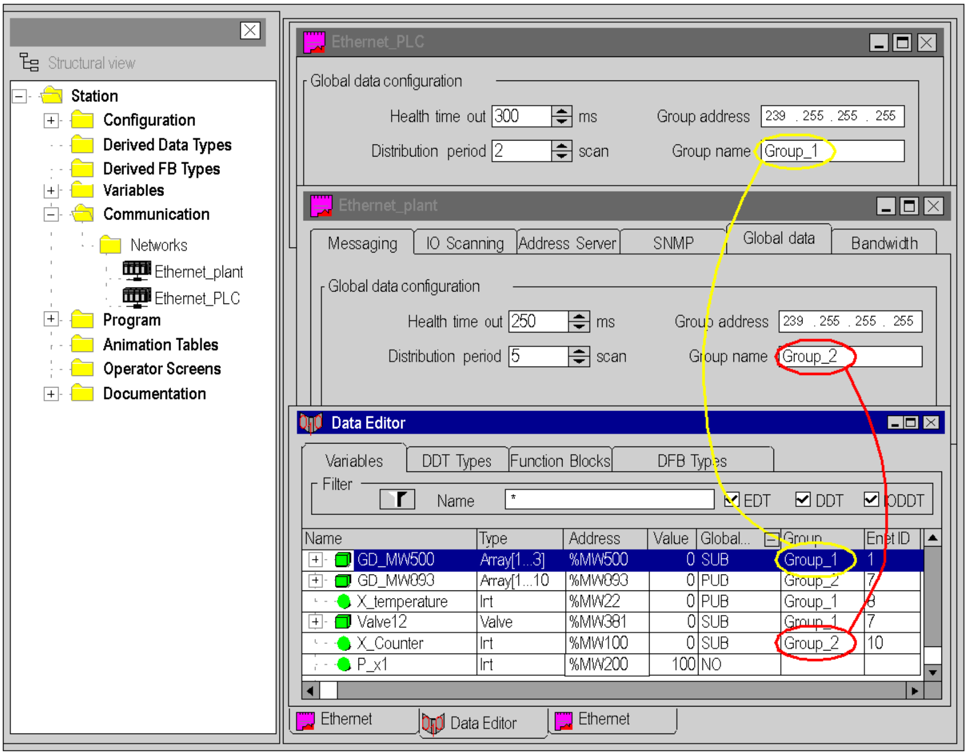
Link between Data and Communication

Network Variables and Groups

The groups of Ethernet network variables are created in the Ethernet network communication folders. An IP domain determines a group. In Unity Pro, one network can support only one group.

In Data Editor, the list of all current groups is provided to select in which group each Ethernet network variables is included. Nevertheless, the group field is also a free entry editing box, in order to give a group name not yet defined in communication folder. The build step checks this link.

The illustration shows corresponding fields in Communication configuration and the Data Editor:



Section 3.3
Selecting the Ethernet Module

Introduction

This chapter contains the software pages for the NOE module.

What Is in This Section?

This section contains the following topics:

Selecting the Quantum NOE Ethernet Module

General Description

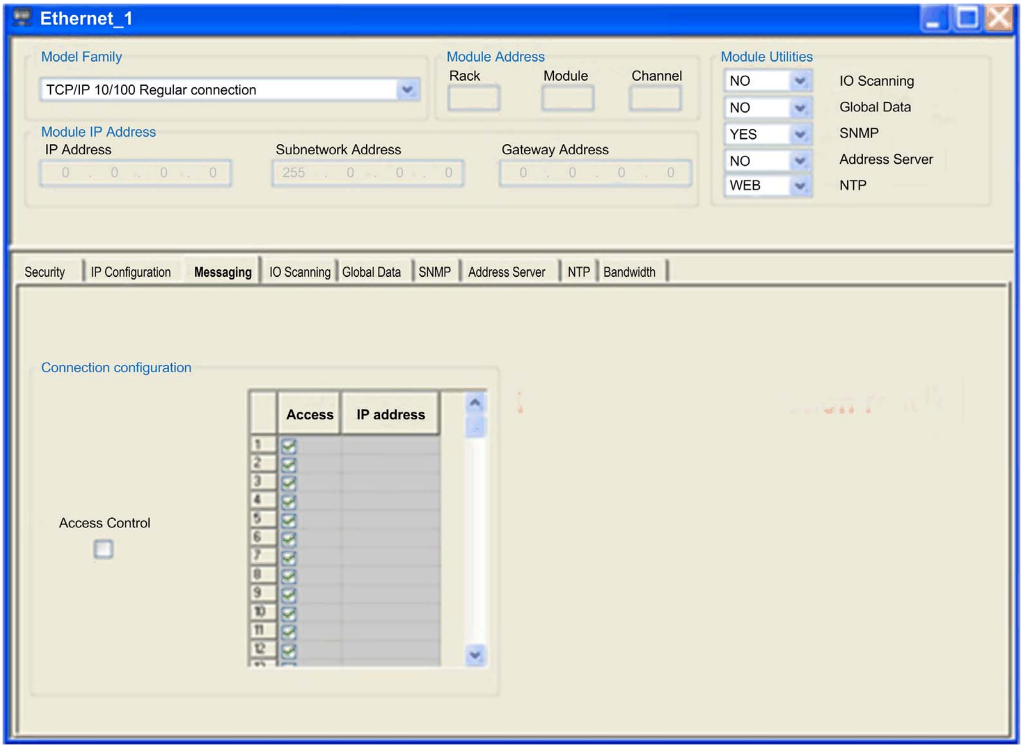
After configuring Ethernet communication *(see page 93)*, the Ethernet module parameters can be configured.

When you select the model family, all the corresponding communication module configuration options are displayed automatically. The module services allow the following settings to be made:

Setting	Description
No	Setting deactivated.
Yes	Setting activated. Parameters are set using the Unity Pro menu window.
Web	Setting activated. Parameters are set using the configured NOE Web pages. Unity Pro menu window deactivated. **Note**: Not available for every model family.

NOTE: The availability of the displayed settings depends on the selected model family and can vary.

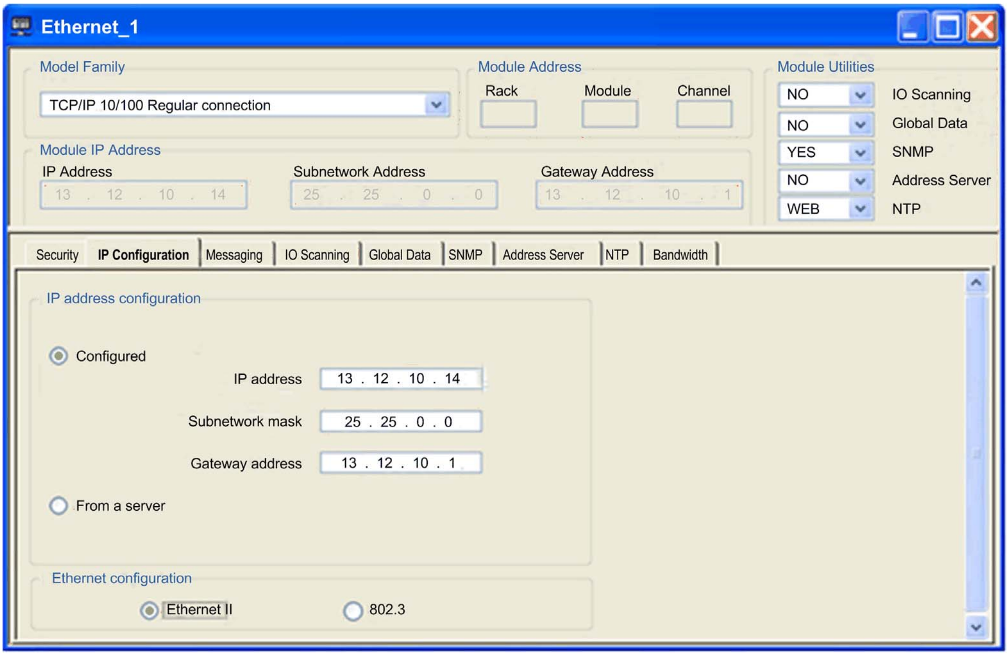
The screen shot shows an example of the menu window of the NOE Ethernet module (TCP/IP 10/100 Regular connection).



Parameter description:

Parameter	Description
Model Family	Quantum NOE Ethernet Module settings
Module Address	Not used
Module Utilities	For module service configuration options, see above
Module IP Address	Overview of the IP address parameter set

After selecting the model family **TCP/IP 10/100 Regular Connection,** the following mask appears. The image also displays the activated module services.



NOTE: The availability of the displayed register depends on the selected model family and can vary.

After selecting the **Yes** option in module services, the tab corresponding to the module is activated.

Security (Enable / Disable HTTP, FTP, and TFTP)

Security and HTTP, FTP, and TFTP Services

The module uses HTTP services to provide access to its embedded Web pages. The module uses FTP and TFTP services to support various features including firmware upgrades, FDR services, and Ethernet remote IO.

The module's HTTP, FTP, and TFTP services can be disabled or enabled using the Unity Pro **Network Editor** →**Security** screen.

HTTP, FTP, and TFTP services are disabled by default in projects created using Quantum EIO Version 1.5 and Unity Pro Version 8 or higher. They are enabled by default in projects created using previous versions of Unity Pro.

You can use Unity Pro to enable or disable HTTP, FTP, and TFTP services as described in the following procedure.

If the HTTP, FTP, or TFTP services have been enabled with Unity Pro, they can also be enabled or disabled at run time using an MBP_MSTR block with operation code 65520 (dec).

Using Unity Pro to Enable and Disable Firmware Upgrade & FDR and Web Access Services

Perform the following steps to enable or disable FTP/TFTP or HTTP services on the module.

Step	Action
1	In the Unity Pro **Project Browser** →**Structural View**, double-click the desired network in the **Communication** →**Networks** directory to open the **Network Editor**.
2	Click the **Security** tab.
3	On the **Security** screen, choose the appropriate setting: (**Enabled** or **Disabled**) for the service or services.

The edits will not take effect until they are successfully downloaded from your PC to the CPU and from the CPU to the communication modules and network devices.

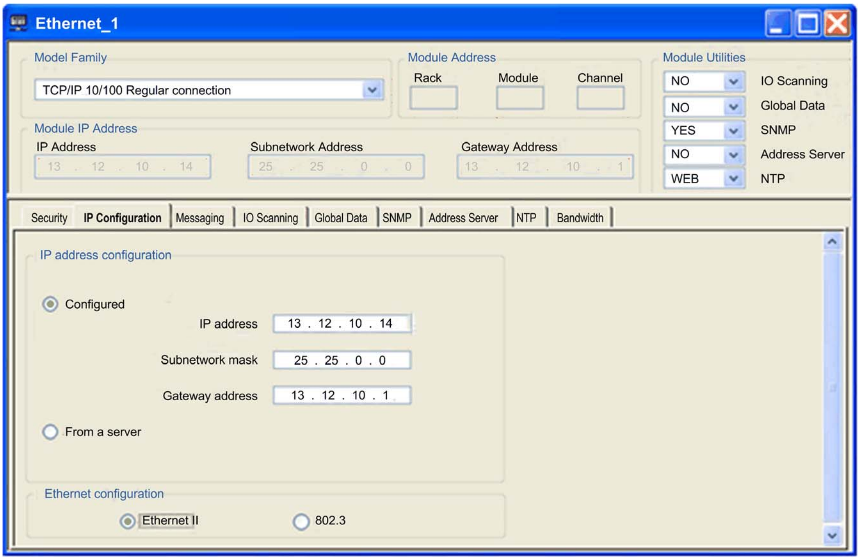
IP Configuration

General Description

The **IP configuration** tab enables you to configure IP address settings for the Quantum Ethernet module. IP address settings become active after:

- the hardware is connected
- the configuration is downloaded to the PLC in the Quantum Ethernet module

The following figure shows the IP configuration for the Quantum Ethernet model family:



Description of the selection properties

Selection	Description
Configured	Activate the IP address, subnet mask, and gateway address. The data is activated after the configuration is downloaded to the PLC.
Client / Server	The Quantum NOE Ethernet module receives its IP address parameter through a BOOTP server on startup.
Ethernet configuration	Select the default protocol as Ethernet or 802.3.

Quantum NOE Ethernet Messaging Configuration

Introduction

Ethernet messaging gives the user the opportunity to send and receive Ethernet messages. Data traffic is handled by the client/server procedure.

Parameter description:

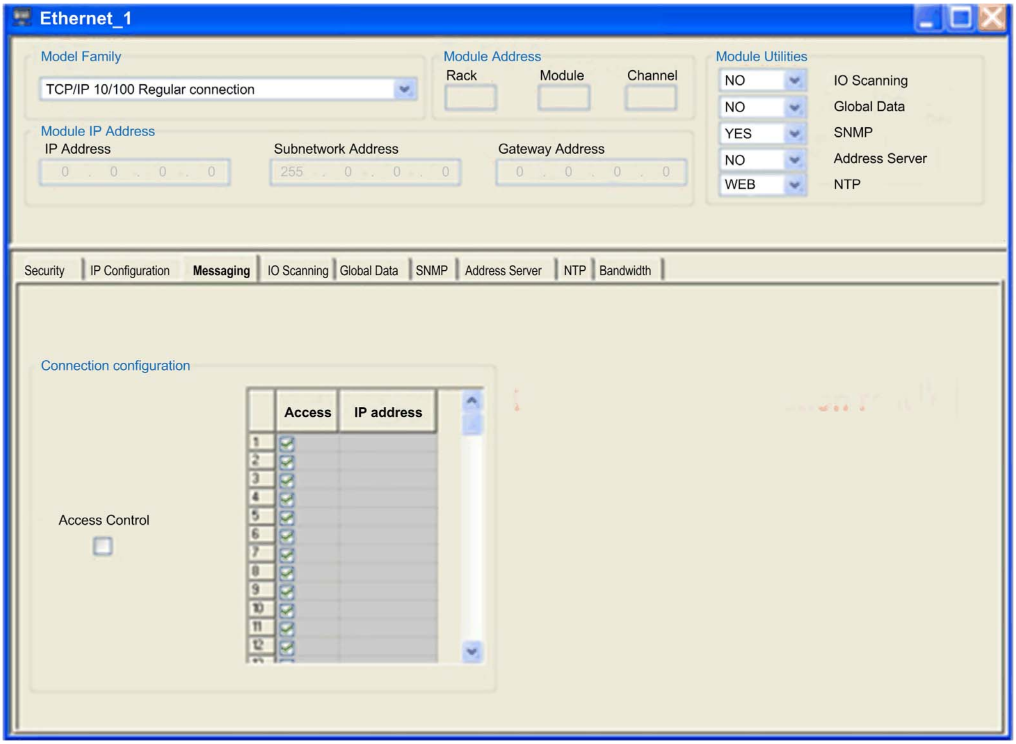
Setting	Description
Connection configuration	Activates general data transfer.
Access	Activates data transfer between specific nodes.
IP Address	Defines the node for the Ethernet Messaging procedure.
Access Control	Activates or deactivates control of remote devices that are attempting to open a TCP connection to the module.

Messaging Tab

The following procedure shows how to access the **Messaging** page:

Step	Action
1	In the Unity Pro project browser, go to the **Structural view**. Expand (**+**) the **Communication** folder until you locate the name of the Ethernet logical network associated with the module.
2	Right click the Ethernet logical network and select **Open**. **Result:** The module configuration screen appears.
3	Select the **Messaging** tab. (See the illustration below.)

The **Messaging** tab is shown below.



Configuration parameters can be accessed in two areas on the Messaging tab screen:
- the **Connection Configuration** area
- the **Access Control** area

Connection Configuration Area

The **Connection Configuration** area is used to:
- activate an access control utility
- list the remote devices that can connect to the module according to a communication protocol

Access Control

The **Access Control** box is used to activate or deactivate control of remote devices that are attempting to open a TCP connection to the module. The functionality depends on whether the box is checked or not:

- **checked:** Access control management is activated and the **Access** column of the table is active (no longer grayed out).
  - The module can only communicate to the addresses entered in the 128 available spaces in the **Slave IP Address** column.
  - With the module in client mode it can only connect to remote devices selected by the **Access** column in the **Connection Configuration** table.
- **unchecked:** Access control management is inoperative and the **Access** column of the table is not active (grayed out).
  - With the module in server mode, remote third-party devices can connect as clients (before communication with the module) without being declared in the table.

NOTE: Access control is only effective on the TCP/IP profile and assists module operations in server and client mode.

Section 3.4
Selecting the Ethernet Coprocessor

Introduction

This section describes configuring the Modicon Quantum with Unity coprocessor, 140 CPU 651 x0.

What Is in This Section?

This section contains the following topics:

Selecting the Modicon Quantum with Unity Ethernet Controller

General Description

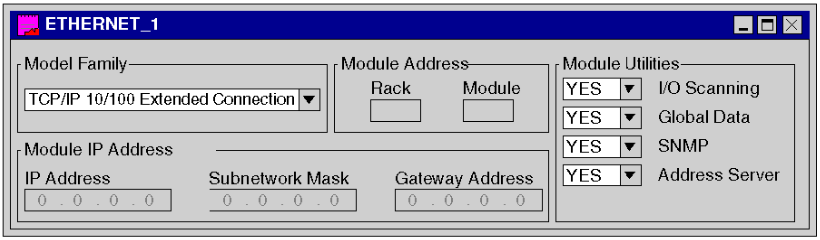
After configuring Ethernet communication *(see page 93)*, the Ethernet module parameters can be configured.

When you select the model family, all the corresponding communication-module configuration options display automatically. The module services allow the following settings to be made:

Setting	Description
No	Setting deactivated.
Yes	Setting activated. Parameters are set using the Unity Pro menu window.

NOTE: The availability of the displayed settings varies and depends on the selected model family.

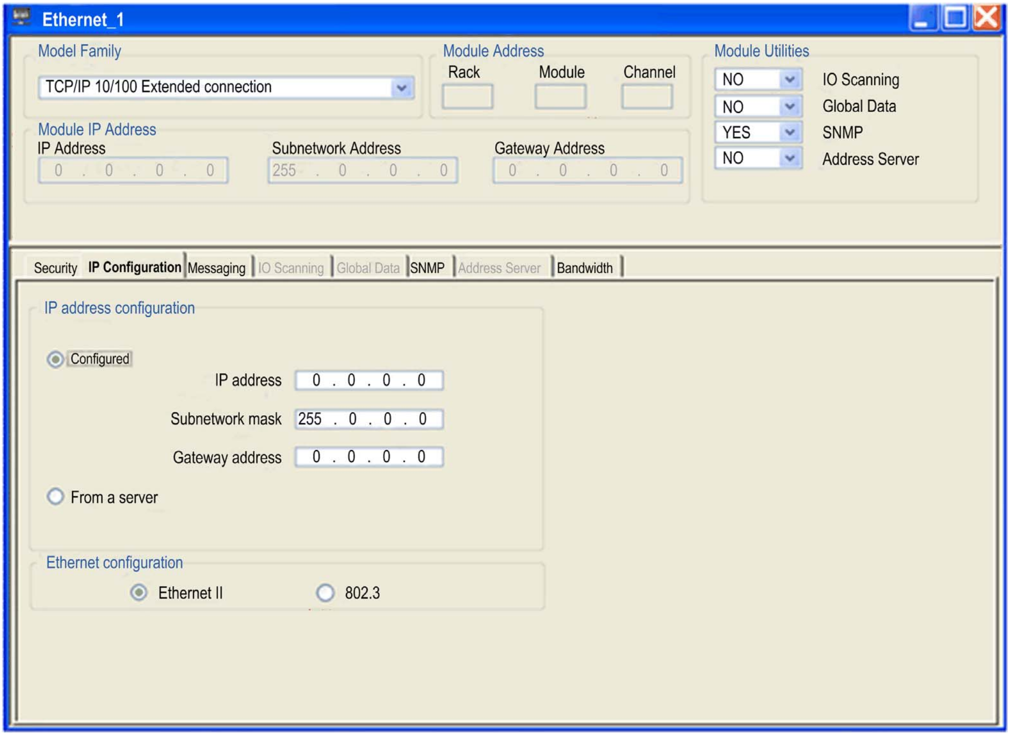
The screen shot shows an example of the menu window of the Ethernet module CPU 651 *x*0 (TCP/IP 10/100 Extended connection).



Parameter description:

Parameter	Description
Model Family	Modicon Quantum with Unity Ethernet module settings.
Module Address	Not used.
Module Utilities	For module service configuration options, see above.
Module IP Address	Overview of the IP address parameter set.

After selecting the model family **TCP/IP 10/100 Extended Connection,** the following mask appears. The image also displays the activated module services.



NOTE: The availability of the displayed register depends on the selected model family and can vary.

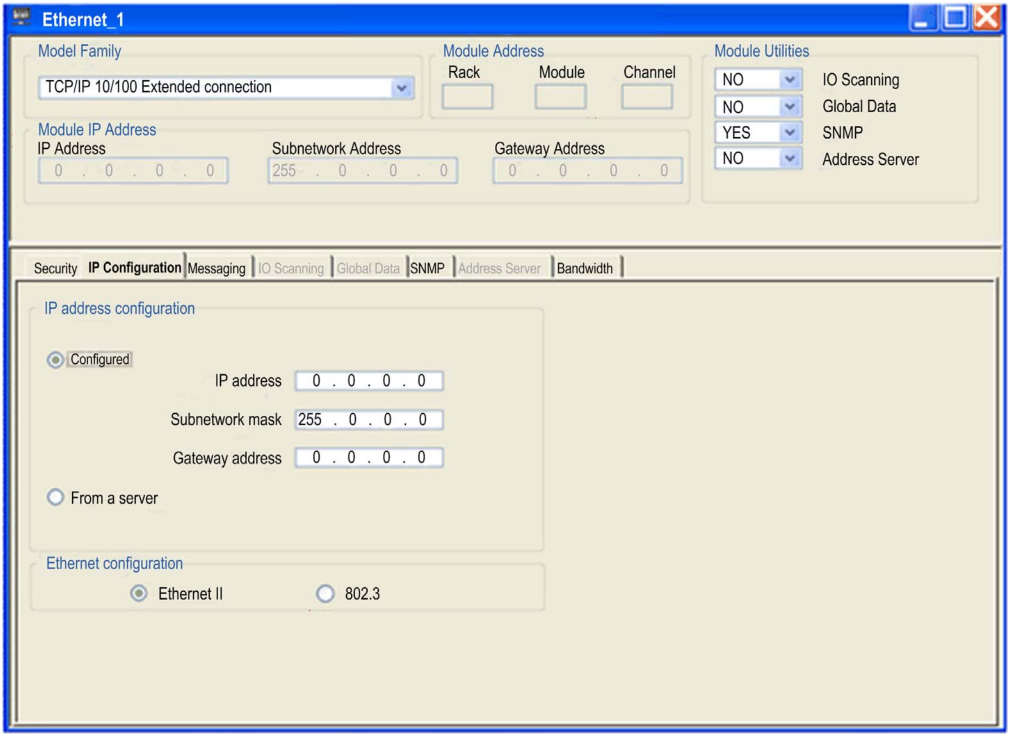
After selecting the **Yes** option in module services, the tab corresponding to the module is activated.

Configuring the IP Address of the Ethernet Controller

General Description

The **IP configuration** tab enables you to configure the IP address settings. The settings are activated after the connection to the hardware and the configuration is downloaded to the Modicon Quantum with Unity Ethernet controller, 140 CPU 651 x0.

The diagram shows the IP configuration for the Modicon Quantum with Unity Ethernet controller, 140 CPU 651 x0.



Description of the selection properties

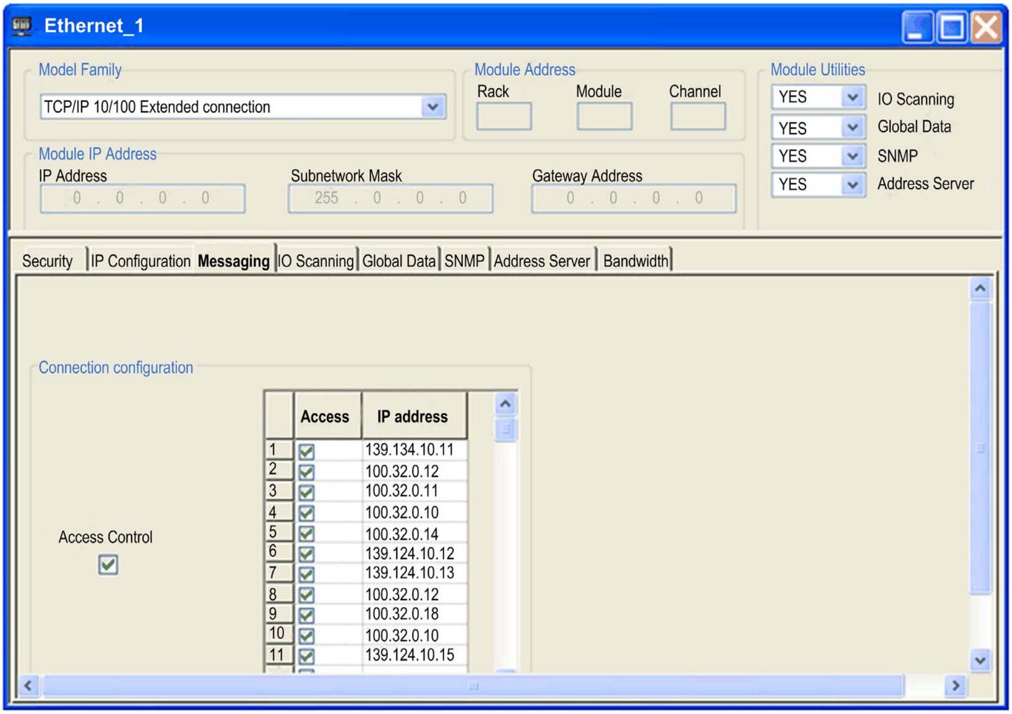
Selection	Description
Configured	Activate the IP address, Subnet mask and Gateway address. The data is activated after the configuration is downloaded to the controller.
Client / Server	The Modicon Quantum with Unity Ethernet controller receives its IP address parameter through a BOOTP server on startup.

Modicon Quantum with Unity Ethernet Controller Messaging Configuration

Introduction

Use the **Messaging** tab in Unity Pro to send and receive messages via Ethernet. The data traffic is handled by the client / server procedure.

The following illustration shows the **Messaging** dialog box.



Parameter description

Setting	Description
Connection configuration	Activates general data transfer.
Access	Activates data transfer between specific nodes.
IP address	Defines the node for the Ethernet messaging procedure.
Access Control	Activates or deactivates control of remote devices that are attempting to open a TCP connection to the module.

Part III
Using the Modicon Quantum with Unity Ethernet Services

Purpose

This part describes how to use the Ethernet services available on Modicon Quantum with Unity Ethernet modules.

What Is in This Part?

This part contains the following chapters:

Chapter 4
Transferring Data Using Communication Blocks

Introduction

This chapter describes how to transfer data to and from nodes on a TCP/IP network using communication blocks. You transfer the data using either a special MBP_MSTR instruction or an IEC Logic function. Operational statistics and error codes for reading and writing the controller information are included.

What Is in This Chapter?

This chapter contains the following sections:

Section 4.1
IEC Data Transfer Functions

Overview

This section describes several IEC function blocks that manage data transfer to and from nodes on a TCP/IP network.

What Is in This Section?

This section contains the following topics:

CREAD_REG

Function Description

The CREAD_REG block reads register data continuously from an addressed node via TCP/IP-Ethernet.

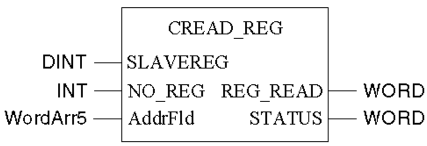
EN and ENO can be projected as additional parameters.

NOTE: About this function block:

- When programming this function, you must be familiar with the routing procedures used by your network.
- For technical reasons, this function block does not allow the use of ST and IL programming languages.

Representation

Block representation:



Parameter Description

Description of parameters:

Parameter	Data Type	Meaning
SLAVEREG	DINT	offset address of the first %MW word (4x register) in the slave to be read from
NO_REG	INT	number of registers to be read from slave
AddrFld	WordArr5	data structure describing the TCI/IP address
REG_READ	WORD	first %MW word (4x register) for read values
STATUS	WORD	error code

Elementary Description for WordArr5 with TCP/IP Ethernet

Elementary description for WordArr5 with TCP/IP Ethernet:

Element	Data Type	Meaning
WordArr5[1]	WORD	Low value byte: MBP on Ethernet Transporter (MET) mapping index High value byte: Slot of the NOE module
WordArr5[2]	WORD	Byte 4 (MSB) of the 32-bit destination IP address

Element	Data Type	Meaning
WordArr5[3]	WORD	Byte 3 of the 32-bit destination IP address
WordArr5[4]	WORD	Byte 2 of the 32-bit destination IP address
WordArr5[5]	WORD	Byte 1 (LSB) of the 32-bit destination IP address

Function Mode of the CREAD_REG Block

Although a large number of CREAD_REG function blocks can be programmed, only sixteen read operations may be active at the same time. In such a case it is insignificant whether they are the result of this function block or others (for example, MBP_MSTR, READ_REG). All function blocks use one data transaction path and require multiple cycles to complete a job.

NOTE: A TCP/IP communication between a Quantum PLC (NOE 771 ••) and a Momentum PLC (all TCP/IP CPUs and all TCP/IP I/O modules) is only possible, when only one read or write job is carried out in every cycle. If several jobs are sent per PLC cycle, the communication stops without generating an error message in the status register of the function block.

NOTE: A TCP/IP communication between a Quantum PLC (NOE 211 00) and a Momentum PLC (all TCP/IP CPUs and all TCP/IP I/O modules) is only possible, when only one read or write job is carried out in every cycle. If several jobs are sent per PLC cycle, the communication stops without generating an error message in the status register of the function block.

The entire routing information is contained in data structure WordArr5 of input AddrFld. The type of function block connected to this input and thus the contents of the data structure depends on the network used.

Please use:
- TCP/IP Ethernet: the function block TCP_IP_ADDR

NOTE: For experts: The WordArr5 data structure can be used with constants as well.

NOTE: This function block puts a heavy load on the network; therefore the network load must be carefully monitored. If the network load is too high, the program logic should be reorganized in order to work with the READ_REG function block, a variation of this function block that does not operate in a continuous mode, but under command control.

SLAVEREG

SLAVEREG is the start of the area in the addressed slave from which the source data is read. The source area always resides within the %MW word (4x register) area. SLAVEREG expects the source reference as offset within that area. (In 4x registers, the leading "4" must be omitted. For example, "59" (contents of the variables or value of the literal) = 40059).

The parameter can be specified as direct address, located variable, unlocated variable, or literal.

NO_REG

NO_REG is the number of registers to be read from the addressed slave (1 ... 100). The parameter can be entered as a direct address, located variable, unlocated variable, or literal.

REG_READ

The REG_READ word parameter addresses the first register in a series of NO_REG registers, listed one after the other, which are used as a destination data area. The parameter must be entered as a direct address or located variable.

STATUS

Error code, see Runtime errors.

The STATUS parameter can be specified as direct address, located variable, or unlocated variable.

CWRITE_REG

Function Description

The CWRITE_REG block writes data to a register area continuously, transferring data from the PLC via TCP/IP Ethernet to an addressed slave.

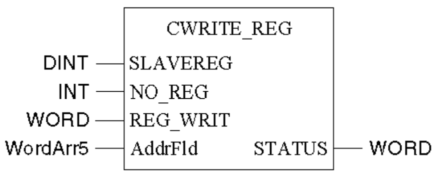
EN and ENO can be configured as additional parameters.

NOTE: About this function block:

- When programming this function, you must be familiar with the routing procedures used by your network.
- For technical reasons, this function block does not allow the use of ST and IL programming languages.

Symbol

Block representation:



Parameter Description

Description of parameters:

Parameter	Data Type	Meaning
SLAVEREG	DINT	offset address of the first %MW word (4x register) in the slave to be written to
NO_REG	INT	number of registers to be written to slave
REG_WRIT	WORD	first %MW word (4x register) of the source data area
AddrFld	WordArr5	data structure for transferring the TCI/IP address
STATUS	WORD	MBP_MSTR error code

Elementary Description for WordArr5 with TCP/IP Ethernet

Elementary description for WordArr5 with TCP/IP Ethernet:

Element	Data Type	Meaning
WordArr5[1]	WORD	low value byte: MBP on Ethernet Transporter (MET) mapping index high value byte: slots of the NOE module
WordArr5[2]	WORD	byte 4 (MSB) of the 32-bit destination IP address
WordArr5[3]	WORD	byte 3 of the 32-bit destination IP address
WordArr5[4]	WORD	byte 2 of the 32-bit destination IP address
WordArr5[5]	WORD	byte 1 (LSB) of the 32-bit destination IP address

CWRITE_REG Block Function Mode

Although a large number of CWRITE_REG function blocks can be programmed, only sixteen write operations may be active at the same time. It makes no difference whether these operations are performed using this function block or others (for example, MBP_MSTR, WRITE_REG). All function blocks use one data transaction path and require multiple cycles to complete a job.

If several CWRITE_REG function blocks are used within an application, they must at least differ in the values of their NO_REG or REG_WRITE parameters.

NOTE: A TCP/IP communication between a Quantum PLC (NOE 771*xx*) and a Momentum PLC (all TCP/IP CPUs and all TCP/IP I/O modules) is only possible, when only one read or write job is carried out in every cycle. If several jobs are sent per PLC cycle, the communication stops without generating an error message in the status register of the function block.

The entire routing information is contained in data structure WordArr5 of input AddrFld. The type of function block connected to this input and thus the contents of the data structure depend on the network used.

Please use:
- TCP/IP Ethernet: the function block TCP_IP_ADDR

NOTE: For experts: The WordArr5 data structure can also be used with constants.

NOTE: This function block puts a heavy load on the network. The network load must therefore be carefully monitored. If the network load is too high, the program logic should be reorganized to work with the WRITE_REG function block, which is a variant of this function block that does not operate in continuous mode but is command driven.

SLAVEREG

SLAVEREG is the start of the area in the addressed slave to which the source data are written. The destination area always resides within the %MW word (4x register) area. SLAVEREG expects the destination address as offset within that area. In 4x registers, the leading 4 must be omitted. For example, 59 (contents of the variables or value of the literal) = 40059.

The parameter can be specified as direct address, located variable, unlocated variable, or literal.

NO_REG

NO_REG is the number of registers to be written to slave processor (1 ... 100). The parameter can be specified as direct address, located variable, unlocated variable, or literal.

STATUS

Error code, see Runtime errors.

The STATUS parameter can be specified as direct address, located variable or unlocated variable.

REG_WRIT

The REG_WRIT word parameter addresses the first register in a series of NO_REG Successive registers used as source data area.

The parameter must be entered as a direct address or located variable.

READ_REG

Function Description

Upon request, the READ_REG block reads a register area once (rising edge of the REQ input). It reads data from an addressed slave via TCP/IP Ethernet.

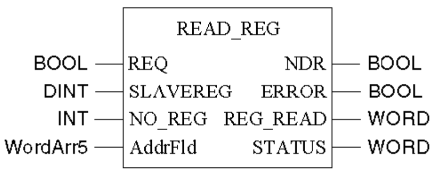
EN and ENO can be projected as additional parameters.

NOTE: About this function block:

- When programming this function, you must be familiar with the routing procedures used by your network.
- For technical reasons, this function block does not allow the use of ST and IL programming languages.

Symbol

Block representation:



Parameter Description

Description of block parameters:

Parameter	Data Type	Meaning
REQ	BOOL	start read operation once
SLAVEREG	DINT	offset address of the first %MW word (4x register) in the slave to be read from
NO_REG	INT	number of registers to be read from slave
AddrFld	WordArr5	data structure describing the TCP/IP address
NDR	BOOL	set to 1 for one cycle after reading new data
ERROR	BOOL	set to 1 for one scan in case of error
STATUS	WORD	error code
REG_READ	WORD	first %MW word (4x register) for read values

Elementary Description for WordArr5 with TCP/IP Ethernet

Elementary description for WordArr5 with TCP/IP Ethernet:

Element	Data Type	Meaning
WordArr5[1]	WORD	low value byte: MBP on Ethernet Transporter (MET) mapping index high value byte: Slot of the NOE module
WordArr5[2]	WORD	byte 4 (MSB) of the 32-bit destination IP address
WordArr5[3]	WORD	byte 3 of the 32-bit destination IP address
WordArr5[4]	WORD	byte 2 of the 32-bit destination IP address
WordArr5[5]	WORD	byte 1 (LSB) of the 32-bit destination IP address

Function Mode of READ_REG Blocks

Although a large number of READ_REG function blocks can be programmed, only 16 read operations may be active at the same time. In such a case, it is insignificant whether they are the result of this function block or of other read operations (for example, MBP_MSTR, CREAD_REG). All function blocks use 1 data transaction path and require multiple cycles to complete a job.

NOTE: A TCP/IP communication between a Quantum PLC (NOE 771*xx*) and a Momentum PLC (all TCP/IP CPUs and all TCP/IP I/O modules) is possible only when 1 read or write job is carried out in every cycle. If several jobs are sent per PLC cycle, the communication stops without generating an error message in the status register of the function block.

The entire routing information is contained in data structure WordArr5 of input AddrFld. The type of function block connected to this input and thus the contents of the data structure depends on the network used.

Please use:

- TCP/IP Ethernet: the function block TCP_IP_ADDR

NOTE: For experts: The WordArr5 data structure can be used with constants as well.

REQ

A rising edge triggers the read transaction.

The REQ parameter can be specified as direct address, located variable, unlocated variable, or Literal.

SLAVEREG

SLAVEREG is the start of the area in the addressed slave from which the source data is read. The source area always resides within the %MW word (4x register) area. SLAVEREG expects the source reference as offset within that area. In 4x registers, the leading 4 must be omitted. For example, 59 (contents of the variables or value of the literal) = 40059.

The parameter can be specified as direct address, located variable, unlocated variable, or literal.

NO_REG

Number of registers to be read from the addressed slave (1 ... 100).

The NO_REG parameter can be specified as direct address, located variable, unlocated variable, or literal.

NDR

Transition to ON state for one program cycle signifies receipt of new data ready to be processed.

The NDR parameter can be specified as direct address, located variable, or unlocated variable.

ERROR

Transition to ON state for one program cycle signifies detection of a new error.

The ERROR parameter can be specified as direct address, located variable, or unlocated variable.

REG_READ

This word parameter addresses the first register in a series of NO_REG registers lying in series used as destination data area.

The REG_READ parameter must be entered as a direct address or located variable.

STATUS

Error code, see Runtime errors.

The STATUS parameter can be specified as direct address, located variable or unlocated variable.

WRITE_REG

Function Description

Upon request, the WRITE_REG block writes a register area once (rising edge of the REQ input). It transfers data from the PLC via TCP/IP Ethernet to an addressed slave.

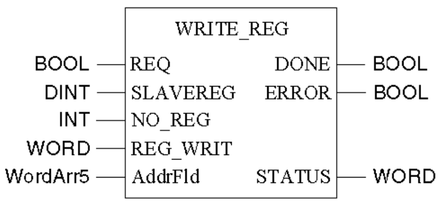
EN and ENO can be configured as additional parameters.

NOTE: About this function block:

- When programming this function, you must be familiar with the routing procedures used by your network.
- For technical reasons, this function block does not allow the use of ST and IL programming languages.

Symbol

Block representation:



Parameter Description

Description of parameters:

Parameter	Data Type	Meaning
REQ	BOOL	start write operation once
SLAVEREG	DINT	offset address of the first %MW word (4x register) in the slave to be written to
NO_REG	INT	number of registers to be written from slave
AddrFld	WordArr5	data structure transferring the TCP/IP address
REG_WRIT	WORD	first %MW word (4x register) of the source data area
DONE	BOOL	set to "1" for one scan after writing data
ERROR	BOOL	set to "1" for one scan in case of error
STATUS	WORD	error code

Elementary Description for WordArr5 with TCP/IP Ethernet

Elementary description for WordArr5 with TCP/IP Ethernet:

Element	Data Type	Meaning
WordArr5[1]	WORD	high value byte: Slot of the NOE module low value byte: MBP on Ethernet Transporter (MET) mapping index
WordArr5[2]	WORD	byte 4 (MSB) of the 32-bit destination IP address
WordArr5[3]	WORD	byte 3 of the 32-bit destination IP address
WordArr5[4]	WORD	byte 2 of the 32-bit destination IP address
WordArr5[5]	WORD	byte 1 (LSB) of the 32-bit destination IP address

Function Mode of the WRITE_REG Module

Although a large number of WRITE_REG function blocks can be programmed, only sixteen write operations may be active at the same time. In such a case, it is insignificant whether they are the result of this function block or of other write operations (for example, MBP_MSTR, CWRITE_REG). All function blocks use one data transaction path and require multiple cycles to complete a job.

If several WRITE_REG function blocks are used within an application, they must at least differ in the values of their NO_REG or REG_WRITE parameters.

NOTE: A TCP/IP communication between a Quantum PLC (NOE 771*xx*) and a Momentum PLC (all TCP/IP CPUs and all TCP/IP I/O modules) is possible only when one read or write job is carried out in every cycle. If several jobs are sent per PLC cycle, the communication stops without generating an error message in the status register of the function block.

The status signals DONE and ERROR report the function block state to the user program.

The entire routing information is contained in data structure WordArr5 of input AddrFld. The type of function block connected to this input and thus the contents of the data structure depend on the network used.

Please use:
- TCP/IP Ethernet: the function block TCP_IP_ADDR

NOTE: For experts: The WordArr5 data structure can also be used with constants.

REQ

A rising edge triggers the read transaction.

The REQ parameter can be specified as direct address, located variable or unlocated variable.

SLAVEREG

SLAVEREG is the start of the area in the addressed slave from which the source data is read. The source area always resides within the %MW word (4x register) area. SLAVEREG expects the source reference as offset within that area. In 4x registers, the leading 4 must be omitted. For example, 59 (contents of the variables or value of the literal) = 40059.

The parameter can be specified as direct address, located variable, unlocated variable, or literal.

NO_REG

Number of registers to be read from the addressed slave (1 ... 100).

The parameter can be specified as direct address, located variable, unlocated variable, or literal.

REG_WRIT

The REG_WRIT word parameter addresses the first register in a series of NO_REG registers used as source data area.

The parameter must be entered as a direct address or located variable.

DONE

Transition to ON state for one program scan signifies data have been transferred.

The DONE parameter can be specified as direct address, located variable or unlocated variable.

ERROR

Transition to ON state for one program scan signifies detection of a new error.

The parameter can be specified as direct address, located variable or unlocated variable.

STATUS

Error code, see Runtime errors.

The parameter can be specified as direct address, located variable, or unlocated variable.

TCP_IP_ADDR

Function Description

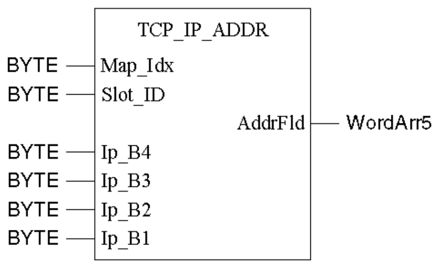
The TCP_IP_ADDR block enables the input of TCP/IP addresses for the READ_REG *(see page 125)*, CREAD_REG *(see page 119)*, WRITE_REG *(see page 128)*, and CWRITE_REG *(see page 122)* blocks. The address is transferred in the form of a data structure.

EN and ENO can be projected as additional parameters.

NOTE: When programming the TCP_IP_ADDR function, you must be familiar with your network's routing procedures.

Symbol

Block representation:



Parameter Description

Description of parameters:

Parameter	Data Type	Meaning
Map_Idx	BYTE	Map-Index MBP on Ethernet Transporter (MET) mapping index
Slot_ID	BYTE	slot ID slot of the NOE module
Ip_B4	BYTE	byte 4 (MSB) of the 32-bit destination IP address
Ip_B3	BYTE	byte 3 of the 32-bit destination IP address
Ip_B2	BYTE	byte 2 of the 32-bit destination IP address
Ip_B1	BYTE	byte 1 (LSB) of the 32-bit destination IP address
AddrFld	WordArr5	data structure used to transfer the TCP/IP address

Elementary Description for WordArr5

Elementary description for WordArr5:

Element	Data Type	Meaning
WordArr5[1]	WORD	high value byte: Slot of the NOE module low value byte: MBP on Ethernet Transporter (MET) mapping index
WordArr5[2]	WORD	byte 4 (MSB) of the 32-bit destination IP address
WordArr5[3]	WORD	byte 3 of the 32-bit destination IP address
WordArr5[4]	WORD	byte 2 of the 32-bit destination IP address
WordArr5[5]	WORD	byte 1 (LSB) of the 32-bit destination IP address

Map_Idx

The MBP on Ethernet Transporter (MET) mapping index is given at the Map_Idx input. That is, if MET is 6, the value appears as follows:

0	0	0	0	0	1	1	0

Slot_ID

If an NOE in the rack of a Quantum controller is addressed as destination node, the value at the Slot_ID input represents the physical NOE slot. That is, if the NOE is plugged in at Slot 7 of the rack, the value appears as follows:

0	0	0	0	0	1	1	1

NOTE: When using an integrated Ethernet CPU module such as the 140 CPU 651 ***x***0, the slot ID must be 254 (FE hex) regardless of the CPU slot.

AddrFld

If an NOE in the rack of a Quantum controller is addressed as a destination node, the value in the High value byte represents the physical slot of the NOE and the Low value byte represents the MBP on Ethernet Transporter (MET) mapping index. That is, if the NOE is inserted in slot 7 of the rack and the MET mapping index is 6, the first element of the data structure looks as follows:

High value byte								Low value byte							
0	0	0	0	0	1	1	1	0	0	0	0	0	1	1	0

High value byte Slots 1 ... 16

Low value byte MBP on Ethernet Transporter (MET) mapping index

Section 4.2
MBP_MSTR

Overview

This section describes the 14 different communication function provided in the MBP_MSTR function block.

What Is in This Section?

This section contains the following topics:

Block Description

Function Description

You can select one of 14 available network communication operations *(see page 137)* using the MBP_MSTR function block.

The MBP_MSTR block is supported on various hardware platforms.

EN and ENO can be configured as additional parameters.

NOTE: You must be familiar with the routing procedures of your network when programming an MBP_MSTR function block. Modbus Plus routing path structures are described in detail in the communication architectures manual and the *Modbus Plus Network Planning and Installation Guide (Reference 31003525)*. If TCP/IP or SY/MAX Ethernet routing is implemented, standard Ethernet IP router products must be used.

CREAD_REG, CWRITE_REG, READ_REG, WRITE_REG and MBP_MSTR function blocks use 1 data transaction path and require multiple cycles to complete an operation. Number of transaction path available is dependent on the communication port used:
- Modbus Plus embedded port or NOM modules support up to 4 blocks at the same time
- TCP/IP Ethernet embedded port support up to 4 blocks at the same time
- TCP/IP Ethernet NOE modules support up to 16 blocks at the same time

More communication function blocks may be programmed on the same communication port. However communication block exceeding the maximum number on that port is not serviced until one of the transaction paths is freed up. When the transaction path resources become free the next block on the same port will become active and begin using freed path.

NOTE: In FBD and LD sections, this function block can be used on the program level and with derived function blocks (DFBs). When using DFBs, the parameters CONTROL and DATABUF must be directly connected to the I/O pins of the DFB.

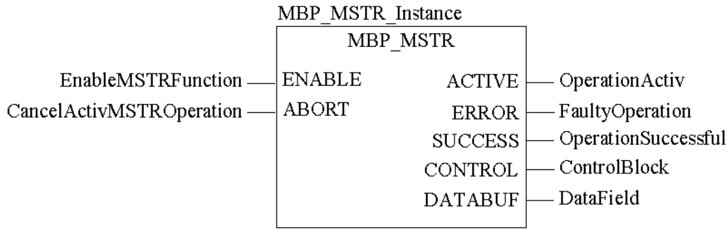
NOTE: A TCP/IP communication between a Quantum PLC and a Momentum PLC is only possible when one read or write job is carried out in every cycle. If several jobs are sent per PLC cycle, the communication stops without generating an error message in the status register of the function block.

NOTE: In Hot Standby, to prevent the former standby CPU from executing communication functions (now in RUN offline state), you must add a condition on the status bits to disable the function if the CPU is offline.

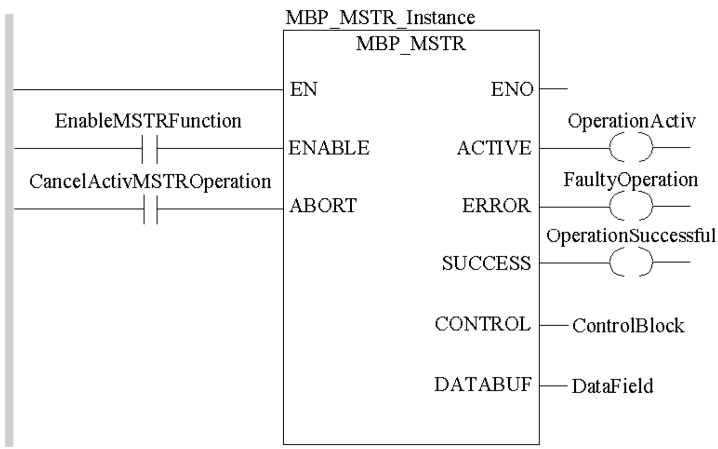
Example:
- You can send MBP_MSTR.Enable:=(HSBY_NOEPLCMSTR_ON) AND (%SW61.1) AND NOT (%SW61.0)
  or
- You can create a boolean variable, primary_state:=(%SW61.1) AND NOT (%SW61.0), and insert it for executing the section

NOTE: Several copies of this function block can be used in the program. However, multiple instancing of these copies is not possible.

Representation in FBD



Representation in LD



Input Parameters

Parameter	Data Type	Description
ENABLE	BOOL	When ON, the operation specified in the first element of the CONTROL register is enabled.
ABORT	BOOL	When ON, the currently active operation *(see page 137)* is aborted.

Output Parameters

Parameter	Data Type	Description
ACTIVE	BOOL	ON when the operation is active.
ERROR	BOOL	ON when the operation is aborted without success.
SUCCESS	BOOL	ON when the operation concludes successfully.
CONTROL	WORD	This field contains the control block. The first element CONTROL[1] contains the number of the operation code of the operation to be performed *(see page 137)*. The content of the sequence register is determined by the operation. The data field must be declared as a located variable. The structure of the control block differs according to the network used *(see page 138)*.
DATABUF	WORD	For operations providing data, e.g. a `write` operation, the data field is the data source. For operations receiving data, e.g. the read operation, the data field is the data destination. With Ethernet CTE `read` and `write` operations, the data field holds the contents of the Ethernet configuration extension table. DATABUF must be defined as an array of at least 10 elements in this case. The data field must be declared as a located variable.

Runtime Error

In the event of an error occurring during an `MBP_MSTR` operation, a hexadecimal error code is displayed in the `CONTROL[2]` register of the control block for one cycle.

NOTE: Function error codes are network-specific

NOTE: For a list of all block error codes and values, refer to the tables of error codes for the communication library.

Operational Function Codes

Valid MBP_MSTR Function Codes

Using the MBP_MSTR block, network communication operations can be triggered via the network. As shown in the following table, each operation has a function code assigned to it. The availability of specific operations depends on both the type of network and the type of module you are using.

Function Code	Operation	Modbus Plus	TCP/IP Ethernet	SY/MAX Ethernet	CIP Ethernet
1	Write data	X	X	X	-
2	Read data	X	X	X	-
3	Get local statistics	X	X	-	-
4	Clear local statistics	X	X	-	-
5	Write global data, peer cop	X	-	-	-
6	Read global data, peer cop	X	-	-	-
7	Get remote statistics	X	X	-	-
8	Clear remote statistics *(see page 156)*	X	X	-	-
9	Peer cop health status	X	-	-	-
10	Reset optional module	-	X	X	-
11	Read CTE (Config extension)	-	X	X	-
12	Write CTE (Config extension)	-	X	X	-
13	Send E-mail *(see page 168)*	-	X	-	-
14	CIP Explicit message	-	-	-	X
15	Send Modbus Request *(see page 170)*	-	X	-	-
16	Close Connection Request *(see page 175)*	-	X	-	-
17	Change Modbus Plus Address *(see page 176)*	X	-	-	-
23	Read / write data *(see page 178)*	-	X	-	-
65520 (dec) FFF0 (hex)	Enable / disable HTTP or FTP/TFTP	-	X	-	-

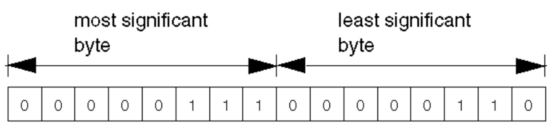
where:
- X indicates *Yes*
- \- indicates *No*

Network Control Block Structures

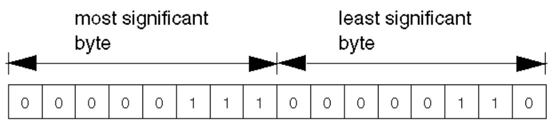
Summary

The structure of the `MBP_MSTR` control block varies according to the type of network you are using. Structures for Modbus Plus, TCP/IP Ethernet, and SyMax Ethernet are described below.

Control Block for Modbus Plus

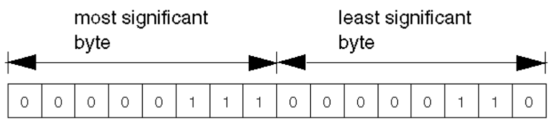
Register	Contents
CONTROL[1]	Indicates an operation that is valid for Modbus Plus
CONTROL[2]	Indicates the error status
CONTROL[3]	Indicates the length, i.e., the number of data units transferred (max. 100)
CONTROL[4]	Indicates `MSTR` operation-dependent information
CONTROL[5]	Routing register 1: used to specify a destination node during network transfer (routing path addresses one of five) Most significant byte: source node address, i.e., the slot for the Modbus Plus Network Options Module (NOM) When using the Modbus Plus Port on the CPU, this byte must be set to 0 (regardless of the CPU slot). Least significant byte: destination node address, i.e., a value that represents a direct or a bridge address. If there is no bridge, this value contains the destination node address. If there is a bridge, this value contains the address of the bridge. If the NOM is inserted in slot 7 on the module rack, the most significant byte of routing register 1 looks as follows (value 0x0706):  **Most significant byte** Slots 1 ... 16 **Least significant byte** Destination address (binary value between 1 and 64 (normal) or 65 to 255 (extended))
CONTROL[6]	Routing register 2, the destination node address (further bridge or Modbus Plus modules). If addressing in the previous routing register has finished, the value is set to 0.
CONTROL[7]	Routing register 3, similar to routing register 2
CONTROL[8]	Routing register 4, similar to routing register 2 (see Routing Register 2)
CONTROL[9]	Routing register 5, similar to routing register 2 (see Routing Register 2)

Control Block for TCP/IP Ethernet

Register	Contents
CONTROL[1]	Indicates an operation that is valid for TCP/IP
CONTROL[2]	Indicates the error status
CONTROL[3]	Indicates the length, i.e., the number of data units transferred (max. 100)
CONTROL[4]	Indicates MSTR operation-dependent information
CONTROL[5]	Routing register: used to specify a destination node during network transfer Most significant byte: source node address, i.e., the NOE slot for the NOE module When using an integrated Ethernet on the CPU, this byte must be set to 254 (FE hex) regardless of the CPU slot. Least significant byte: destination node address, i.e, a value that represents a direct or bridge address. If there is no bridge the value in the least significant byte is set to 0. If there is a bridge, this value contains the MBP for the Ethernet mapping index (MET). If the NOE is inserted in slot 7 on the module rack and the Ethernet mapping index (MET) is 6, the routing register looks as follows (value 0x0706):  **Most significant byte** Slots 1 ... 16 **Least significant byte** MBP on Ethernet Transporter (MET) mapping index
CONTROL[6]	Byte 4, MSB of the 32-bit destination IP address
CONTROL[7]	Byte 3 of the 32-bit destination IP address
CONTROL[8]	Byte 2 of the 32-bit destination IP address
CONTROL[9]	Byte 1, LSB of the 32-bit destination IP address
CONTROL[10]	Indicates MSTR operation-dependent information
CONTROL[11]	Indicates MSTR operation-dependent information

NOTE: CONTROL[10] and CONTROL[11] are used when configuring the MBP_MSTR block for a Read/Write Data operation (function code 23).

Control Block for SY/MAX Ethernet

Register	Contents
CONTROL[1]	Indicates an operation that is valid for SY/MAX
CONTROL[2]	Indicates the error status
CONTROL[3]	Indicates the length, i.e., the number of registers transferred (max. 100)
CONTROL[4]	Indicates `MSTR` operation-dependent information
CONTROL[5]	Routing register: used to specify a destination node during network transfer Most significant byte: source node address, ie.e, the slot for the NOE module Least significant byte: destination node address, i.e, a value that represents a direct or bridge address. If there is no bridge the value in the least significant byte is set to 0. If there is a bridge, this value contains the MBP for the Ethernet mapping index (MET). If NOM is inserted in slot 7 on the module rack and the Ethernet mapping index (MET) is 6, the routing register looks as follows (value 0x0706):  **Most significant byte** Slots 1 ... 16 **Least significant byte** MBP on Ethernet Transporter (MET) mapping index
CONTROL[6]	Destination drop number (or set to FF hex)
CONTROL[7]	Terminator (set to FF hex)

Modbus Plus, SY/MAX, and Ethernet TCP/IP Error Codes

Form of the Function Error Code

Function error codes for Modbus Plus and SY/MAX Ethernet transactions appear as **Mmss**, where:

- **M** is the high code
- **m** is the low code
- **ss** is a subcode

Modbus Plus and SY/MAX Ethernet Network Errors

Hexadecimal error codes for Modbus Plus and SY/MAX Ethernet:

Hex. Error Code	Description
1001	Abort by user
2001	An operation type that is not supported has been specified in the control block
2002	One or more control block parameters were modified while the `MSTR` element was active (this only applies to operations which require several cycles for completion). Control block parameters my only be modified in inactive `MSTR` components.
2003	Invalid value in the length field of the control block
2004	Invalid value in the offset field of the control block
2005	Invalid value in the length and offset fields of the control block
2006	Unauthorized data field on slave
2007	Unauthorized network field on slave
2008	Unauthorized network routing path on slave
2009	Routing path equivalent to their own address
200A	Attempt to get more global data words than available
200C	Bad pattern for change address request
200D	Bad address for change address request
200E	The control block is not assigned, or parts of the control block are located outside of the %MW (4x) range.
30ss	Exceptional response by Modbus slave *(see page 144)*
4001	Inconsistent response by Modbus slave
5001	Inconsistent response by the network
6007	Invalid slot ID
6mss	Routing path error *(see page 144)* The subfield m shows where the error occurred (a 0 value means local node, 2 means 2nd device in route, etc).

TCP/IP Ethernet Network Errors

Hexadecimal error codes for TCP/IP Ethernet:

Hex. Error Code	Meaning
5004	Interrupted system call
5005	I/O error
5006	No such address
5009	Socket descriptor is invalid
500C	Not enough memory
500D	Permission denied
5011	Entry exists
5016	Argument is invalid
5017	Internal table has run out of space
5020	Connection is broken
5028	Destination address required
5029	Protocol wrong type for socket
502A	Protocol not available
502B	Protocol not supported
502C	Socket type not supported
502D	Operation not supported on a socket
502E	Protocol family not supported
502F	Address family not supported
5030	Address already in use
5031	Cannot assign requested address
5032	Socket operation on a non-socket
5033	Network is unreachable
5034	Network dropped connection on reset
5035	Network caused connection abort
5036	Connection reset by peer
5037	No buffer space available
5038	Socket already connected
5039	Socket not connected
503A	Cannot send after socket shutdown
503B	Too many references, cannot splice
503C	Connection timed out (see note below)
503D	Connection refused

Hex. Error Code	Meaning
503E	Network down
503F	Text file busy
5040	Too many levels of links
5041	No route to host
5042	Block device required
5043	Host is down
5044	Operation now in progress
5045	Operation already in progress
5046	Operation would block
5047	Function not implemented
5048	Hardware length is invalid
5049	Route specified cannot be found
504A	Collision in select call: these conditions have already been selected by another task
504B	Task ID is invalid
5050	No network resource
5051	Length error
5052	Addressing error
5053	Application error
5054	Client in bad state for request
5055	No remote resource -- may indicate no path to remote device (see note below)
5056	Non-operational TCP connection
5057	Incoherent configuration

NOTE:

- Error code 5055 can occur before a 503C error.
- No remote device takes precedence over a timeout.

ss Hexadecimal Value in 30ss Error Code

ss hexadecimal value in 30ss error code:

ss Hex. Value	Description
01	Slave does not support requested operation
02	Non-existing slave registers were requested
03	An unauthorized data value was requested
05	Slave has accepted a lengthy program command
06	Function cannot currently be carried out: lengthy command running
07	Slave has rejected lengthy program command

ss Hexadecimal Value in 6mss Error Code

NOTE: Subfield m in error code 6mss is an `Index` in the routing information that shows where an error has been detected (a 0 value indicates the local node, 2 means the second device in the route, etc.).

The ss subfield in error code 6mss is as follows:

ss Hex. Value	Description
01	No response reception
02	Access to program denied
03	Node out of service and unable to communicate
04	Unusual response received
05	Router-node data path busy
06	Slave out of order
07	Wrong destination address
08	Unauthorized node type in routing path
10	Slave has rejected the command
20	Slave has lost an activated transaction
40	Unexpected master output path received
80	Unexpected response received
F001	Wrong destination node was specified for the `MSTR` operation

CTE Error Codes for SY/MAX and TCP/IP Ethernet

CTE Error Codes for SY/MAX and TCP/IP Ethernet

The following error codes are displayed in the `CONTROL[1]` register of the control block, if there is a problem with the Ethernet configuration extension table (CTE) in your program configuration.

CTE error codes for SY/MAX and TCP/IP Ethernet:

Hex. Error Code	Description
7001	There is no Ethernet configuration extension.
7002	The CTE is not available for access.
7003	The offset is not valid.
7004	Offset + length are not valid.
7005	Bad data field in the CTE.

SY/MAX-Specific Error Codes

SY/MAX-Specific Error Codes

When utilizing SY/MAX Ethernet, three additional types of errors may appear in the `CONTROL[1]` register of the control block ().

The error codes have the following meaning:

- 71xx Error: Errors found by the SY/MAX remote device
- 72xx Error: Errors found by the server
- 73xx Error: Errors found by the Quantum translator

SY/MAX-Specific Hexadecimal Error Codes

SY/MAX-specific hexadecimal error codes:

Hex. Error Code	Description
7101	Invalid opcode found by the SY/MAX remote device
7103	Invalid address found by the SY/MAX remote device
7109	Attempt to write to a write protected register found by the SY/MAX remote device
F710	Receiver overflow found by the SY/MAX remote device
7110	Invalid length found by the SY/MAX remote device
7111	Remote device not active, no connection (occurs when retry attempts and time-out have been used up), found by the SY/MAX remote device
7113	Invalid parameter in a read operation found by the SY/MAX remote device
711D	Invalid route found by the SY/MAX remote device
7149	Invalid parameter in a write operation found by the SY/MAX remote device
714B	Invalid drop number found by the SY/MAX remote device
7101	Invalid opcode found by the SY/MAX server
7203	Invalid address found by the SY/MAX server
7209	Attempt to write to a write protected register found by the SY/MAX server
F720	Receiver overflow found by the SY/MAX server
7210	Invalid length found by the SY/MAX server
7211	Remote device not active, no connection (occurs when retry attempts and time-out have been used up), found by the SY/MAX server
7213	Invalid parameter in a read operation found by the SY/MAX server
721D	Invalid route found by the SY/MAX server
7249	Invalid parameter in a write operation found by the SY/MAX server
724B	Invalid drop number found by the SY/MAX server

Hex. Error Code	Description
7301	Invalid opcode in an `MSTR` block request from the Quantum translator
7303	Read/Write QSE module status (200 route address out of range)
7309	Attempt to write to a write protected register when a status write is carried out (200 route)
731D	Invalid route found by the Quantum translator. Valid routes: ● dest_drop, 0xFF ● 200, dest_drop, 0xFF ● 100+drop, dest_drop, 0xFF ● All other routing values produce an error
734B	One of the following errors occurred: ● No CTE (configuration extension table) has been configured ● No CTE table entry has been made for the QSE model slot number ● No valid drop has been specified ● The QSE module has not been reset after the creation of the CTE. Note: After writing and configuring the CTE and downloading to the QSE module, the QSE module must be reset for the modifications to become effective. ● When using an `MSTR` instruction no valid slot or drop has been specified

Read Data

Description

A read operation transfers data from a specified slave source device to a master destination device on the network. It uses a master transaction path and may require several cycles to complete. To program an `MBP_MSTR` block to perform a write operation, use function code 2 *(see page 137)*.

NOTE: Do not attempt to program an `MBP_MSTR` to read to its own station address. This action causes the function block to generate an error in the CONTROL[2] register of the control block *(see page 138)*.

NOTE: You can perform a read operation on a nonexistent slave register. The slave detects the status and logs it. This can last for several cycles.

NOTE: For Quantum PLC communication with a Premium/Atrium PLC the addressing must be made with an offset of 1. In order to access an address object **n** of a Premium PLC, the communication function of the Quantum PLC must use the **n+1** address. The reason is that IEC addressing starts at 0 and Modbus addressing starts at 1.

Network Implementation

The read operation can be performed on Modbus Plus, TCP/IP Ethernet, and SY/MAX Ethernet networks.

Control Block Usage for Modbus Plus

Register	Meaning
CONTROL[1]	2 = Read data
CONTROL[2]	Indicates the error status.
CONTROL[3]	Number of registers to be read from the slave
CONTROL[4]	Determines the %MW starting register in the slave from which the data is read, e.g., 1 = %MW1, 49 = %MW49).
CONTROL[5] ... CONTROL[9]	Routing register 1 is used to specify the address (routing path address 1 of 5) of the node during a network transfer. The last byte in the routing path that is not 0 is the destination node.

Control Block Usage for TCP/IP Ethernet

Register	Meaning
CONTROL[1]	2 = read data
CONTROL[2]	Indicates the error status.
CONTROL[3]	Number of addresses to be read from the slave
CONTROL[4]	Determines the %MW starting register in the slave from which the data is read, e.g., 1 = %MW1, 49 = %MW49)
CONTROL[5]	Routing register: Most significant byte: network adapter module slot Least significant byte: MBP on Ethernet Transporter (MET) mapping index
CONTROL[6] ... CONTROL[9]	Each address contains 1 byte of the 32-bit IP address, where the MSB is in CONTROL[6] and the LSB is in CONTROL[9].

Control Block Usage for SY/MAX Ethernet

Register	Meaning
CONTROL[1]	2 = Read data
CONTROL[2]	Indicates the error status.
CONTROL[3]	Number of addresses to be read from the slave
CONTROL[4]	Determines the %MW starting register in the slave to which the data is written, e.g., 1 = %MW1, 49 = %MW49).
CONTROL[5]	Routing register Most significant byte: network adapter module slot Least significant byte: destination drop number
CONTROL[6] ... CONTROL[9]	Terminator: FF hex

Write Data

Description

A write operation transfers data from a master source device to a specified slave destination device on the network. It uses a master transaction path, and may require several cycles to complete. To program an `MBP_MSTR` block to perform a write operation, use function code 1 *(see page 137)*.

NOTE: Do not attempt to program an `MBP_MSTR` to write to its own drop address. This action causes the function block to generate an error in the CONTROL[2] register of the control block *(see page 138)*.

NOTE: You can perform a write operation to a nonexistent slave register. The slave detects the status and logs it. This can last for several cycles.

NOTE: For Quantum PLC communication with a Premium/Atrium PLC the addressing must be made with an offset of 1. In order to access an address object **n** of a Premium PLC, the communication function of the Quantum PLC must use the **n+1** address. The reason is that IEC addressing starts at 0 and Modbus addressing starts at 1.

Network Implementation

The write operation can be performed on Modbus Plus, TCP/IP Ethernet, and SY/MAX Ethernet networks.

Control Block Usage for Modbus Plus

Register	Meaning
CONTROL[1]	1 = write data
CONTROL[2]	Indicates the error status
CONTROL[3]	Number of addresses sent to the slave
CONTROL[4]	Determines the %MW starting register in the slave to which the data is written, e.g. 1 = %MW1, 49 = %MW49
CONTROL[5] ... CONTROL[9]	Routing register 1 is used to specify the address (routing path address 1 of 5) of the node during a network transfer. The last byte in the routing path that is not 0 is the destination node.

Control Block Usage for TCP/IP Ethernet

Register	Meaning
CONTROL[1]	1 = write data
CONTROL[2]	Indicates the error status
CONTROL[3]	Number of addresses sent to the slave
CONTROL[4]	Determines the %MW start address of the slave to which the data is written
CONTROL[5]	Routing register Most significant byte: network adapter module slot Least significant byte: MBP on Ethernet transporter (MET) mapping index
CONTROL[6] ... CONTROL[9]	Each address contains 1 byte of the 32-bit IP address.

Control Block Usage for SY/MAX Ethernet

Register	Meaning
CONTROL[1]	1 = Write data
CONTROL[2]	Indicates the error status
CONTROL[3]	Number of addresses sent to the slave
CONTROL[4]	Determines the %MW starting register in the slave to which the data is written, e.g., 1 = %MW1, 49 = %MW49)
CONTROL[5]	Routing register Most significant byte: network adapter module slot Least significant byte: destination drop number
CONTROL[6] ... CONTROL[9]	Terminator: FF hex

Get Local Statistics

Description

A `Get Local Statistics` operation reads the data from the local node in one cycle and does not require a master transaction path.To program an `MBP_MSTR` block to `Get Local Statistics`, use function code 3 *(see page 137)*.

Network Implementation

A `Get Local Statistics` operation can be performed on Modbus Plus and TCP/IP Ethernet networks *(see page 157)*.

Control Block Usage for Modbus Plus

Register	Meaning
CONTROL[1]	3 = get local statistics
CONTROL[2]	Indicates the error status
CONTROL[3]	Number of addresses to be read from local statistics (1 ... 54) **Note:** The size of databuf must be at least the size of this entry.
CONTROL[4]	First address from which the statistics table must be read (Reg1=0)
CONTROL[5]	Routing register 1 is used to specify the address (routing path address 1 of 5) of the node during a network transfer. The last byte in the routing path that is not 0 is the destination mode.

Control Block Usage for TCP/IP Ethernet

Register	Meaning
CONTROL[1]	3 = get local statistics
CONTROL[2]	Indicates the error status
CONTROL[3]	Number of addresses to be read from local statistics (1 ... 38) **Note:** The size of databuf must be the size of this entry.
CONTROL[4]	First address from which the statistics table must be read (Reg1=0)
CONTROL[5]	Routing register Most significant byte: Network adapter module slot
CONTROL[6] ... CONTROL[9]	Not used

Clear Local Statistics

Description

A clear local statistics operation clears the values from words 13 ... 22 in the statistics table of the local node. The operation is carried out in one cycle and does not require a master transaction path. To program an `MBP_MSTR` block to clear local statistics, use function code 4 *(see page 137)*.

Network Implementation

A clear local statistics operation can be performed on Modbus Plus and TCP/IP Ethernet networks *(see page 157)*.

Control Block Usage for Modbus Plus

Register	Meaning
CONTROL[1]	4 = clear local statistics
CONTROL[2]	Indicates the error status
CONTROL[3]	Reserved
CONTROL[4]	Reserved
CONTROL[5]	Routing register 1 is used to specify the address (routing path address 1 of 5) of the node during a network transfer. The last byte in the routing path that is not 0 is the destination mode.

Control Block Usage for TCP/IP Ethernet

Register	Meaning
CONTROL[1]	4 = clear local statistics
CONTROL[2]	Indicates the error status
CONTROL[3]	Reserved
CONTROL[4]	Reserved
CONTROL[5]	Routing register Most significant byte: network adapter module slot
CONTROL[6] ... CONTROL[9]	Reserved

Get Remote Statistics

Description

A get remote statistics operation can be used to read data from remote nodes on the network. With each query, the remote communications processor supplies a complete table of statistics even if the query does not refer to the entire table. It then copies only the words that you queried into identified $MW addresses.

An operation can take several cycles to complete; it does not require a master data transaction path. To program an `MBP_MSTR` block to get remote statistics, use function code 7 *(see page 137)*.

Network Implementation

A get remote statistics operation can be performed on Modbus Plus and TCP/IP Ethernet networks.

Control Block Usage for Modbus Plus

Register	Meaning
CONTROL[1]	7 = get remote statistics
CONTROL[2]	Indicates the error status
CONTROL[3]	Number of addresses to be read from the statistics data field (1 ... 54). **Note:** The size of `databuf` must be at least the size of this entry.
CONTROL[4]	First address from which the node statistics must be read. The number of available statistics registers cannot be exceeded.
CONTROL[5] ... CONTROL[9]	Routing address 1 ... 5 of the node. The last byte in the routing path that is not 0 is the destination node.

Control Block Usage for TCP/IP Ethernet

Register	Meaning
CONTROL[1]	7 = get remote statistics
CONTROL[2]	Indicates the error status
CONTROL[3]	Number of addresses to be read from the statistics data field (1 ... 38). **Note:** The size of `databuf` must be at least the size of this entry.
CONTROL[4]	First address from which the node statistics must be read. The number of available statistics registers cannot be exceeded.

Register	Meaning
CONTROL[5]	Routing register Most significant byte: network adapter module slot
CONTROL[6] ... CONTROL[9]	Each address contains 1 byte of the 32-bit IP address, where the value in CONTROL[6] is the MSB and the value in CONTROL[9] is the LSB.

Clear Remote Statistics

Description

A clear remote statistics operation clears remote-node values from words 13 ... 22 in the statistics table of the local node. It uses a master transaction path and may require several cycles to complete. To program an MBP_MSTR block to perform a clear remote statistics operation, use function code 8 *(see page 137)*.

Network Implementation

A clear remote statistics operation can be performed on Modbus Plus and TCP/IP Ethernet networks *(see page 157)*.

Control Block Usage for Modbus Plus

Register	Meaning
CONTROL[1]	8 = clear remote statistics
CONTROL[2]	Indicates the error status
CONTROL[3]	Reserved
CONTROL[4]	Reserved
CONTROL[5] ... CONTROL[9]	Routing register 1 is used to specify the address (routing path address 1 of 5) of the destination node during a network transfer. The last byte in the routing path that is not 0 is the destination mode.

Control Block Usage for TCP/IP Ethernet

Register	Meaning
CONTROL[1]	8 = clear remote statistics
CONTROL[2]	Indicates the error status
CONTROL[3]	Reserved
CONTROL[4]	Reserved
CONTROL[5]	Routing Register Most significant byte: network adapter module slot
CONTROL[6] ... CONTROL[9]	Each address contains one byte of the 32-bit IP address, where the MSB is in CONTROL[6] and the LSB is in CONTROL[9].

TCP/IP Ethernet Network Statistics

TCP/IP Ethernet Network Statistics

A TCP/IP Ethernet module replies to local and remote statistics commands from the `MBP_MSTR` block with the the contents of the `databuf` array (see the information in the table below):

<table>
<tr><th>Word</th><th colspan="3">Meaning</th></tr>
<tr><td>00 to 02</td><td>MAC address
e.g. MAC address 00 00 54 00 12 34 is displayed as follows:</td><td>Word
00
01
02</td><td>Contents
00 00
54 00
12 34</td></tr>
<tr><td>03</td><td colspan="3">Board status (refer to the following table)</td></tr>
<tr><td>04 and 05</td><td colspan="3">Number of receiver interrupts</td></tr>
<tr><td>06 and 07</td><td colspan="3">Number of transfer interrupts</td></tr>
<tr><td>08 and 09</td><td colspan="3">Transfer timeout error count</td></tr>
<tr><td>10 and 11</td><td colspan="3">Collision detection error count</td></tr>
<tr><td>12 and 13</td><td colspan="3">Omitted packets</td></tr>
<tr><td>14 and 15</td><td colspan="3">Memory error count</td></tr>
<tr><td>16 and 17</td><td colspan="3">Number of restarts performed by the driver</td></tr>
<tr><td>18 and 19</td><td colspan="3">Receive framing error count</td></tr>
<tr><td>20 and 21</td><td colspan="3">Overflow error count receiver</td></tr>
<tr><td>22 and 23</td><td colspan="3">Receive CRC error counter</td></tr>
<tr><td>24 and 25</td><td colspan="3">Receive buffer error counter</td></tr>
<tr><td>26 and 27</td><td colspan="3">Transfer buffer error counter</td></tr>
<tr><td>28 and 29</td><td colspan="3">Transfer bin underflow counter</td></tr>
<tr><td>30 and 31</td><td colspan="3">Late collision counter</td></tr>
<tr><td>32 and 33</td><td colspan="3">Lost carrier counter</td></tr>
<tr><td>34 and 35</td><td colspan="3">Number of retries</td></tr>
<tr><td>36 and 37</td><td>IP address
e.g. the IP address 198.202.137.113 (or C6 CA 89 71) is represented as follows:</td><td>Word
36
37</td><td>Contents
89 71
C6 CA</td></tr>
</table>

Board Status Word Bit Definition

NOTE: It is best to view the board status word in binary format.

The following table describes the bit definitions of the board status word:

- 140 NOE 771 *x*1, versions 2.0, 3.0, 3.1, 3.3 and 3.6 or higher
- 140 NOE 771 *x*0, versions 3.0, 3.3 and 3.4 or higher

Bit #	Definition
15	0 = Link LED off 1 = Link LED on
14	0 = Appl LED off 1 = Appl LED on
13	0 = twisted pair 1 = fiber
12	0 = 10 Mbit 1 = 100 Mbit
11 ... 8	(Reserved)
7 ... 4	Module type (see table, below)
3	(Reserved)
2	0 = half duplex 1 = full duplex
1	0 = not configured 1 = configured
0	0 = not running 1 = running

NOTE: Bits are numbered from right to left, starting with bit 0 (low bit). For example, **PLC running =** 0000 0000 0000 0001 and **LED connection =** 1000 0000 0000 0000.

The following table describes the word bit definitions for board status for the:

- 140 NOE 771 *x*1, version 3.5
- 140 NOE 771 *x*0, versions 1.02 and 2.0
- 140 CPU 651 *x*0

Bit #	Definition
15 ... 12	Module type (see table below)
11	(Reserved)
10	0 = half duplex 1 = full duplex
9	0 = not configured 1 = configured
8	0 = PLC not running 1 = PLC/NOE running
7	0 = Link LED off 1 = Link LED on
6	0 = Appl LED off 1 = Appl LED on

5	0 = twisted pair 1 = fiber
4	0 = 10 Mbit 1 = 100 Mbit
3 ... 0	(Reserved)

NOTE: Bits are counted from right to left, starting with bit 0 (low bit). For example, **PLC running =** 0x0100, **Application LED =** 0x0040, and **LED Connection =** 0x0080.

Board Status Word Bit Definition by Module Type

The following table describes the values of the module types:

Value of Bits 7...4 or 15...12 **Note: See the previous tables for the bit range that applies to your module's software version.**	**Module Type**
0	NOE 2x1
1	ENT
2	M1E
3	NOE 771 00
4	ETY
5	CIP
6	(reserved)
7	140 CPU 651 x0
8	(reserved)
9	(reserved)
10	NOE 771 10
11	NOE 771 01
12	NOE 771 11
13 ... 15	(reserved)

TCP/IP Ethernet Error Codes

TCP/IP Ethernet Error Codes

An error in an `MSTR` routine via TCP/IP Ethernet may produce one of the following errors in the `MSTR` control block:

The error code appears as **Mmss**, where:

- **M** is the high code
- **m** is the low code
- **ss** is a subcode

Hexadecimal Error Codes TCP/IP Ethernet

Hexadecimal error codes TCP/IP Ethernet:

Hex. Error Code	Meaning
1001	Abort by user
2001	An operation type that is not supported has been specified in the control block
2002	One or more control block parameters were modified while the `MSTR` element was active (this only applies to operations which require several cycles for completion). Control block parameters my only be modified in inactive `MSTR` components.
2003	Invalid value in the length field of the control block
2004	Invalid value in the offset field of the control block
2005	Invalid value in the length and offset fields of the control block
2006	Unauthorized data field on slave
2008	Unauthorized network routing path on slave
200E	The control block is not assigned, or parts of the control block are located outside of the %MW (4x) range.
3000	Generic Modbus failure code
30ss	Exceptional response by Modbus slave *(see page 160)*
4001	Inconsistent response by Modbus slave

ss Hexadecimal Value in 30ss Error Code

ss hexadecimal value in 30ss error code:

ss hex. Value	Meaning
01	Slave does not support requested operation
02	Non-existing slave registers were requested

ss hex. Value	Meaning
03	An unauthorized data value was requested
05	Slave has accepted a lengthy program command
06	Function cannot currently be carried out: lengthy command running
07	Slave has rejected lengthy program command

Hexadecimal Error Codes TCP/IP Ethernet Network

An error on the TCP/IP Ethernet network itself may produce one of the following errors in the `CONTROL[1]` register of the control block.

Hexadecimal error codes TCP/IP Ethernet network:

Hex. Error Code	Meaning
5004	Interrupted system invocation
5005	I/O error
5006	No such address
5009	The socket descriptor is not valid
500C	Not enough storage space
500D	Authorization denied
5011	Entry exists
5016	An argument is not valid
5017	An internal table has no more space
5020	There is interference on the connection
5023	This operation was blocked and the socket is non-blocking
5024	The socket is non-blocking and the connection cannot be closed down
5025	The socket is non-blocking and a previous connection attempt has not been concluded
5026	Socket operation on a non-socket
5027	The destination address is not valid
5028	Message too long
5029	Wrong type of protocol for the socket
502A	Protocol not available
502B	Protocol not supported
502C	Socket type not supported
502D	Operation not supported at socket
502E	Protocol family not supported
F502	Address family not supported

Hex. Error Code	Meaning
5030	Address is already in use
5031	Address not available
5032	Network is out of order
5033	Network cannot be reached
5034	Network shut down the connection during reset
5035	The connection was terminated by the peer
5036	The connection was reset by the peer
5037	An internal buffer is required, but cannot be assigned
5038	The socket is already connected
5039	The socket is not connected
503A	Cannot transmit after the socket has been shut off
503B	Too many references; cannot splice
503C	Connection timed out
503D	The connection attempt was denied
5040	Host is out of order
5041	The destination host could not be reached from this node
5042	Directory not empty
5046	NI_INIT returned -1
5047	The MTU is not valid
5048	The hardware length is not valid
5049	The route specified cannot be found
504A	Collision when invoking Select; these conditions have already been selected by another job
504B	The job ID is not valid
5050	No Network Resource
5051	Length Error
5052	Addressing Error
5053	Application Error
5054	Client cannot process request
5055	No Network Resource
5056	Non-Operational TCP connection
5057	Incoherent configuration
6003	FIN or RST not expected
F001	In reset mode
F002	Component not fully initialized

Reset Optional Module

Description

A reset optional module operation causes a Quantum NOE Ethernet communications module or the Ethernet port on a 140CPU65150/60 CPU module to enter a cycle that resets its working environment. To program an MBP_MSTR block to perform a reset option module operation, use function code 10 *(see page 137)*.

Network Implementation

A reset optional module operation can be performed on TCP/IP Ethernet *(see page 157)* and SY/MAX Ethernet networks.

Control Block Usage for TCP/IP Ethernet

Register	Meaning
CONTROL[1]	10 = reset optional module
CONTROL[2]	Indicates the error status
CONTROL[3]	No significance
CONTROL[4]	No significance
CONTROL[5]	Routing register *(see page 139)*
CONTROL[6] ... CONTROL[9]	No significance

Control Block Usage for SY/MAX Ethernet (CONTROL)

Register	Meaning
CONTROL[1]	10 = reset optional module
CONTROL[2]	Indicates the error status
CONTROL[3]	No significance
CONTROL[4]	No significance
CONTROL[5]	Routing register *(see page 140)*
CONTROL[6] ... CONTROL[9]	No significance

Read CTE

Description

A read CTE operation reads a specified number of bytes from the Ethernet configuration extension table in the specified buffer of PLC memory. These bytes start with a byte offset at the CTE table start. The CTE table contents are displayed in the `DATABUF` output parameter *(see page 135)*. To program an MBP_MSTR block to perform a clear remote statistics operation, use function code 11 *(see page 137)*.

Network Implementation

A read CTE operation can be performed on TCP/IP Ethernet and SY/MAX Ethernet networks.

Control Block Usage for TCP/IP Ethernet

Register	Meaning
CONTROL[1]	11 = read CTE
CONTROL[2]	Indicates the error status
CONTROL[3]	The length setting: a value from 12 to 37
CONTROL[4]	No significance
CONTROL[5]	Routing register Least significant byte = mapping index Either a value displayed in the byte of the register or is not used. or Most significant byte = network adapter module slot
CONTROL[6]... CONTROL[9]	No significance

Control Block Usage for SY/MAX Ethernet

Register	Meaning
CONTROL[1]	11 = read CTE
CONTROL[2]	Indicates the error status
CONTROL[3]	Number of words transferred
CONTROL[4]	Byte offset in the PLC register structure, specifying from where the CTE bytes are read
CONTROL[5]	Routing register MSB: slot of the NOE module
CONTROL[6] ... CONTROL[9]	Terminator: FF hex

CTE Indicator Implementation (DATABUF)

The values in the CTE table are displayed in the DATABUF output when a CTE read operation is implemented. The registers display the following CTE data:

CTE indicator implementation (DATABUF):

Parameter	Register	Contents
Frame type	DATABUF[0]	1 = 802.3 2 = Ethernet
IP address	DATABUF[1]	First byte of the IP address
	DATABUF[2]	Second byte of the IP address
	DATABUF[3]	Third byte of the IP address
	DATABUF[4]	Fourth byte of the IP address
Lower netmask	DATABUF[5]	Most significant word
	DATABUF[6]	Least significant word
Gateway	DATABUF[7]	First byte of the gateway
	DATABUF[8]	Second byte of the gateway
	DATABUF[9]	Third byte of the gateway
	DATABUF[10]	Fourth byte of the gateway

Write CTE

Description

A write CTE operation writes the CTE configuration table from the specified data (`DATABUF`) to a specified Ethernet configuration extension table or to a specific slot. To program an MBP_MSTR block to perform a write CTE operation, use function code 12 *(see page 137)*.

Network Implementation

A write CTE operation can be performed on TCP/IP Ethernet *(see page 157)* and SY/MAX Ethernet networks.

Control Block Usage for TCP/IP Ethernet

Register	Meaning
CONTROL[1]	12 = write CTE
CONTROL[2]	Indicates the error status
CONTROL[3]	The length setting: a value from 12 to 37.
CONTROL[4]	No significance
CONTROL[5]	Routing register Least significant byte = mapping index Either a value displayed in the byte of the address or is not used. or Most significant byte = network adapter module slot
CONTROL[6] ... CONTROL[9]	No significance

Control Block Usage for SY/MAX Ethernet

Register	Meaning
CONTROL[1]	12 = Write CTE (Config extension table)
CONTROL[2]	Indicates the error status
CONTROL[3]	Number of words transferred
CONTROL[4]	Byte offset in the PLC address structure specifying where the CTE bytes are written
CONTROL[5]	Routing register Most significant byte = NOE module slot Least significant byte = Destination drop number

Register	Meaning
CONTROL[6]	Terminator: FF hex
CONTROL[7] ... CONTROL[9]	No significance

CTE Indicator Implementation (DATABUF)

The values in the Ethernet configuration extension table are displayed in the DATABUF output field when a write CTE operation is implemented. The registers are used to transfer the following CTE data:

CTE indicator implementation (DATABUF):

Parameter	Register	Contents
Frame type	DATABUF[0]	1 = 802.3 2 = Ethernet
IP address	DATABUF[1]	First byte of the IP address
	DATABUF[2]	Second byte of the IP address
	DATABUF[3]	Third byte of the IP address
	DATABUF[4]	Fourth byte of the IP address
Lower netmask	DATABUF[5]	Most significant word
	DATABUF[6]	Least significant word
Gateway	DATABUF[7]	First byte of the gateway
	DATABUF[8]	Second byte of the gateway
	DATABUF[9]	Third byte of the gateway
	DATABUF[10]	Fourth byte of the gateway

Send Email

Description

The electronic mail notification service allows controller-based projects to report alarms or events. The controller monitors the system and dynamically creates an electronic mail message, which alerts local or remote users.

A user-defined event or condition triggers the MSTR block to create a message. Each message uses one of three user-defined headers. Each message sent from the controller can contain text and variable information (with a maximum of 238 bytes).

The project selects the appropriate header. Each header contains:
- sender name
- list of recipients
- subject

To program an MBP_MSTR block to send email, use function code 13 *(see page 137)*.

Network Implementation

A send email operation can be performed on a TCP/IP Ethernet network.

Control Block Usage for TCP/IP Ethernet

Register	Meaning
CONTROL[1]	13 = send Email
CONTROL[2]	Indicates the email-specific error codes
CONTROL[3]	Number of words transferred
CONTROL[4]	Not used
CONTROL[5]	High byte: slot address of the NOE module or 0xFE for the 140 CPU 651 60
	Low byte: always 0
CONTROL[6] ... CONTROL[9]	Not used

DATABUF Parameter Description

Register	Contents
DATABUF 1	The mail header is the least significant byte with a value of 1, 2, or 3.
	The most significant byte contains the number (*n*) of characters in the subject, a value between 0 and 238.
DATABUF 2 ... DATABUF 119	The data (in ASCII format) that will be copied into the Email message. The first *n* characters are added to the configured Email subject. The remaining characters (2 * *N* -2 - *n*) are part of the message body, where *N* is the number of words transferred.

Send Modbus Request

At a Glance

Use MSTR operation 15 to send generic Modbus requests on the network.

NOTE: This operation is not available on Modbus Plus ports (embedded port on CPU or NOM modules) and the embedded Ethernet port on a CPU.

Block Operation

The MBP_MSTR block can send requests and receive responses up to 253 bytes long.

For the operation, refer to Block Operation *(see page 134)*:

- When the ENABLE input pin is turned ON, operation 15 begins.
- If the ABORT input pin is turned ON or if the ENABLE input pin is turned OFF, the operation ends.
- The ACTIVE output pin is ON during the operation.
- The ERROR output pin turns ON if the operation aborts without success.
- The SUCCESS output pin turns ON if the operation completes with success.
- The CONTROL and DATABUF output pins define the operation (refer to the Control Block *(see page 170)* and Data Buffer *(see page 171)*).
- EN and ENO can be configured as additional parameters.

Control Block

The format of the Control block is described in the following table:

Word	Description
CONTROL[1]	15 = Send Modbus Request
CONTROL[2]	Indicates detected error status
CONTROL[3]	DATABUF length (WORDs)
CONTROL[4]	Offset for the beginning of the response in the DATABUF (WORDs). **NOTE:** To avoid overwriting the request, the Response Offset value multiplied by 2 must be greater than the Request Length (CONTROL[10]).
CONTROL[5]	Routing register: High byte = Ethernet communication module slot Low byte = MBP on Ethernet transporter (MET) mapping index (also known as Unit ID)
CONTROL[6]	Byte 4 of the IP address (MSB)
CONTROL[7]	Byte 3 of the IP address
CONTROL[8	Byte 2 of the IP address

Word	Description
CONTROL[9]	Byte 1 of the IP address (LSB)
CONTROL[10]	Length of the DATABUF Request data (bytes)
CONTROL[11]	Length of the DATABUF Response received (bytes) **NOTE:** This is read only, it is set by option module after operation completion.

Data Buffer

The MODBUS protocol defines a simple protocol data unit (PDU) independent of the underlying communication layers.

The data buffer (DATABUF) consists of contiguous registers that include both the Modbus Request PDU and the Modbus Response PDU:

DATABUF *Data Buffer Length* is set in the CONTROL[3] word.	**Modbus Request** PDU: *Data Request Length* is set in the CONTROL[10] word.
	Modbus Response PDU: Data Response Start (*Response offset*) is set in the CONTROL[4] word. **NOTE:** To avoid overwriting the request, the Response Offset value multiplied by 2 must be greater than the Request Length (CONTROL[10]). *Data Response Length* is set in the CONTROL[11] word.

⚠CAUTION

LOSS OF DATA

Verify that the *Response Offset* is greater than the *Data Request Length*.

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

NOTE: Both the Request and Response must be structured in little endian order. Each word of 2 bytes of data in little endian format, where the least significant byte is stored in the smallest memory address.

Modbus Request

The **Modbus Request** PDU format is:

Byte offset	Field	Data type	Description
1	Function code	BYTE	Modbus function code
2	Request data	BYTE area	This field is function code dependent and contains information such as variable references, variable counts, data offsets, sub-function codes and so on.

Modbus Response

The **Modbus Response** PDU format is:

Byte offset	Field	Data type	Description
1	Function code	BYTE	Modbus function code
2	Response data	BYTE area	This field is function code dependent and contains information such as variable references, variable counts, data offsets, sub-function codes and so on.

Modbus Exception Response

The **Modbus Exception Response** PDU format is:

Byte offset	Field	Data type	Description
1	Function code	BYTE	Modbus function code + 80 (hex)
2	Exception code	BYTE	This field is defined in the *MODBUS Application Protocol Specification*.

Modbus Specification

The standard Modbus function codes are defined in the *MODBUS Application Protocol Specification*, V1.1b, which can be downloaded from www.modbus-ida.org.

Example of a Send Modbus Request

To read 4 contiguous Input registers (Modbus Function code 4) starting at register 100 (64 hex) in a remote device. A **Modbus Request** has to be sent on the network with the following **Modbus Response**.

Request Field Name	Value
Function code	04
Starting address, High	00

Request Field Name	Value
Starting address, Low	64
Number of registers, High	00
Number of registers, Low	04

Response Field Name	Value
Function code	04
Byte count	08
Register 100 value, High	00
Register 100 value, Low	01
Register 101 value, High	00
Register 101 value, Low	02
Register 102 value, High	00
Register 102 value, Low	03
Register 103 value, High	00
Register 103 value, Low	04

Using the following CONTROL word values:
- CONTROL[1] (MSTR operation) = 15
- CONTROL[2] (Error Code) = refer to CONTROL block description (read-only)
- CONTROL[3] (DATABUF length) = 11
- CONTROL[4] (Response Offset) = 5
- CONTROL[5] (Routing Register) = refer to CONTROL block description
- CONTROL[6] (IP1) = refer to CONTROL block description
- CONTROL[7] (IP1) = refer to CONTROL block description
- CONTROL[8] (IP1) = refer to CONTROL block description
- CONTROL[9] (IP1) = refer to CONTROL block description
- CONTROL[10] (Request Length) = 5
- CONTROL[11] (Response Length) = 10

the data encoding in the Data Buffer is as follows:

	Field	Value (hex)	Description
Request	DATABUF[1]	0400	Function code + Starting address, High
	DATABUF[2]	6400	Starting address, Low + Number of registers, High
	DATABUF[3]	0400	Number of registers, Low + NULL
	DATABUF[4]	0000	Null
	DATABUF[5]	0000	Null

	Field	Value (hex)	Description
Response	DATABUF[6]	0408	Function code + Byte count
	DATABUF[7]	0001	Register 100 value, High + Register 100 value, Low
	DATABUF[8]	0002	Register 101 value, High + Register 101 value, Low
	DATABUF[9]	0003	Register 102 value, High + Register 102 value, Low
	DATABUF[10]	0004	Register 103 value, High + Register 103 value, Low
	DATABUF[11]	0000	Null

Close Connection Request

At a Glance

Use MBP_MSTR operation 16 to close a Modbus TCP/IP connection on the network.

NOTE: This operation is not available on Modbus Plus ports (embedded port on CPU or NOM modules).

Block Operation

The MBP_MSTR block can send requests and receive responses up to 253 bytes long. For block operation, refer to the block description *(see page 134)*:

- When the ENABLE input pin is turned ON, operation 15 begins.
- If the ABORT input pin is turned ON or if the ENABLE input pin is turned OFF, the operation ends.
- The ACTIVE output pin is ON during the operation.
- The ERROR output pin turns ON if the operation aborts without success.
- The SUCCESS output pin turns ON if the operation completes with success.
- The CONTROL and DATABUF output pins *(see page 136)* define the operation.
- EN and ENO can be configured as additional parameters.

Control Block

The format of the CONTROL block is described below:

Register	Function	Description
CONTROL[1]	Operation	16 = close connection
CONTROL[2]	Error status	Indicates detected error status (read only)
CONTROL[3]	(not used)	–
CONTROL[4]	(not used)	–
CONTROL[5]	Routing Register	High byte = Ethernet communication module slot
		Low byte = MBP on Ethernet transporter (MET) mapping index (also known as Unit ID)
CONTROL[6]	IP Address	Byte 4 of the IP address (MSB)
CONTROL[7]		Byte 3 of the IP address
CONTROL[8]		Byte 2 of the IP address
CONTROL[9]		Byte 1 of the IP address (LSB)

Change Modbus Plus Address

Description

Use MSTR **Operation 17** (Change Modbus Plus Address) to change the Modbus Plus address associated with the embedded Modbus Plus port of the CPU.

On Low End CPUs, this function overrides the address switch (the hardware rotary switch behind the CPU) as long as the CPU is powered on. The address of hardware switch is restored when the CPU is restarted.

On High End CPUs, a new address set with this function is saved in the configuration table and is restored on subsequent CPUs starts.

Setting an address to a station that is already used by another station on the network disconnects the station already on the network and may cause unpredictable operation on the network.

⚠CAUTION

UNINTENTIONAL EQUIPMENT OPERATION

Ensure that the Modbus Plus address to be set is not assigned to another station before using Operation 17.

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

Network Implementation

The Operation 17 can only be performed for a Modbus Plus CPU embedded port. This operation is not available on the Modbus Plus ports of NOM modules.

Control Block

The following table describes the format of the Operation 17 Control Block:

Register	Meaning
Control[1]	17 (Change Modbus Plus Address operation)
Control[2]	Detected error status, see Modbus Plus, SY/MAX, and Ethernet TCP/IP Error Codes *(see page 141)*
Control[3]	16#1234
Control[4]	16#5678
Control[5]	16#00AA
Control[6	16#00BB
Control[7]	16#00CC

Register	Meaning
Control[8]	16#00DD
Control[9]	16#00EE

NOTE: The constant values in CONTROL[3] to CONTROL[9] consist of a signature that prevents unintentional use of this operation.

Data Buffer

The following table describes the format of the Operation 17 Data Buffer:

Register	Meaning
DADABUF[1]	Low byte: new address (range: 1 to 64)

Read/Write Data

Introduction

In a single transaction, the MSTR read and write operations can transfer data from a master source device to a specified slave destination device, then transfer data from this specified slave source to the master. It uses a master transaction path and may require several cycles to complete. To program an `MBP_MSTR` block to perform a combined read/write operation, use function code 23 *(see page 137)*.

The combined read/write operation can be used only with these models:
- Quantum: NOE 771 01 (3.0 or later) or NOE 771 11 (3.0 or later)
- Momentum Unity: M1E Unity CPU (1.0 or later)

Control Block Usage

Register	Content
CONTROL[1]	23 = read/write data.
CONTROL[2]	Indicates the error status.
CONTROL[3]	Number of registers to be sent to the slave.
CONTROL[4]	Specifies the %MW starting register in the slave to which the data will be written, e.g. 1 = %MW1, 49 = %MW49.
CONTROL[5]	Routing register: Most significant byte: network adapter module slot. Least significant byte: MBP on Ethernet Transporter (MET) mapping index.
CONTROL[6] ... CONTROL[9]	Each address contains 1 byte of the 32-bit IP address, where the MSB is in CONTROL[6] and the LSB is in CONTROL[9].
CONTROL[10]	Number of registers to be read from slave.
CONTROL[11]	Specifies the %MW starting register in the slave from which the data is read, e.g. 1 = %MW1, 49 = %MW49.

NOTE:

When configuring the `MBP_MSTR` block for a read/write data operation, note that
- The `DATABUF` output parameter is used to store, in the following sequence, both:
  1 the data to be written
  2 the data to be read
- The size of the `DATABUF` output parameter must equal the combined size of the data to be written and the data to be read; if the size is smaller, data will be overwritten and may be lost.
- Both the `CONTROL` and `DATABUF` parameters must be stored at located addresses, for example %MW addresses.

Enable / Disable HTTP or FTP/TFTP Services

Description

An `Enable Disable HTTP or FTP/TFTP Services` operation changes the enabled state of the module's HTTP or FTP/TFTP services.

Network Implementation

An `Enable Disable HTTP or FTP/TFTP Services` operation can be performed on TCP/IP Ethernet networks *(see page 157)*.

Control Block Usage for TCP/IP Ethernet

When HTTP or FTP/TFTP has been enabled using Unity Pro configuration tools *(see page 105)*, an MSTR block can be used to change the enabled state of the service while the application is running. The MSTR block cannot change the state of the HTTP or FTP/TFTP services if the service was disabled using one of the configuration tools.

Register	Meaning
CONTROL[1]	65520 (dec) FFF0 (hex) = enable / disable HTTP or FTP/TFTP
CONTROL[2]	Indicates the error status. Possible return codes include: 0x000 (Success): MSTR block with operational code 65520 was called and the enabled state of HTTP or FTP/TFTP was changed. 0x5068 (Busy): MSTR block with operational code 65520 was called within 2 seconds of the previous call (regardless of return code from previous call). 0x4001 (Same state): MSTR block with operational code 65520 was called to change the enabled state of HTTP and FTP/TFTP to the states they were already in. 0x2004 (Invalid data): MSTR block with operational code 65520 was called and the data in the control block did not match the specifications. 0x5069 (Disabled): If the HTTP or FTP/TFTP service was already disabled via the Unity Pro interface when the MSTR block with operational code 65520 was called to change the state of the disabled service.
CONTROL[3]	Set this register to 1
CONTROL[4]	
CONTROL[5]	Module slot number and destination ID High byte = Module slot number communication module slot Low byte = Destination ID
CONTROL[6]	Request mode Bit 0 (LSB) = 1: Enable FTP/TFTP Bit 0 (LSB) = 0: Disable FTP/TFTP Bit 1 = 1: Enable HTTP Bit 1 = 0: Disable HTTP

Register	Meaning
CONTROL[7]	Set this register to 0
CONTROL[8]	
CONTROL[9]	

HTTP, FTP, and TFTP service state changes made by MSTR with operation code 65520 (dec) are overridden by the configured value when the module is power-cycled or reset and when a new application is downloaded to the module.

Here are some examples:

State Configured By Unity Pro	Action attempted using MSTR with operation code 65520) (dec)	Result
Disabled	Any	MSTR returns error code 0x5069 (service was already disabled by configuration)
Enabled	Disable	MSTR returns code 0x000 (success). ● Another MSTR block action enables the service --OR-- ● The module is reset or power-cycled --OR-- ● A new application is downloaded with the service disabled by configuration
	Enable	MSTR returns error code 0x4001 (same state). No change made.

Chapter 5
Global Data (Publish/Subscribe) Utility

Introduction

The material in this section presents the Global Data (Publish/Subscribe) utility available on the following modules.

- 140 NOE 771 01
- 140 NOE 771 11
- 140 CPU 651 x0

For more information on the publish-subscribe model, go to this URL:

`http://www.isa.org/journals/intech/feature/printable/1,1171,596,00.html`

What Is in This Chapter?

This chapter contains the following topics:

Planning the Global Data (Publish/Subscribe) System

Overview

Global Data service is a real-time publisher/subscriber mechanism that provides the most efficient data exchange for PLC application coordination.

Devices that support Global Data are arranged in a distribution group for the purpose of application variable exchange and synchronization. Each Global Data device can publish up to one network (application) variable and subscribe to up to 64 network (application) variables.

The Quantum NOE's embedded **Global Data Configuration** Web page provides a configuration screen *(see page 188)* to determine which and how many application variables are exchanged with this service. After configuration, exchanges between all stations belonging to the same distribution group are done automatically.

The Global Data service uses %MW (4x registers) or unlocated variables for Global Data exchanges.

Key Features of Global Data

The main features for Global Data are:
- One publisher and many subscribers
- A device can publish one network variable of up to 512 %MW words (4x registers) or unlocated variables
- A device can subscribe to up to 64 network variables of up to 2048 %MW words (4x registers) or unlocated variables
- A device subscribes to the complete network variable
- One distribution group per network IP address
- Application-defined publication rate
- Up to 64 Global Data Network variables (numbered from 1 to 64) can be part of the data distribution group
- An NOE has only one multicast address. Consequently, it can only publish and subscribe within the group
- A device can participate in several distribution groups by using multiple NOEs in the rack

Global Data has an advantage over client/server services when more than one subscriber is receiving the same data since only one transaction is necessary for all subscribers to receive the data. There are two benefits:
- the reduction of overall network traffic
- tighter synchronization of multiple subscribers

Planning Your System Configuration

The Global Data (publish/subscribe) utility is a powerful function incorporated into the NOE product line. Implementing Global Data requires a configuration that spans many PLCs throughout the system. Therefore, we recommend pre-planning your installation before implementation. Pre-planning saves time and money by:

- reducing errors, which circumvents a debugging effort
- ensuring system consistency

Go to paper before computer.

We offer the following table to assist in system planning. The **Global Data Planning Spreadsheet** below is a graphic representation of a recommended configuration table for system planning. You can create your own table using the format below or you can download a Microsoft *Excel*™ spreadsheet template, which is available on the Schneider public Web site.

Here is the graphic representation of the **Global Data Planning Spreadsheet**:

<table>
<tr><th>Parameter Checking</th><th>Variable ID</th><th>Symbol*</th><th>Length (Registers)</th><th colspan="4">Device Number</th><th>Variable Public. Status</th></tr>
<tr><td></td><td></td><td></td><td></td><td>1</td><td>2</td><td>...</td><td>3</td><td></td></tr>
<tr><td></td><td>1</td><td>VALVE_STATUS</td><td>20</td><td>PUB</td><td>SUB</td><td></td><td>NONE</td><td>OK</td></tr>
<tr><td></td><td>2</td><td>VALVE_CONTROL</td><td>10</td><td>SUB</td><td>NONE</td><td></td><td>PUB</td><td>OK</td></tr>
<tr><td></td><td>...</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td>64</td><td>PUMP_CONTROL</td><td>50</td><td>SUB</td><td>PUB</td><td></td><td>NONE</td><td>OK</td></tr>
<tr><td colspan="4">Device Publication Status:</td><td>OK</td><td>OK</td><td></td><td>OK</td><td rowspan="3"></td></tr>
<tr><td colspan="4">Total Publication Size per Node:</td><td>20</td><td>50</td><td></td><td>10</td></tr>
<tr><td colspan="4">Total Subscription Size per Node:</td><td>60</td><td>20</td><td></td><td>0</td></tr>
<tr><td colspan="2">Group IP Address</td><td colspan="2">239.255.255.0</td><td colspan="5" rowspan="6"></td></tr>
<tr><td colspan="2">Multicast Filtering Enabled</td><td colspan="2">OFF</td></tr>
<tr><td colspan="2">Default %MW Address for Health</td><td colspan="2">%MW100</td></tr>
<tr><td colspan="2">Distribution Period</td><td colspan="2">10</td></tr>
<tr><td colspan="2">Health Timeout</td><td colspan="2">1000</td></tr>
<tr><td colspan="2">Data Zone</td><td colspan="2">%MW200</td></tr>
<tr><td colspan="9">*Entries or changes to the symbol (description) do NOT affect or change a variable or the system. The symbol used in the Quantum product line has no relation to the Concept/Unity product line symbol(s).</td></tr>
</table>

Table of **Global Data Limits**:

Parameter	Limit
Maximum number of publish variables per device	1
Maximum size for the publish variable	512 registers = 512 Words (16 bits) = 1024 bytes
Maximum number of subscription variables per device	64 (63 if this device is publishing)
Maximum size for the subscribe variables per device	2048 registers = 2048 Words (16 bits) = 4096 bytes

NOTE: We recommend that you consider the following when planning.

- Allow for a 10 to 20 % increase in growth of any variable.
- Add variables at the end of the configuration where they do not affect the existing application address. Therefore, you avoid changing the existing addresses in your configuration, which can be a time consuming process.

Table of **Global Data Planning Spreadsheet**

Parameter	Description
Parameter Checking	Reserved
Variable Id	Represents the Data ID on the NOE's **Global Data Configuration** Web page
Symbol	Symbolic name for Global Data exchange.
Length Words (Registers)	Length of Global Data information. Number of %MW words (4x registers) or unlocated variables.
Device Number	Number of devices (up to 64) for the Global Data network.
Variable Public. Status	Automatic information of the correct publication status of the Global Data network. Only by using the Microsoft *Excel*™ spreadsheet. Information per symbol.
Device Publication Status	Automatic information of the correct publication status of the Global Data network. Only by using the Microsoft *Excel*™ spreadsheet. Information per device.
Total Publication Size per Node	Publication size for the specific node. The maximum publication size is 512 words (registers) per node
Total Subscription Size per Node	Subscription size for the specific node. The maximum subscription size is 2048 words (registers) per node
Group IP Address Enabled	IP address for multicast networking. Identifies the stations distribution group. The address range is from 224.0.0.0 to 239.255.255.255
Multicast Filtering Enabled	A check box for Ethernet switches that support multicast filtering.
Default Address for Health%MW (4x register)	%MW (4x register) address for the Health bits. This is the memory area where the Health bits are stored. It has the size of 4 words (registers).

Parameter	Description
Distribution Period	Is the minimum number of controller scan times before an update will occur.
Health Timeout	Is the maximum time between received subscriptions before a subscription is declared unhealthy (faulty). The value is measured in milliseconds and can be set from 50 to 15000 ms in 50 ms increments.
Data Zone	The starting address for the data. This are the registers where the data information are stored.

Multicast Filtering

Overview

Your NOE may offer the multicast filtering functionality.

The global data service synchronizes several stations located in a distribution group. A distribution group is a set of stations identified by using the same IP multicast address for all stations in the group. By using the same IP address for multiple devices, multicast exchanges can be used to distribute global data. Several independent distribution groups can coexist on the same sub-network. Each distribution group possesses its own unique IP multicast address.

Early versions of switches treat multicast packets as a broadcast, thereby broadcasting to all nodes and suppressing all benefits of both switching and multicasting. Newer versions of switches provide automatic multicast filtering, and consequently only forward multicast traffic to ports that are connected to registered end stations.

Multicast Filtering Protocols

The following multicast filtering protocols are supported by Ethernet modules for its global data service.

- GARP Multicast Registration Protocol (GMRP)
  GMRP provides a mechanism that allows bridges and end stations to dynamically manage the membership of multicast groups.
  NOTE: GMRP is defined in the IEEE 802.1D-1998 Standard, which is available as a free download at: http://IEEE802.org.
- Internet Group Management Protocol (IGMP)
  IGMP is a communications protocol used to manage the membership of internet protocol multicast groups. IGMP is used by IP hosts and adjacent multicast routers to establish multicast group memberships.

Reducing Traffic

Multicast filtering helps to reduce the traffic on a network, because broadcasts are sent only to interested, or subscribed, devices.

For distributed applications and one to many communications, multicast affords advantages over unicast because it:

- utilizes the network bandwidth more efficiently
- sends a single transmission instead of multiple transmissions.
- reduces collisions
- optimizes the performance of Ethernet module processing

Using Multicast Filtering

These ConneXium switches support multicast filtering. Other switches from alternate vendors also support multicast filtering.

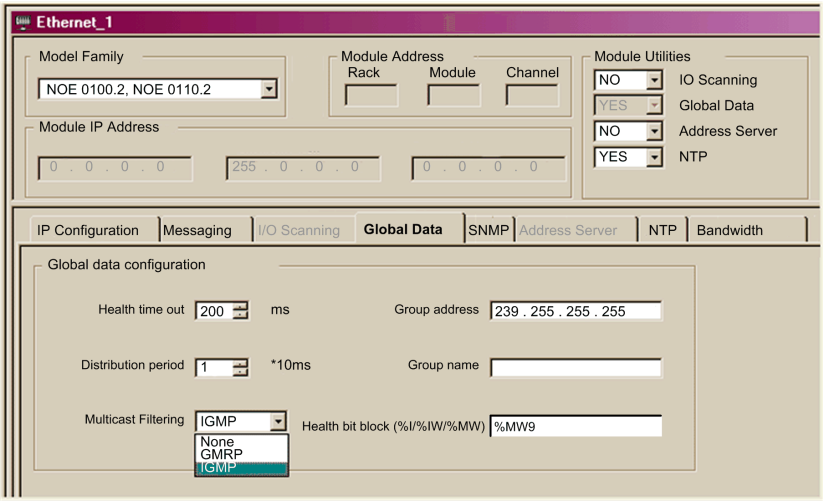
Switch	Description
499NES17100	Managed Switch with 7 ports 10/100 BASE-TX
499NOS17100	Managed Switch with 5 ports 10/100 BASE-TX and 2 ports 100 BASE-FX

Quantum NOE Global Data Configuration

Introduction

Global data configuration is carried out in the network configuration as well as the data editor. The variables for the publish/subscribe procedure are configured in the data editor.

The screen shot shows the network configuration global data configuration settings:



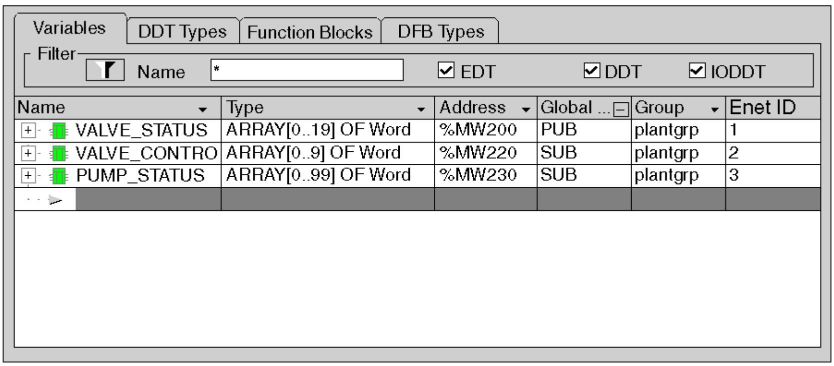
Parameter description:

Parameter	Description
Health time out	After this time period expires, the received data becomes invalid.
Group address	Class D multicast IP address. All nodes in the global data procedure use the same multicast address for distributing or receiving data. The address range is: 224.0.0.0 to 239.255.255.255.
Distribution period	Time after which the data is received or sent. Minimum scan time of the PLC.
Group name	Logical name. Defines the varibale allocation to different communication configurations in the variable editor.
Health bit block	Address for retrieving the status information of the global data procedure.

Parameter	Description
Multicast filtering	Activates an Ethernet module that supports multicast filtering. From the drop down list, select: ● None: disable both GMRP & IGMP (Data will be sent to all end devices in the network.) ● GMRP Make sure your client, server and switches, and routers support and enable GMRP. ● IGMP V1 Make sure your client, server and switches, and routers support and enable IGMP. **Note:** The following modules support IGMP V1: ● 140 NOE 771 01 V4.4 or later ● 140 NOE 771 11 V4.4 or later ● 140 CPU 651 50/60 V2.7 or later **NOTE:** The **IGMP** and **None** features are only available in Unity 4.1 or later.

NOTE: A Quantum PLC does not update health bits in `STOP` mode.

The screen shot shows an image of the data editor:



Parameter description:

Parameter	Description
Name	Variables symbols
Type	Variable type
Address	Variable address
Global Data	Type of Global Data Variable. Options: No/Publish/Subscribe

Parameter	Description
Group	Group name for allocating the variables of the existing network description. When creating the different Ethernet networks, a logical connection is arranged here between the network and the variable declaration.
ID	Variable ID

Configuration of Global Data (Publish/Subscribe) by the Web for NOE 771 x1

Modeling the System Configuration

There are two methods of configuring a system:

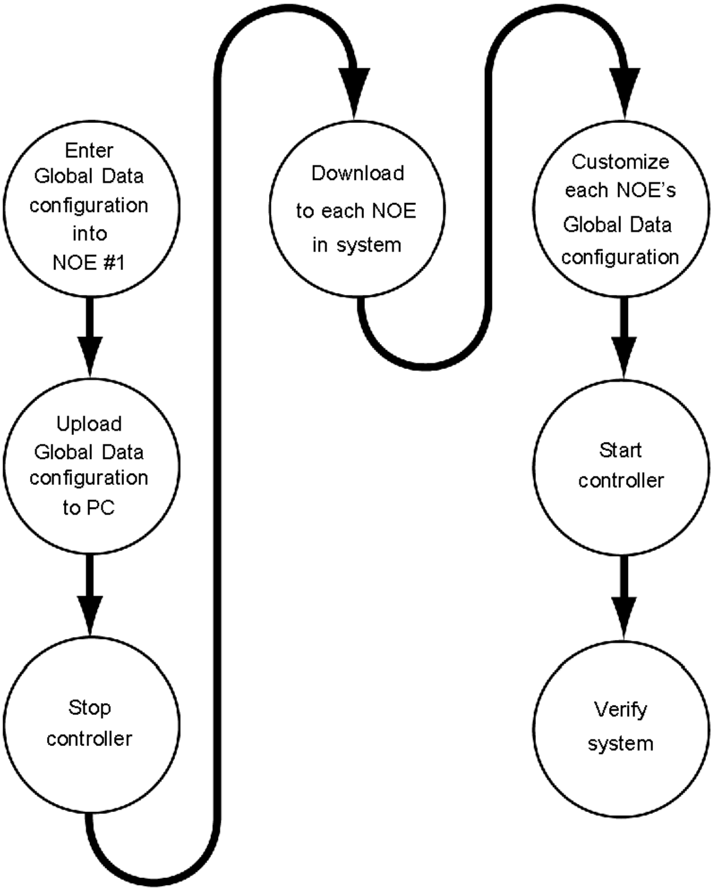
- Configuring each device separately
  Configure each device through the **Global Data Configuration** Web page *(see page 291)*. Repeat for each device in the system. Entry errors may occur because the same information must be reentered on each and every device, possibly as many as 64 times.
- Copying the configuration (preferred)
  Configure all variable IDs, symbols (descriptions), and lengths on one NOE, upload to your PC, download the same configuration to all devices and finish with a small customization to each node.

NOTE: The **Copy Configuration** method reduces entry errors, allows for verifying the configuration, and permits you to determine if the system is satisfactory for your needs before implementing the configuration system wide.

NOTE: You enter the variable ID, symbol (description), and length only once, thus maintaining consistency.

When your planning is complete, follow the *Configuring Your NOE* steps below. We present these steps both as a picture and as instructions.

The following illustration shows the *Configuring Your NOE* steps:



Configuring the NOE

Step	Action
1	Select one NOE.
2	Using your browser, navigate to that NOE's **Global Data Configuration** page. Follow these links: | **Setup** | **Global Data** |
3	Enter the configuration's variable IDs, symbols (description), and lengths.

Step	Action
4	Click the **Update Global Data Configuration** button to update the file. This creates the file `glbdata.ini`. Full path follows: `ftp://NOE_IP_ADDRESS/wwwroot/conf/glbdata/glbdata.ini` **Note:** You substitute `NOE_IP_ADDRESS` with an address like 192.168.000.010. Please check the address with your system administrator.
5	Using the FTP process, upload the `glbdata.ini` file to a PC. (See the information below at **Uploading a glbdata.ini file**.)
6	Stop each controller before you do the customization.
7	Using the same path, download the `glbdata.ini` file to the other devices. (See the information below at **Downloading a glbdata.ini file**.)
8	Connect your Web browser to each device to customize start address and the Publish/Subscribe setting.

NOTE: The Global Data Configuration page is populated with data from the `glbdata.ini` configuration file.

Uploading a glbdat.ini File to a PC

Step	Action
1	At the DOS prompt type **FTP** followed by the IP address and press **Enter**.
2	At the User prompt type **FTP** Username and press **Enter**.
3	At the password prompt enter your FTP Password and press **Enter**.
4	At the FTP prompt type **cd wwwroot/conf/glbdata** and press **Enter**.
5	At the FTP prompt type **get** and press **Enter**.
6	At the local file prompt type **glbdata.ini** and press **Enter**.
7	At the remote file prompt type **glbdata.ini** and press **Enter**.

Downloading a glbdat.ini File to another NOE

Step	Action
1	At the DOS prompt type **FTP** followed by the IP address and press **Enter**.
2	At the User prompt type the FTP username and press **Enter**.
3	At the password prompt enter your FTP Password and press **Enter**.
4	At the FTP prompt type **cd wwwroot/conf/glbdata** and press **Enter**.
5	At the FTP prompt type **put** and press **Enter**.
6	At the local file prompt type **glbdata.ini** and press **Enter**.
7	At the remote file prompt type **glbdata.ini** and press **Enter**.

Verifying System Operation

Step	Action
1	Verify all controllers are running.
2	Look at the health of all variables using the **Global Data Diagnostic** page. Follow these links: | **Diagnostics** | **NOE Diagnostics** | **Global Data**

Chapter 6
I/O Scanner

Introduction

This chapter discusses the Ethernet modules' I/O scanner capabilities.

What Is in This Chapter?

This chapter contains the following topics:

I/O Scanner Concepts

Introduction

The I/O Scanner is a feature that resides in Ethernet (NOE 771 00, -01, -11, and CPU 651 *x*0) modules. The I/O Scanner allows, through a controller, repeated reading from and/or writing to scanned devices.

Use the I/O scanner to transfer data between network devices.

Configure the I/O Scanner with Schneider Electric's programming package Unity Pro. (I/O Scanner configuration information *(see page 200)* appears elsewhere in this guide.)

Health Block Bits

Each bit corresponds to an entry in the I/O Scanner table. Each entry in the table represents one logical device.

The bits contain the health status for the Modicon Quantum I/O Scanner.

NOTE: A Quantum PLC does not update health bits in `STOP` mode.

Device Control Block

With the Device Control Block, disable an individual entry/device in the I/O Scanner table by checking the box and setting the associated bit in Device Control Block to 1.

Repetitive Rate Step

In the Repetitive Rate Step field, enter a rate of time for how often you want the I/O scanner to send a query to the device after the rate has timed out.

I/O Scan List

The I/O scan list is a configuration table that identifies the targets to which repetitive communications are authorized. While the controller is running, the Ethernet module transfers data to and from the controller's registers as indicated by the I/O scan list.

I/O Scanner Parameters

The I/O Scanner parameters are described in the following table:

Parameter	Value
Max. No. of Devices	64: 140 NOE 771 00 (Version 2.2 or earlier) 128: 140 NOE 771 00 (Version 3.0 or later), 140 NOE 771 01, and 140 NOE 771 11 only 128: 140 CPU 651 x0
Max. No. of Input Words	4000
Max. No. of Output Words	4000

<table>
<tr><th>Parameter</th><th>Value</th></tr>
<tr><td>Health Timeout Value</td><td>User configured: 1...65535 ms in 1 ms increments</td></tr>
<tr><td>Last Value (Input)</td><td>User configured (Zero or Hold)</td></tr>
<tr><td>IP Address</td><td>User configured
IP address of scanned device (Slave IP)</td></tr>
<tr><td>Local and Remote Register Reference</td><td>User configured</td></tr>
<tr><td rowspan="1">Repetitive Rate</td><td>The rate at which data will be scanned, from 0...65535 in multiples of:
If you are running Unity Pro V2.0 or higher
140 NOE 771 01/11: 16 ms
140 CPU 651 x0: 10 ms
If you are running Unity Pro V4.0
140 NOE 771 01/11: 5 ms (min), 200 ms (max)
140 CPU 651 x0: 5 ms</td></tr>
<tr><td>Unit ID</td><td>User configured
Configure ID only if using a bridge</td></tr>
<tr><td>Gateway/Bridge Device</td><td>To allow slower TCP/IP network devices (i.e., gateways and bridges) to be compatible with the I/O Scanner:
Select the check box to enable this feature. Defines a new bit, and sets it to high (1).
Deselect the check box to disable this feature (default). Defines a new bit, and sets it to zero (0).</td></tr>
<tr><td rowspan="2">Operation through a bridge</td><td>Modbus bridge: Supported</td></tr>
<tr><td>Modbus Plus bridge: Supported</td></tr>
</table>

Elsewhere in this guide is information about I/O scan response times for high-performance communications modules *(see page 225)*.

Using the I/O Scanner with an IP Router

NOTE: The I/O Scanners in the 140 NOE 771 x1 and 140 CPU 651 x0 modules send out requests with a Time To Live (TTL) of 32, which allows passage through multiple routers.

Device Control Block

Important information about using the Device Control Block:

Registers	The Device Control Block consists of registers either 8 single words or 4 double words. Contents of the registers are mapped in the controller's memory. Each bit corresponds to an entry in the table (see the tables below.)
Disabling Devices	Each I/O Scanner device can be disabled. To disable individual devices: **1.** Select the Device Control Block option on the I/O Scanner tab in Unity Pro. (Insert a check mark in the box.) **2.** Set the associated bit = 1.
Mapping Device Control Block Bits to I/O Scanner Entry Numbers (#)	See the table for mapping entry numbers to bits. Each entry number represents a logical device on the network.
Setting Bits	If Device Control Block bit is set to ● 0 = Device is enabled ● 1 = Device is disabled

Mapping Device Control Block Bits to I/O Scanner Entry Numbers (#)

Single Word (W) Register (%MD*x*:4)																	
W1 %MW [x+1]	Table Entry #	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
	Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
W2 %MW [x+2]	Table Entry #	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
	Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
W3 %MW [x+3]	Table Entry #	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48
	Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
							Word 4 through Word 7 (Table Entry 49 through 112)										
W8 %MW [x+8]	Table Entry #	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127	128
	Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

NOTE: Bits are counted from right to left starting from bit 0 (low bit). Examples: to configure %MD1:4 as a device control block in the I/O Scanner table, use %MW2, bit 15 to enable or disable table entry 1. Use %MW3, bit 15 to enable or disable table entry 17.

NOTE: Quantum and Premium co-processors follow Premium Ethernet module control bit operation.

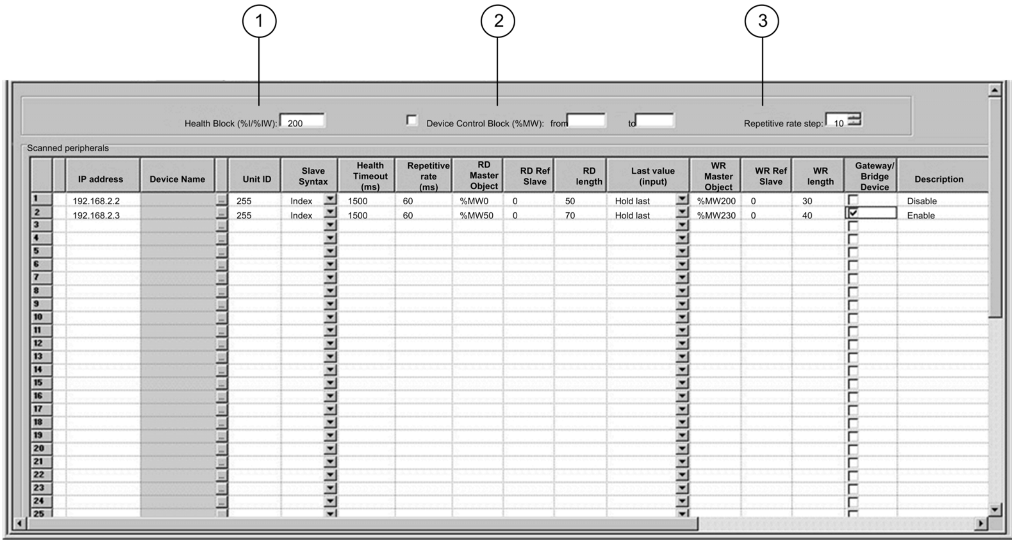
Double Word (DW) Register (%MD*x*:4)																	
DW1%MD x[0]	Table Entry #	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
	Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DW2 %MD x[1]	Table Entry #	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
	Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Table Entry #	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48
	Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
							Word 3 through Word 4 (Table Entry 49 through 112)										
DW %MD x[4]	Table Entry #	113	114	115	116	117	118	119	120	12	122	123	124	125	126	127	128
	Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

NOTE: Bits are counted from right to left starting from bit 0 (low bit). For example, to configure %MD1:4 as a device control block in the I/O scanner table, use %MD1[0], bit 31 to enable or disable table entry 1. Use %MD1[1], bit 15 to enable or disable table entry 17.

Configuring the Quantum I/O Scanner

The I/O Scanner Configuration Dialog

When you click the **I/O Scanning** tab in the Ethernet module configuration screen, the **I/O Scanner** screen opens:



NOTE: This screen shows I/O Scanner services for an 140 NOE 771 01 or -11 module. The **I/O Scanner** screen is slightly different for the CPU 651 x0 module, which does not display a **Health Block** field.

Health Block

The **Health Block** field (number 1 in previous figure) lets you define the first word or bit in a health table. That table can be up to 8 words (%IW) or 128 bits (%I) in length. Each bit in the table represents the status of an individual device. The following table shows how the bits are displayed based on the data type you use.

Bit	Data Type	
	%I	**%IW**
1	%I1	%IW1.15
2	%I 2	%IW1.14
...		
16	%I16	%IW1.0
17	%I17	%IW2.15

By default, the table comprises words starting at %IW1.0. If you want to set the table to bits, you need to specify a %I value in an increment of 16 (%I1, %I16, %I32, etc.).

NOTE: The **Health Block** field is available only for the NOE 771 00, -01, and -11. It is not available for the CPU 651 x0.

NOTE: For Safety Monitor applications, the **Health Block** is mapped on %M/%MW from the UMA zone instead of %I/%IW.

Device Control Block

The **Device Control Block** (number 2 in previous figure) lets you disable any scanned device by setting a bit associated with that device to 1. The I/O scanner closes the connection and sets the Health Bit to 0 (unhealthy state).

To enable the **Device Control Block**, select the check box in the **I/O Scanner configuration** dialog (item 2 in previous figure).

NOTE: To enable the **Device Control Block**, use:

- Unity Pro at V2.0 or later
- an 140 NOE 771 01 or 140 NOE 771 11 at version 3.5 or later
- a 140 CPU 651 x0 at version 2.0 or later

NOTE: If you attempt to enable the **Device Control Block** with an earlier version of the firmware, the I/O scanning service is disabled.

NOTE: If the check box is not selected, the I/O scanner service is enabled, and you cannot disable the I/O scanning of individual devices.

NOTE: Disabling I/O scanning with the **Device Control Block** is equivalent to ending communication between the scanner and the device.

Therefore:

- The fallback positions are applied to the inputs by the scanner.
- Communication with the device stops.
- All entries in the IN and OUT tables still transfer between the CPU and the scanner on each scan.

NOTE: As a consequence of the last point above, if you modify a %MWi attached to an input, this %MWi is overwritten by the values coming from the I/O scanner in the next scan (with either 0 or the last input value).

It is possible (but meaningless) to access %MW attached to the outputs because they are not transmitted to the device.

Repetitive Rate Step

The **Repetitive Rate Step** (number 3 in previous figure) is set in multiples of 5 ms (the minimum) through 200 ms (the maximum).

The **Repetitive Rate** column is where you enter a rate of time for how often you want the I/O scanner to send a query to the device after the rate has timed out.

NOTE: The repetitive rate of the I/O scanner table is a multiple of the rate displayed in the **Repetitive Rate Step**. The real repetitive rate being executed by the I/O scanner service is shown in the **Repetitive Rate** column.

NOTE: An entry in the **Repetitive Rate** column is rounded up to the next multiple that was entered in the **Repetitive Rate Step** if the entry is not a multiple of the **Repetitive Rate Step**.

NOTE: For example, if the entry in the **Repetitive Rate Step** is 5 and you enter a 7 in the **Repetitive Rate** column, the 7 is rounded up to 10; if you change the **Repetitive Rate Step** to 6 and enter a 7 in the **Repetitive Rate** column, the 7 is rounded up to 12.

I/O Scanner Table Parameters

The I/O Scanner parameters are described in the table below:

Parameter	Description	Example
Entry #	This is the first column; it has no name. Valid range: 1 ... 128 Each entry represents an I/O Scanning exchange on the network.	
IP address	This is the IP address of the scanned Ethernet slave device.	192.168.1.100
Device Name	To configure a device (Advantys island or DTM), click the **...** button to open the **Property** box *(see page 217)* to start the device configuration software. For an introduction to this procedure for Advantys, go here *(see page 209)*. For an introduction to this procedure for DTMs, go to FDT Container. **NOTE:** While the **Property** box is open, **I/O scanning** cannot be edited.	MySTB1 or Master_PRM_DTM_10
Unit ID	This field associates the slave address of the device connected to an Ethernet/Modbus gateway with the IP address of that gateway: ● Value range: 1 to 255 ● Default value: 255 When using a bridge, enter the bridge index (1 to 255) in this field.	255

Parameter	Description	Example

Slave Syntax	Use this drop-down menu to pick the way **RD Ref Slave** and **WR Ref Slave** values are displayed. There are 4 types of display available:
	● Index: 100
	● Modbus: 400101
	  ● (Modbus register)
	● IEC 0: %MW100
	  ● M340 and Premium PLC slaves
	● IEC 1: %MW101
	  ● Ouantum PLC slaves
		Index (default value)

Health Timeout (ms)	This field sets the maximum interval between the responses from a remote device. After this time period expires, the received data is invalid. The **Health Timeout** should be longer than the **Repetitive rate** time (ms). For a Quantum NOE Ethernet module, it also should be longer than the CPU scan time. For the **Health Timeout**: ● Range: 1ms to 65535 ms ● Interval: 1ms	1500 ms

Repetitive rate (ms)	The rate at which data is scanned, from 0...65535 in multiples of the **Repetitive Rate Step**:
	● If you are running Unity Pro V3.1 or earlier with the following firmware versions:
	  ● 140 NOE (V4.0 or earlier): 16 ms
	  ● CPU (V2.5 or lower): 10 ms
	● If you are running Unity Pro V4.0 or later with the following firmware versions:
	  ● 140 NOE (V4.3 or later): 5 - 200 ms
	  ● CPU (V2.6 or later): 5 - 200 ms
		60 ms

RD Master Object*	Destination address in the master PLC where, from each device, newly read information is stored	%mw10
RD Slave Ref.**	Source address index in the slave/remote device	The format of this value depends on the **Slave Syntax**: ● Index: 5 ● Modbus: 400006 ● IEC 0: %MW5 ● IEC 1: %MW6
RD length	Number of words to read	10

Parameter	Description	Example
Last value (Input)	This field configures the behavior of inputs in the event of an access error in relation to the remote device (for example: inoperative network or device power supply, etc.): ● Set to 0: fall back to 0 ● Hold last: maintain last value	Hold last
WR Master Object*	Source address of the master PLC whose data is being written into the slave/remote device. Write operations are always performed at the word level.	%mw20
WR Slave Ref.**	The address of the first word written into the slave/remote device.	The format of this value depends on the **Slave Syntax**: ● Index: 1 ● Modbus: 400002 ● IEC 0: %MW1 ● IEC 1: %MW2
WR length	Number of words to be written	10
Gateway/Bridge Device	To allow slower TCP/IP network devices (i.e., gateways and bridges) to be compatible with the I/O Scanner: ● Select the check box to enable this feature. Defines a new bit, and sets it to high (1). ● Deselect the check box to disable this feature (default). Defines a new bit, and sets it to zero (0).	
Description	Additional information	
*Master refers to the client PLC that makes the request. **Slave refers to the server from which data is read or to which data is written.		

NOTE: For more information, refer to the Contextual Menu for Copy/Cut/Paste topic *(see page 205)*.

NOTE: For more information, refer to the I/O Scanning with Multiple Lines topic *(see page 207)*.

I/O Scanning Contextual Menu for Copy/Cut/Paste

At a Glance

A right-click on a line in the **I/O Scanning** table opens the **I/O Scanning Contextual Menu**. Use this menu to perform common operations on the lines of the **I/O Scanning** table, such as, delete a device, copy & paste, cut & paste, insert a new line, etc.

Contextual Menu

The following illustration is the **I/O Scanning** contextual menu:

The following table describes the menu functions:

Menu Item	Description
Delete Device	For an ACS or PRA configuration, **Delete Device** permanently deletes the **Device Name** and all its data (and associated ACS symbols). For a **PRM Master DTM**, its link to the **I/O Scanning** table is deleted, **NOTE:** Deleting a **PRM Master DTM** link from the **I/O Scanning** table does not delete the corresponding DTM from the connectivity tree in the DTM Browser.
Cut line(s)	**Cut line(s)** copies and deletes the selected I/O Scanning lines. The lines are copied without the **Device Name** information. For an ACS or PRA configuration, it permanently deletes the **Device Name** and all its data (and associated ACS symbols). For a **PRM Master DTM**, the link between the DTM and the **I/O Scanning** line is removed.
Copy line(s)	**Copy line(s)** copies the selected lines, but without the **Device Name**.
Paste line(s)	**Paste line(s)** has 2 actions depending on its target line: ● If the line is empty, it fills the line with the copied line (without a **Device Name)** ● If the line is not empty, it replaces the line with the copied line (without a **Device Name**). Be careful, it also permanently deletes the **Device Name** link to the **I/O Scanning** table and, for an ACS or PRA configuration, all its data (and associated ACS symbols) of the old line before replacing it with the copied line.

Menu Item	Description
Insert copied line(s)	**Insert copied line(s**) inserts the copied line between the selected line and the line just above it. Be careful with ACS or DTM configurations, all the lines below the inserted line become desynchronized. To synchronize these lines, open and close the device configuration tool, then do an **Update** from the Property box *(see page 217)*.
Insert empty line	**Insert empty line** inserts an empty line above the line selected line. Inserting an empty line does not desynchronize the devices below the line, but using this line for a new device can, depending the number of words needed, desynchronize the devices below the line.
Pack all lines	**Pack all lines** removes any empty lines between the top of the **I/O Scanning** table and the last non-empty of the table.

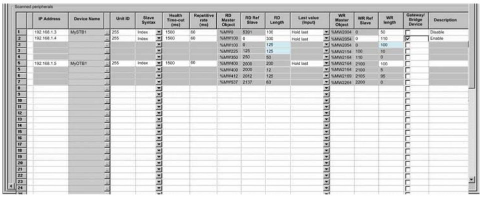
I/O Scanning with Multiple Lines

At a Glance

Modbus exchanges are limited to a maximum of 125 input words and 100 output words. If an application needs to exchange more than these limits for a device, more than one **I/O Scanning** line can be used: multiple lines for one device.

When the length is higher than the authorized limit for one Modbus exchange, the length is divided into 2 or more Modbus exchanges. New lines are created for each Modbus exchanges with the PLC.

The following **I/O Scanning** table is used for the multiple device lines example:



NOTE: This example shows an Advantys island, but DTM and PRA devices work the same way.

Multiple Line Length Configuration Example

In this example, the *first* (the main) **I/O Scanning** line 2 contains all the information for the exchanges with the device including the totals for the **RD length** and **WR length**.

The *second* line 2 contains the specific word lengths (125 and 100) needed so that it can also be used for part of the exchanges.

Line 2 needs a **RD length** of 300 word and a **WR length** of 110 words. How many extra lines are needed?
- **RD length** = 300/125 = 2.72 = 3 lines needed
- **WR length** = 110/100 = 1.10 = 2 lines needed

The larger of the 2 numbers is used:
- Three lines are needed to accommodate the **RD length**: 125 words, 125 words, 50 words for a total of 300 words
- The 3 lines for the **WR length** are: 100 words, 10 words, 0 words for a total of 110 words

The *second* line 2, line 3, and line 4 correspond to the Modbus exchange queries.

When multiple lines are used, only the **RD length** and **WR length** columns of these new lines can be edited. In the case of Advantys or DTM, the software supplies the **RD length** and the **WR length**, and they cannot be changed in the **I/O Scanning** table.

NOTE: It is not necessary to have a **Device Name** defined to use multiple lines.

The total number of words allowed in an **I/O Scanning** table is:
- 4 KW for Premium extended and Quantum networks
- 2 KW for Premium ETY and M340 NOE modules

Line Length for Multiple Word Variables

When using variables with 2 or more words, adjust the **RD** and **WR lengths** so that a variable is not partly on one **I/O Scanning** line and partly on the next. Because the 2 newly created lines result in 2 independent Modbus exchanges that can be sent non-synchronized to the device. The variables can receive the wrong values (if the 2 parts are received at different times). It may be necessary to use a **RD length** < 125 and a **WR length** < 100 for some of the scanned lines, in order to get each variable on only one exchange line.

⚠WARNING

UNEXPECTED SYSTEM BEHAVIOR

Verify that multiple word variables are completely on the same **I/O Scanning** line to avoid sending parts of a variable data in 2 non-synchronized **I/O Scanning** Modbus exchanges.

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

Introduction to Configuring Advantys from Unity Pro

At a Glance

The Advantys Configuration Software (ACS) is integrated in Unity Pro. This allows you to configure Advantys STB and OTB islands from the Unity Pro Ethernet I/O scanning tab.

Configuring an Advantys Island

⚠WARNING

UNEXPECTED SYSTEM BEHAVIOR

Always launch ACS from Unity Pro in order to synchronize variables and data between Unity Pro and ACS.

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

⚠WARNING

UNEXPECTED SYSTEM BEHAVIOR

Stop the PLC before transferring an ACS configuration and/or I/O Scanning modifications.

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

The following procedure configures an Advantys STB or OTB island:

Step	Action	Results
1	Open the Ethernet network configuration screen.	
2	In the **Module Utilities** zone select **YES** for **IO Scanning**.	
3	Select the **I/O Scanning** tab.	**I/O Scanning** screen opens.
4	Enter, on a free line, the IP address for the connection you want to use to communicate with the Advantys island.	
5	Enter **RD length** and **WR length** on the same line. The lengths must be long enough for the expected Advantys configuration.	
6	Validate the **I/O Scanning** screen.	
7	Click on the **...** button (that is next to **Device Name** cell on the same line).	The **Property** box *(see page 217)* opens.

Step	Action	Results
8	Select **STB** or **OTB** in the **Device Type** drop-down menu.	
9	Enter a **Device Name** (following the naming rules *(see page 220)*).	
10	You have 2 choices: **1.** If you want to go to ACS **now** to configure an island, click on the **Launch Advantys Configuration Software** button. Click on **Yes** in the “*Confirm device name and type*” Message Box and go to Step 11. **2.** If you want to configure the Advantys island **later**, click on the **OK** button. Click on **Yes** in the “*Confirm device name and type*” Message Box. To open the ACS later: ● Carry out Step 7. ● Click on the **Launch Advantys Configuration Software** button.	Results for both 1. and 2. are: ● A Unity Pro Message Box opens: “*The device name and device type won’t be modifiable. Do you want to confirm the device name and device type?*” ● The **Device Type** and **Device Name** are verified and saved. ● The **Property** box closes.
11	After ACS opens, configure your Advantys island. **NOTE:** While the ACS is open Ethernet screen is locked and cannot be edited, but the other Unity Pro services can be edited. **NOTE:** The `User Defined Label` must be filled in the `IO image`. If not, the Advantys varialble will no be added in the Unity Data Editor.	
12	When your Advantys island has been built and validated, close ACS.	A Unity Pro Message Box opens “*Do you want to update your symbols now?*”
13	You have 2 choices: **1.** Click on **Yes** in the “*update*” Message Box and go to Step 14. **2.** Click on **No** in the “*update*” Message Box. You are returned to the **I/O Scanning** screen without carrying out the **Yes** results. Later, when you want to update the Advantys symbols into Unity Pro: ● Carry out Step 7 ● In the **Property** box, click on the **Update** button and go to Step 14.	If you clicked on **No**: ● You are returned to **I/O Scanning** without carrying out the results in Step 14. ● The **Device Name** is displayed in the **I/O Scanning** in red. This indicates that the island configuration has not been synchronized with Unity Pro.

Step	Action	Results
14	Your Advantys island configuration is being **synchronized** with Unity Pro. After the synchronization is finished, you are returned to **I/O Scanning**. Verify that the **Device Name** is now displayed in black.	The results are: ● The Advantys island modifications are synchronized with the Unity Pro application. ● The Advantys island symbols are imported into the Unity Pro Data Editor. ● The Advantys **Device Name** is displayed in the **I/O Scanning** in black. This indicates that the island configuration is synchronized.
15	Build your Unity Pro application.	
16	STOP the PLC.	
17	Transfer: ● Unity Pro application to the PLC ● STB or OTB configuration to the Advantys island using ACS	
18	RUN your application in the PLC.	

Copy an Existing Island

This following procedure copies an existing Advantys island file (*.isl) into a new Advantys island configuration:

Step	Action
1	From Unity Pro, open a new Advantys island in ACS.
2	In ACS, select **File menu** →**Copy Island Contents**.
3	In the **Open island** window, select the island file (*.isl) to copy.
4	Click on **Yes** in the “Do you want to proceed?” message box.
5	The message “Island file has been saved.” in the **Log Window** verifies that the operation was successful.

Copy an Island File to a New Location

The following procedure copies an Advantys island file (*.isl) to a new directory:

Step	Action
1	In ACS, open an island configuration, for example, STB1.
2	Select **File** menu => **Copy STB1 Contents**
3	In the **Copy STB1.isl to** window, select the target directory.
4	The message “A copy of the island file has been saved with another name.” includes in the **Log Window** verifies that the operation was successful. The name is new because its path has changed.

Introduction to Configuring the PRM Master DTM

At a Glance

The **PRM Bus Master** uses the Unity Pro **I/O Scanner** to communicate with the CPU through an Ethernet port. This requires configuring the **PRM Master DTM** in the Unity Pro Ethernet **I/O Scanning** tab.

Configuring a PRM Master DTM

The following procedure configures a **PRM Master DTM** in the **I/O Scanner**:

Step	Action
1	Install the **PRM Master DTM** on the Host PC. **NOTE:** After installing new DTMs, the **Hardware Catalog** must be updated.
2	Add a **PRM Master DTM** to the connectivity tree in the DTM Browser using the contextual **Device menu** service.
3	In the DTM Browser, select the **PRM Master** and use the contextual **Device menu** function to open the DTM **PRM Offline Parameter** screen.
4	In the **General Setting** part of this screen set the IP address of the **PRM device.**
5	Open the **I/O Scanning** configuration editor (tab).
6	In the **Module Utilities** zone select **YES** for **IO Scanning**.
7	Select the **I/O Scanning** tab. **Results: I/O Scanning** configuration editor opens.
8	Enter, on a free line, the IP address for the connection to be used to communicate with the **PRM Bus Master**.
9	Set correct values for the **Read Ref.** and **Write Ref.** parameters.
10	Enter **RD length** and **WR length** for the IP address line (within the **Read Ref.** and **Write Ref.** constraints). **NOTE:** The lengths must be long enough for the expected configuration **PRM Master DTM** and its subnode DTMs.
11	Validate the **I/O Scanning** screen.
12	Click on the **...** button (next to **Device Name** cell). Results: The **Property** box *(see page 217)* opens.
13	Select **DTM** in the **Device Type** drop-down menu.
14	Select the protocol in the **DTM Protocol** drop-down menu.
15	Select a **PRM Master DTM** in the **DTM Name** drop-down menu.
16	Click on **OK** to validate the choices you made. **Results:** ● The **Device Type**, **Device Protocol** and **Device Name** are verified and saved. ● The **Property** box closes.

Step	Action
17	Update the **I/O Scanning** line, refer to Updating I/O Scanning for a PRM Master DTM *(see page 213)*.
18	Build the Unity Pro application.
19	Stop the PLC.
20	Transfer the Unity Pro application to the PLC.
21	In the DTM Browser, right click on **PRM Master** and select the **Connect** function.
22	In the DTM Browser, right click on **PRM Master** and select the **Store data to device** function.
23	**Run** the application in the PLC.

Updating I/O Scanning for a PRM Master DTM

The following procedure updates the **I/O Scanning** information for a **PRM Master DTM**:

Step	Action
1	Configure and validate the **PRM Bus Masters** in the DTM Browser using the contextual **Device menu** function.
2	Open the **I/O Scanning** configuration editor (tab).
3	Click on the **...** button (that is next to the **Device Name** of the **PRM Master DTM** to update).
4	In the open **Property** box *(see page 217)*, click on the **Update** button. **Results:** ● The **PRM Master DTM** modifications are synchronized with the Unity Pro application. ● The **PRM Master DTM** symbols are imported into the Unity Pro Data editor. ● The **DTM Name** is displayed in the **I/O Scanning** configuration tab in black. This indicates that the PRM configuration is synchronized. ● The **Property** box closes.

Introduction to Configuring a BMX PRA 0100 from Unity Pro

At a Glance

Unity Pro allows configuration of BMX PRA 0100 modules through the Ethernet I/O scanning tab. The PRA device configuration is done in a **second** instance of Unity Pro.

Configuring a PRA

The following procedure configures a PRA device:

Step	Action	Results
1	Open the Ethernet network configuration screen.	
2	In the **Module Utilities** zone select **YES** for **IO Scanning**.	
3	Select the **I/O Scanning** tab.	**I/O Scanning** screen opens.
4	Enter, on a free line, the IP address for the connection you want to use to communicate with the **PRA**. **NOTE:** The IP address in the **I/O Scanning** table must be the same as the IP address of the **PRA** device.	
5	Enter **RD length** and **WR length** on the same line.	
6	Validate the **I/O Scanning** screen.	
7	Click on the **...** button (that is next to **Device Name** cell on the same line).	The **Property** box *(see page 217)* opens.
8	Select **PRA** in the **Device Type** drop-down menu.	
9	Enter a **Device Name** (following the naming rules *(see page 220)*).	

<table>
<tr><th>Step</th><th>Action</th><th>Results</th></tr>
<tr><td>10</td><td>You have 2 choices:
1. If you want to now configure a PRA, click on the Launch PRA button.
Click on Yes in the “Confirm device name and type” Message Box and go to Step 11.

2. If you want to configure a PRA later, click on the OK button.
Click on Yes in the “Confirm device name and type” Message Box.
NOTE: The Device Name becomes red in the I/O Scanning table. This indicates that a PRA has not been configured for the table line that contains the Device Name

To configure a PRA later:
● Carry out Step 7.
● Click on the Launch PRA button.

NOTE: While the second PRA instance of Unity Pro is running no changes can be made to the Ethernet Editor in the first (master) instance of Unity Pro.</td><td>Results for both 1. and 2. are:
● A Unity Pro Message Box opens: “The device name and device type won’t be modifiable. Do you want to confirm the device name and device type?”
● The Device Type and Device Name are verified and saved.
● The Property box closes.</td></tr>
<tr><td>11</td><td>After the second instance of Unity Pro opens:
● File menu →Open
● Change the file type to .XEF
● Open the PRA application template, PRA_Template.XEF</td><td></td></tr>
<tr><td>12</td><td>When your PRA application is configured:
● If desired, you can build the PRA application now.
● Save the application.
NOTE: The Save As function is not available. To copy your PRA application use the Export or Save Archive function.

● Close this instance of Unity Pro.
NOTE: You are asked if you want to save the PRA application in the master application *.stu file. If you select No, all changes are lost.

NOTE: Later you can build your PRA application by carrying out Step 7. Because the PRA application is saved (embedded) in the master application *.stu file, it is opened. You can then build the PRA application.

NOTE: If there is no PRA application in the master *.stu, an empty application is opened (as happens the first time the Launch PRA button is used in the Property box)</td><td></td></tr>
<tr><td>13</td><td>Build your Unity Pro application.</td><td></td></tr>
<tr><td>14</td><td>STOP the PLC.</td><td></td></tr>
</table>

Step	Action	Results
15	Transfer: ● Unity Pro application to the PLC ● **PRA** configuration to the PRA device **NOTE:** There are no imported variables, the user must ensure the synchronization of the data exchange.	
16	RUN your application in the PLC.	

NOTE: When the second (**PRA**) instance of Unity Pro is closed, there is no indication if the **PRA** application has been built or not.

Copy an Existing PRA Application

This following procedure copies an existing **PRA** application:

Step	Action
1	From the Unity Pro **I/O Scanning** table using the **...** button, open an existing **PRA** application.
2	In the second Unity Pro instance, save the existing **PRA** application with a new name as a *.sta or .xef file.
3	Close this second Unity Pro instance.
4	In the Unity Pro **I/O Scanning** table create a new **PRA** application on a new line.
5	Import or Open the *.xef or *.sta file previously saved.
6	If desired, build the new **PRA** application and transfer it to the **PRA** device.
7	Close the second Unity Pro instance.

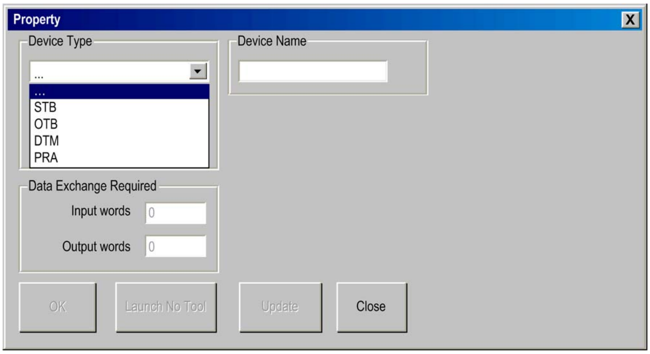
Property Box

At Glance

The **Property** box is the link between Unity Pro and a device configuration tool. It is used to select and name a device and to launch the configuration tool for the device.

Property Box

The following illustration is the **Property** box before selecting the **Device Type**.

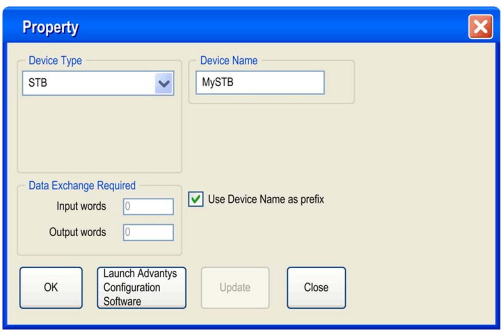


For details on how top use the **Property** box, refer to:
- Advantys *(see page 217)*
- DTM *(see page 220)*
- BMX PRA 0100 *(see page 219)*

Property Box for Advantys

This **Property** box allows you to choose the name and type of Advantys island to be configured using the Advantys Configuration Software (ACS).

The following illustration is the **Property** box for Advantys *after* **Device Type** and **Device Name** entered:



Property Box for Advantys Elements

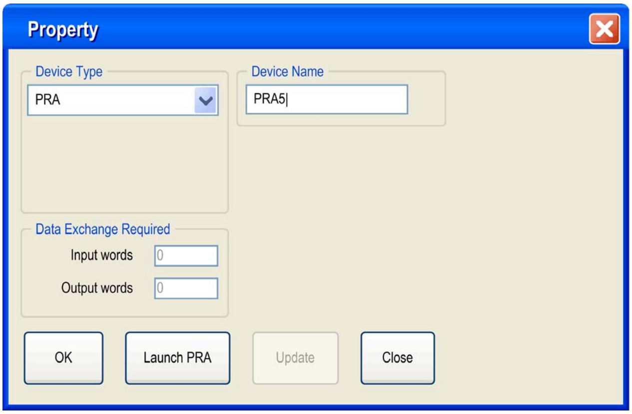
The elements of the Advantys **Property** box are:

Element	Description
Device Type	Selection of a STB or OTB device is made from this drop-down list. After the first validation (using the **OK** or **Launch Advantys** button) the **Device Type** cannot be changed.
Device Name	The **Device Name** *(see page 220)* is used as a prefix to all variables created for an Advantys island in ACS. This allows unique variables for duplicated islands. After validation (using the **OK** or **Launch Advantys** button) the **Device Name** cannot be changed.
Data Exchange Required	These are the minimum number of words necessary for communication between the Unity Pro module and the Advantys island. These values cannot be changed via the **Property** box. **NOTE:** Increasing the RD/WR lengths via the **I/O Scanning** tab leaves enough data exchange words for the future expansion of your Advantys island. Expanding an island that does not use the last line in the **I/O Scanning** table requires changing the values for all the lines below the line that needs the additional exchange words.
Use Device Name as prefix	If this checkbox is unchecked, the user is in charge of giving unique names to the variables and symbols in **all** Advantys islands. This checkbox is only available for ASC V5.5 or higher. For versions less than 5.5 the Device Name is automatically added to all variables and symbols in all Advantys islands.
OK	This button is only available after entering the **Device Type** and **Device Name**. When clicked, the **Device Type** and **Device Name** are checked to see if they are valid. If there is a problem, a message box opens explaining the why they were not valid. **OK** is only available during the first use of the **Property** box for a new island.
Launch Advantys Configuration Software	This button is only available if both: ● The **Device Type** and **Device Name** have been entered ● ACS is installed This button does two things: ● It carries out the action of the **OK** button ● If there is no problem during validation, it launches ACS
Update	When clicked, the ACS modifications are synchronized with your Unity Pro application (after these modifications have been validated in ACS). It also imports and updates all ACS symbols and variables into the Unity Pro variable manager. **NOTE:** All variables modified in ACS are deleted and rewritten in the Unity Pro Data Editor. But they are not updated in the program.
Close	This button closes the **Property** box without saving anything.

Property Box for BMX PRA 0100

This **Property** box allows you to choose the name for the PRA module to be configured.

The following illustration is the **Property** box for the PRA *after* **Device Name** validation:



Property Box for PRA Elements

The elements of the PRA **Property** box are:

Element	Description
Device Type	Selection of the PRA device is made from this drop-down list. After the first validation (using the **OK** or **Launch PRA** button) the **Device Type** cannot be changed.
Device Name	The **Device Name** *(see page 220)* is the name of PRA application.
Data Exchange Required	This is not used when configuring a PRA device.
OK	This button is only available after entering the **Device Type** and **Device Name**. When clicked, the **Device Type** and **Device Name** are checked to see if they are valid. If there is a problem, a message box opens explaining why they are not valid. **OK** is only available during the first use of the **Property** box for a new PRA configuration.
Launch PRA	This button is only available if the **Device Type** and **Device Name** has been entered. This button does two things: ● It carries out the action of the **OK** button ● If there is no problem during validation, it launches another instance of Unity Pro, which is used to do the actual configuration of the PRA.
Close	This button closes the **Property** box without saving anything.

Valid Name

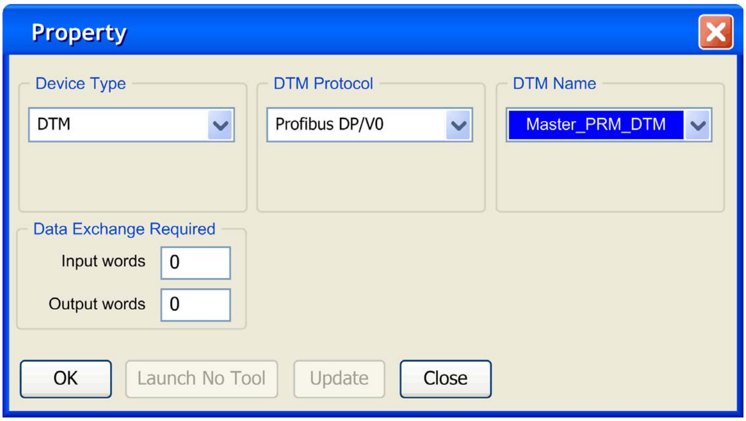
A valid **Device Name** for a configuration:

- Does not already exist in the application
- Is not a empty name
- Starts with a letter
- Has a maximum of 8 characters
- Only ASCII characters, not Unicode characters
- Has no spaces
- Follows the Windows file naming conventions: no slashes, question marks, etc.
- Follows Unity Pro variable naming conventions

Property Box for a PRM Master DTM

This **Property** box allows you to choose the type and protocol for a **PRM Master DTM**:

The following illustration is the **Property** box *after* selecting the **Device Type**, **DTM Protocol** and **DTM Name**:



Property Box PRM Master DTM Elements

The elements of the DTM **Property** box are:

Element	Description
Device Type	Selection of DTM device type is made from this drop-down list.
DTM Protocol	Select the protocol to be used from this drop-down list. This list contains the DTM protocols of all the DTMs in the DTM Browser that can be linked with I/O Scanning.
Device Name	Select a **PRM Master DTM** from this drop-down list.This list uses the DTM Browser Alias names. This list contains all the **PRM Master DTMs** in the DTM Browser that support the selected DTM Protocol. To validate the choices, click on the **OK** button.
Data Exchange Required	These are the minimum number of words necessary for communication between Unity Pro and the **PRM Master DTMs**. These values cannot be changed via the **Property** box. **NOTE:** Increasing the RD/WR lengths via the **I/O Scanning** tab leaves enough data exchange words for the future expansion of your DTM topology tree. Expanding a tree that does not use the last line in the **I/O Scanning** table requires changing the values for all the lines below the line that needs the additional exchange words.
OK	The **OK** button is only available after selecting the **Device Type**, **DTM Protocol** and **DTM Name**. When clicked, the **DTM Protocol** and **DTM Name** are checked to see if they are valid. If there is a problem, a message box opens explaining the why they were not valid. The **OK** button is only available during the first use of the **Property** box for a new **PRM Master DTM**.
Launch No Tool	This button is never available for **PRM Master DTMs**.
Update	Use the **Update** button after validating or changing the configuration of the linked **PRM Master DTM**. Refer to Update I/O Scanning for a PRM Master DTM *(see page 213)*.
Close	The **Close** button closes the **Property** box without saving anything.

Saving an Advantys Configuration in an Unity Pro Application

At a Glance

ACS saves an island configuration in an *.isl file. To add the island to an application, it is necessary for Unity Pro to know the location of the island configuration information.

Saving the Configuration

The recommended way to save your island configuration information is to save your Unity Pro application as a *.stu or *.sta file. The *.isl file is automatically included in these files.

Uploading or Importing

There are 2 situations where the information contained in the *.isl file is not available:

1. Uploading the application running in the PLC
2. Importing an *.xef file

In these 2 cases, if ACS is launched from the **Property** box *(see page 217)*, it automatically tries to open the latest **Device Name**.isl file the Unity Pro **General Path** => **Project Path** directory:

- If the same PC is used for the import (upload) and export (download) and the Unity Pro **Project Path** has not changed, the island configuration is synchronized with ACS.
- If the same PC is not used for the import (upload) and export (download) or if the Unity Pro Project Path has changed, either:
  - Create a new island
  - Use the **File** menu => **Copy Island Contents** function
    NOTE: The new **Device Name**.isl file is copied to the **Project Path** directory.

Managed Variables

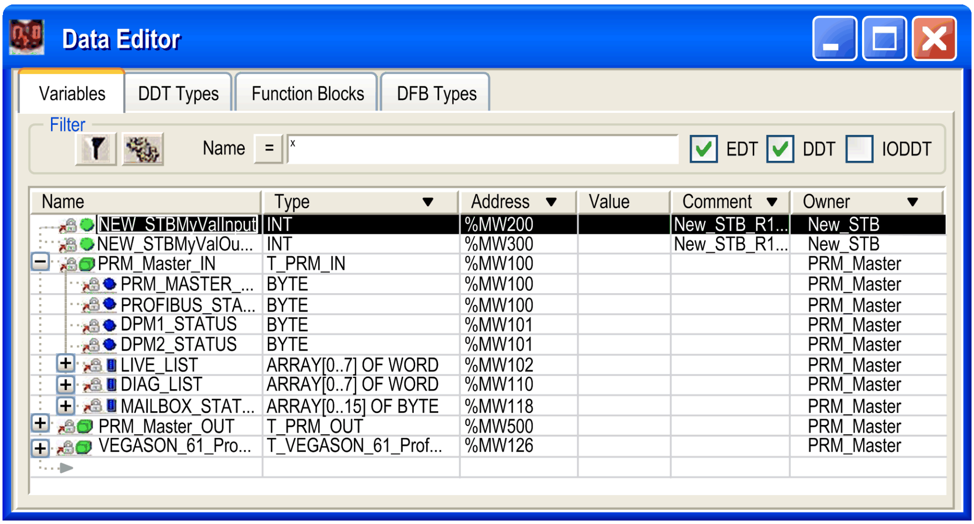
At a Glance

Variables of devices that are linked to Unity Pro through **I/O Scanning** or **DTM** are *managed variables*. They are created by the device configuration tool or by the DTM and are imported into Unity Pro. They are named as a concatenation of the Property box *(see page 217)* **Device Name** + device symbol name.

Managed Variables in the Data Editor

Advantys symbols become managed variables when imported into Unity Pro. An Advantys managed variable name is a concatenation of:
the Advantys island name + Advantys symbol name.

This **Data Editor** illustration shows managed variables with their device name **prefixes** and their **Owner** attributes:



Managed variables follow the usual rules for Unity Pro and ACS or DTM naming.

The optional **Owner** column lists the owner attribute of the managed variables. This allows you to filter the variables according to their **Device Name**.

The Unity Pro managed variables are locked and cannot be modified through the **Data Editor**. You have to use the device configuration tool (ACS or the DTM) to modify these variables.

Importing Managed Variables from a Device (Advantys, DTM)

Using the **Update** button on the Property box *(see page 217)* imports the device Symbols into the Data Editor as Unity Pro managed variables.

In the case of a conflict between an device Symbol and an existing variable in the Data Editor:
- If the Unity Pro variable *is not* managed, a message box allows you to replace this variable with the managed variable coming from the ACS- or DTM-controlled device).
- If the Unity Pro variable *is already* managed, the update is cancelled.

For an already managed variable, there are 2 options, either:
1. Use the device configuration tool (ACS or the DTM) to rename the variable.
2. Delete the old managed variable using the tool that manages the variable, then use the tool to perform an Update.

After performing one of these options, use the **Update** button again on the device being updated to complete the import without a conflict.

Permanent Deletion of a Managed Variable

Managed variables cannot be deleted directly from the Data Editor.

Removing a managed variable from a configuration must be done from the tool (ACS or the DTM) that manages the device (either delete the device using the DTM or delete the Symbol using the ACS).

NOTE: During an **Update**, all managed variables are deleted and recreated during synchronization between Unity Pro and the device.

Partial Import of a Managed Variable

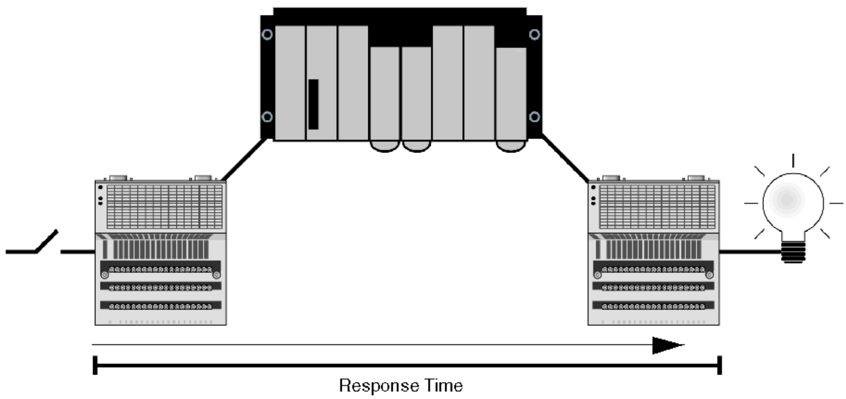
Starting with Unity Pro V5.0, the managed variables become *unmanaged* during a partial import from an .XSY file. This allows deletion of the variables if the linked device is not also imported.

After importing variables from an .XSY file, an **Update** is needed to resynchronize the managed variables linked to a device. During this **Update**, a conflict box appears to allow validation of the replacement managed variables.

I/O Scanner Response Times: Remote Input to Remote Output

Measurement Setup

The set of curves below illustrates Quantum PLC response times when a signal is sent from a remote input module to a remote output module through a Quantum PLC:

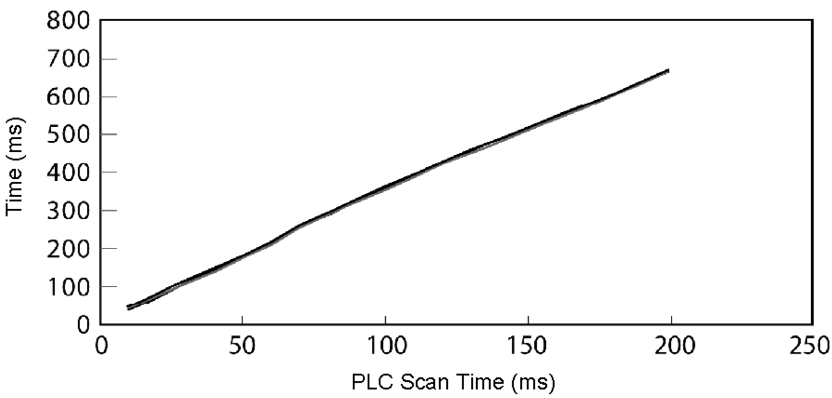


The signal is:

1 triggered by a Momentum input module with a response time of ~2 ms
2 scanned into the Quantum PLC at a repetitive rate of 0 ms
3 copied to another internal variable within the PLC
4 written to a Momentum output module with a response time of ~2 ms

140 CPU 434 12A with 140 NOE 771 x1 Module

The 140 CPU 434 12A with an NOE 771 x1 (v3.3) was used for the following measurements:



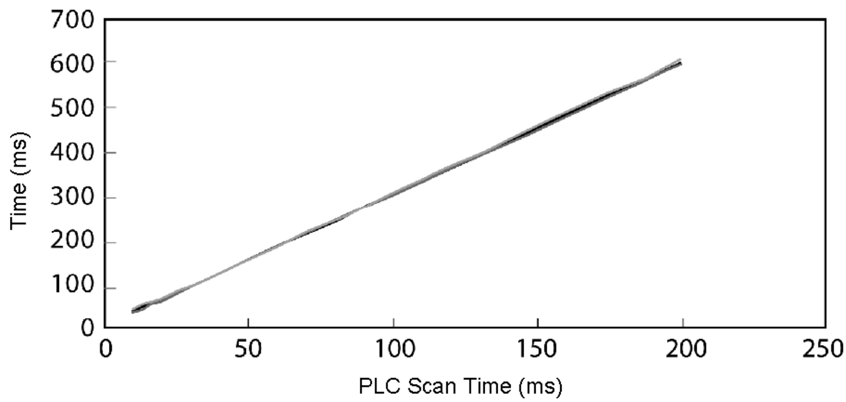
The plot shows four lines representing the number of scanned devices:
- 1 device
- 8 devices
- 16 devices
- 32 devices

The four lines are indistinguishable at this scale because the response times are so similar. You can see the similarity in the following table of data from which the above graph was plotted:

Number of Devices to Scan	Time from Scanned Device Input to Scanned Device Output (ms)				
434 12A + NOE 771 x1 (v3.3)	10 ms scan	20 ms scan	50 ms scan	100 ms scan	200 ms scan
1 device	41	73	179	358	665
8 devices	42	75	180	360	666
16 devices	44	77	182	361	668
32 devices	46	79	185	364	671

140 CPU 65 150 with NOE 771 x1 (v3.3)

The 140 CPU 65 150 with an NOE 771 x1 (v3.3) is used used for the following measurements:



The plot shows four lines representing the number of scanned devices:
- 1 device
- 8 devices
- 16 devices
- 32 devices

The four lines are indistinguishable at this scale because the response times are so similar. You can see the similarity in the following table of data from which the above graph was plotted:

Number of Devices to Scan	**Time from Scanned Device Input to Scanned Device Output (ms)**				
65150 + NOE 771x1 (v3.3)	10 ms scan	20 ms scan	50 ms scan	100 ms scan	200 ms scan
1 device	35	61	153	302	602
8 devices	36	62	154	303	603
16 devices	38	64	155	305	606
32 devices	40	66	157	307	609

Chapter 7
Address Server Configuration/Faulty Device Replacement

Introduction

This chapter covers the Address Server Configuration/Faulty Device Replacement service available on the NOE 771 01 and -11 (Transparent Factory/Real Time modules) and HE CPU modules. The Faulty Device Replacement service offers you a method of handling device replacement without disrupting the system or service.

What Is in This Chapter?

This chapter contains the following topics:

Address Server Configuration/Faulty Device Replacement

Overview

The address server provides two capabilities:

1. **Standard BOOTP server behavior:** Enter the MAC address and IP configuration. The NOE BOOTP server will provide the IP configuration when the device sends a BOOTP request.
2. **Faulty Device Replacement (FDR) behavior:** Enter the role name or the MAC address of the device. The device will send its role name or the MAC address with its DHCP request. With the DHCP response from the NOE, the device will receive its IP Configuration, plus the name and location of a configuration file.
   The next step for an FDR-compliant device is to download its configuration from the NOE.

NOTE: Consult your Schneider Electric sales representative for the current list of FDR-compliant devices.

The address server in the NOE supports both modes at the same time. You select a mode by entering either the MAC address or the role name in the Address Server Node Configuration *(see page 294)* page. You may enter only one or the other, but not both.

The Faulty Device Replacement capability allows automatic configuration of FDR-compliant devices.

Identifying a Role Name

Role names play an important role in Faulty Device Replacement. A role name is a logical name that the user assigns to a device, a logical name that has a meaning within the application.
Example role names might be:

- **ENT_6**: The sixth Momentum ENT in your application.
- **OUTPUT_VALVE_2**: The second output valve in your application

NOTE: Role names are case-sensitive.

Role Name

The logical role name should be written on devices. The technician can get the new device from stores, enter the corresponding role name into the device, and place the device in the system. The device automatically gets its configuration and starts running with no further input from the technician. This process is designed to get your machine up and running quickly. All the technician has to do for any FDR compliant device is to enter the role name into the new device.

Address Server Limits

This table displays the parameters and limits of the address server:

Parameter	Limit
Maximum number of address server entries	128
Maximum size of the configuration file per device	4K bytes
Total size of Faulty Device Replacement storage	512K bytes
Maximum role name size	16 Characters

NOTE: For the DHCP server to work correctly the following must be observed.

- Address class and subnet class configured for the devices must match.
- Address class of the NOE and of the devices must be the same.

Operating on a Corporate Network

Keep these points in mind when operating on a corporate network:

- Before placing the NOE on a corporate network, Schneider Electric recommends that you discuss the installation with your MIS department. It is likely that your company's corporate network has at least one DHCP server running already. If the NOE's DHCP server is running on the same network, it may disturb the network.
- To avoid any possible problem related to the NOE's DHCP server on the corporate network, you have to ensure that the DHCP server is not running in the NOE by not having address entries in the configuration. If there are no configured devices in the address server Configuration page, then the NOE will not start the DHCP server.

Available FDR Agents

Three FDR agents are available:

- Momentum ENT
- Micro ETZ
- ATV58

The `role-name.prm` configuration files are stored in the NOE in non-volatile memory. Therefore, after a power failure all configurations will be available.

BOOTP and DHCP Compatible Devices

Use either the MAC address or the role name *(see page 295)* to assign IP addresses. Therefore, you may use the DHCP server with devices that support BOOTP only, such as Momentum ENT v1.

Understanding Faulty Device Replacement

Understanding Confirmation and Error Messages

In addition to highlighting errors, the system provides confirmation information and error messages.

Confirmation Message If you successfully added, modified, or removed and entry, the following alert message appears:

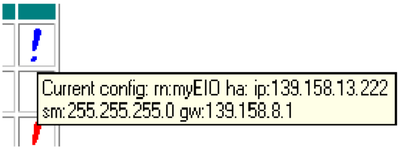
Error Messages Error messages, appearing as icons in the seventh column, display on the Address Server Configuration page *(see page 294)*, or in a dialog box.

Error Icon If an entry is not loaded in the DHCP server or loaded with a different configuration, an icon of an exclamation point displays in the seventh column. The icon informs you of the difference between the current and stored information.

- Not loaded in the server: A red icon displays. 
- Duplicate configuration: A blue icon displays. 

Place the pointer over the icon and a pop-up window appears and displays a message

- Not Loaded in the server:



- Present configuration:

Error Dialog Box If you entered an existing role name or MAC address, you will receive an alert message asking you to correct the entry.

Modifying the Database

If you need to add or modify an entry in the database, use the Address Server Configuration page *(see page 294)*. Three fields must be filled in:

- Device IP address
- Subnet Mask
- Gateway

Choose either the **Role Name** or **Device MAC address** field. When one field is selected, the other is made unavailable.

Adding Entries If you are adding a device, the page appears with values. You need to enter either a Role Name or a MAC Address.
If you are adding an entry, submit your selection using the **Add the Entry** button.

Modifying Entries If you are modifying an entry, the **Device IP address**, **Subnet Mask**, and **Gateway** fields display with the current configuration.
If you are modifying an entry, submit your selection using the **Reset the Form** button.

Each field of the **Address Server Node Configuration** page has restrictions, which are detailed below.

- **Role Name:** Each role name must be unique. Only letters, numerals, and underscores are allowed. No more than 16 characters are allowed, and no spaces are allowed.
- **Device MAC Address:** This address must be in hexadecimal format and six (6) bytes long (6x2 characters). The MAC address can be entered with or without a delimiting character between each pair of lower- or upper-case hexadecimal characters. The delimiting characters improve readability. There are three choices for a delimiting character:
  - Space " " (spacebar)
  - Colon ":"
  - Hyphen "-"
- **Device IP Address:** The IP address must use the standard numeric Internet Protocol address that uniquely identifies each computer on the network. The IP address is a 32-bit identifier made up of four groups of numbers (from 0 through 255), each number separated by a period, for example `123.456.78.9`.
- **Subnet Mask:** The subnet mask must be entered in IP address format.
- **Gateway:** The gateway must be entered in IP address format. The gateway must be on the same subnet as the device.

Configuring Faulty Device Replacement

Configuring the Address Server

To configure the Address Server you use Web pages generated by the embedded Web server. The first page that appears is the **Address Server Configuration** page. The first column contains buttons used for selecting devices. The **Address Server Configuration** page displays configuration information for each device in the system and has seven columns in the table.

Displayed on this page is information about:
- Role Name
- MAC Address
- IP Address
- Subnet Mask
- Gateway

An additional, unnamed column indicates any difference between the current and stored configuration. If a difference exists, an exclamation point is displayed.

This is the **Address Server Configuration** page. In this sample page, the configured and stored configurations are the same for each device.

Address Server Configuration

	Role Name	MAC Address	IP Address	Subnet Mask	Gateway	
○	myNIP		192.168.3.11	255.255.255.0	192.168.3.11	
○	ENT_1st_floor		192.168.5.14	255.255.0.0	192.168.2.1	

Refresh Address Server Database Table

Add a New Entry | Change an Entry | Delete an Entry

On the Address Server Configuration page you can:
- Add a New Entry
- Change an Entry
- Delete an Entry
- Refresh the data table

Choosing Options

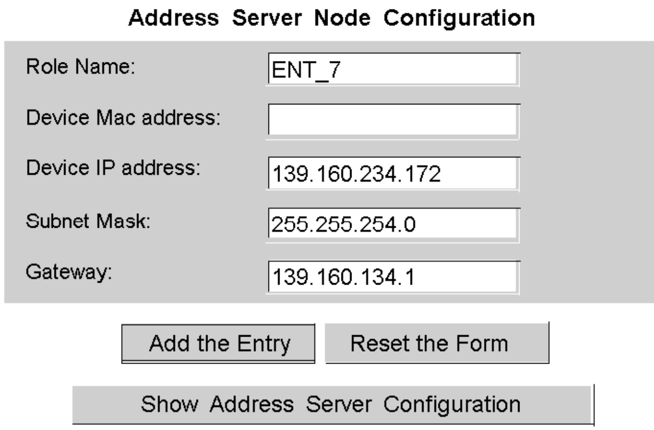
The Address Server Configuration page allows you to choose different options for adding or altering the configurations of your NOE. The options available to you are:
- select an entry
- add an entry
- change an entry
- delete an entry

Below we describe the method and options chosen to perform any of the four options listed above. Screen shots are presented to accompany the *Adding an entry* section.

Selecting an Entry When the page displays, by default no entries are selected. Use the radio buttons in the first column to select an entry. Only one entry may be selected at a time.

Adding an Entry When the **Add a New Entry** button is selected, the **Address Server Node Configuration** page appears. This page displays information about a device.
If you selected a device, this page displays the device's configuration. Configuration information appears in four of the five fields of the dialog box. Only the Role information field is blank. You should enter a Role name, for example *ENT_7*.
If no device is selected, default values appear in each field.

Changing an Entry Before using this button, you must select an entry in the database by choosing one of the radio buttons in the first column. If you do not choose an entry, a message appears.
When the **Change an Entry** button is selected, the **Address Server Node Configuration** page appears. The information displayed is for the device selected.



Deleting an Entry Before using this button, select an entry in the database by choosing one of the radio buttons in the first column. If you do not choose an entry, a message appears.
The entry selected will be removed from the database. Before completely deleting an entry, a warning message appears. Click **Yes** if you want to delete the entry, **No** if you don't.

If you click **Yes**, a dialog box appears.

Click **OK**. Another dialog box appears notifying you that the deletion was successful.

Highlighting Anomalies

If there are anomalies with the entered configuration parameter information, the system indicates anomalies using a highlighting mechanism. Valid device configurations appear in purple and italic. A device with configuration anomalies appears in red and bold.

The system detects the following anomalies:

- Bad **Role Name**
  The valid **Role Name** characters are:
  - a through z (lower case)
  - A through Z (upper case)
  - "_" (underscore)
- Bad **MAC Address**
  The valid **MAC Address** characters are:
  - 0 through 9
  - A through F
- Wrong **IP Address**
- Wrong **Subnet Mask**
- Wrong **Gateway**
- Double Entry
  Each entry needs a unique **Role Name** or **MAC Address**. If a duplicate **Role Name** or **MAC Address** is entered, the system highlights the duplication.

Erroneous entries are not loaded into the DHCP server. Therefore, correct erroneous before loading. There are two ways of correcting the erroneous entries:

- **Correcting through Web page:** Make the changes on the Web page and submit the change.
- **Correcting through the Address Server configuration file:** Make the changes in the file and reboot the server.

Chapter 8
Network Management Service (SNMP)

Introduction

The following material describes SNMP and the Schneider private MIB. Under the Schneider private MIB is the Transparent Factory Ethernet private MIB.

What Is in This Chapter?

This chapter contains the following topics:

SNMP

Introduction

This topic describes the Simple Network Management Protocol (SNMP), which is configured on your NOE or 140 CPU 651 x0. Network management software allows a network manager to:

- monitor and control network components
- isolate problems and identify their causes
- query the status of devices such as a host computer, routers, switches, and bridges
- obtain statistics about the networks to which devices are connected

Manager/Agent Paradigm

Network management software follows the conventional client-server model. To avoid confusion with other network communication protocols that use the client/server terminology, network management software uses the following terms:

- *manager*: the client application that runs on the manager's computer
- *agent*: the application that runs on a network device

The manager uses conventional transport protocols (for example, TCP or UDP) to establish communication with the agent. Managers and agents then exchange requests and responses according to the network management protocol.

Simple Network Management Protocol

Your NOE module is configured with SNMP, which is the standard protocol used to manage a local area network (LAN). SNMP defines exactly how a *manager* communicates with an *agent*.

The SNMP defines the format of the requests that a manager sends to an agent and the format of the replies that the agent returns to the manager.

The MIB

Each SNMP object has to be defined and given a unique name. Both the manager and agent programs must agree on the names and the meanings of the fetch and store operations. The set of all objects SNMP can access is known as a *Management Information Base (MIB*).

Private MIB

Schneider obtained a private MIB, Groupe_Schneider (3833). Under the Groupe Schneider private MIB is a Transparent Factory Ethernet (TFE) private MIB. The Transparent Factory SNMP embedded component controls the Schneider private MIB function.

Choosing an SNMP Manager

If you already have a working SNMP manager, you may continue to use it. Any of the many SNMP version 1-compliant managers on the market will work.

If you do not currently use an SNMP manager in your organization and are evaluating SNMP managers for purchase, then we recommend that you consider HiVision with the ConnexView Add-On developed for use with Schneider Electric PLCs.

Please contact your Schneider Electric sales office for availability and pricing of HiVision and ConnexView.

Using an SNMP Manager

The process for obtaining an SNMP manager:

Step	Action
1	Get Schneider `.mib` file from the NOE Web page. Find the `.mib` file as a packed file under **/wwwroot/SchneiderTFE.zip** on your NOE module.
2	Load `.mib` file to the SNMP manager.
3	When you are done, you will see the Schneider private MIB manager in your manager.

Other SNMP Resources

SNMP and related subjects are well-documented on Web sites and in many books:

- As of this writing, a useful description appears on Microsoft's *Technet* pages. Browse to *http://www.microsoft.com/technet*. Use the **Search** function to find "Network Management for Microsoft Networks Using SNMP."
- Use an Internet search engine to search for an SNMP introduction, tutorial, or other resource.
- The SNMP FAQs from the news group `comp.protocols.snmp` appear on many `.com` and `.org` Web pages. Search for the combination of "`comp.protocols.snmp`" and "`FAQ`."

ASN.1 Naming Scheme

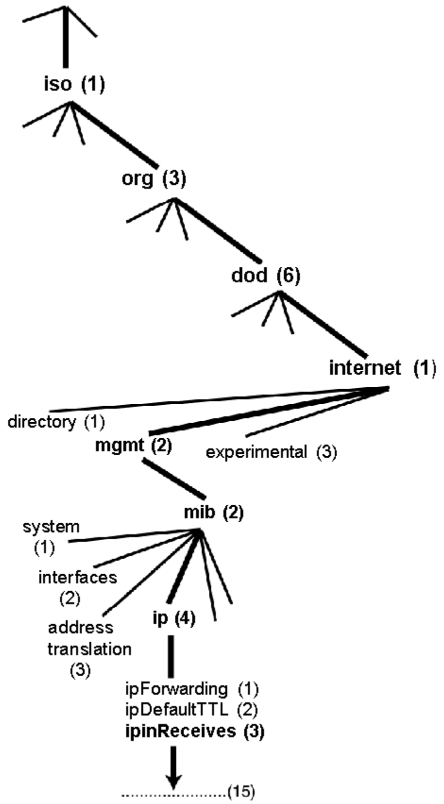
ASN.1 Overview

Abstract Syntax Notation One (ASN.1) is a formal language for abstractly describing messages to be exchanged between distributed computer systems.

An Example

Objects in a MIB are defined with the ASN.1 naming scheme that assigns each object a long prefix that guarantees that the name will be unique. For example, an integer that counts the number of IP datagrams that a device has received is named: *iso.org.dod.internet.mgmt.mib.ip.ipinReceives*.

The following figure depicts the ASN.1 naming scheme example.



This object name is represented in an SNMP message by assigning each part an integer. So, the above message would appear as 1.3.6.1.2.2.4.3.

Each integer has the following meaning:

- 1 = ISO (International Organization for Standardization)
- 3 = identified organization (one of branches under the ISO root)
- 6 = U. S. Department of Defense (DoD) (one of the children under branch1.3)
- 1 = the Internet subtree under 1.3.6
- 2 = the mgm branch — (one of seven) of the Internet subtree. It is managed by the Internet Assigned Numbers Authority, and includes the standard MIBs
- 2 = mib-2(1) group of managed objects
- 4 = ip (the mib-2(1) IP group (one of 11))
- 3 = ipinReceives (the MIB object)

Configuring an NOE with SNMP

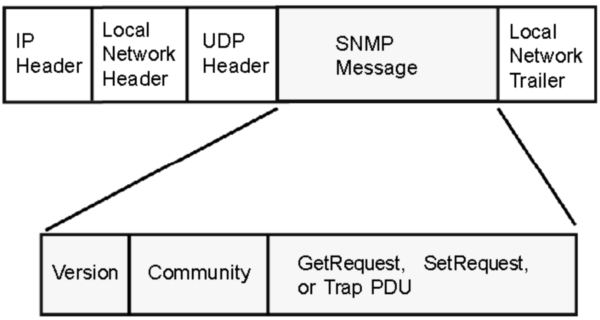
Object Identifier (OID)

In the ASN.1 Naming Scheme example *(see page 240)*, the MIB object identified by the notation 1.3.6.1.2.2.4.3 is referred to as the Object Identifier or OID. All OIDs can be seen as part of a tree structure that begins at the root (ISO) and branches out with each subtree identified by an integer.

SNMP Protocol Data Units

SNMP uses protocol data units (PDUs) to carry the requests and responses, between the manager and the agents, for the information contained in an OID.

As the following figure shows, the SNMP message is the innermost part of a typical network transmission frame.



The PDUs within the SNMP initiate the communication between the manager and the agents.

The SNMP installed on your NOE module uses the following three PDUs:
- GetRequest
- SetRequest
- Trap

GetRequest PDU

The GetRequest (shortened to Get) PDU is used by the SNMP manager to retrieve the value of one or more objects (OIDs) from an agent.

SetRequest PDU

The SetRequest (shortened to Set) PDU is used by the SNMP manager to assign a value to one or more objects (OIDs) residing in an agent.

Trap PDU

The Trap PDU is used by the agent to alert the manager that a predefined event has occurred.

Version & Community Identifiers

The version identifies the version number of the SNMP software being used by the manager and the agent. Your NOE supports Version 1 of the SNMP. The community is an identifier that you assign to your SNMP network. If community names for the manager and the agent do not agree, the agent will send an authentication failure trap message to the manager. If the community names and version number agree, the SNMP PDU will be processed.

What Can Be Configured

Your NOE module can be configured to send an authentication trap to two SNMP managers if it receives a community name in a Get/Set request that does not match the configured name. Also, you can configure the SysContact and SysLocation via the configuration page in the module's embedded web pages. After making changes in the SNMP Configuration Web page and to set those changes, reboot the module using hot swap.

Configuring an NOE with TFE Private MIB

Introduction

A MIB, a Management Information Base, is an element used in network management. Network management services are based on the need to monitor and manage:

- performance
- fault occurrences
- security

Each MIB contains a finite number of objects. Manage your MIB with a management station running an SNMP management application. The management application uses **GETs** and **SETs** to retrieve system information and to set system environment variables.

Schneider Private MIB

Schneider Electric obtained a Private Enterprise Number (PEN) from the Internet Assigned Numbers Authority (IANA). That number represents a subtree in the SNMP MIB, a number that is a unique identifier used for Groupe Schneider.

The object identifier for the root of the Groupe Schneider subtree is `1.3.6.1.4.1.3833` and represents a path to the subtree as follows:

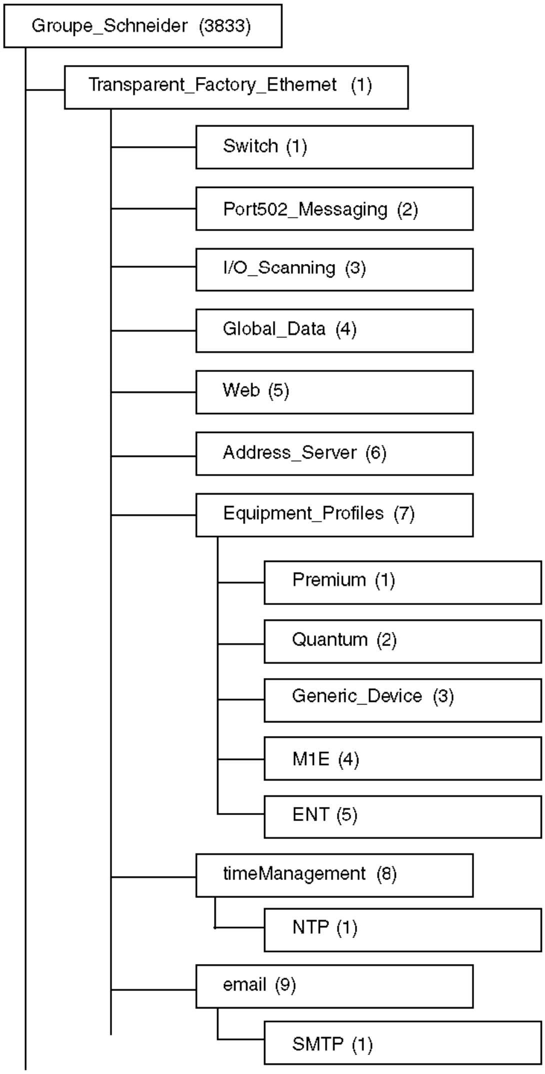
```
iso(1)
    org(3)
        dod(6)
            internet(1)
                private(4)
                    enterprise(1)
                        GroupeSchneider(3833)
                            Transparent_Factory_Ethernet(1)
```

Under the GroupeSchneider private MIB is a Transparent Factory Ethernet (TFE) private MIB, **Transparent_Factory_Ethernet(1)**.

TFE Private MIB

The Transparent Factory SNMP-embedded component controls the Schneider private MIB function. The Schneider private MIB, and associated services, perform Network Management on all system components. The Transparent Factory private MIB provides the data to manage the main Transparent Factory communication services for all the communication components of the Transparent Factory architecture (ETYs, NOEs, third party toolkit, ENTs, M1Es). The Transparent Factory private MIB does not define the specific management applications and policies.

The diagram following illustrates the Schneider Electric (Groupe_Schneider (3833)) private enterprise MIB subtree.



The **Groupe_Schneider (3833)** subtree is the root of Groupe Schneider's private MIB in the Structure of Management Information (SMI) used by SNMP and defined in RFC-1155, which is a specification that defines the structure and identification of management information for TCP/IP-based networks.

Transparent Factory Ethernet Subtree

The **Transparent_Factory_Ethernet (1)** subtree defines groups that support the Transparent Factory Ethernet services and devices.

Service	Subtree Definition
Switch(1)	brand of switches labeled: ConneXium switches private MIB
Port502_Messaging(2)	objects for managing explicit client/server communications supporting applications, such as HMI, SCADA, or programming tools
I/O_Scanning(3)	objects for managing I/O device communications that use the I/O Scanner mechanism with the MB/TCP protocol
Global_Data(4)	objects for managing the application coordination service using a publish/subscribe protocol
Web(5)	objects for managing the activity of the embedded Web servers
Address_Server(6)	objects for managing the activity of the BOOTP and (or) DHCP servers
Equipment_Profiles(7)	objects for each type of device in Transparent Factory Ethernet's product portfolio
timeManagement(8)	objects for managing the UTC time stamp service
email(9)	objects for managing the mail service

Device subtrees, or groups, will be defined for the following devices:
- **Premium(1)**
- **Quantum(2)**
- **Generic_Device(3)**
- **M1E(4)**
- **ENT(5)**

As devices are added to Schneider's catalog, Schneider's private MIB will be extended in the following manner:
- If needed, a Transparent Factory, communication-service object will be added for the new device in the corresponding **Equipment_Profiles(7)** subtree. This subtree can hold as many objects as are required.
- If needed, a new branch will be added at the same level as **Transparent_Factory_Ethernet(1)**. This subtree will be created for product-specific objects (such as the ATV58 object under the **IndustrialControlProducts (3) subtree**).

When a new device is created, a corresponding object description is created in the ASN.1 format. The ASN.1 file(s) are then given to producers of SNMP manager software for inclusion in their products.

Port502 Messaging Subtree

The Port502_Messaging (2) subtree, or group, provides connection management and data flow services. The following list describes the function of each object.

Service	Indicates . . .
port502Status(1)	status of the service (idle, operational)
port502SupportedProtocol(2)	supported protocols (MODBUS, Xway)
port502IpSecurity(3):	status of the Port502 IP Security service (enabled/disabled)
port502MaxConn(4)	maximum number of TCP connections supported by the Port502 entity
port502LocalConn(5)	TCP connection number currently opened by the local Port502 entity
port502RemConn(6)	TCP connection number currently opened by the remote entity to the local Port502 entity
port502IpSecurityTable(7)	a table containing the number of unsuccessful TCP connection open tries from a remote TCP entity
port502ConnTable(8)	a table containing Port502 TCP specific information (MsgIn, MsgOut)
port502MsgIn(9)	total number of Port502 messages received from the network
port502MsgOut(10)	total number of Port502 messages sent from the network
port502MsgOutErr(11)	total number of error messages built by the Port502 messaging entity and sent to the network
port502AddStackStat(12)	the support of additional port502 stack statistics: ● 1 - disabled ● 2 - enabled
port502AddStackStatTable(13)	additional stack statistics for Port502 (optional)

I/O Scanning Subtree

The I/O_Scanning (3) subtree, or group, contains the objects related to I/O scanning device management and associated MODBUS communications on Port502.

Service	Indicates . . .
ioScanStatus(1)	global status of the I/O scanning service: ● 1 - idle ● 2 - operational ● 3 - stopped
ioScanMaxDevice(2)	maximum number of devices supported by the I/O scanning entity
ioScanPolledDevice(3)	number of devices currently polled by the I/O scanning entity
ioScanTransSend(4)	total number of transactions sent by the I/O scanning entity
ioScanGlbHealth(5)	global status of health for the I/O scanning service: ● 2 - OK: Every remote I/O device is responding ● 4- Warning: At least one remote I/O device is not responding
ioScanDeviceTable(6)	a table containing information on each remote devices polled by the I/O scanning entity

Global Data Subtree

The Global_Data (4) subtree, or group, contains the objects related to the Global Data service.

Service	Indicates . . .
glbDataStatus(1)	global status of the Global Data service: ● 1 - idle ● 2 - operational ● 3 - stopped
glbDataMaxPub(2)	maximum number of published variables configured by the Global Data entity
glbDataMaxSub(3)	maximum number of subscribed variables configured by the Global Data entity
glbDataPub(4)	total number of publications sent to the network
glbDataSub(5)	total number of subscriptions received from the network
glbDataPubErr(6)	total number of publication errors detected by the local entity
glbDataSubErr(7)	total number of subscription errors detected by the local entity
glbDataGlbSubHealth(8)	global status of health for the Global Data subscribed variables: ● 2 - OK: The health status of all subscribed variables is OK ● 4 - Warning: At least one subscribed variable has a health fault
glbDataPubTable(9)	a table containing information on each published variable (the number of publications, the source IP address, the number of errors)
glbDataSubTable(10)	a table containing information on each subscribed variable (the number of subscriptions, the source IP address, the number of errors, Health)

Web Subtree

The Web (5) subtree, or group, contains the objects related to the Web server service.

Service	Indicates . . .
webStatus(1)	global status of the Web service: ● 1 - idle ● 2 - operational
webPassword (2)	switch to enable or disable the use of Web passwords: ● 1 - disabled ● 2 - enabled
webSuccessfullAccess (3)	total number of successful accesses to the Web site
webFailedAttempts (4)	total number of unsuccessful accesses to the Web site

Address Server Subtree

The Address_Server (6) subtree, or group, contains the objects related to the Address Server service. The Address Server can be either a BOOTP server or a DHCP server.

Service	Indicates . . .
addressServerStatus(1)	global status of the address server service: ● 1 - idle ● 2 - operational

Equipment Profile Subtree

The Equipment_Profiles (7) subtree contains a set of common objects.

Service	Indicates . . .
profileProductName(1)	the commercial name of the communication product in a string form (for example, 140 NOE 771 11)
profileVersion(2)	the software version of the communication product in a string form (for example, Vx.y or V1.1)
profileCommunicationServices(3)	the communication services supported by the profile (Port502Messaging, I/O scanning Messaging, Global Data, Web, and Address Server)
profileGlobalStatus(4)	the global status of the communication module: ● 1 - NOK ● 2 - OK
profileConfigMode(5)	the IP configuration mode of the communication module: ● 1 - Local: The IP configuration is created locally ● 2 - dhcpServed: The IP configuration is created by a remote DHCP server
profileRoleName(6)	rthe ole name for the IP address management if it exists (empty string if there is none)
profileBandwidthMgt(7)	the status of Bandwidth Management: ● 1 - disabled ● 2 - enabled
profileBandwidthDistTable(8)	the CPU time distribution between Global Data, Port502 Messaging, I/O scanning
profileLedDisplayTable(9)	a table giving the name and the state of each module's LEDs
profileSlot(10)	the position of the communication module inside the rack if there is one. If there is no rack, the profileSlot value will be zero
profileCPUType(11)	the host for which that communication module is a part when a CPU type exists (if there is no host, the string is empty)
profileTrapTableEntriesMax(12)	the maximum numbers of entries in the Trap Table (equal to the number of possible remote managers)
profileTrapTable(13)	a table allowing you to enable or disable the private traps for each of the communication services

Service	Indicates . . .
profileSpecificId(14)	a unique Profile Specific Identification inside the equipmentProfile object of the Schneider Transparent Factory MIB (for example, the PLC Premium family is 100)
profileIpAddress(15)	the IP address of the SNMP agent
profileIpNetMask(16)	the subnet mask associated with the IP address of the SNMP agent (the value of the mask is an IP address with all the network bits set to 1 and all the host bits set to 0)
profileIpGateway(17)	the default Gateway IP address of the SNMP agent
profileMacAddress(18)	the Ethernet media-dependent address of the SNMP agent

NTP Subtree

The NTP (1) subtree contains a set of common objects.

Service	Indicates . . .
ntpStatus(1)	the status of NTP service (not server): **1.** 1 = Idle no configuration **2.** 2 = Operational
ntpSrvAddr(2)	the IP address of NTP server in dot notation format
ntpLnkSrvStatus(3)	the status of link between module and NTP server: **1.** 1 = NOK (module can not reach NTP server) **2.** 2 = OK
ntpReqCnt(4)	the number of requests sent to NTP server
ntpRespCnt(5)	the number of responses received from NTP server
ntpErrCnt(6)	the total number of communication errors
ntpDate(7)	date of the day
ntpTime(8)	time of the day
ntpTimeZone(9)	current time zone
ntpDSTStatus(10)	Daylight Savings Time status: **1.** 1 = ON (Daylight Savings Time) **2.** 2 = OFF (Standard Time)
ntpLastErr(11)	Last error code generated by system

SMTP Subtree

The SMTP (1) subtree contains a set of common objects.

Service	Indicates . . .
emailIndex(1)	the index value in the email service table
smtpStatus(2)	the status of SMTP service (not server): ● 1 = Idle (no configuration) ● 2 = operational
smtpSrvAddr(3)	the IP address of SMTP server in dot notation format
smtpMailSentCnt(4)	the total number of emails sent to the network and successfully acknowledged by the server
smtpErrCnt(5)	the total number of email messages that could not be sent to the network or that have been sent but not acknowledged by the server
smtpLastErr(6)	the error code of the last error that occurred while trying to send an email message to the network
smtpLastMailElapsedTime(7)	the number of elapsed seconds since last successful email was sent to the server
smtpLnkSrvStatus(8)	the status of link with SMTP server: **1.** 1 = NOK (not OK), link is down; module failed to contact SMTP server **2.** 2 = OK
smtpSrvChkFailCnt(9)	the number of times the link to SMTP server is detected as ‘down.’

See the Electronic Mail Notification Service subtree table *(see page 273)*.

NOTE: A diagram of the Schneider Electric private enterprise MIB subtree appears in Simple Network Management Service (SNMP) *(see page 237)*.

Private Traps and MIB Files

Traps are used to signal status changes to the manager. Using traps helps to avoid adding traffic.

The status changes signaled by the trap are for the:
- LEDs
- communication ports
- I/O scanning health values
- Global Data health
- NTP service
- SMTP service

The following list describes the characteristics of private traps, which means that they can:

- send messages to the two managers whose IP addresses are configured in the SNMP configuration (either the PL7 or the Web page)
- use the community name given to this configuration
- enable or disable each of the Transparent Factory Ethernet Private MIB groups: Switch (1), Port502_Messaging (2), I/O_Scanning (3), Global_Data (4), Web (5), Address_Server (6), Equipment_Profiles (7), NTP (8), and SMTP (9)

Private traps are described in the MIB ASN.1 description, which is contained in a `.mib` text file.

NTP Traps

1. **DST Change Trap:** notifies the manager that the NTP server time has changed either from (a) standard time to daylight saving time or (b) daylight saving time to standard time
2. **NTP Status Change Trap:** sent when the NTP component status changes (ntpStatus(1))
3. **Leap Second Trap:** sent when leap seconds are inserted

SMTP Traps

1. **SMTP Status Change Trap:** sent when SMTPStatus changes
2. **SMTP Link to Server Status Change:** sent when tSMTPLnkSrvStatus changes. Trap is sent when service tries to send an email. Every 30 minutes a periodic test checks the connection to the SMTP server.

Chapter 9
NTP Time Synchronization Service

Introduction

This chapter describes the NTP time synchronization service, which provides an accurate local clock using NTP.

What Is in This Chapter?

This chapter contains the following topics:

Introducing the NTP Time Synchronization Service

General

The NTP time service synchronizes computer clocks over the Internet. For example, the time of one client is synchronized either with another server or to a referenced time source like a radio or satellite receiver.

Typical time service configurations utilize redundant servers and diverse network paths to achieve high accuracy and reliability. Time service accuracy can be within a millisecond on LANs and up to a few tenths of milliseconds on WANs.

Use the time service for:
- event recording (sequence events)
- event synchronization (trigger simultaneous events)
- alarm and I/O synchronization (time stamp alarms)

Features of the NTP Time Service

Some features of the NTP time synchronization service are:
- periodic time correction obtained from the reference-standard time server
- automatic switch over to a backup time server if a problem occurs with the normal time server system
- controller projects use a function block to read the accurate clock, a feature that allows project events or variables to be time stamped
- estimated time stamping accuracy of:
  - 5 mSec for 140 CPU 651 *x*0’s and higher
  - 10 mSec for other CPUs
- local time zone is configurable and customizable including daylight savings time
- Web page diagnostics for the time synchronization service

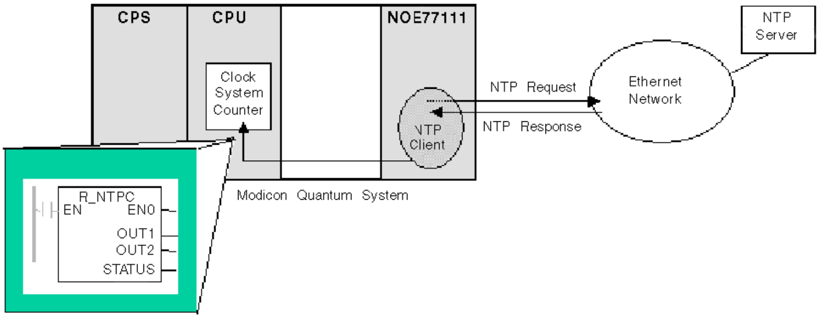
NTP Time Synchronization and Time Stamps

The Schneider Electric Ethernet interface module NOE 771 11 provides the source time-synchronization signal sent to a controller. The module’s firmware includes an NTP client, which provides time synchronization.

Action	Result
NTP Client requests a time synchronization signal from an NTP server. (Request is sent over an Ethernet network.)	NTP Server responds with a signal.
NTP Client stores the time.	
NTP Client sends a message to the controller’s clock system counter.	The controller updates its internal clock with the following granularity: ● 1 ms

To read the clock from the PLC application:
- Use the R_NTPC function block *(see page 257)* in either MAST, FAST, or Interrupt sections.

On an Ethernet network, all controllers should be synchronized with the same NTP server.



NTP Time Synchronization Definitions

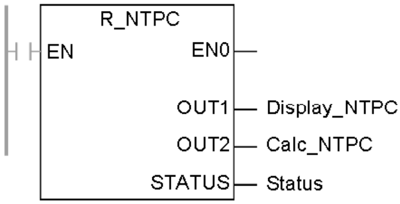
Term	Description of Service
Local clock offset	Accurate local time is maintained by adjusting the time by using a local clock offset. The local clock offset is calculated by the formula: `( (T4 - T1) + (T3 - T2) ) / 2` The formula's variables are: ● T1 = Time at which the NTP request is transmitted from the module ● T2 = Time at which the NTP Server receives the request (provided by the module in response) ● T3 = Time at which the NTP Server transmits the response (provided to the module in the response) ● T4 = Time at which the NTP Response is received by the module
Time accuracy	Local Time error is < 10 ms compared to the reference NTP server: ● Typical: under 5 ms ● Worst case: <10 ms Note: **1.** Settling Time: The maximum accuracy is obtained after 2 updates from the NTP server. **2.** Polling Period Dependency: The accuracy is dependent upon the polling period. Less than 10 mSec of error is guaranteed for polling periods of 120 seconds or less. To obtain the best possible accuracy and if your network bandwidth will allow, reduce the polling period to a small value. **3.** For example, choosing a polling time of 5 seconds provides better accuracy than a time of 30 seconds.
Time zone	Default format: Universal Time, Coordinated (UTC) Optional: Configure the service to use a local time zone. For example, GMT+1 for Barcelona or Paris

Term	Description of Service
Daylight savings time	Automatic: module adjusts time in spring and fall
Leap Second	Automatic: module inserts or deletes a second Note: **1.** To compensate for the deceleration of the earth's rotation, a leap second is inserted in the UTC time every 18 months by an INTERNATIONAL EARTH ROTATION SERVICE (IERS). **2.** Leap seconds will be inserted automatically as needed. If needed, leap seconds are inserted at the end of the last minute in June or December, as commanded by the NTP server.

Using the R_NTPC Block for Time Synchronization

R_NTPC Representation

The block reads the Ethernet network system time and transfers it into specified parameters. The additional parameter EN should be configured.



R_NTPC block has a 16-bit status word.

R_NTPC Parameter Description

Description of parameters:

Parameter	Data Type	Description
Display_NTPC (OUT1)	DT + INT	NTP clock value displayed in: ● year, month, day, hours, minutes, and seconds using the DT format ● milliseconds as an INT
Calc_NTPC (OUT2)	UDINT+INT	NTP clock value displayed in: ● seconds as an UDINT ● fractions of a second as an INT

<table>
<tr><th>Parameter</th><th>Data Type</th><th colspan="3">Description</th></tr>
<tr><td rowspan="8">Status</td><td rowspan="8">INT</td><td>Low Byte</td><td>High Byte</td><td>Description</td></tr>
<tr><td>0</td><td>0</td><td>un-initialized state</td></tr>
<tr><td>1</td><td>0</td><td>illegal</td></tr>
<tr><td>0</td><td>1</td><td>The CPU is out of synchronization with the NTP server, but the clock has been updated at least once by an external server.</td></tr>
<tr><td>1</td><td>1</td><td>normal operation</td></tr>
<tr><td colspan="3">The low byte is managed by the controller
● Set = 0
  ● The clock value is NOT available.
  ● The date/time is NOT updated within last two minutes.
● Set = 1
  ● The date/time is updated within the last two minutes.
  ● The date/time is acceptable.
The high byte is managed by the NOE.
● Set = 0
  ● The NTP server clock value is not available.
● Set = 1
The updated date/time is received from server and sent to the module (at least once).
  ● within two-minute time interval
  ● acceptable (10 ms or less error)</td></tr>
<tr><td colspan="3">For a valid time in the CPU, the low and high bytes of the STATUS parameter must be set to 1.</td></tr>
</table>

NTP Time Synchronization Service Operating Conditions

Introduction

This material describes special operating conditions related to:
- powering up
- stop or run PLC
- downloading applications
- configuring the time service

Power up

To establish the accurate Ethernet system network time, the system does the following at power-up:
- requires the Ethernet module to boot
- uses the Ethernet module to obtain the time from the NTP server
- requires a predefined interval until time is accurate; user's configuration determines how long before time is accurate
- may require several updates to achieve peak accuracy

Once an accurate time is received, the service sets the STATUS in the associated time service register.

The time-service clock value starts at 0 until fully updated from the module.

Model	Starting Date
Modicon Quantum with Unity	January 1st 1980 00:00:00.00

Stop or Run PLC

- Stop and Run have no effect on the accuracy of the clock.
- Stop and Run have no effect on the update of the clock.
- A transition from one mode to the other has no effect on the accuracy of the system Ethernet network time.

Download Application

The status clock value associated with the time service register in the CPU is reinitialized after an application is downloaded or after an NTP server swap.

There will be two polling periods before the time is accurate.

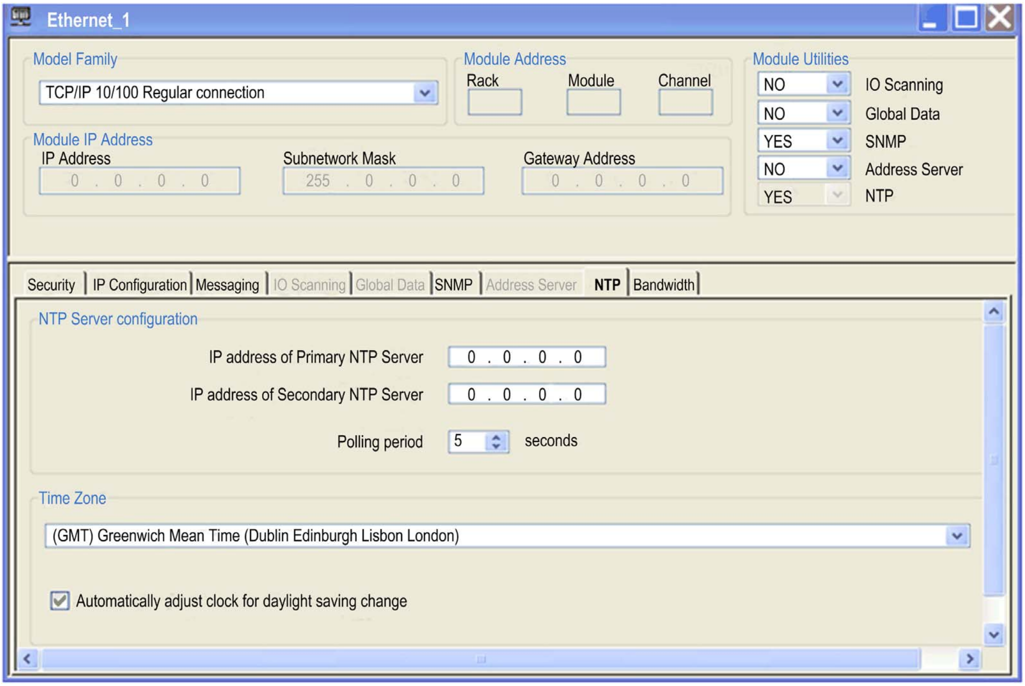
NTP Configuration

Elsewhere in this section is information about the NTP configuration.

Configuring the NTP Time Service

Configuring the NTP Time Service

In Unity Pro, click the **NTP** tab to configure the following parameters.



Field	Parameter	Action
NTP Server Configuration		
	IP address of primary NTP server	Enter a valid IP address
	IP address of secondary NTP server	Enter a valid IP address
	Polling period (in seconds	Enter a value ● min = 1 sec ● max = 120 sec ● default = 5 sec
Time Zone		Select from the drop-down menu: ● Universal Time, Coordinated (GMT) = default ● Custom time zone
	Automatically adjust clock for daylight saving change	This parameter is selected by default (check mark appears) if daylight saving time is chosen.

Changing Time Service Parameters

To make any changes to the time synchronization service:

Step	Action
1	Enter changes in the appropriate field on the NTP configuration page for one or all of the configurable parameters.
2	Click **Save**.

Important Information about the Time Service

NOTE: Regarding the time service:

1. **Automatically adjust clock for daylight savings change parameter**: If this check box is selected, the module automatically corrects the local time to account for daylight saving time. Therefore, no action is required, since the daylight saving time start and end are automatically changed each year.
2. **Polling Time Parameter**: The time (in seconds) is the time between time updates from the NTP server. The default is 5 seconds.
3. **Storing the Time Service Configuration**: The last time service configuration is saved internally in the Ethernet module.
4. **Replacing the Ethernet Module**: If the Ethernet module has to be replaced, the stored configuration is lost, and the system returns to the default configuration.

Customizing Time Zone Parameters

If you want a time zone not listed in the time zone table:

Step	Action	Comment
1	Write the text rules for the custom time zone.	
2	Using an FTP client, store your rules in the file: /FLASH0/wwwroot/conf/NTP/customrules user ID: ntpupdate password: ntpupdate	Root directory to store 'customrules' is set by the FTP server as: /FLASH0/wwwroot/conf/NTP
3	When the rules are written, choose the drop down menu on the **NTP Configuration** web page, and configure (or reboot) the module by selecting Time Zone = Custom	The NTP component looks for customrules, calls the tz compiler and generates a new file called 'tz_custom'. This file is binary file and should not be edited. If the tz compiler detects a syntax error in customrules, the error is logged in the file: /FLASH0/wwwroot/conf/NTP/error.log **1.** NTP component is not launched **2.** NTP Status field in diagnostic web page displays NOT OK.
4	If you want more information, the syntax to write those rules along with a few examples are found in the module in: /FLASH0/wwwroot/conf/NTP/instructions.txt	

Time Zone Parameters

Select a time zone from the drop-down menu:

Time Zone	Description	DST Available
Custom		Yes
(GMT-12:00)	Dateline Standard Time [Eniwetok Kwajalein]	No
(GMT-11:00)	Samoa Standard Time [Midway Is Samoa]	No
(GMT-10:00)	Hawaiian Standard Time [Hawaii Honolulu]	No
(GMT-09:00)	Alaskan Standard Time [Anchorage]	Yes
(GMT-08:00)	Pacific Standard Time [Los Angeles Tijuana]	Yes
(GMT-07:00)	Mexican Standard Time [Chihuahua La Paz Mazatlan]	Yes
(GMT-07:00)	Mountain Standard Time [Arizona Phoenix]	No
(GMT-07:00)	Mountain Standard Time [Denver]	Yes
(GMT-06:00)	Central Standard Time [Chicago]	Yes
(GMT-06:00)	Mexico Standard Time [Tegucigalpa]	No
(GMT-06:00)	Canada Central Standard Time [Saskatchewan Regina]	No
(GMT-06:00)	Central America Standard Time [Mexico_city]	Yes
(GMT-05:00)	SA Pacific Standard Time [Bogota Lima Quito]	No
(GMT-05:00)	Eastern Standard Time [New York]	Yes
(GMT-05:00)	Eastern Standard Time [Indiana (East)] [Indianapolis]	No
(GMT-04:00)	SA Western Standard Time [Caracas La Paz]	No
(GMT-04:00)	Pacific SA Standard Time [Santiago]	Yes
(GMT-03:30)	Newfoundland Standard Time [Newfoundland St Johns]	Yes
(GMT-03:00)	E. South America Standard Time [Brasilia Sao_Paulo]	Yes
(GMT-03:00)	SA Eastern Standard Time [Buenos Aires Georgetown]	No
(GMT-02:00)	Mid-Atlantic Standard Time [South_Georgia]	No
(GMT-01:00)	Azores Standard Time [Azores Cape Verde Island]	Yes
(GMT)	Universal Coordinated Time [Casablanca, Monrovia]	No
(GMT0)	Greenwich Mean Time [Dublin Edinburgh Lisbon London]	Yes
(GMT+01:00)	Romance Standard Time [Amsterdam CopenHagen Madrid Paris Vilnius]	Yes
(GMT+01:00)	Central European Standard Time [Belgrade Sarajevo Skopje Sofija Zagreb]	Yes
(GMT+01:00)	Central Europe Standard Time [Bratislava Budapest Ljubljana Prague Warsaw]	Yes
(GMT+01:00)	W. Europe Standard Time [Brussels Berlin Bern Rome Stockholm Vienna]	Yes
(GMT+02:00)	GTB Standard Time [Athens Istanbul Minsk]	Yes
(GMT+02:00)	E. Europe Standard Time [Bucharest]	Yes

Time Zone	Description	DST Available
(GMT+02:00)	Egypt Standard Time [Cairo]	Yes
(GMT+02:00)	South Africa Standard Time [Johannesburg Harare Pretoria]	No
(GMT+02:00)	FLE Standard Time [Helsinki Riga Tallinn]	Yes
(GMT+02:00)	Israel Standard Time [Israel Jerusalem]	Yes
(GMT+03:00)	Arabic Standard Time [Baghdad]	Yes
(GMT+03:00)	Arab Standard Time [Kuwait Riyadh]	No
(GMT+03:00)	Russian Standard Time [Moscow St. Petersburg Volgograd]	Yes
(GMT+03:00)	E. Africa Standard Time [Nairobi]	No
(GMT+03:30)	Iran Standard Time [Tehran]	Yes
(GMT+04:00)	Arabian Standard Time [Abu Dhabi Muscat]	No
(GMT+04:00)	Caucasus Standard Time [Baku Tbilisi]	Yes
(GMT+04:00)	Afghanistan Standard Time [Kabul]	No
(GMT+05:00)	Ekaterinburg Standard Time [Ekaterinburg]	Yes
(GMT+05:00)	West Asia Standard Time [Islamabad Karachi Tashkent]	No
(GMT+05:30)	India Standard Time [Bombay Calcutta Madras New Delhi]	No
(GMT+06:00)	Central Asia Standard Time [Almaty Dhaka]	Yes
(GMT+06:00)	Sri Lanka Standard Time [Columbo]	No
(GMT+07:00)	SE Asia Standard Time [Bangkok Hanoi Jakarta]	No
(GMT+08:00)	China Standard Time [Beijing Chongqing Hong Kong Urumqi]	No
(GMT+08:00)	W. Australia Standard Time [Perth]	No
(GMT+08:00)	Singapore Standard Time [Singapore]	No
(GMT+08:00)	Taipei Standard Time [Taipei]	No
(GMT+09:00)	Tokyo Standard Time [Osako Sapporo Tokyo]	No
(GMT+09:00)	Korea Standard Time [Seoul]	No
(GMT+09:00)	Yakutsk Standard Time [Yakutsk]	Yes
(GMT+09:30)	Cen. Australia Standard Time [Adelaide]	Yes
(GMT+09:30)	AUS Central Standard Time [Darwin]	No
(GMT+10:00)	E. Australia Standard Time [Brisbane]	No
(GMT+10:00)	AUS Eastern Standard Time [Canberra Melbourne Sydney]	Yes
(GMT+10:00)	West Pacific Standard Time [Guam Port Moresby]	No
(GMT+10:00)	Tasmania Standard Time [Hobart]	Yes
(GMT+10:00)	Vladivostok Standard Time [Vladivostok]	Yes
(GMT+11:00)	Central Pacific Standard Time [Magadan Solomon Is New Caledonia]	Yes

Time Zone	Description	DST Available
(GMT+12:00)	New Zealand Standard Time [Auckland Wellington]	Yes
(GMT+12:00)	Fiji Standard Time [Fiji Kamchatka Marshall Is]	No

Chapter 10
Electronic Mail Notification Service

Introduction

This chapter describes the electronic mail notification service, which uses SMTP to allow the controller's project to send e-mail messages.

What Is in This Chapter?

This chapter contains the following topics:

Introducing the Electronic Mail Notification Service

General

The electronic mail notification service allows controller-based projects to report alarms or events. The automation controller monitors the system and can automatically create an electronic mail message alert with data, alarms, and/or events. Mail recipients can be either local or remote.

- Based on predefined events or conditions, messages are created using the MSTR function block.
- The email message is constructed from predefined headers plus variables and text (a maximum of 238 bytes). This message is sent directly from the automation system to the local email server.
- Mail headers contain common predefined items—recipient list, sender name, and subject. These items can be updated by an authorized administrator.

Mail System Types

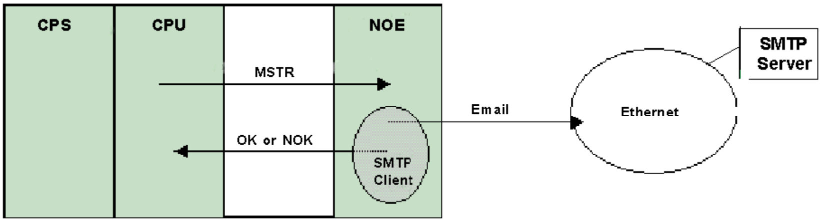
The Simple Mail Transfer Protocol (SMTP) provides two mechanisms for the transmission of email messages:

- direct connection
- relay system

Mechanism	Condition	Result
Direct connection	Sender and receiver are connected to the same transport service.	Email messages are sent to host.
Relay system	Sender and receiver are NOT connected to the same transport service.	Email messages are relayed from one server to another server. To relay messages, the SMTP server must be supplied with the address of the: ● destination host ● destination mailbox

Mail Service Client

The Quantum Ethernet module includes an SMTP client. When the module receives a specific request from the project, it sends an email message to the mail server.



Operating Modes and Sending Requests

Because the project sends the email request, a controller cannot send an email message either while in the stopped mode or while downloading a project. As soon as the controller is in RUN mode, the function block sends a request during the first project scan.

Diagnostic counters are reset to 0 after either a power-up, a project download, or a reconfiguration of the mail service.

Mail Service

Configuring the Mail Service

As an authorized administrator, use the SMTP Configuration Web page to:

- configure the service
- set the IP address of the mail server

Elsewhere in this guide is detailed information about configuring the electronic mail service *(see page 303)*.

NOTE: Default TCP Port: The default TCP port number for SMTP is 25. Ensure that you configure the port specified by your local mail server.

Message Creation and Delivery

A user-defined event or condition triggers the MSTR block to create a message. Each message uses one of three user-defined headers. Each message sent from the controller can contain text and variable information (with a maximum of 238 bytes).

The project selects the appropriate header. Each header contains:

- sender name
- list of recipients
- subject

Header Examples

The text and variable information can be defined and updated by an authorized administrator using an embedded Web page (SMTP Configuration page). Define mail headers to indicate different levels of importance. For example:

- Header 1 could be "Urgent problem reported by PLC 10"
- Header 2 could be "NOTIFICATION from substation 10"
- Header 3 could be "INFO message from water system"

Listing different recipients in each of the three headers ensures that the right information quickly flows to the right recipients. The project adds pertinent information such as the specific device, process, or location. This pertinent information is added to the body of the mail message. Then the complete message is sent to an electronic mail server for distribution to recipients. These recipients could be engineers, managers, or process owners.

Security (Authentication)

An optional login (system ID) and password can be used to authenticate the connection to the SMTP mail server. The SMTP-supported authentication method is LOGIN.

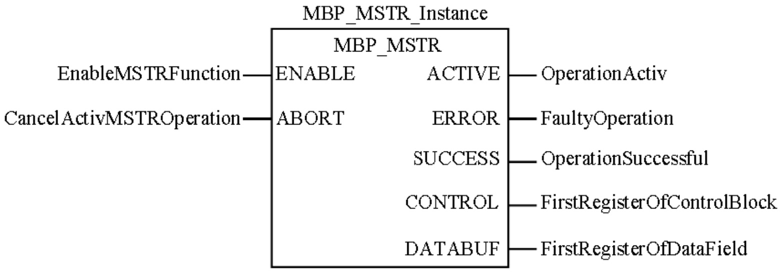
System Diagnostics

The SMTP Diagnostic Web page displays the status of the service. Diagnostic information is also available for remote management using the SNMP network management standard.

Using the MBP_MSTR Block for Mail Service Communication

Block Representation

Each operation is designated by a code. To send an email message, use the MBP_MSTR block with function code 13 *(see page 168)*.



Parameter Description

The following table describes the MBP_MSTR parameters:

Parameter	Data Type	Description
ENABLE	None	ON = enables selected MBP_MSTR operation
ABORT	None	ON = terminates active MBP_MSTR operation
ACTIVE	None	ON while the instruction is active
ERROR	None	ON if the MBP_MSTR operation is terminated prior to completion
SUCCESS	None	ON = operation successful

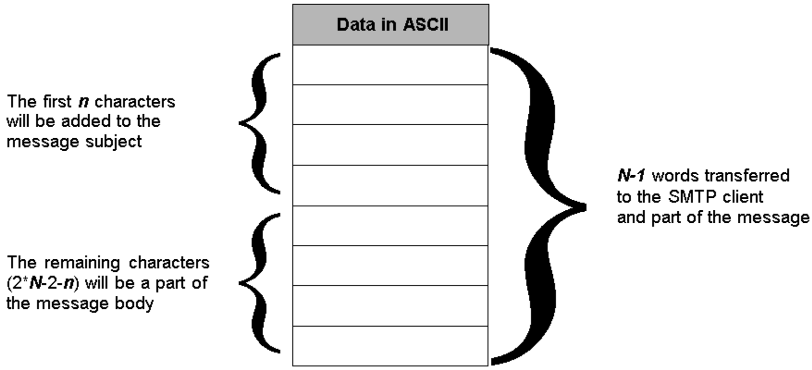
<table>
<tr><th>Parameter</th><th>Data Type</th><th colspan="2">Description</th></tr>
<tr><td rowspan="12">CONTROL</td><td rowspan="12">INT, UINT</td><td colspan="2">Control block is the first of several network-dependant, contiguous, holding words, and the first of nine contiguous words entered in the top node.
The remaining eight words are:</td></tr>
<tr><td colspan="2">Displayed: Identifies one of the MBP_MSTR = 13</td></tr>
<tr><td>First implied</td><td>displays error status (see Mail Service Error Codes (see page 272))</td></tr>
<tr><td>Second implied</td><td>displays length (number of words transferred)</td></tr>
<tr><td>Third implied</td><td>(not used)</td></tr>
<tr><td rowspan="2">Fourth implied</td><td>high byte: slot address of the NOE module or 0xFE for the 140 CPU 651 x0</td></tr>
<tr><td>low byte: 0 (not used)</td></tr>
<tr><td>Fifth implied</td><td>(not used)</td></tr>
<tr><td>Sixth implied</td><td>(not used)</td></tr>
<tr><td>Seventh implied</td><td>(not used)</td></tr>
<tr><td>Eighth implied</td><td>(not used)</td></tr>
<tr><td colspan="2"></td></tr>
<tr><td>DATABUF</td><td>INT, UINT</td><td colspan="2">The DATABUF parameter is the address of the buffer including the data inserted into the body of the email message. The data should be in ASCII format starting at the second word.
Note:
1. Least significant byte of the first word: contains a valid (configured) mail header (should be of value 1, 2, or 3).
2. Most significant byte of the first word: contains the length of the dynamic part of the message’s subject field. The length must be between 0 and 238 characters.</td></tr>
</table>

DATABUF Parameter Description

The first word of the DATABUF parameter contains the following information:

Byte Number	Description	Value
1 (least significant byte)	Mail header	{1, 2, 3}
2 (most significant byte)	Nb **n** of extra characters in subject	User-defined (between 0 and 238)

The second and subsequent words (maximum 119) contain the data (ASCII format) that will be copied into the email message. The first ***n*** characters will be added to the configured email subject. The remaining characters (2****N***-2-***n***) will be part of the message body. ***N*** represents the number of words transferred.



Mail Service Error Codes

Error Codes

The electronic mail notification service supports the following error codes:

Hex. Error Code	Description
5100	Internal error.
5101	SMTP component not operational.
5102	Mail header not configured.
5103	Invalid mail header value (should be 1, 2, or 3).
5104	Cannot connect to SMTP server.
5105	Error in transmitting content of email body to SMTP server.
5106	Closing SMTP connection with the server returned an error.
5107	SMTP HELO request failed.
5108	SMTP mail request failed. SMTP server may require authentication.
5109	SMTP RCPT request failed.
510A	No recipient has been accepted by the SMTP server.
510B	SMTP data request failed.
510C	Send email request contains an invalid length.
510D	Authentication failed.
510E	A reset component request has been received while the connection was open.

Electronic Mail Notification Service Subtree

Summary

The electronic mail delivery service subtree contains the following objects:

Service	Description
emailIndex (1)	index value in the email service table
smtpStatus (2)	global status of the SMTP service: ● idle(1): no configuration ● operational(2): operational and running ● stopped(3): stopped
smtpSrvAddr (3)	IP address of the remote SMTP server
smtpMailSentCnt (4)	total number of emails sent to the network and successfully acknowledged by server
smtpErrCnt (5)	total number of emails: ● not sent to the network ● sent but not successfully acknowledged by server (The smtpLastErr (6) object details the errors.)
smtpLastErr (6)	last error code (see details at Configuring an NOE with TFE Private MIB)
smtpLastMailElapsedTime (7)	number of seconds elapsed since last successful email sent
smtpLnkSrvStatus (8)	status of link between communication module and remote SMTP server: ● NOK (1) = SMTP server can NOT be reached ● OK (2) = SMTP server can be reached
smtpSrvChkFailCnt (9)	number of times link to SMTP server has been detected as down

Chapter 11
Embedded Web Pages

Introduction

This chapter presents the contents of the embedded Web pages contained in the Quantum 140 NOE 771 xx modules. These Web pages enable you to access diagnostic information, view configuration information, and change the online configurations for the module.

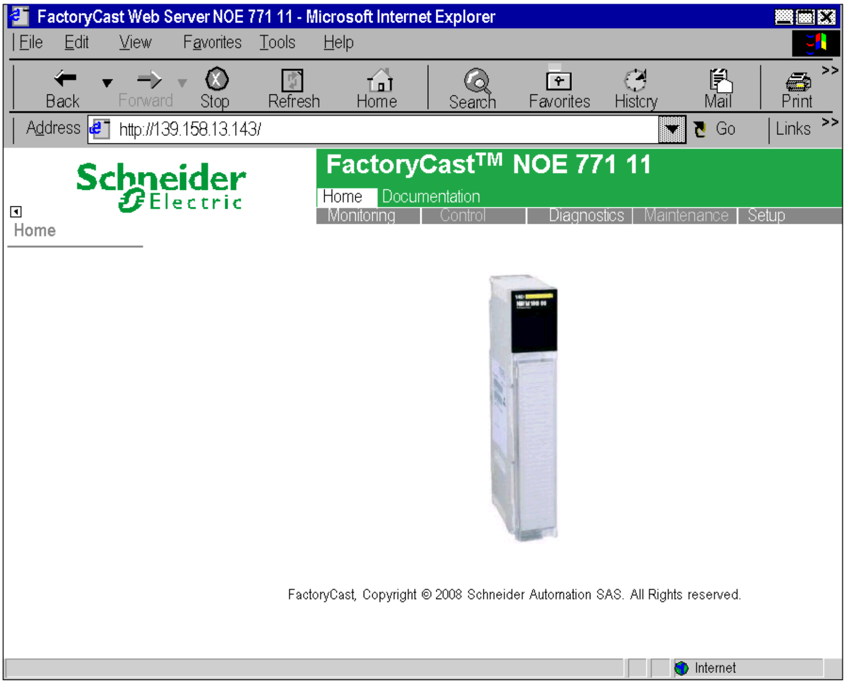
What Is in This Chapter?

This chapter contains the following topics:

Quantum Home Page

Home Page

To access the Quantum home page enter the IP address of the module in his web browser. No password is required to display the home page:



Links

From the Quantum home page, you can access the following pages:

- Monitoring *(see page 277)*
- Diagnostics *(see page 279)*
- Setup *(see page 280)*
- Documentation

Enter a user name and a password to access the services on these pages.

Language

In the Quantum home page, you can display the Quantum web pages in either English or Chinese.

Monitoring Home Page

Home Page

This page lists the various viewer services supported by the default Web site of the module and provides links for accessing the services you require.

Illustration

The **Monitoring** home page looks like this:

To access a service, click on a link. The services include:

- Data editor: for creating variable data tables to determine their values when the table is animated.
- Data editor lite: for creating variable data tables to determine their values when the table is animated. (This editor contains fewer features than the standard Data editor.)
- Graphic editor: for creating graphics to determine the values of variables when the graphic is animated.
- Graphic viewer: for viewing graphics to determine the values of variables when the graphic is animated.

- PLC Program viewer: Display UnityPro programs in run mode using a web browser.
- Password-protected custom pages: for restricting access to web pages created by the user.
- Custom pages without password protection: for unrestricted access to web pages created by the user.
- Silverlight pages: for viewing user-defined pages created with *Microsoft Expression Blend*® software that have been added to the web site

Diagnostics Home Page

Home Page

This page lists the various services supported by the default Web site of the module and provides links for accessing the services you require.

Illustration

The **Diagnostics** home page looks like this:

Links

To access the service you require, click on a link:
- Rack Viewer
- Controller Status *(see page 284)*
- RIO Status *(see page 286)*
- Alarm Viewer
- Ethernet *(see page 305)*
- Properties *(see page 314)*

Setup Home Page

Home Page

The NOE 771 11 **Setup** page lists the various services used to configure the module. You can navigate to this page from the link on the Home Page *(see page 276)*.

Illustration

The **Setup** page looks like this:

Setup Page Links

These links are on the **Setup** page:

Link	Result
SNMP	Provides the ability to configure the SNMP Agent in the NOE
Address Server *(see page 294)*	Configure the IP assignments, including showing the BOOTP and DHCP database
Global Data *(see page 191)*	Displays the **Global Data Configuration** page. Configure the Group Address, Multicast filtering, Distribution period, Health Time Out, Health Bits, and Data Zones. The **Global Data Configuration** page also displays a Variable Table.
NTP *(see page 297)*	Configure the IP address of the primary and secondary NTP server. Set a polling period. Select a time zone from the drop-down list, and automatic adjustment for daylight savings time.
Email *(see page 83)*	Configure the email server's IP and Port (default port is 25). If security is required, select **Enable** and set a Login and Password. Create three mail headers.

Link	Result
Security	View and edit: ● the username and password for accessing the home page ● the password for writing variables to the data editor
FTP	View and edit the username and password for the FTP service.

Accessing the Web Utility Home Page

Introduction

Each Modicon Quantum 10-/100-Megabit Ethernet module contains an embedded Web server that allows you to access diagnostics and online configurations for the module and its associated controller (PLC).

Pages on the embedded Web site display the following information:
- Configurable menus of the Address Server both BOOTP and DHCP and for SNMP *(see page 294)*
- Ethernet statistics for the node *(see page 275)*
- Controller's configuration (Controller Status on menu) *(see page 284)*
- Controller's register values
- Remote I/O status and configuration *(see page 286)*
- Remote I/O register values
- Remote I/O distributed values

The FactoryCast/Real Time modules *(see page 88)* offer these additional pages:
- Configuration and status for Global Data (publish/subscribe) *(see page 291)*
- Bandwidth monitoring *(see page 84)*
- I/O Scanner status *(see page 306)*
- MODBUS Messaging status *(see page 77)*
- NTP configuration and status *(see page 309)*
- SMTP configuration and status *(see page 310)*

The web pages can be viewed using Internet Explorer 4.0 or higher. Either browser supports JRE 1.4.2_04 or higher.

For information about the additional functionality provided by the FactoryCast system in the Ethernet modules, see the *FactoryCast Manual* (31001229).

Accessing the Module's Home Page

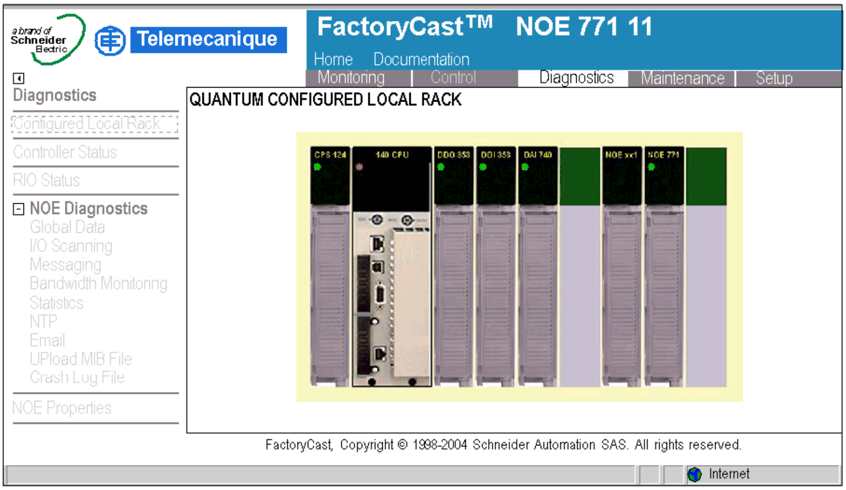
Do the following.

Step	Action
1	Obtain the full IP address or URL from your system administrator.
2	Type the address or URL in the address or location box in the browser window. The Schneider Electric Web Utility home page displays.
3	On the Quantum home page *(see page 276)*, click **Diagnostics**.
4	You will be requested to supply a user name and password.
5	After supplying the user name, password, and clicking **OK**, the Quantum Welcome Page appears. The Quantum Welcome Page provides the links to all the Quantum configuration and diagnostic pages and to the **Data Editor / Data Monitor**. **NOTE:** The default User Name is USER, and the default password is USER. Both should be changed by the system administrator during module installation.

Configured Local Rack Page

Overview

The Configured Local Rack page shows the current configuration:



Configured Local Rack Page Overview

The following table details the links on the Configured Local Rack page. To view the pages related to each of these topics, click the topic.

Link	Display
Global Data *(see page 305)*	general diagnostics of the global data and a table of all published/subscribed variables in a distribution group
I/O Scanning *(see page 306)*	general diagnostics of the I/O scanning utility and a summary of the status of all modules
Messaging *(see page 307)*	current information on open TCP connections on Port502
Bandwidth Monitoring *(see page 308)*	load distribution of the Embedded Server module between the global data utilities, I/O scanning, messaging, and other utilities
Statistics *(see page 308)*	Ethernet module statistics with the reset counters link
NTP *(see page 309)*	time synchronization service parameters
Email *(see page 310)*	Email notification service parameters
Upload MIB file	upload the private MIB source file
Crash Log file *(see page 311)*	crash message (after crash) and status message (normal operations)

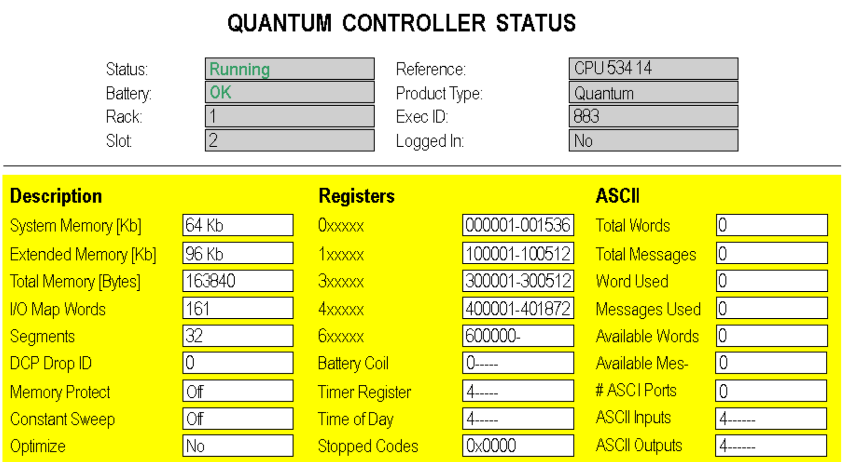
Controller Status Page

Overview

The **Controller Status Page** provides up-to-date information about the controller and its configuration. Access this page by selecting the CPU model from the Rack Viewer *(see page 283)* or the hyperlink Controller Status.

Sample Page

Here is an example of a **Controller Status** page.



QUANTUM CONTROLLER STATUS

Status:	Running	Reference:	CPU 534 14
Battery:	OK	Product Type:	Quantum
Rack:	1	Exec ID:	883
Slot:	2	Logged In:	No

Description		Registers		ASCII	
System Memory [Kb]	64 Kb	0xxxxx	000001-001536	Total Words	0
Extended Memory [Kb]	96 Kb	1xxxxx	100001-100512	Total Messages	0
Total Memory [Bytes]	163840	3xxxxx	300001-300512	Word Used	0
I/O Map Words	161	4xxxxx	400001-401872	Messages Used	0
Segments	32	6xxxxx	600000-	Available Words	0
DCP Drop ID	0	Battery Coil	0-----	Available Mes-	0
Memory Protect	Off	Timer Register	4-----	# ASCI Ports	0
Constant Sweep	Off	Time of Day	4-----	ASCII Inputs	4------
Optimize	No	Stopped Codes	0x0000	ASCII Outputs	4------

Dynamic Data

Some of the data provided on this page is dynamic. Dynamic data is constantly refreshed at a rate determined by the performance of the Embedded Server, network, and client CPU.

CPU Configuration Screen: Data Field Descriptions

Description Fields

The following table describes the description fields on the CPU Configuration Screen *(see page 284)*:

Field	Description
System Memory [Kb]	Amount of system memory used
Extended Memory [Kb]	Amount of extended memory used
Total Memory [Bytes]	Total memory used in bytes
I/O Map Words	Number of I/O words mapped.
Segments	Number of segments
DCP Drop ID	Drop number for distributed control
Memory Protect	Position of the memory protect switch
Constant Sweep	Current status of constant sweep
Optimize	Current status of optimization

Word Fields

The following table describes the Word fields on the CPU Configuration Screen *(see page 284)*:

Field	Description
%M	valid address of %M
%I	valid address of %I
%IW	valid address of %IW
%MW	valid address of %MW
Battery Coil	address of battery bit
Timer Register	address of timer word
Time of Day Clock	address of timer of day clock
Stopped Codes	reason for controlled stopping

ASCII Fields

The ASCII column on the CPU Configuration Screen *(see page 284)* contains information about the ASCII fields.

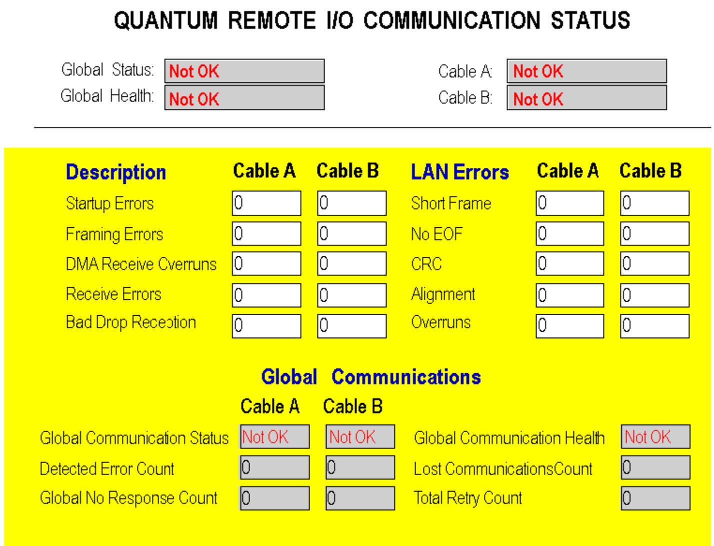
Remote I/O Status

Overview

The **Remote I/O Status** page gives an overview of the status and health of the Remote I/O network communications. Access this page by selecting the CRP Drop down menu item **Remote I/O Status** or using the hyperlink **RIO Status**.

Sample Page

Here is an example of a **Remote I/O Status** page.



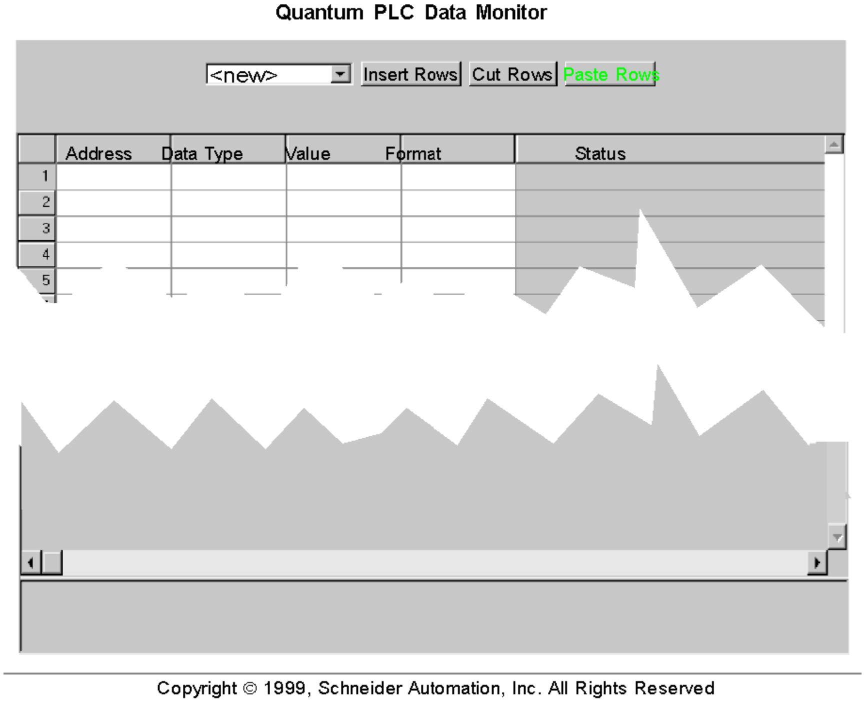
Dynamic Data

Some of the data provided on this page is dynamic. Dynamic data is constantly refreshed at a rate determined by the performance of the embedded server, network, and client CPU.

Quantum PLC Data Monitor Page

Introduction to the PLC Data Monitor Page

The following figure shows the Web page that allows you to display Quantum PLC data.



You can add, delete, and copy Quantum PLC data as described in the following list:

- Insert additional rows of data by clicking on the **Insert Rows** button.
- Delete specific rows of data by clicking on the **Cut Rows** button.
- Copy in rows of data by clicking on the **Paste Rows** button.

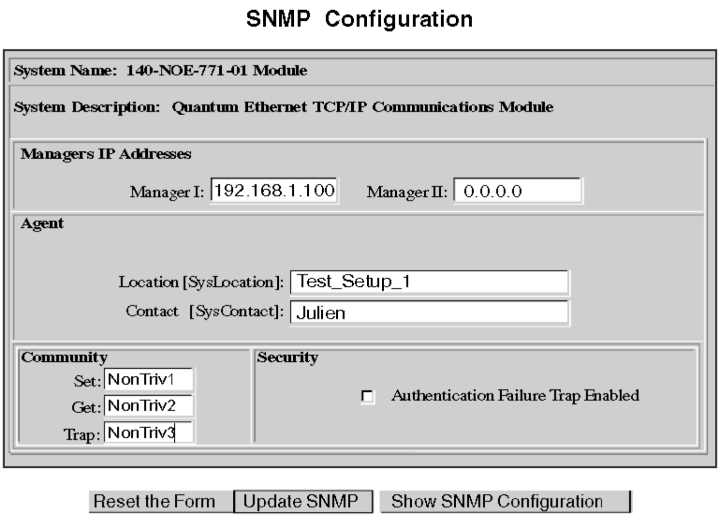
SNMP Configuration

Overview

This topic describes SNMP configuration for the Quantum Ethernet module through the **SNMP Configuration** page.

SNMP Configuration Page

Click the **SNMP** link in the **Setup** home page to display the Configure SNMP page:



The following table describes the SNMP configuration tasks that you can perform through the Web page:

Task	How To
To display the current SNMP configuration	Click **Show SNMP Configuration**.
To clear the fields	Click **Reset the Form**.
To change the SNMP configuration	Change the information on the page and click **Update SNMP**.

SNMP Page Fields

The following table describes the SNMP fields that you can modify.

Field	Information To Be Supplied
Manager I	IP Address of first SNMP Manager
Manager II	IP Address of second SNMP Manager
Location [SysLocation]	Location of the module
Contact [SysContact]	Name of the responsible systems engineer
Set	Designation of user level who can set the configuration
Get	Designation of user level who can view the configuration
Trap	Designation of user level who can capture information
Authentication Failure Trap Enabled	Turns on Community Name Checking

After you make your modifications to any of the above parameters, click **Update SNMP**. A new page displays the following message: "Successfully updated SNMP database." Note that this page contains the same links as those on the **Configure SNMP** page.

NOTE: Reset the module to implement the changes.

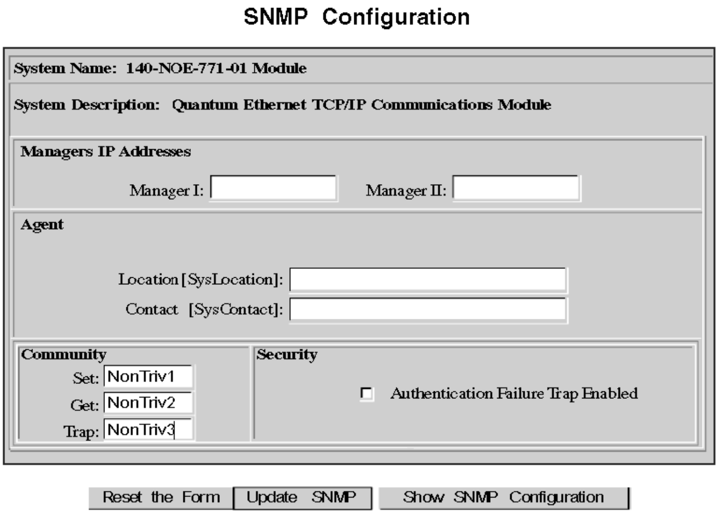
SNMP Community Strings

Use strings to restrict access to the SNMP Agent. These strings should be set to non-trivial names during module installation.

Modifying the SNMP Community Strings

To configure the SNMP community strings:

Step	Action
1	Enter the following URL into your browser: **http://hostname/secure/embedded/builtin?submit=Configure+SNMP** or click the **SNMP** link in the **Setup** home page to navigate to the **SNMP Configuration** page.

<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>2</td><td>Enter the Community names for Set, Get, and Trap into the SNMP Configuration page as shown below.

</td></tr>
<tr><td>3</td><td>Click Update SNMP.</td></tr>
<tr><td>4</td><td>To set your changes, use hot swap to reboot the module.</td></tr>
</table>

Global Data (Publish/Subscribe) Utility Configuration

Overview

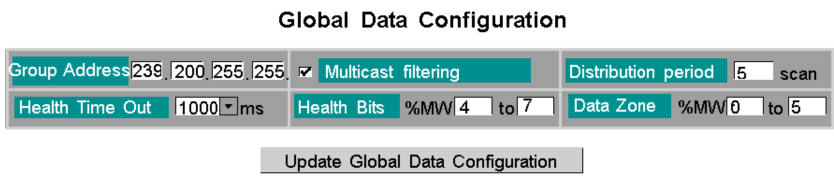
Whether you use either the Configure Each Device Separately or the Copy Configuration method *(see page 191)*, the procedure to configure individual parameters is the same. Therefore, in order to use the Global Data (publish/subscribe) utility in the NOE, you need to configure the Global Data parameters including:

- Distribution period
- Multicast filtering
- Health bit location
- Global Data base address
- Group IP address

The following sections describe in detail the exact steps to configure each parameter on the **Global Data Configuration** page.

Illustration

You can change the configuration in the **Global Data Configuration** page:



Variable Table

Data ID	Type	Symbol	Address	Length
1	SUB	var_01	%M 48	2
2	PUB	var_02	%M 60	2
3	SUB	var_03	%M 44	2
4	NON		%M	
5	NON		%M	
6	NON		%M	
7	NON		%M	
8	NON		%M	
9	NON		%M	

Configuring Global Data

After you have completed the Modeling System Configuration process *(see page 191)* using the second method, Copy Configuration, you modify the following parameters:

- Distribution period
- Health Time Out
- Health Bits location

- Start address
- Type: Pub / Sub / None

Please **do not** change Symbol (description), and Length.

To change the Global Data variables of the group box on the **Global Data Configuration** page, follow the instructions below.

Step	Action
1	Adjust the **Distribution Period** cycle. Enter a value from 1 through 50. **Note: Distribution Period** is the minimum number of controller scan times before an update occurs.
2	Before entering a value in the **Group Address** field, identify the station's distribution group. The **Group Address** entry will be an IP address from `224.0.0.0` through `239.255.255.255`. **Group Address:** the Class D Multicast IP address used for a distribution group. Members of this distribution group are configured to use the same group address; therefore, members can communicate with each other using Global Data.
3	Set the timeout in the **Health Time Out** field. This value is measured in milliseconds and can be set to a value that ranges from 50 through 15000 ms (in 50 ms increments) **Note: Health Time Out** is the maximum time between received subscriptions before a subscription is declared unhealthy (inoperable).
4	In the 4x starting address, set the **Data Zone** field.
5	If you are connected to an Ethernet switch that supports multicast filtering, click the **Multicast filtering** check box.
6	Enter %MW word (4x register) location for the Health Bits. This is the location for storing health bits.

NOTE: Health bits run in different directions.

- I/O scanner health bits run left to right.
- Global Data health bits run right to left.

Changing Global Data Variables

To change the Global Data variables that appear in the **Variable Table** area, follow the instructions below.

Step	Action
1	Highlight the identification number in the **Data ID** column.
2	In the **Type** column select the publish/subscribe variable type from the drop down list. Three options are available publish, subscribe, or none, displayed as: ● **NONE** ● **SUB** ● **PUB**
3	In the **Symbol** column you may enter text to describe the variable.

Step	Action
4	In the **Address** column you see the application address for this variable. **Note:** This is a read only field.
5	In the Length column for each row, type a value, which represents the number of 4x registers. The ending 4x register field is automatically updated. If you are using the second method, **Copy Configuration**, update **Length** the first time only.
6	When you are finished, click the **Update Global Data Configuration** button.

Verifying System Operation

To verify that the system is operational, do the following:

Step	Action
1	Verify that every controller is running.
2	Look at the health of variables using the **Global Data Diagnostics** page. Follow these links: **Diagnostics** →**Ethernet** →**Global Data**

Configure Address Server Page

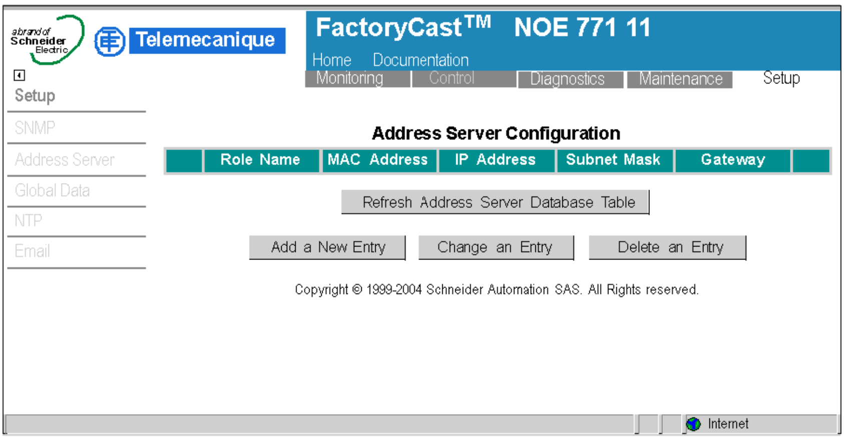
Overview

This topic describes the DHCP and BOOTP address server configuration for the Transparent Factory Ethernet modules.

NOTE: To configure the address server for the 140 NOE 771 *x* FactoryCast Web server modules, see Address Server Configuration/Faulty Device Replacement *(see page 230)*, which describes the BOOTP process.

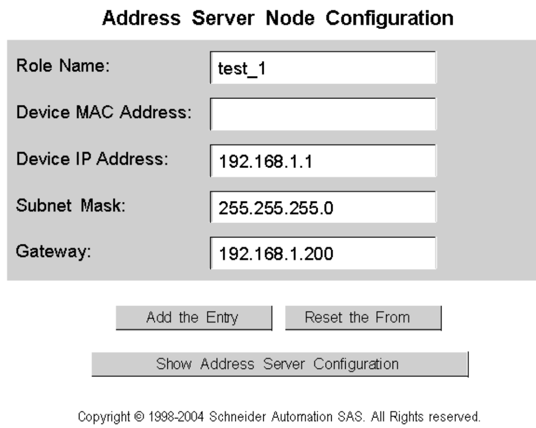
Address Server Page

The Address Server Configuration page appears below:



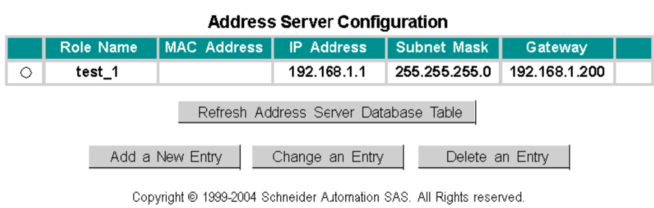
Adding Entries

Create new address server configurations with these steps:

<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>1</td><td>Press Add a New Entry. A dialog appears in the web frame:


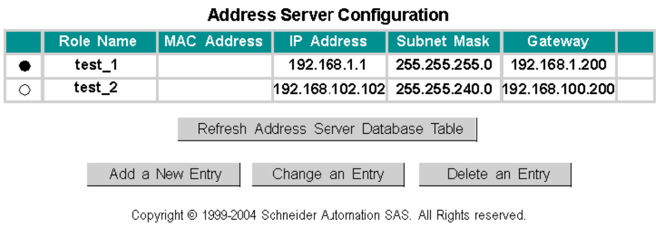
Note: The system does not allow you to enter text in both the Role Name and Device MAC Address text fields. Create either a DHCP configuration (Role Name) or a BOOTP (Device MAC Address) configuration.</td></tr>
<tr><td>2</td><td>For the address server configuration type (DHCP or BOOTP), enter text in the appropriate field:
● Role Name: variable name for DHCP address server configuration
● Device MAC Address: MAC address for BOOTP address server configuration</td></tr>
<tr><td>3</td><td>Enter an address in the Device IP Address field. (We have provided an example address.)</td></tr>
<tr><td>4</td><td>Enter an address in the Subnet Mask field. (We have provided an example address.)</td></tr>
<tr><td>5</td><td>Enter an address in the Gateway field. (We have provided an example address.)</td></tr>
<tr><td>6</td><td>Press the Add the Entry button.</td></tr>
</table>

When you press the **Add the Entry** button, a new entry that corresponds to your input appears in the table in the web frame:



	Role Name	MAC Address	IP Address	Subnet Mask	Gateway	
○	test_1		192.168.1.1	255.255.255.0	192.168.1.200	

When you use the same **Add the Entry** procedure to add subsequent entries, multiple entries appear in the table:



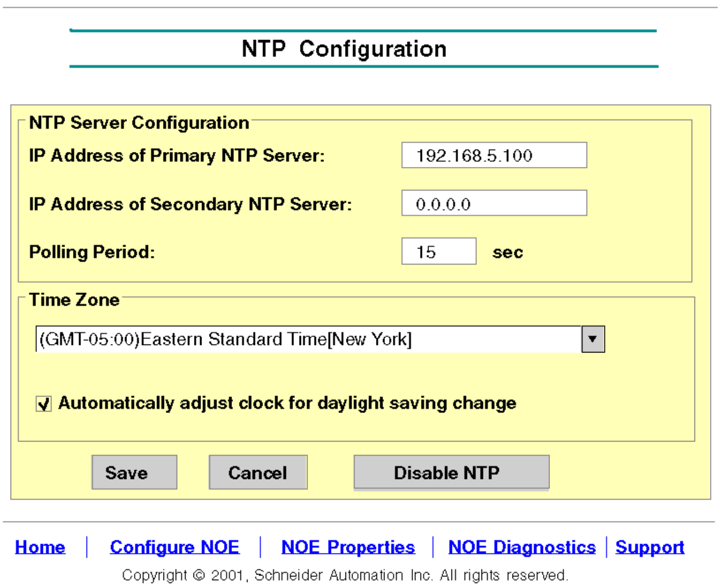
In the above figure, a user has highlighted the radio button for *test_1*. That item can now be edited with the last row of buttons:

Button	Function
Change an Entry	Change the table data for the entry.
Delete an Entry	Delete the entry from the table data for the entry.
Note: The selected radio button has no effect on the **Add a New Entry** function. **Note:** You can press the **Refresh Address Server Database Table** button to refresh the table data at any time.	

Configuring the Time Synchronization Service

Configuring the Time Service

You can configure the time service using either the module’s NTP Configuration page in Unity Pro or the embedded Web page.



Time Service Command Buttons

Execute the following commands:

Button	Description
Save	Stores new NTP (time service) configuration. Previous configuration is no longer valid.
Cancel	Cancels new NTP (time service) configuration. Previous configuration is valid.
Disable NTP	IP of Primary and Standby set = 0. NTP server not polled. Time in controller not updated.

Configurable Time Service Parameters

Configure or change the following parameters on the **NTP Configuration** page.

1. IP address of primary NTP server
   - Enter a valid IP address
2. IP address of secondary NTP server
   - Enter a valid IP address
3. Polling Period (in seconds)
   Enter a value
   - min = 1sec
   - max = 120sec
   - default = 5 sec
4. Time Zone
   - Select from drop-down menu
     Universal Time, Coordinated (GMT) = default
   - Custom time zone
5. Automatically adjust clock for daylight saving change
   - Parameter is selected by default (check mark appears) if daylight saving time is chosen.

Changing Time Service Parameters

To make any changes to the time synchronization service:

Step	Action
1	Enter changes in the appropriate field on the NTP Configuration page for one or all of the configurable parameters.
2	Click **Save**.

Important Information about the Time Service

NOTE: About the time service:

1. **Enable/Disable Daylight Savings Time parameter**: If the Enable/Disable check box is selected, the module automatically corrects the local time to account for daylight savings time. Therefore, no action is required, since the daylight saving time start and end are automatically changed each year.
2. **Polling Time Parameter**: The time (in seconds) is the time between time updates from the NTP server. The default is 5 seconds.
3. **Storing the Time Service Configuration**: The last time service configuration is saved internally in the Ethernet module.
4. **Replacing the Ethernet Module**: If the Ethernet module has to be replaced, the stored configuration is lost, and the system returns to the default configuration.

Customizing Time Zone Parameters

If you want a time zone not listed in the time zone table:

Step	Action	Comment
1	Write the text rules for the custom time zone.	
2	Using an FTP client, store your rules in the file: /FLASH0/wwwroot/conf/NTP/customrules user ID: ntpupdate password: ntpupdate	Root directory to store 'customrules' is set by the FTP server as: /FLASH0/wwwroot/conf/NTP
3	When the rules are written, choose the drop down menu on the **NTP Configuration** web page, and configure (or reboot) the module by selecting Time Zone = Custom	The NTP component looks for customrules, calls the tz compiler and generates a new file called 'tz_custom'. This file is binary file and should not be edited. If the tz compiler detects a syntax error in customrules, the error is logged in the file: /FLASH0/wwwroot/conf/NTP/error.log **1.** NTP component is not launched **2.** NTP Status field in diagnostic web page displays NOT OK.
4	If you want more information, the syntax to write those rules along with a few examples are found in the module in: /FLASH0/wwwroot/conf/NTP/instructions.txt	

Time Zone Parameters

Select a time zone from the drop-down menu.

Time Zone	Description	DST Available
Custom		Yes
(GMT-12:00)	Dateline Standard Time [Eniwetok Kwajalein]	No
(GMT-11:00)	Samoa Standard Time [Midway Is Samoa]	No
(GMT-10:00)	Hawaiian Standard Time [Hawaii Honolulu]	No
(GMT-09:00)	Alaskan Standard Time [Anchorage]	Yes
(GMT-08:00)	Pacific Standard Time [Los Angeles Tijuana]	Yes
(GMT-07:00)	Mexican Standard Time [Chihuahua La Paz Mazatlan]	Yes
(GMT-07:00)	Mountain Standard Time [Arizona Phoenix]	No
(GMT-07:00)	Mountain Standard Time [Denver]	Yes
(GMT-06:00)	Central Standard Time [Chicago]	Yes
(GMT-06:00)	Mexico Standard Time [Tegucigalpa]	No
(GMT-06:00)	Canada Central Standard Time [Saskatchewan Regina]	No
(GMT-06:00)	Central America Standard Time [Mexico_city]	Yes

Time Zone	Description	DST Available
(GMT-05:00)	SA Pacific Standard Time [Bogota Lima Quito]	No
(GMT-05:00)	Eastern Standard Time [New York]	Yes
(GMT-05:00)	Eastern Standard Time [Indiana (East)] [Indianapolis]	No
(GMT-04:00)	SA Western Standard Time [Caracas La Paz]	No
(GMT-04:00)	Pacific SA Standard Time [Santiago]	Yes
(GMT-03:30)	Newfoundland Standard Time [Newfoundland St Johns]	Yes
(GMT-03:00)	E. South America Standard Time [Brasilia Sao_Paulo]	Yes
(GMT-03:00)	SA Eastern Standard Time [Buenos Aires Georgetown]	No
(GMT-02:00)	Mid-Atlantic Standard Time [South_Georgia]	No
(GMT-01:00)	Azores Standard Time [Azores Cape Verde Island]	Yes
(GMT)	Universal Coordinated Time [Casablanca, Monrovia]	No
(GMT0)	Greenwich Mean Time [Dublin Edinburgh Lisbon London]	Yes
(GMT+01:00)	Romance Standard Time [Amsterdam CopenHagen Madrid Paris Vilnius]	Yes
(GMT+01:00)	Central European Standard Time [Belgrade Sarajevo Skopje Sofija Zagreb]	Yes
(GMT+01:00)	Central Europe Standard Time [Bratislava Budapest Ljubljana Prague Warsaw]	Yes
(GMT+01:00)	W. Europe Standard Time [Brussels Berlin Bern Rome Stockholm Vienna]	Yes
(GMT+02:00)	GTB Standard Time [Athens Istanbul Minsk]	Yes
(GMT+02:00)	E. Europe Standard Time [Bucharest]	Yes
(GMT+02:00)	Egypt Standard Time [Cairo]	Yes
(GMT+02:00)	South Africa Standard Time [Johannesburg Harare Pretoria]	No
(GMT+02:00)	FLE Standard Time [Helsinki Riga Tallinn]	Yes
(GMT+02:00)	Israel Standard Time [Israel Jerusalem]	Yes
(GMT+03:00)	Arabic Standard Time [Baghdad]	Yes
(GMT+03:00)	Arab Standard Time [Kuwait Riyadh]	No
(GMT+03:00)	Russian Standard Time [Moscow St. Petersburg Volgograd]	Yes
(GMT+03:00)	E. Africa Standard Time [Nairobi]	No
(GMT+03:30)	Iran Standard Time [Tehran]	Yes
(GMT+04:00)	Arabian Standard Time [Abu Dhabi Muscat]	No
(GMT+04:00)	Caucasus Standard Time [Baku Tbilisi]	Yes
(GMT+04:00)	Afghanistan Standard Time [Kabul]	No
(GMT+05:00)	Ekaterinburg Standard Time [Ekaterinburg]	Yes
(GMT+05:00)	West Asia Standard Time [Islamabad Karachi Tashkent]	No
(GMT+05:30)	India Standard Time [Bombay Calcutta Madras New Delhi]	No
(GMT+06:00)	Central Asia Standard Time [Almaty Dhaka]	Yes

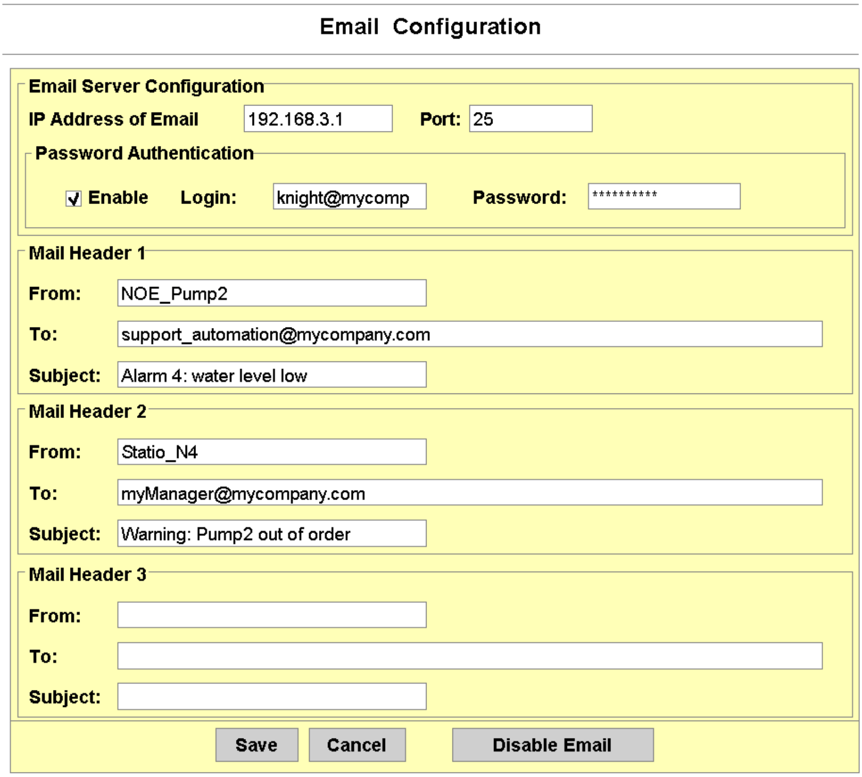
Time Zone	**Description**	**DST Available**
(GMT-05:00)	SA Pacific Standard Time [Bogota Lima Quito]	No
(GMT-05:00)	Eastern Standard Time [New York]	Yes
(GMT-05:00)	Eastern Standard Time [Indiana (East)] [Indianapolis]	No
(GMT-04:00)	SA Western Standard Time [Caracas La Paz]	No
(GMT-04:00)	Pacific SA Standard Time [Santiago]	Yes
(GMT-03:30)	Newfoundland Standard Time [Newfoundland St Johns]	Yes
(GMT-03:00)	E. South America Standard Time [Brasilia Sao_Paulo]	Yes
(GMT-03:00)	SA Eastern Standard Time [Buenos Aires Georgetown]	No
(GMT-02:00)	Mid-Atlantic Standard Time [South_Georgia]	No
(GMT-01:00)	Azores Standard Time [Azores Cape Verde Island]	Yes
(GMT)	Universal Coordinated Time [Casablanca, Monrovia]	No
(GMT0)	Greenwich Mean Time [Dublin Edinburgh Lisbon London]	Yes
(GMT+01:00)	Romance Standard Time [Amsterdam CopenHagen Madrid Paris Vilnius]	Yes
(GMT+01:00)	Central European Standard Time [Belgrade Sarajevo Skopje Sofija Zagreb]	Yes
(GMT+01:00)	Central Europe Standard Time [Bratislava Budapest Ljubljana Prague Warsaw]	Yes
(GMT+01:00)	W. Europe Standard Time [Brussels Berlin Bern Rome Stockholm Vienna]	Yes
(GMT+02:00)	GTB Standard Time [Athens Istanbul Minsk]	Yes
(GMT+02:00)	E. Europe Standard Time [Bucharest]	Yes
(GMT+02:00)	Egypt Standard Time [Cairo]	Yes
(GMT+02:00)	South Africa Standard Time [Johannesburg Harare Pretoria]	No
(GMT+02:00)	FLE Standard Time [Helsinki Riga Tallinn]	Yes
(GMT+02:00)	Israel Standard Time [Israel Jerusalem]	Yes
(GMT+03:00)	Arabic Standard Time [Baghdad]	Yes
(GMT+03:00)	Arab Standard Time [Kuwait Riyadh]	No
(GMT+03:00)	Russian Standard Time [Moscow St. Petersburg Volgograd]	Yes
(GMT+03:00)	E. Africa Standard Time [Nairobi]	No
(GMT+03:30)	Iran Standard Time [Tehran]	Yes
(GMT+04:00)	Arabian Standard Time [Abu Dhabi Muscat]	No
(GMT+04:00)	Caucasus Standard Time [Baku Tbilisi]	Yes
(GMT+04:00)	Afghanistan Standard Time [Kabul]	No
(GMT+05:00)	Ekaterinburg Standard Time [Ekaterinburg]	Yes
(GMT+05:00)	West Asia Standard Time [Islamabad Karachi Tashkent]	No
(GMT+05:30)	India Standard Time [Bombay Calcutta Madras New Delhi]	No
(GMT+06:00)	Central Asia Standard Time [Almaty Dhaka]	Yes

Time Zone	Description	DST Available
(GMT+06:00)	Sri Lanka Standard Time [Columbo]	No
(GMT+07:00)	SE Asia Standard Time [Bangkok Hanoi Jakarta]	No
(GMT+08:00)	China Standard Time [Beijing Chongqing Hong Kong Urumqi]	No
(GMT+08:00)	W. Australia Standard Time [Perth]	No
(GMT+08:00)	Singapore Standard Time [Singapore]	No
(GMT+08:00)	Taipei Standard Time [Taipei]	No
(GMT+09:00)	Tokyo Standard Time [Osako Sapporo Tokyo]	No
(GMT+09:00)	Korea Standard Time [Seoul]	No
(GMT+09:00)	Yakutsk Standard Time [Yakutsk]	Yes
(GMT+09:30)	Cen. Australia Standard Time [Adelaide]	Yes
(GMT+09:30)	AUS Central Standard Time [Darwin]	No
(GMT+10:00)	E. Australia Standard Time [Brisbane]	No
(GMT+10:00)	AUS Eastern Standard Time [Canberra Melbourne Sydney]	Yes
(GMT+10:00)	West Pacific Standard Time [Guam Port Moresby]	No
(GMT+10:00)	Tasmania Standard Time [Hobart]	Yes
(GMT+10:00)	Vladivostok Standard Time [Vladivostok]	Yes
(GMT+11:00)	Central Pacific Standard Time [Magadan Solomon Is New Caledonia]	Yes
(GMT+12:00)	New Zealand Standard Time [Auckland Wellington]	Yes
(GMT+12:00)	Fiji Standard Time [Fiji Kamchatka Marshall Is]	No

Mail Service Configuration

Configuring the Mail Service with the Email Configuration Page

Use the module’s embedded Web page to configure the Electronic Mail Notification service. No other method is available.



Mail Service Command Buttons

Button	Description
Save	Saves the new Email configuration. **Note**: The previous configuration is no longer valid and it is not stored.
Cancel	Cancels the entries in the fields. The previous configuration is valid.
Disable Email	Clears the stored configuration, and disables the email service. **Note**: The next time the service is enabled, a new configuration is required.

Configurable Mail Service Parameters

<table>
<tr><th>Parameter</th><th>Description</th></tr>
<tr><td>IP address of Email</td><td>Enter a valid IP address. (This parameter identifies the SMTP server.)</td></tr>
<tr><td>Port</td><td>Default = 25 (If necessary, you may enter a new value.)</td></tr>
<tr><td>Password Authentication</td><td>If you want to restrict access, enable Password Authentication by entering a check mark in the box.
Enter values for:
Login:
Any printable character allowed
64 character maximum
Password:
Any printable character allowed
64 character maximum</td></tr>
<tr><td>3 mail headers</td><td>Each header contains:
sender’s ID in the From: field
32 character maximum; no spaces
list of recipients in the To: field
Separate each email address with a comma.
128 character maximum
fixed part of message in the Subject: field
(32 character maximum)</td></tr>
<tr><td colspan="2">The Subject field consists of two parts:
1. Fixed (32 character maximum)
2. Dynamic (206 character maximum)</td></tr>
</table>

Ethernet Module Diagnostic Pages

Overview

The **Ethernet** menu contains a list of links for accessing the different diagnostic pages for the Ethernet module:

- Global Data
- I/O Scanning *(see page 195)*
- Messaging
- Bandwidth Monitoring *(see page 84)*
- Statistics
- NTP (network time protocol)
- Email
- Upload MIB File
- Crash Log File

Click a link to access the desired diagnostics utility.

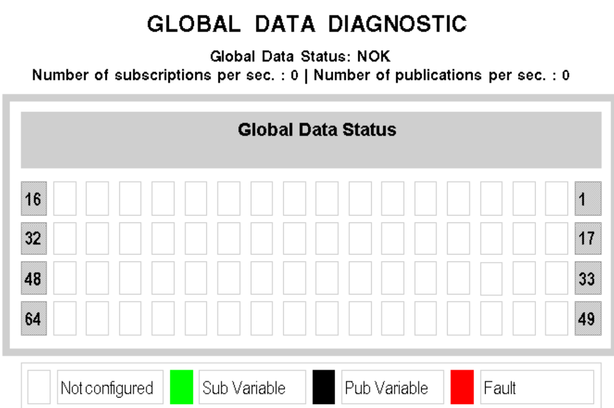
Global Data Page

Information on the general diagnostics of Global Data can be found at the top of this page:

- Status
- Number of publications per second
- Number of subscriptions per second

This page also shows a table of published and subscribed variables in the same distribution group. Each variable is identified by its Identifier:

- Green for the subscribed variables
- Black for the published variables
- White for unconfigured variables
- Red for variables with communication interruptions



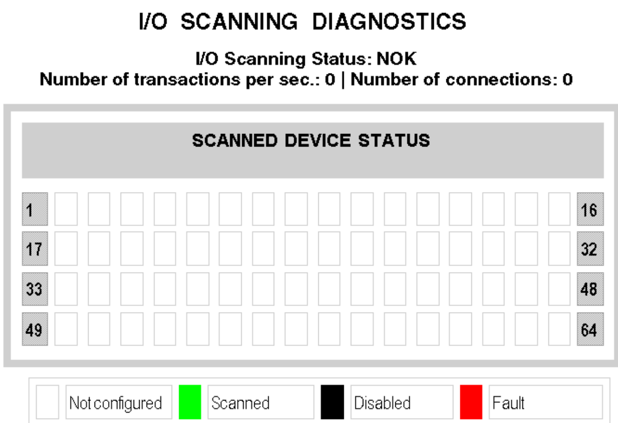
I/O Scanning Page

General diagnostics for the I/O scanning service are shown at the top of this page:
- the I/O scanning status
- the number of transactions per second
- the number of connections

A value of NOK in the **I/O Scanning Status** field indicates that the local system is not scanning. In this case, any data that appears in the **Scanned Device Status** display is meaningless.

A value of OK in the **I/O Scanning Status** field indicates that the values in the **Scanned Device Status** display are reporting the state of scanned devices.

In the **Scanned Device Status** display, the colors that appear in each block indicate the following states for specific remote devices:
- Green indicates that a device is being scanned
- Black indicates that I/O scanning of the specific device has been intentionally disabled via the Device Control Block
- White indicates an unconfigured device
- Red indicates a suspect device



NOTE: Be aware that the green **Scanning** indicator may incorrectly remain green for a remote scanned device after the Ethernet cable gets removed from that device. This inaccurate indication occurs when the health timeout value in the I/O Scanning configuration screen is set to 0 (*see page 200*).

NOTE: You should configure an operational health timeout value in the range 1...65535 (in 1 ms increments). If the health timeout value is set to any value in this range, the **Scanning** indicator reports I/O scanning health correctly.

Messaging Page

This page provides current information on the open TCP connections on port 502.

The number of sent/received messages on the port can be found at the top of this page.

A table provides, for each connection (numbered from 1 to 64):
- remote IP Address
- remote TCP port
- local TCP port
- number of messages sent from this connection
- number of messages received from this connection
- detected event number on this connection

MESSAGING DIAGNOSTICS

Number of Messages sent: 2007 | Number of Messages received: 2007

Conn.	Remote address	Remote port	Local Port	Mess. sent	Mess. received	Error sent.
1	127 . 0 . 0 . 1	1	502	40	40	0
2	192 . 168 . 2 . 10	1240	502	356	356	0
3	139 . 168 . 2 . 10	1247	502	56	56	0

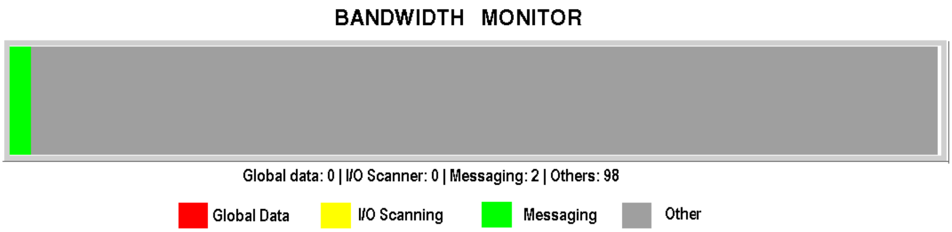
NOTE: Following a request to close a connection, the PLC may hold the connection open in its memory for a few minutes, during which the table will reflect the open connection.

NOTE: Number of Messages received is not reset after a port 502 connection is closed. Therefore, the count indicates the total number of messages that have been received since the module was started.

NOTE: The remote address ‘127.0.0.1’ is used as Private System Connection For Diagnostic Feature or SOAP Communications.

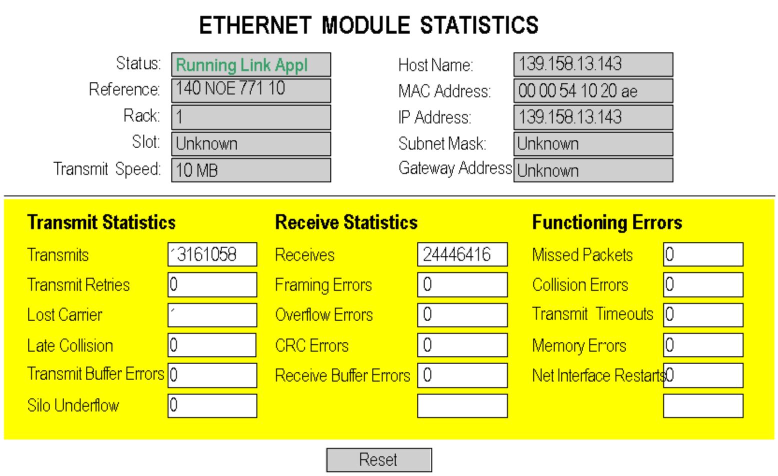
Bandwidth Monitoring Page

This page displays the load distribution of the Embedded Server module between the Global Data utilities, I/O Scanning, Messaging, and other utilities:



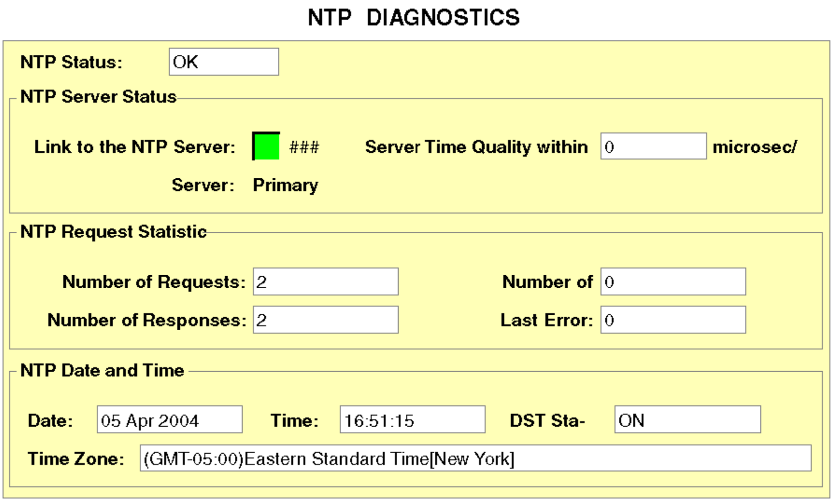
Ethernet Statistics Page

The **Ethernet Module Statistics** page provides information about the status, transmit and receive statistics, and detected errors for the Embedded Server module. Access this page by selecting the NOE module from the local rack or use the hyperlink Statistics. The following graphic is an example **Ethernet Module Statistics** page:



NOTE: Refer to *Modicon Quantum Ethernet TCP/IP Module User Guide* (840 USE 107) and *Modicon Quantum Ethernet TCP/IP Module User Guide* (840 USE 115) for definitions of terms.

NTP Diagnostics Page



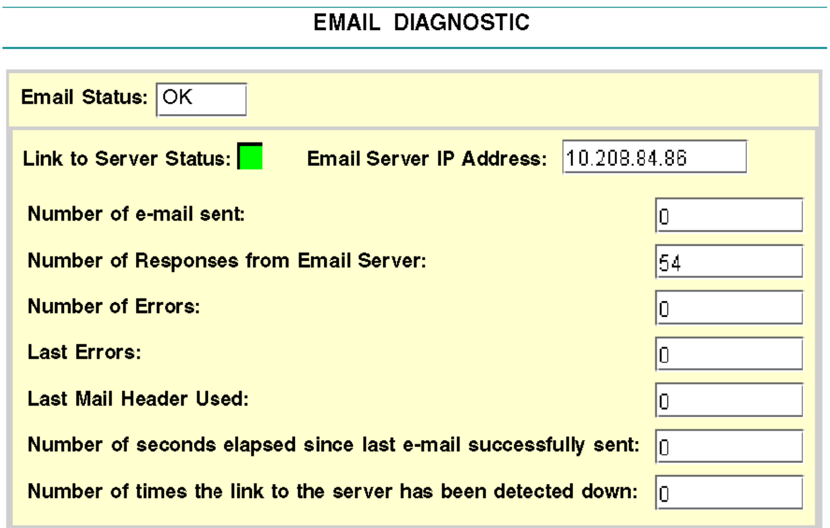
Time synchronization service parameters:

Parameter	Description
NTP status	Service is correctly configured (OK)
NTP server status	NTP client is connected to the NTP server, and if the server is Primary or Standby
NTP requests	Total number of client requests sent to the NTP server
NTP responses	Total number of server responses sent from the NTP server
Number of errors	Total number of unanswered NTP requests
Last error code	Last detected error code received from the NTP client
Date	Date in d/m/y format
Time	Time
Time zone	Time zone plus or minus Universal Time, Coordinated (UTC)
DST	Daylight saving time (DST) parameter is either **1.** on (enabled) **2.** off (disabled)

Last Error field displays values, which indicate the type of detected event.

Type of detected event	Value
Component OK and running	0
Excessive network traffic or server overload	1
Bad parameters in the configuration	3
Component is disabled	4
Incorrect IP	9
Time zone file absent	14
Incorrect syntax in the *customrules* file	15

Email Diagnostics Page

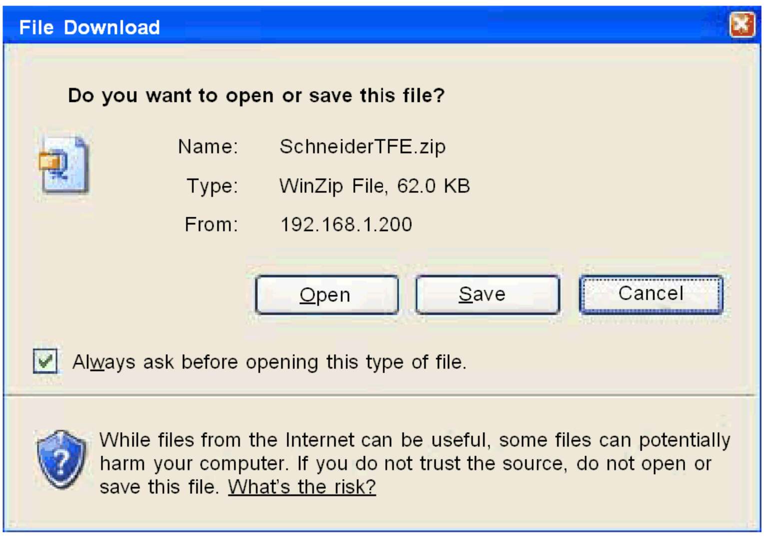


Parameter	Description
Email status	Email service is correctly configured (OK).
Link to Server Status	Ethernet module is connected to the SMTP server. Status is checked at start-up and at least every 30 minutes after start-up: ● Green = module connected to server ● Red = module NOT connected to server
SMTP Server IP Address	IP address of the SMTP server
Number of e-mails sent	Total number of e-mails sent successfully

Parameter	Description
Number of Responses from SMTP Server	Total number of SMTP messages received from the SMTP server
Number of Errors	Total number of e-mails not sent because of a detected error
Last Errors	Reason for the last detected error with a code in hexadecimal. 0 indicates no detected error.
Last Mail Header Used	Last header used by giving the number.
Number of seconds elapsed since last e-mail successfully sent	Counts the number of seconds since the last email was successfully sent.
Number of times the link to the server has been detected down	Number of times the SMTP server could not be reached. (Link checked every 30 minutes.)

Upload MIB File

When you select **Upload MIB File**, the **File Download** dialog box appears. You are asked if you want to save the MIB file or open it:



Crash Log Diagnostics Page

The **Crash Diagnostics** Page displays a crash log file when an unrecoverable event has occurred, and a status message otherwise.

Press **Clear Crash Log File** to clear the log file *(see page 341)*.

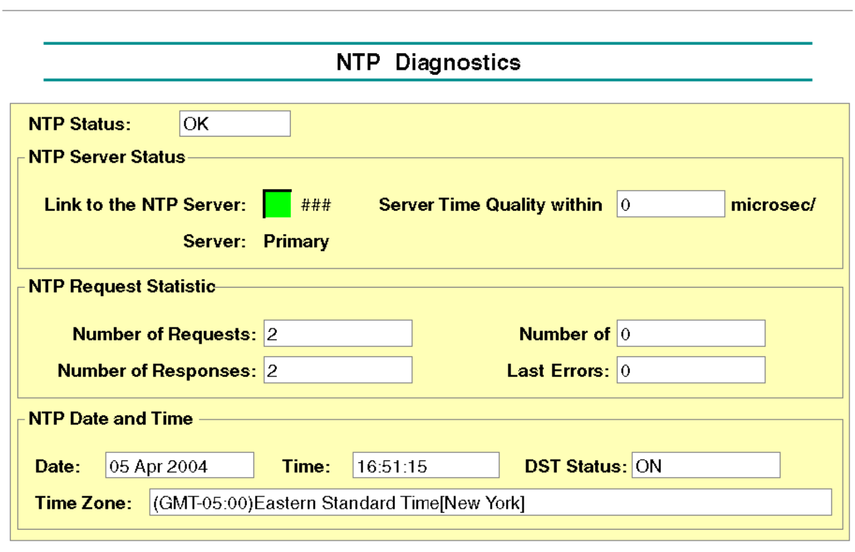
NTP Diagnostics Page

NTP Diagnostic Dialog

Time synchronization service parameters are in the table:

Parameter	Description
NTP status	Service is correctly configured (OK)
NTP server status	NTP client is connected to the NTP server, and if the server is Primary or Standby
NTP requests	Total number of client requests sent to the NTP server
NTP responses	Total number of server responses sent from the NTP server
Number of errors	Total number of unanswered NTP requests
Last error code	Last error code received from the NTP client
Date	Date in d/m/y format
Time	Time
Time zone	Time zone plus or minus Universal Time, Coordinated (UTC)
DST	Daylight saving time (DST) parameter is either **1.** on (enabled) **2.** off (disabled)

The dialog:



Last Error Field

Last Error field displays values, which indicate the type of error.

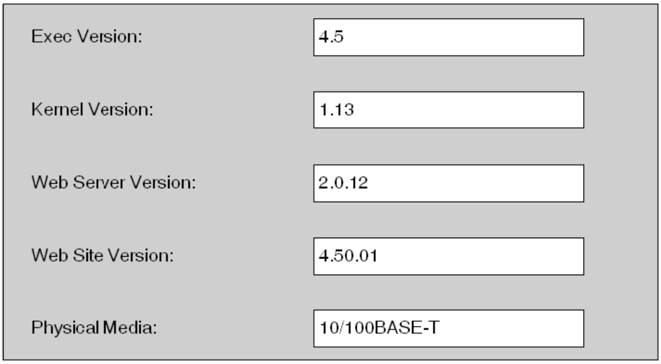
Type of Error	Value
Component OK and running	0
Excessive network traffic or server overload	1
Bad parameters in the configuration	3
Component is disabled	4
Incorrect IP	9
Time zone file absent	14
Syntax error in the customrules file	15

Properties Page

Introduction to the Properties Page

You can navigate to the **Properties** page from the Diagnostics page.The **Properties** page displays the versions of the Exec, Kernel, Web Server, Web Pages and the Physical Media:

PROPERTIES

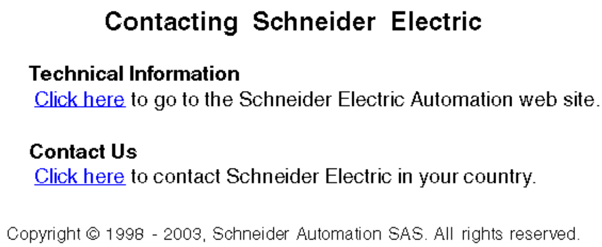


NOTE: This page only reports this information. The fields cannot be changed.

Contacting Schneider Electric Page

Schneider Electric Contact Page

The following figure shows the Contacting Schneider Electric page, which contains information about how to obtain support for the NOE 771 *xx* modules.



Chapter 12
Hot Standby

Overview

The Quantum Ethernt module offers a Hot Standby configuration available for Quantum controllers.

What Is in This Chapter?

This chapter contains the following topics:

Quantum Hot Standby for Unity Pro

The Hot Standby Solution

⚠ CAUTION

RISK OF INCREASE OF TIME TO SWAP

Whenever possible, use of a switch (not a hub) to connect the NOE modules to each other or to the network.

Schneider Electric offers switches. Contact a local sales office for more information.

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

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

Design your application so that unmonitored modules support communication only to noncritical parts of the application.

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

In the hot standby solution, two identically configured PLC systems are set up to control the same application. One PLC, the primary, runs the application and updates the other secondary (standby) PLC. The standby maintains awareness of the application status but does not perform any control functions. In the event of a failure in the primary PLC, the standby PLC takes over the primary PLC responsibilities. When the PLC that has failed becomes operational again, it comes back up in the hot standby system as the new secondary PLC.

The NOEs coordinate the swapping of IP addresses. After closing both the client and the server connections, each NOE sends a swap UDP message to its peer NOE. The sending NOE then waits a specified timeout (500 ms) for the peer swap of UDP messages. Either after receiving the messages or after a timeout, the NOE changes its IP address.

NOTE: NOEs must communicate with each other in order to swap IP addresses. Schneider Electric recommends that you connect the primary and secondary NOEs to the same switch because:

- Communication failures between the NOEs increases the time to swap.
- Connecting 2 NOEs to the same switch minimizes the probability of a communication failure.

NOTE: Schneider Electric recommends that you use a switch (not a hub) to connect the NOEs to each other or to the network. Schneider Electric offers switches; please contact a local sales office for more information.

The NOE waits for either a change in the controller's Hot Standby state or the swap of UDP messages. Then the NOE performs 1 of 2 Hot Standby actions.

If the NOE . . .	Then
Detects that the new Hot Standby state is either primary or standby	The NOE changes the IP address.
Receives a swap UDP message	The NOE transmits a Swap UDP message and swaps the IP address.

All client/server services (I/O scanner, global data, messaging, FTP, SNMP, and HTTP) continue to run after the switch from the old to the new primary NOE.

NOTE: Failure of an NOE module is not a condition for the primary system to leave the primary state.

Hot Standby and NOE Module Functionality

The following table identifies the Ethernet services that are available and unavailable in a hot standby solution.

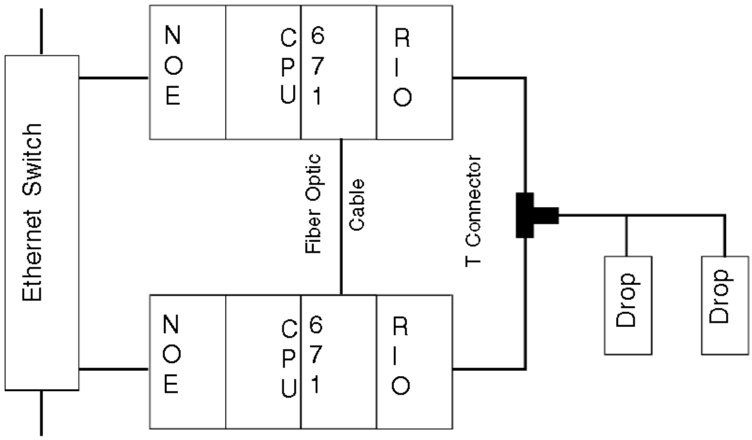
Service	NOE 771 Family
I/O Scanning	Available
Global Data	Available
Modbus Messaging	Available
FTP/TFTP	Available
SNMP	Available
HTTP Server	Available
DHCP	Unavailable

NOTE: Only the 140 NOE 771 01 or 140 NOE 771 11 (TCP/IP Ethernet modules) support a Modicon Quantum Hot Standby with Unity V2.0 system.

Hot Standby Topology

Hot Standby Interconnection

The following diagram shows a Hot Standby system and the relationship between the 2 redundant systems. Two 140 CPU 67• 6• controllers are connected via a link created with fiber optic cable. The RIOs are connected both to each other (through the fiber optic cable) and to the RIO drops.



NOTE: The following three items are required.

1. Two identical systems
2. identical order of modules in each rack
3. identical software revisions

The NOEs are connected to the same switch. Connecting to the same switch is recommended because the NOEs communicate with each other in order to swap the IP address.

There are two reasons to connect to the same switch:
- If a failure to communicate between the NOEs occurs, the time to swap increases.
- To minimize the probability of a failure, connect the two NOEs to the same switch.

The other requirement for the switches is that they are on the same sub-network.

NOE Module Configuration and Hot Standby

TCP/IP Configuration

When the NOE module goes into service the first time, it attempts to get its IP address from a BOOTP server. If a BOOTP server is not available, the NOE module derives its IP address from its MAC address. Connecting to a BOOTP server or deriving the IP address from a MAC address gives you a connection to the NOE, and you can then download a project to the PLC.

All standard rules apply to IP addressing with the additional restriction that the IP address cannot be greater than 253 or broadcast address minus 2. Also, no other device can be assigned the configured IP + 1 address.

140 NOE 771 x1 / 140 NOC 78• 00 IP Address Assignment

Configuring a 140 NOE 771 •1 / 140 NOC 78• 00 Module

Since the primary and standby PLCs in a Quantum Hot Standby system have identical configurations, the configured 140 NOE 771 •1 / 140 NOC 78• 00 module IP addresses are the same. The current local Hot Standby mode determines the IP address.

This table shows how the 140 NOE 771 •1 / 140 NOC 78• 00 module IP addresses are assigned:

Hot Standby State	IP Address
primary CPU	IP address configured in Unity Pro
standby CPU	IP address configured in Unity Pro + 1
transition from primary to offline	IP address configured in Unity Pro, if peer controller **does not** go to primary
transition from standby to offline	IP address configured in Unity Pro + 1

IP Address Restrictions

Do not use either the **broadcast IP address -1** or **broadcast IP address - 2** to configure a 140 NOE 771 •1 / 140 NOC 78• 00 module.

For example, do not configure the primary CPU address as `nnn.nnn.nnn.254`. This causes the standby CPU IP address to be `nnn.nnn.nnn.255`. The standby CPU would then return the diagnostic code **Bad IP configuration**.

IP Address Transparency

⚠WARNING

UNINTENDED EQUIPMENT OPERATION

For a Quantum Hot Standby configuration:
- Do not use the IP address configured in Unity Pro + 1.
- Do not use consecutive IP addresses of the IP address configured in Unity Pro.

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

When a switchover occurs, the new primary PLC takes the IP address of the old primary PLC.

When the PLC that has stopped becomes operational again and rejoins the Hot Standby system, it takes the IP address of the standby PLC.

The new primary 140 NOE 771 •1 / 140 NOC 78• 00 module has the same IP address as the former primary module. The IP address in the secondary module is IP address + 1.

The 140 NOE 771 •1 / 140 NOC 78• 00 modules integrated into the Quantum Hot Standby configuration coordinate this swapping of IP addresses with the management of Ethernet services used.

140 NOE 771 •1 IP Address Swap Time

The following table details the 140 NOE 771 •1 module IP address swap time:

Service	Typical Swap Time	Maximum Swap Time
swap IP address	6 ms	500 ms
I/O scanning	1 initial cycle of I/O scanning	500 ms + 1 initial cycle of I/O scanning

140 NOC 78• 00 IP Address Swap Time

The following table details the 140 NOC 78• 00 module IP address swap time:

Maximum swap time	500 ms (IP address swapping) + connection establishment time (3 s)
Recommended setting for implicit message	Set RPI to 1/2 of MAST cycle time (50 ms maximum)

Timeout multiplier setting for EtherNet/IP (EIP) scanner connections:

MAST Cycle Time (ms)	Recommended RPI (ms)	Timeout Multiplier	Connection Timeout (ms)
20	10	16	160
50	25	8	200
100	50	4	200
200	50	4	200
255	50	4	200

NOTE: The maximum swap time may increase if the end device does not respond in a timely manner.

NOTE: During the swap, there may be disruption in communication between the 140 NOE 771 •1 / 140 NOC 78• 00 module and the end device. Confirm that the application can tolerate this communication disruption.

140 NOE 771 x1 / 140 NOC 78• 00 Operating Modes in Quantum Hot Standby System

Operating Modes

The 140 NOE 771 x1 / 140 NOC 78• 00 module modes are:

- primary mode:
  The Hot Standby mode is primary CPU and client/server services are active.
- standby mode:
  The Hot Standby mode is standby CPU and server services are active except DHCP.
- standalone mode:
  The 140 NOE 771 x1 / 140 NOC 78• 00 module is in a non-redundant system, or the CPU is not present or not healthy.
- offline mode:
  The CPU is stopped.

The Quantum Hot Standby and the 140 NOE 771 x1 / 140 NOC 78• 00 operating modes are synchronized by these conditions:

CPU Module Status	Hot Standby State	140 NOE 771 x1 / 140 NOC 78• 00 Operating Mode
present and healthy	primary CPU	primary
present and healthy	standby CPU	standby
present and healthy	offline	offline
present and healthy	unassigned	standalone
Not present or unhealthy	N/A	standalone

Any of the following events affect the 140 NOE 771 x1 / 140 NOC 78• 00 operating mode:

- a 140 NOE 771 x1 / 140 NOC 78• 00 module is powered up
- a 140 NOE 771 x1 / 140 NOC 78• 00 module executes a Hot Standby switchover
- a 140 NOE 771 x1 / 140 NOC 78• 00 module goes to offline mode
- a new application is downloaded to the 140 NOE 771 x1 / 140 NOC 78• 00 module

IP Address Assignment at Power-Up

A 140 NOE 771 x1 / 140 NOC 78• 00 module obtains its IP address assignment at power-up as follows:

If the Hot Standby state is...	Then the IP address assigned is...
unassigned	Default IP address
primary CPU	IP address configured in Unity Pro
standby CPU	IP address configured in Unity Pro + 1
unassigned to offline transition	See the *Offline Mode at Power-up Sequence* topic in the next table.

If two 140 NOE 771 x1 / 140 NOC 78• 00 modules power-up simultaneously, then:
- Their mode (primary or standby) is determined by the mode of CPU in their rack.
- The primary 140 NOE 771 x1 / 140 NOC 78• 00 has the IP address configured in Unity Pro.
- The standby 140 NOE 771 x1 / 140 NOC 78• 00 has the IP address configured in Unity Pro + 1.

Offline Mode at Power-up Sequence	Result
Controller A powers up before controller B.	• IP address of controller A is the IP address configured in Unity Pro. • IP address of controller B is the IP address configured in Unity Pro + 1.
Both controller A and controller B power up at the same time.	The resolution algorithm assigns the configured IP address to controller A, and it assigns the configured IP address + 1 to controller B.

The 140 NOE 771 x1 / 140 NOC 78• 00 module detects a duplicate IP address, the IP address remains at the default IP and the 140 NOE 771 x1 / 140 NOC 78• 00 module blinks a diagnostic code.

If no IP configuration exists, 140 NOE 771 x1 / 140 NOC 78• 00 modules remains in the standalone mode. The IP address must be obtained for the:
- 140 NOE 771 x1 via a BOOTP server
- 140 NOC 78• 00 based on the MAC address

Ethernet Services at Power Up

The following table shows how the status of a 140 NOE 771 x1 / 140 NOC 78• 00 module's service is affected by the Quantum Hot Standby state:

Hot Standby State	Status of 140 NOE 771 x1 / 140 NOC 78• 00 Services					
	Client Services		Client/Server Services	Server Services		
	EIP Scanner	Global Data	Modbus/EIP Messaging	FTP	SNMP	HTTP
unassigned	not running	run	run	run	run	run
primary CPU	run	run	run	run	run	run
standby CPU	stop	stop	run	run	run	run
offline	stop	stop	run	run	run	run

Hot Standby Switchover

The following table describes how the 140 NOE 771 x1 / 140 NOC 78• 00 modules coordinate a Hot Standby switchover. The 140 NOE 771 x1 module is used as the example in the following procedure.

Step	Action
1	In a Hot Standby configuration NOE A is running in the primary PLC and NOE B is in the standby PLC.
2	NOE A detects that its PLC has changed from primary CPU to the offline mode.
3	NOE A changes from primary NOE to offline with the same Ethernet services running and starts its watchdog timer (with a 500 ms time-out setting). It waits for a message to swap IP addresses from NOE B.
4	NOE B detects that its PLC has changed state from standby PLC to primary CPU.
5	NOE B stops all its Ethernet services, sends a request to NOE A for the synchronization of the IP address swap, starts its watchdog timer (with a 500 ms time-out setting) and waits for a response from NOE A.
6	When NOE A receives the request from NOE B (or after the NOE A watchdog timer times out), it stops all its Ethernet services: ● If it has received a request, NOE B sends a response to NOE A. ● If its watchdog timer has timed out, NOE B does not send a response. NOE A then swaps its IP address and starts the secondary services.
7	NOE B swaps IP addresses and starts Ethernet services as the primary NOE.
8	After NOE A senses that its local CPU changes from offline to standby, it takes the secondary IP address.
9	NOE B now becomes the primary NOE.
10	NOE B opens all client connections, listens for all server connections, and reestablishes those connections.
11	NOE A listens for all server connections and reestablishes those connections.

NOTE: During the Hot Standby switchover, there is a loss of communication during 500 ms between the PLC and the HMI and/or Unity Pro.

Going to Offline

When either the CPU stops or the Hot Standby CPU goes to offline mode, 2 events occur:

1. the 140 NOE 771 x1 / 140 NOC 78• 00 module goes to the offline mode
2. the 140 NOE 771 x1 / 140 NOC 78• 00 module uses the IP address of the present configuration

The IP address assignment when going offline:

Hot Standby State	IP Address Assigned Is...
primary CPU to offline	configured IP address, if other controller **does not** go to primary CPU mode
standby CPU to offline	configured IP address + 1

NOTE: For more information, refer to the 140 NOE 771 x1 / 140 NOC 78• 00 IP Address Assignment topic *(see page 322)*.

NOE IP Address Swap Times

Description

The following table details the 140 NOE 771 •1module address swap times, such as the time to close connections, time to swap IP addresses or time to establish connections:

Service	Typical Swap Time	Maximum Swap Time
Swap IP addresses	6 ms	500 ms
I/O Scanning	1 initial cycle of I/O scanning	500 ms + 1 initial cycle of I/O scanning
Global data	For swap times, please see the *Quantum NOE 771xx Ethernet Modules User Guide* (840 USE 116).	500 ms + 1 CPU scan
Client messaging	1 CPU scan	500 ms + 1 CPU scan
Server messaging	1 CPU scan + the time for the client to reestablish the connection	500 ms + the time for the client to reestablish the connection
FTP/TFTP server	The time for the client to reestablish the connection	500 ms + the time for the client to reestablish the connection
SNMP	1 CPU scan	500 ms + 1 CPU scan
HTTP server	The time for the client to reestablish the connection	500 ms + the time for the client to reestablish the connection

Network Effects of Modicon Quantum Hot Standby with Unity Solution

Overview

The Modicon Quantum Hot Standby with Unity Pro solution is a powerful feature of NOEs, a feature that increases the reliability of your installation. Hot Standby uses a network, and using the Hot Standby feature over a network can affect the behavior of:

- browsers
- remote and local clients
- I/O scanning service
- global data service
- FTP/TFTP server

The following are factors you may encounter while using the Modicon Quantum Hot Standby with Unity solution.

Browsers

If a browser requests a page and during the process of downloading that page an IP address switchover occurs, the browser either hangs or times out. Click the **Refresh** or **Reload** button.

Remote Clients

Hot Standby switchover affect remote clients.

A NOE module resets under the following conditions:

Remote Connection Request during Hot Standby : If a remote client establishes a TCP/IP connection during a Hot Standby switchover, the server closes the connection using a TCP/IP reset.

Hot Standby Switchover during Remote Connection Request : If a remote client makes a connection request and a Hot Standby switchover occurs during the connection request, the sever rejects the TCP/IP connection by sending a reset.

Outstanding Requests : If there is an outstanding request, the NOE module does not respond to the request, but the NOE module does reset the connection.

The NOE module does a Modbus logout if any connection has logged in.

Local Clients

During a switchover, the NOE module resets all client connections using a TCP/IP reset.

I/O Scanning Service

⚠WARNING
UNINTENDED EQUIPMENT OPERATION - DEVICES GO TO THEIR FALLBACK STATES DURING SWITCHOVER
Configure Ethernet output devices to their Hold Last Value fallback state whenever possible. Output devices that support only a Set to Zero fallback state may produce a pulse during switchover.
Failure to follow these instructions can result in death, serious injury, or equipment damage.

I/O scanning provides the repetitive exchange of data with remote TCP/IP nodes I/O devices. While the PLC is running, the primary CPU NOE sends Modbus read/write, read, or write requests to remote I/O devices, and transfers data to and from the PLC memory. In the secondary controller, the I/O scanning service is stopped.

When the Hot Standby switchover occurs, the primary CPU NOE closes all connections with I/O devices by sending a TCP/IP reset. The I/O scanning service in this NOE is standby CPU.

After the switchover, the new primary CPU NOE re-establishes the connection with each I/O devices. It restarts the repetitive exchange of data with these re-connections.

The module provide the I/O scanning feature. Configure this feature with the Unity Pro software.

Using either method, the configuration and transfer of data between network addresses can be done without using the MSTR/IEC function block.

NOTE:

You must account for the following Ethernet I/O scanning considerations during a switchover.

- If MSTR/IEC function block is used for TCP/IP, only some of the opcode is used. Therefore, the block does not complete its transaction, and it returns error code 0•8000.
- While the NOE module is in the process of performing the transaction, a new MSTR/IEC function block may become active.
- The output states of the scanned I/Os follows the state defined in the last value option configured in the I/O scanning table of the NOE module (in Unity Pro software).
  These 2 states are either:
  a. set to 0
  b. hold last

NOTE: With the above considerations, we recommend using switchover with Ethernet I/O scanning for less critical applications.

Global Data (Publish/Subscribe) Service

The Hot Standby primary CPU NOE is 1 station within a distribution group. Distribution groups exchange application variables. Exchanging application variables allows the system to coordinate all the stations in the distribution group. Every station publishes local application variable in a distribution group for all other stations and can subscribe to remote application variables independent of the location of the producer.

The communication port has only 1 multicast address.

In this network service, the Modicon Quantum Hot Standby with Unity controllers are viewed like only 1 station. The primary CPU NOE publishes the Hot Standby application variables and receives the subscription variables. The secondary NOE global data service is in a stopped state.

When the Hot Standby switchover occurs, the primary CPU NOE stops the global data service. The NOE module does not publish the local variable during a switchover. And after the switchover, the new primary CPU NOE starts to publish application variables and to receive the subscription variables.

FTP/TFTP Server

The FTP/TFTP server is available as soon as the module receives an IP address. Any FTP/TFTP client can logon to the module. Access requires the correct user name and password. Modicon Quantum Hot Standby with Unity allows only 1 active FTP/TFTP client session per NOE module.

When the Hot Standby switchover occurs, the primary CPU and secondary NOEs close the FTP/TFTP connection. If you send an FTP/TFTP request during the switchover, the communication is closed.

Whenever you re-open communication, you must re-enter a user name and a password.

Appendices

Introduction

The appendices provide supplementary reference information for the Quantum 140 NOE 771 *xx* series of modules and the 140 NWM 100 00 module.

What Is in This Appendix?

The appendix contains the following chapters:

Appendix A
Maintenance

Introduction

This chapter details information about system maintenance including accessing and clearing the crash log and downloading the new NOE exec.

What Is in This Chapter?

This chapter contains the following topics:

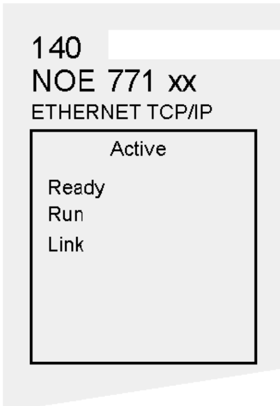
Responding to Errors

Overview

The following information describes how to respond to errors on the NOE module.

Detecting Errors

When faults occur, the NOE module’s LED display can help you determine what went wrong. The following figure shows the pattern that the LEDs should display during normal operation.



The **Run** indicator will be solid. The **Coll** (collision) LED may flash, indicating that collisions are occurring on the Ethernet network. Such collisions are normal.

If a fault occurs, the normal LEDs may be extinguished or other indicators may light. This topic discusses errors reported by the **Active**, **Ready**, **Coll**, **Link**, **Kernel**, **Appl** and **Fault** indicators.

For each type of error, try the suggested remedies in the order given. If no remedy suggested here overcomes the error, call your local service representative or call Schneider Electric customer service at 1-800-468-5342 for further directions.

Procedure for Responding to an Active LED Error Indicator

If the Active LED fails to light, the NOE module is not communicating with the backplane . The following procedure describes the steps to perform to respond to an Active LED error.

Step	Action
1	Make sure the NOE module and the controller are installed properly.
2	Verify that the controller is working; if it is not, replace it.
3	If neither the new controller nor the NOE module functions, replace the backplane.

Step	Action
4	Make sure that the number of network option modules (including NOE, NWM, NOM, and CRP 811 modules) in the backplane is not exceeded: ● 140 CPU 311 10: 2 modules ● 140 CPU 670 60: 3 modules ● 140 CPU 434 12A, 140 CPU 534 14A, 140 CPU 651 *x*0, 40 CPU 652 60, 140 CPU 658 60, 140 CPU 671 60, 140 CPU 672 60, 140 CPU 672 61, 140 CPU 678 61: 6 modules
5	Check the version of the controller executive. You must have version 2.0 or later to support the Ethernet module. Earlier versions do not recognize the module.
6	If steps 4 and 5 above check out ok, replace the NOE module.

Procedure for Responding to a Ready LED Error Indicator

If the **Ready** LED fails to light, the NOE module has failed internal diagnostic tests. The following procedure describes the steps to perform.

Step	Action
1	Make sure that power has been applied to the backplane.
2	If step 1 checks out ok, replace the NOE module.

Procedure for Responding to a Link LED Error Indicator

If the **Link** LED fails to light, the NOE module is not communicating with the Ethernet hub/switch. The following procedure describes the steps to perform to respond to a **Link** LED error.

Step	Action
1	Make sure that the cable has been installed correctly and the module is functioning properly.
2	Verify that the hub/switch is working properly.
3	If steps 1 and 2 check ok, replace the NOE module.

Kernel LED Error

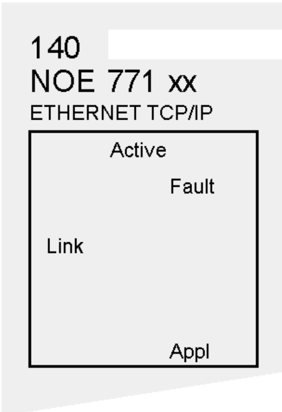
The following table describes the **Kernel** LED errors that may occur and how to respond to them.

If	Then
The **Ready** LED is on and the **Kernel** LED is flashing	The module has detected an invalid software image.
The **Ready** LED is on and the **Kernel** LED is shining steadily,	An attempt to download a software image has failed and the module is in kernel mode.
Either of the above conditions exists.	Download a new NOE Exec *(see page 342)*.

Fault LED

The **Fault** LED will flash briefly following an error as the module attempts to recover.

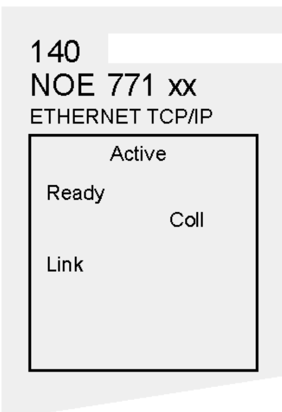
The following figure shows the **Fault** LED.



Collision LED Error

If the twisted pair cable has not been connected properly, the **Coll** LED will shine steadily and the **Link** LED will be extinguished. (This condition does not occur with fiber optic modules.)

The following figure shows the Collision LED.



Procedure for Responding to a Collision LED Error

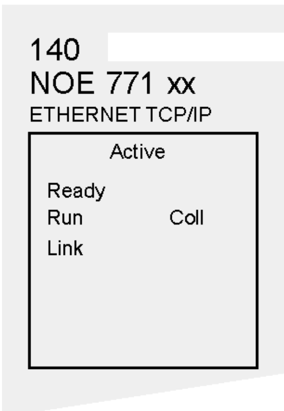
If the Collision LED fails to light, use the following procedure.

Step	Action
1	Make sure that the cable has been installed and is working properly.
2	Verify that the Ethernet Hub/Switch is working properly.

Collision LED Normal Condition

If the **Coll** LED is flashing, the module is reporting collisions on the Ethernet network. While such collisions are normal, the frequency of the flashes is an indication of the volume of traffic on the network. The flashes may be so frequent that the LED appears to be shining steadily. Heavy traffic will slow communications. If response time is important to your application, you should consider segmenting your network to reduce the frequency of collisions.

The following figure shows the Collision LED under normal conditions.



Run LED

The following table describes the action to be taken if the **Run** LED is flashing. The action depends on the number of flashes in sequence.

Number of Flashes in Sequence	Action
Three	Check Ethernet connection
Four	Change IP address
Five	Provide IP address
Six	Connect using default IP address and configure
Seven	Download a new NOE Exec *(see page 342)*

Application LED

If the module crashes, it will note the reason in a log. If the module is able to recover, the **Appl** LED will light, indicating that an entry has been made in the crash log. You can read and clear the crash log *(see page 341)*.

Reading and Clearing the Crash Log

Overview

The following information describes the crash log.

Introduction

The crash log provides you with the ability to capture conditions that lead to an anomalous condition. By providing the crash log to Schneider Electric technical support, you can facilitate their assistance in resolving your problems.

NOTE: The crash log is provided with the understanding that, with a complex product in thousands of customer applications, there may be conditions that require advance diagnostics. The crash log is one of the tools used to solve complex problems.

The Crash Log

If the **Appl** (application) indicator is on, entries have been made in the crash log. The log may hold up to 64K of entries.

Reading the Crash Log

The crash log can be read from the embedded web pages or via FTP.

Procedure for Reading the Crash Log via FTP

To access the crash log via FTP:

Step	Action
1	Log in to the module’s FTP server.
2	Change the directory to *wwwroot/conf/diag*.
3	Perform an FTP to get the crash log file: `get crash.log`

Clearing the Crash Log

The crash log can be cleared from the embedded web pages or via FTP.

Procedure for Clearing the Crash Log via FTP

To access the crash log via FTP:

Step	Action
1	Log in to the module’s FTP Server.
2	Change the directory to *wwwroot/conf/diag.*
3	Perform an FTP to delete the crash log file: `rm crash.log`

Downloading a New NOE Exec

Introduction

The following tools can be used to download a new NOE Exec:

- Schneider Electric programming packages (see corresponding manuals)
- FTP

Use the OS loader to update the NOE Executive and web pages. (Refer to the Unity Pro documentation.)

Downloading a New NOE Exec via FTP

Exec Version

Please check the current NOE Exec file version on the **NOE Properties** Web page.

Follow these links: | **Diagnostics** | **NOE Properties** |

Procedure

The following procedure describes the steps to use to download a new NOE Exec via FTP. An example follows the procedure.

Step	Action
1	At the DOS prompt, type `FTP`, followed by the IP Address and press **Enter**.
2	At the User prompt, type: `USER` and press **Enter**.
3	At the password prompt, enter your FTP Password and press **Enter**.
4	At the FTP prompt, type `cd wwwroot/conf/exec` and press **Enter**.
5	At the FTP prompt, type `put` and press **Enter**. **Note:** Pay attention that the NOE771xx.bin has to be to the local path on your PC (default path: `c:\`).
6	At the local file prompt, type `NOE771xx.bin` and press **Enter**.
7	At the remote file prompt, type `NOE771xx.bin` and press **Enter**.
8	After the transfer is complete you must reboot the NOE to allow the new EXEC to become operational. **Note:** The file name is case-sensitive and must be entered with the name in uppercase and the extension in lowercase as shown in the figure below. For example: `NOE771xx.bin`

Sample FTP Session

The following FTP session was used to download an NOE Exec.

```
Command Prompt - ftp 205.217.193.173

C:\noe77100>ftp    205.217.193.173
331  Password  required
Password:
230  User  logged  in
ftp>  cd  wwwroot/conf/exec
250  Changed  directory  to  "/FLASH0/wwwroot/conf/exec"
ftp>  put
(local-file)   NOE77100.bin
(remote-file)   NOE77100.bin
200  Port  set  okay
150  Opening  BINARY  mode  data  connection
226  Transfer  complete
485376  bytes  sent  in  3.06  seconds  (158.41  Kbytes/sec)
ftp>  dir
200  Port  set  okay
150  Opening  BINARY  mode  data  connection
-rwx---A--  1  user                                   2  kerVer
-rwx---A--  1  user                          485376  NOE77100.bin
226  Transfer  complete
86  bytes  received  in  0.01  seconds  (8.60  Kbytes/sec)
ftp>  _
Connected  to  205.217.193.173.
220  VxWorks  FTP  server  (VxWorks  5.3.1)  ready.
User  (205.217.193.173:(none)):  USER
```

NOTE: The NOE Kernel can not be downloaded via FTP.

Reboot Information after FTP

NOTE: After downloading by FTP, reboot the module.

NOTE: Perform a download or a reboot when your system can tolerate these actions.

Downloading a New NOE Kernel

Procedure

The NOE Executive (Exec) adds a new feature that allows updating of the low level Kernel within the NOE module's firmware. For the proper installation of new kernel firmware, use the following procedure.

Step	Action
1	Check the current version of the NOE module's Executive firmware (Exec file).
2	If the Exec is not the appropriate version, the Exec must be updated before updating the Kernel.
3	Use the EXECLoader to load the latest version of the EXEC.
4	After loading the new Exec and before loading the Kernel, make sure to cycle power to the NOE module.
5	Load the Kernel using the EXECLoader.
6	After the transfer displays as Successful, the NOE module needs approximately 1 minute to burn the new Kernel into its Flash memory.
7	The NOE module goes go through a reboot sequence.

Kernel Version

⚠CAUTION

UNINTENDED EQUIPMENT OPERATION

Failure to perform the preceding update procedure will render the NOE module inoperable.

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

The NOE Kernel can not be downloaded via FTP.

Please check the current NOE Kernel version on the NOE Properties *(see page 314)* Web page.

Follow these links: | **Diagnostics** | **NOE Properties** |

Appendix B
Specifications

Specifications

140 NOE 771 •• Specification Table

The main specifications for the Quantum 140 NOE 771 xx Ethernet module are described in the following table:

Communication Ports	One auto-sensing 10/100 BASE-T shielded twisted pair (RJ-45 connector) port and one 100 BASE-FX (MT-RJ connector) port. Both ports transmit and receive Modbus commands encapsulated in TCP/IP protocol
Bus Current Required	750 mA
Power Dissipation	3.8 W
Fuse	None
Programming Software	
Type and version	Concept, Ver. 2.2, or higher
	Unity Pro, Ver. 1.0, or higher
	Modlink, Ver. 2.0, or higher
	Modsoft, Ver. 2.6, or higher
	ProWORX NxT, Ver. 2.1, or higher
Firmware	
CPU Type and version	Quantum Executive, Ver. 2.0, or higher
NOE Upgradeable	Field Upgradeable via FTP or Programming Panel.
Operating Conditions	
Temperature	0 to +60° C
Humidity	0 to 95% Rh non condensing @ 60° C
Altitude	15,000 ft (4500 m)
Vibration	10-57 Hz @ 0.0075 mm d.a
	57-150 Hz @ 1 g
Storage Conditions	
Temperature	-40 to +85° C
Humidity	0 to 95% Rh non condensing @ 60° C
Free Fall	1 m unpackaged
Shock	3 shocks / axis, 15 g, 11 ms

140 NWM 100 00 Specification Table

The main specifications for the Quantum 140 NWM 100 00 Ethernet module are described in the following table:

Specification	Description
Communication Ports	One auto-sensing 10/100 BASE-T shielded twisted pair (RJ-45 connector) port and one 100 BASE-FX (MT-RJ connector) port. Both ports transmit and receive Modbus commands encapsulated in TCP/IP protocol
Bus Current Required	900 mA
Power Dissipation	4.5 W
Fuse	None
Programming Software	
Type and version	Concept, Ver. 2.6, or higher
	Unity Pro, Ver. 1.0, or higher
Firmware	
CPU Type and version	Quantum Executive, Ver. 2.6, or higher
NOE Upgradeable	Field Upgradeable via FTP or Programming Panel.
Operating Conditions	
Temperature	0 to +60° C
Humidity	0 to 95% Rh non-condensing @ 60° C
Altitude	15,000 ft (4500 m)
Vibration	10-57 Hz @ 0.0075 mm d.a
	57-150 Hz @ 1 g
Storage Conditions	
Temperature	-40 to +85° C
Humidity	0 to 95% Rh non-condensing @ 60° C
Free Fall	1 m unpackaged
Shock	3 shocks/axis, 15 g, 11 ms
Immunity	
International Standard	IEC 61131-2
US Standard	UL 508
European Standard	EN61131-2, EN50081-2
Canadian Standard	CAN/CSA C22.2 No. 142
Agency Approvals	UL: UL 508 CSA: CSA 142 CE: EN61131-2 Factory Mutual Class 1 Division 2

Appendix C
Quantum Ethernet TCP/IP Modbus Application Protocol

Introduction

This chapter describes the Quantum Ethernet TCP/IP Modbus Application Protocol.

What Is in This Chapter?

This chapter contains the following topics:

Modbus Application Protocol Overview

Introduction

The following information describes the Modbus Application Protocol (MBAP).

The Modbus Application Protocol (MBAP) is a layer-7 protocol providing peer-to-peer communication between programmable logic controllers (PLCs) and other host-based nodes on a LAN. Collectively, these nodes implement all or part of a control application used for industrial automation applications in the automotive, tire and rubber, food and beverage, and utilities industries, to name a few

Client-Server Communications

Modbus protocol transactions are typical request-response message pairs between a client node and a server node. These nodes function as follows

Client : The node that initiates a data transaction is called a *client*. The Modicon Quantum Ethernet module provides the user with the capability to transfer data to and from nodes on a TCP/IP network using a communication instruction. All PLCs that support networking communication capabilities over Ethernet can use either the MBP_MSTR Ladder Logic instruction to read or write controller information or IEC communication blocks.

Server : The node that receives an inquiry is the *server*. Using the standard Modbus/TCP protocol, the Modicon Quantum Ethernet module provides access to controller data. Any device, PC, HMI package, another PLC, or any Modbus/TCP compliant device can access data from the PLC. The Modbus/TCP server also allows programming panels to log into the controller over Ethernet.

Modbus requests contain function codes representing several classes of service including data access, online programming, and program download and upload classes. Modbus responses can be ACKs with and without data, or NACKs with error information.

The Modbus Application Protocol can be transmitted over any communication system that supports messaging services. However, the current Quantum implementation transports Modbus Application Protocol PDUs over TCP/IP. The Quantum PLC accommodates both Ethernet II and IEEE 802.3 framing, although Ethernet II framing is the default.

Limitations

The Modicon Quantum Ethernet module supports up to 64 simultaneous Modbus/TCP server connections. To guarantee consistency of changes to the controller configuration, the module allows only one programming panel to be logged in at a time.

The module supports these Modbus/TCP commands:

- Read Data
- Write Data
- Read/Write Data
- Get Remote Statistics
- Clear Remote Statistics
- Modbus 125 Commands (used by programming panels to download a new exec to the module

For More Information

For more information about Modbus communications, consult the *Modbus Protocol Reference Guide* (PI-MBUS-300). For more information about communication blocks, consult Chapter 4, Transferring Data Using Communication Blocks *(see page 117)*.

Modbus Application Protocol PDU

Overview

The following information describes the structure and content of the Modbus Application Protocol PDU.

Description

The Modbus Application Protocol PDU, mbap_pdu, is received at TCP port number 502. The current maximum size of the mbap_pdu for this class of services is 256 bytes. The structure and content of the mbap_pdu is defined to be:

```
mbap_pdu::={inv_id[2], proto_id[2], len[2],dst_idx[1], data=mb_pdu}
```

The header is seven bytes long and includes the fields listed in the following table:

Field	Description
inv_id	[2 bytes] invocation id used for transaction pairing
proto_id	[2 bytes] used for intra-system multiplexing, default is 0 for Modbus services
len	[2 bytes] the len field is a byte count of the remaining fields, and it includes the dst_id and data fields

The remainder of the pdu includes two fields:

Field	Description
dst_idx	[1 byte] destination index is used for intra-system routing of packets (currently not implemented)
data	[n bytes] this is the service portion of the Modbus pdu, mb_pdu, and it is defined below

The service portion of the Modbus Application Protocol, called mb_pdu, contains two fields:

```
mb_pdu::={func_code[1], data[n]}
```

The following table describes the fields in mb_pdu.:

Field	Description
func_code{1 byte	Modbus function code
data	[n bytes] this field is function code dependent and usually contains information such as variable references, variable counts, and data offsets

The size and content of the data field are dependent on the value of the function code.

Example

Here are the values for a sample mbap_pdu for reading a register:

```
00 01 00 00 00 06 01 03 00 00 00 01
```

The following table shows the structure and content for this example:

inv_id	00 01	
	proto_id	00 00
	len	00 00
	dst_idx	01
	func_code	03
	data	00 00 00 01

Modbus Application Protocol Service Classes

Introduction

There are several classes of service that are part of the Modbus Application Protocol. Each of these classes is described below.

Data Access

Read/write both discrete and analog data values from PLC register files.

Online Programming

Services make relatively minor alterations to ladder logic programs with a highly controlled introduction of these changes into the executing program.

Image Download/Upload

Image download services support the downloading of a ladder logic control program to the PLC. Image upload services support the uploading of a ladder logic control program from a PLC to PC host for archival/backup purposes.

Configuration

Configuration services allow the user to define parameter values which affect the PIC’s register files, I/O map, communication port configuration and scan attributes, to name a few.

Device Execution State Control

The class of service allows the user to start/stop the PLC scan execution. These services require the user to be in an application login context which is obtained through other Modbus services.

Modbus Application Protocol PDU Analysis

Overview

The following information provides an analysis of the Modbus Application Protocol.

Analysis

The Modbus Application Protocol PDU is transmitted over a TCP/IP Ethernet stack. Both Ethernet II and IEEE 802.3 framing will be accommodated. Ethernet II framing is the default.

```
    . . .   from the wire in for IEEE 802.3 framing   . . .
    . . .   is IEEE 802.3 framing if length <=1500   . . .802.3_pdu ::=
{dst_addr[6], src_addr[6], length[2], data=802.2_pdu}    *an IEEE 802.3
PDU has a maxFrameSize of 1518 octets
   *an IEEE 802.3 PDU has a minFrameSize of 64 octets802.2_pdu : {dsap[1],
ssap[1], frm_cntrl[1], snap_hdr[5], data=ip_pdu}    *the snap_hdr is
associated with a "well-known" 802.2 sap snap_hdr
     ::={org_code[3], ethertype[2] }

    *the snap hdr (sub network access protocol) allows the older style
    Ethernet protocols to run on the newer IEEE 802.2 interface.  The
    ethertype parameter indicates the service, ex. ip or arp.  IP has a
value
    0x800.    . . .   from the wire in for Ethernet II framing   . . .
    . . .   is Ethernet II framing if length >1500   . . .802.3_pdu ::=
{dst_addr[6], src_addr[6], length[2], data=ip_pdu}    . . .   the common
part of the packet begins here   . . .ip_pdu ::= {ip_hdr[20],
data=tcp_pdu}tcp_pdu ::= {tcp_hdr[24], data=appl_pdu=mbap_pdu}
```

The mbap_pdu is the Modbus Application Protocol whose messages are received at a well-known port. The current maximum size of the mbap_pdu for this class of services in 256 bytes.

Structure and Content

The structure and content of the mbap_pdu is defined to be:

```
mbap_pdu ::={ inv_id[2], proto_id[2], len[2], dst_idx[1], data=mb_pdu
}The header is 7 bytes long, and includes the following fields:
```

```
   inv_id         [2 bytes] invocation id used for transaction pairing
proto_id       [2 bytes] used for intra-system multiplexing, default is
0 for Modbus
    services    len               [2 bytes] the len field is a byte count
of the remaining fields and
    includes the dst_id and data fields.
```

The remainder of the pdu includes two fields:

```
    dst_idx        [1 byte] destination index is used for intra-system
routing of
    packets. (currently not implemented)    data            [n bytes]
this is the service portion of the Modbus pdu, mb_pdu, and is
    defined below
```

The service portion of the Modbus Application Protocol, called mb_pdu, contains 2 fields:

```
mb_pdu ::= { func_code[1], data[n] }
```

```
    func_code    [1 byte] MB function code    data            [n bytes]
this field is function code dependent and usually contains
    information such as variable references, variable counts, and data
offsets.
```

The size and content of the data field are dependent on the value of the function code.

TCP/IP Specific Issues

Overview

The following information describes some TCP/IP specific issues.

Broadcast/Multicast

Although broadcast and/or multicast are supported by both IP network address and IEEE 802.3 MAC address, the Modbus Application Protocol does not support either broadcast or multicast at the application layer.

Schneider Electric’s Quantum PLCs use broadcast addressing because they use ARP as the means of locating the destination node. The client interface to the Modbus Application Protocol service on the PLC, the MBP_MSTR block, requires the user to provide the destination IP address. Also the embedded stack does use a pre-configured default gateway IP address in the case where ARP does not succeed.

TCP Port Number

Schneider Electric has obtained a well-known system port from an Internet Authority. Schneider Electric’s well-known system port number is 502. The Internet Authority assigned the system port number 502 to asa-appl-proto with Dennis Dubé as the company point of contact.

This port number allows Schneider Electric to transport various application protocols over with TCP or UDP. The particular protocol is indicated by the value of the proto_id parameter in the mbap_pdu. Currently the only assignment is 0 meaning Modbus Application Protocol.

Closing a TCP Connection

Overview

A TCP connection can be closed by one of the following:

- a remote station (which closes a connection by sending a TCP/IP connection closure message)
- the local PLC

When the Maximum Number of Connections are Open

If the PLC receives a request to open a new connection when the maximum number of connections has been reached, the PLC:

- closes an open but inactive connection
  - then -
- opens a new connection

To identify which connection to close, the PLC examines groups of connections for inactive open connections, in the following sequence:

1 non-referenced connections to devices that are not configured as part of the Unity application
2 client connections
3 server connections

If the PLC discovers one or more inactive connections in the first group, it closes the oldest inactive connection in that group then opens a new connection.

If no inactive connection is discovered in the first group, the PLC examines the second group of connections and, if it discovers one or more inactive connections in the second group, the PLC closes the oldest inactive connection in that group and opens a new connection.

If no inactive connection is discovered in the first and second groups, the PLC examines the third group and, if it discovers one or more inactive connections in the third group, the PLC closes the oldest inactive connection in that group and opens a new connection.

If the PLC fails to discover an inactive connection in any of the three groups, no open connection is closed and no new connection can be opened.

NOTE: Closure of a connection is indicated to the application by means of a status report (message refused) on any exchanges in progress.

Reference Documents

Overview

The following information provides a list of reference documents that you may find helpful.

Introduction

Following is a list of related documentation.

- ANSI/IEEE Std 802.3-1985, ISO DIS 8802/3, ISBN - 0-471-82749-5, May 1988
- ANSI/IEEE Std 802.2-1985, ISO DIS 8802/2, ISBN 0-471-82748-7, Feb 1988
- RFC793, TCP (Transmission Control Protocol) DARPA Internet Program Protocol Specification, Sep 1981
- RFC 791, IP (Internet Protocol) DARPA Internet Protocol Specification, Sep 1981
- RFC826, An Ethernet Address Resolution Protocol (ARP), David Plummer, NIC Sep 1982
- RFC1042, A Standard for the Transmission of IP Datagrams over IEEE 802.2 Networks, Postel & Reynolds, ISI, Feb 1988
- RFC 792, ICMP (Internet Control Message Protocol) DARPA Internet C Control Message Protocol Specification, Jon Postel, Sep 1981
- RFC951, BOOTSTRAP PROTOCOL (BOOTP), Bill Croft and John Gilmore, September 1985
- RFC783, The Trivial File Transfer Protocol (TFTP) rev 2, K.R. Solons MIT, June 1981

Appendix D
Installation and Configuration of a Modicon Quantum Platform

Overview

This quick start guide describes how to install and configure a Modicon Quantum Ethernet module. It also sets up the I/O scanning service to allow data transfer to occur between the PLC and a remote slave device. Instructions for connecting to the module's embedded web server pages are also provided at the end of the guide.

What Is in This Chapter?

This chapter contains the following topics:

Overview

Introduction

This quick start guide explains how to install and configure Modicon Quantum Ethernet modules and set up a I/O scanning communication service. The following types of modules are applicable to this guide:

- 140 NOE 771 01
- 140 NOE 771 11
- 140 CPU 651 50
- 140 CPU 651 60

Hardware/Software Requirements

For the example discussed in this guide, the following Modicon Quantum Ethernet modules are required:

- 140 CPS 114 x0 power supply
- 140 CPU 651 50 CPU
- 140 NOE 77101 communication module

Also, a PC running Windows 2000 or XP with Schneider's Unity Pro configuration software installed on it is required.

Finally, either a USB or a Modbus cable is required to connect the PC to the network PLC.

Intended Audience

This user guide is intended for anyone who is involved in installing and configuring Modicon Quantum Ethernet modules in a network arrangement that can perform a variety of communication services.

Anyone reading this guide should:

- be familiar with Ethernet networks and the TCP/IP protocol
- understand the operation of PLCs

Installation

Introduction

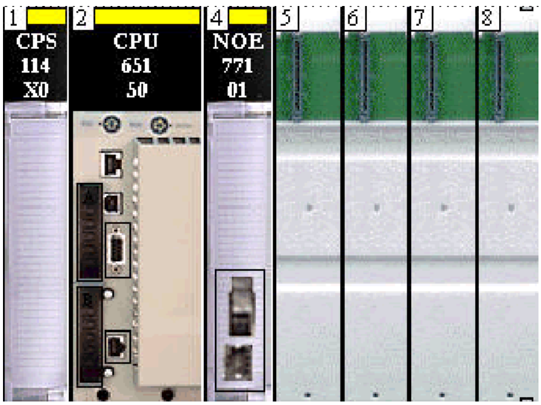
The Modicon Quantum Ethernet modules used for the example in this guide may vary from the ones available at your site. You can substitute the appropriate power supply, CPU, and Ethernet communication module(s) and other Quantum modules to make up a rack similar to the one described below.

Assembling the Rack

Assemble the modules into the rack as follows:

Step	Action
1	Insert the power supply into the leftmost slot (1) on the rack.
2	Add the CPU to the next two slots (2 & 3)on the right of the power supply.
3	Place the remaining Quantum communication modules, beginning with slot 4, to complete your installation.

In our example (see below), we use a CPS 114 x0 power supply, a CPU 651 50 processor, and a single NOE 771 01 Ethernet module to make up our rack.



Configuring the Rack with Unity Pro

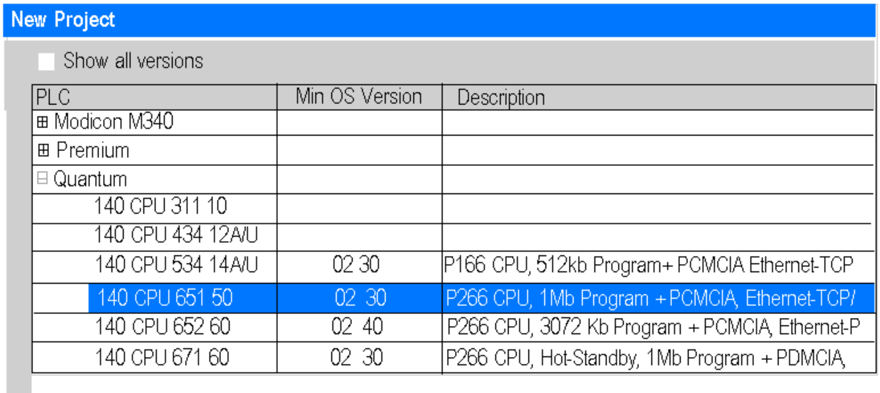
Introduction

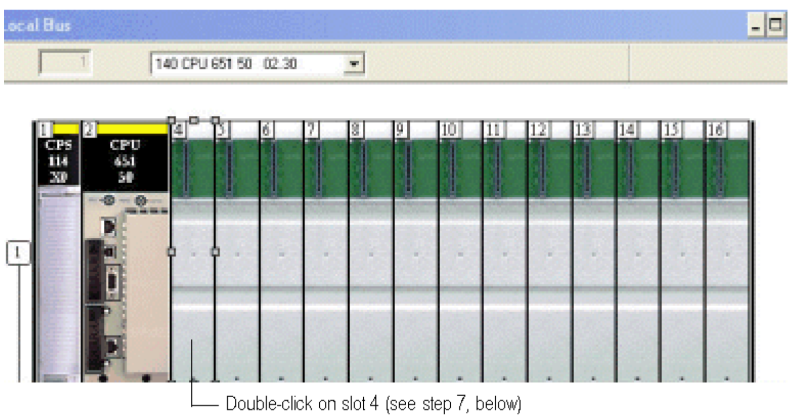
We begin this procedure by configuring the Modicon Quantum Ethernet module rack using Schneider Electric's Unity Pro configuration program.

Configuring the Rack

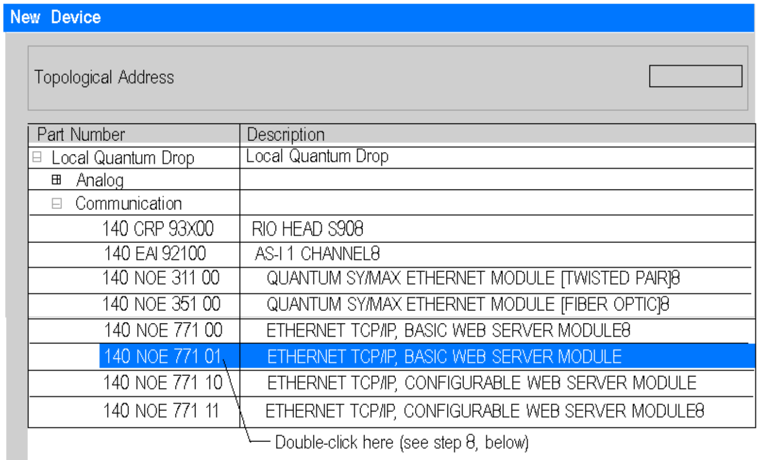
Using a PC loaded with Unity Pro software, proceed as follows:

Step	Action
1	Click **Start**.
2	Select **Programs**.
3	Then select **Schneider Electric** →**Unity Pro** →**Unity Pro XL**.  **Note:** The name of your Unity Pro package may be different. It may be Unity Pro M, Unity Pro L, Unity Pro XL, etc.
4	Select **New** in the File menu to create a New Project dialog box.

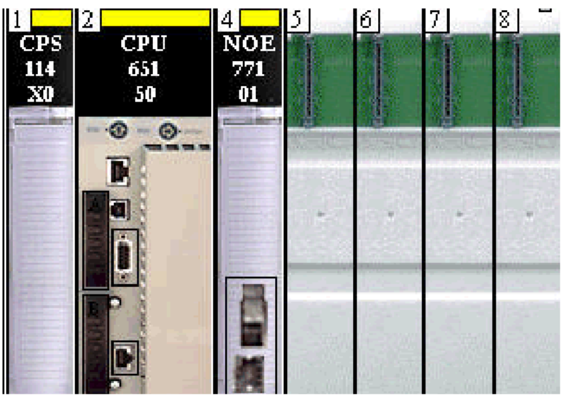
<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>5</td><td>In the New Project dialog box, expand the Quantum family to select the installed processor (the 140 CPU 65150 in our example).

</td></tr>
<tr><td>6</td><td>In the project browser, double-click Station →Configuration →PLC bus to access the configuration of the local rack (shown below).



</td></tr>
</table>

<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>7</td><td>Double-click slot 4 to bring up the New Device menu to show the hardware catalog.


</td></tr>
<tr><td>8</td><td>Double-click the module that goes in slot 4 (140 NOE 771 01 in our example, shown above).
Note: Alternately, you can click the module and drag it to the selected slot on the rack.</td></tr>
<tr><td>9</td><td>Repeat step 8 for each module included in your configuration.</td></tr>
</table>

The figure below shows the completed rack assembly for our example with the 140 NOE 771 01 module in slot 4.



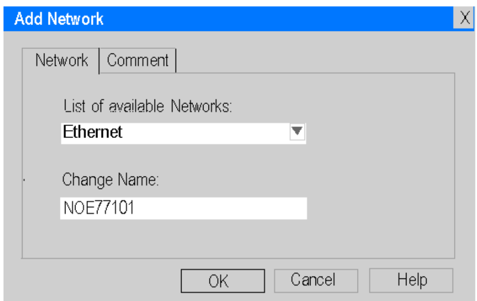
Configuring the Ethernet Network with Unity Pro

Introduction

The following procedure describes how to add a new Ethernet network and link it to the Modicon Quantum module we configured in the previous section.

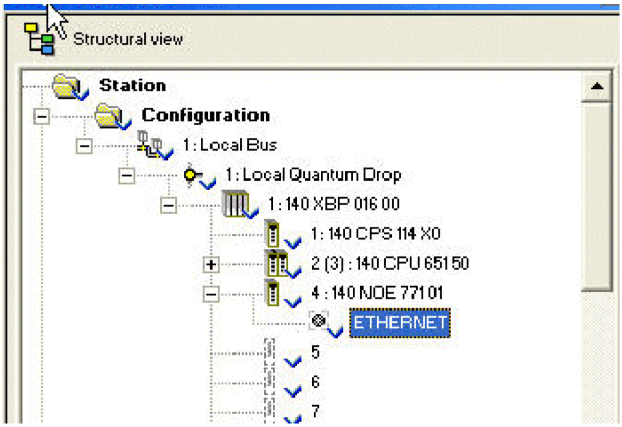
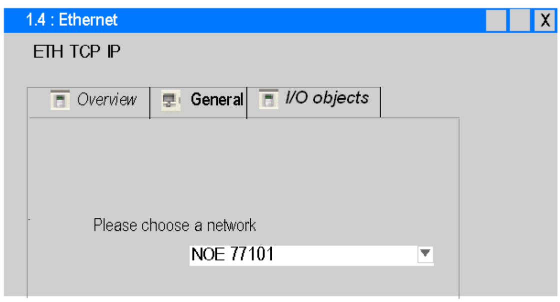
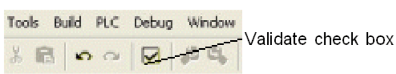
Setting Up the Network

Perform the following steps to add the Ethernet network:

Step	Action
1	Locate the **Communications** directory in the Project browser.
2	Right-click the **Network** subdirectory located under the Communications directory.
3	Select the **New Network** option to bring up the Add Network dialog box.
4	Scroll to **Ethernet** in the List of available Networks box.
5	Enter a meaningful name for your network in the **Change Name** field (NOE77101 was used in our example). 
6	Click **OK**.

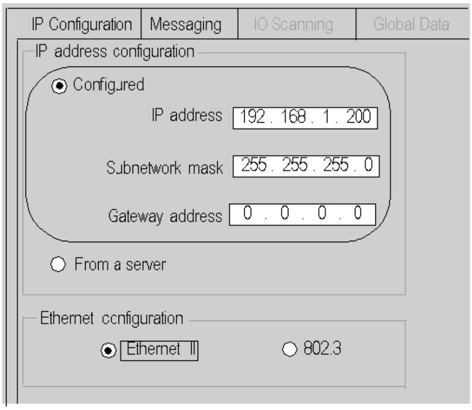
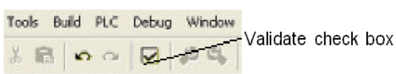
Linking the Network to the NOE 77101 Module

Perform the following steps to link the new logical Ethernet network with the NOE 77101 module.

Step	Action
1	Double-click **Local Bus** in the Project browser to show the rack configuration.
2	Expand the **Local Quantum Drop** to show the **ETHERNET** port under the 140 NOE 77101. 
3	Double-click the **ETHRNET** port to bring up the network link screen.
4	Under **Please choose a network**, scroll to locate the name of your logical network (**NOE77101** in our example). 
5	Click the validate check box in the upper toolbar to confirm the network link configuration. 

Assigning an IP Address to the 140 NOE 771 01 Module

Perform the following steps to assign an IP address to the 140 NOE 771 01 communication module.

Step	Action
1	Locate the **Communications\Networks** directory in the Project browser.
2	Double-click your new logical network (NOE77101 in our example) to open the NOE077101 configuration screen. 
3	Click **Configured** in the IP Address Configuration group (circled area shown above). **Note:** Be sure to contact your network administrator and request the IP, Subnetwork mask, and Gateway addresses prior to performing the next step.
4	Enter the appropriate values in the IP address, Subnetwork mask, and Gateway address text fields.
5	Click the validate check box in the upper tool bar to confirm the IP configuration settings. 

Configuring the I/O Scanning Service

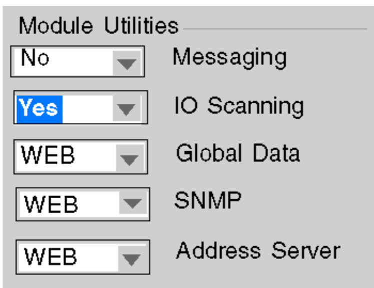
Introduction

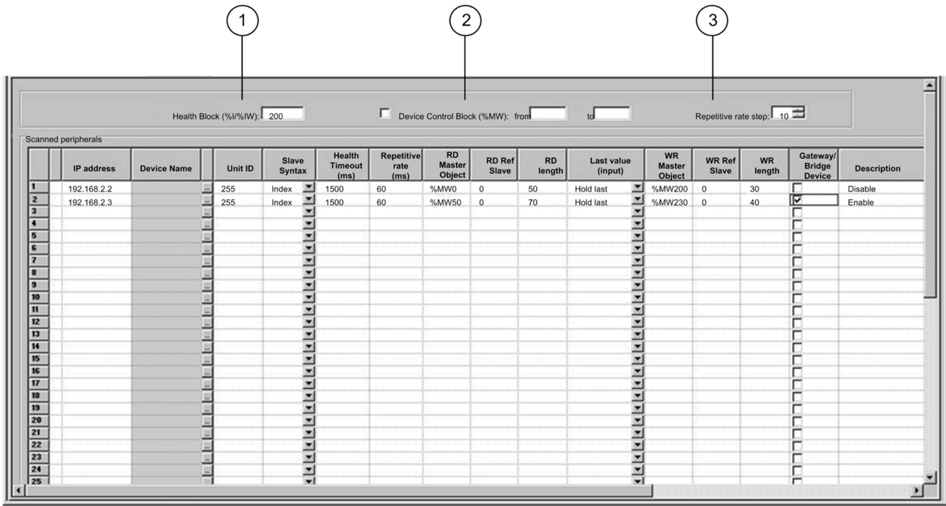
The Modicon **140 NOE 771 01** module supports Ethernet communication services such as I/O scanning, Global Data, Modbus messaging, SNMP, etc. This example shows you how to configure the I/O scanning service. This service is used to:

- transfer data between network devices
- allow a CPU to regularly read data from and write data to scanned devices

Selecting the I/O Scanning Parameters

Perform the following steps to setup the I/O scanning parameters:

Step	Action
1	Open the Unity Pro program on your PC.
2	In the **Project Browser**, locate the **Communication** →**Networks** sub-directory.
3	Click the network name (140 NOE 771 01 in our example) to open the module configuration table.
4	In the **Module Utilities** dialog, select **Yes** in the I/O Scanning field: 

Step	Action
5	Click the **I/O Scanning** tab to display the I/O scanning configuration screen: 
6	Type in the parameter settings under each of the column headings for the I/O Scanner Configuration. Refer to the parameter definitions below.
7	Click the validate check box in the upper tool bar to confirm the **I/O scanning** parameter settings.  Validate check box

Health Block

The **Health Block** (number 1 in previous figure) lets you define the first word or bit in a health table. That table can be up to 8 words (%IW) or 128 bits (%I) in length. Each bit in the table represents the status of an individual device. The table below shows how the bits are displayed based on the data type you use.

Bit	Data Type	
	%I	%IW
1	%I1	%IW1.0
2	%I 2	%IW1.1
...		
16	%I16	%IW1.15
17	%I17	%IW2.0

By default, the table comprises words starting at %IW1.0. If you want to set the table to bits, you need to specify a %I value in an increment of 16 (%I1, %I16, %I32, etc.).

NOTE: The **Health Block** is available only for the NOE 771 00, -01, and -11. It is not available for the CPU 651 x0.

NOTE: For Safety Monitor applications, the **Health Block** is mapped on %M/%MW from the UMA zone instead of %I/%IW.

Device Control Block

The **Device Control Block** (number 2 in previous figure) lets you disable any scanned device by setting a bit associated with that device to 1. The I/O scanner closes the connection and sets the Health Bit to 0 (unhealthy state).

To enable the **Device Control Block** select the check box in the **I/O Scanner configuration** dialog (number 2 in previous figure).

NOTE:

To enable the **Device Control Block**, use:
- Unity Pro at V2.0 or later
- a 140 NOE 771 01 or 140 NOE 771 11at version 3.5 or later
- a 140 CPU 651 x0 at version 2.0 or later

NOTE: If you attempt to enable the **Device Control Block** with an earlier version of the firmware, the I/O scanning service is disabled.

NOTE: If the check box is not selected, the I/O scanner service is enabled, and you cannot disable the I/O scanning of individual devices.

Disabling I/O scanning with the **Device Control Block** is equivalent to ending communication between the scanner and the device. Therefore:
- The fallback positions are applied to the inputs by the scanner.
- Communication with the device stops.
- All entries in the IN and OUT tables transfer between the CPU and the scanner on each scan.

NOTE: As a consequence of the last point above, if you modify a %MWi attached to an input, this %MWi is overwritten by the values coming from the I/O scanner in the next scan (with either 0 or the last input value).

It is possible (but meaningless) to access %MW attached to the outputs because they are not transmitted to the device.

Repetitive Rate Step

The **Repetitive Rate Step** (number 3 in previous figure) is set in multiples of 5 ms (the minimum) through 200 ms (the maximum).

The **Repetitive Rate** column is where you enter a rate of time for how often you want the I/O scanner to send a query to the device after the rate has timed out.

NOTE: The repetitive rate of the I/O scanner table is a multiple of the rate displayed in the **Repetitive Rate Step**. The real repetitive rate being executed by the I/O scanner service is shown in the **Repetitive Rate** column.

Note: An entry in the **Repetitive Rate** column is rounded up to the next multiple that was entered in the **Repetitive Rate Step** if the entry is not a multiple of the **Repetitive Rate Step**.

For example, if the entry in the **Repetitive Rate Step** is 5 and you enter a 7 in the **Repetitive Rate** column, the 7 is rounded up to 10; if you change the **Repetitive Rate Step** to 6 and enter a 7 in the **Repetitive Rate** column, the 7 is rounded up to 12.

I/O Scanner Table Parameters

The I/O Scanner parameters are described in the table below:

Parameter	Description	Example
Entry #	This is the first column; it has no name. Valid range: 1 ... 128 Each entry represents an I/O Scanning exchange on the network.	
IP address	This is the IP address of the scanned Ethernet slave device.	192.168.1.100
Device Name	To configure a device (Advantys island or DTM), click the **...** button to open the **Property** box *(see page 217)* to start the device configuration software. For an introduction to this procedure for Advantys, go here *(see page 209)*. For an introduction to this procedure for DTMs, go FDT Container. **NOTE:** While the **Property** box is open, Property cannot be edited.	MySTB1 or Master_PRM_DTM_10
Unit ID	This field associates the slave address of the device connected to an Ethernet/Modbus gateway with the IP address of that gateway: • Value range: 1 to 255 • Default value: 255 When using a bridge, enter the bridge index (1 to 255) in this field.	255

Parameter	Description	Example

Slave Syntax	Use this drop-down menu to pick the way **RD Ref Slave** and **WR Ref Slave** values are displayed. There are 4 types of display available:
	● Index: 100
	● Modbus: 400101
	    ● (Modbus register)
	● IEC 0: %MW100
	    ● M340 and Premium PLC slaves
	● IEC 1: %MW101
	    ● Quantum PLC slaves
		Index (default value)

Health Timeout (ms)	This field sets the maximum interval between the responses from a remote device. After this time period expires, the received data is invalid. The **Health Timeout** must be longer than the **Repetitive Rate** time (ms). For a Quantum 140 NOE771 01 Ethernet module, it also should be longer than the CPU scan time. For the **Health Timeout**: ● Range: 1 ms to 50 s ● Interval: 1 ms	1500 ms
Repetitive Rate (ms)	The rate at which data is scanned, in multiples of 16 ms.	64 ms
RD Master Object*	Destination address in the master PLC where, from each device, newly read information is stored	%MW10
RD Slave Ref.**	Source address index in the slave/remote device	The format of this value depends on the **Slave Syntax**: ● Index: 5 ● Modbus: 400006 ● IEC 0: %MW5 ● IEC 1: %MW6
RD length	Number of words to read	10
Using these examples, **Master Object** 140 NOE 771 01 reads data from address 192.168.1.100 starting at index 5. It puts the data into the NOE address starting at %MW10 using a read size of 10 words.		
Last value (Input)	This field configures the behavior of inputs in the event of an access error in relation to the remote device (for example: inoperative network or device power supply, etc.): ● Set to 0: fall back to 0 ● Hold last: maintain last value	Hold last
WR Master Object*	Source address of the master PLC whose data is being written into the slave/remote device. Write operations are always performed at the word level.	%MW20

Parameter	Description	Example
WR Slave Ref.**	The address of the first word written into the slave/remote device.	The format of this value depends on the **Slave Syntax**: ● Index: 100 ● Modbus: 400101 ● IEC 0: %MW100 ● IEC 1: %MW101
WR length	Number of words to be written	1
Using these examples, **Master Object** 140 NOE 771 01 writes data from address %MW20 to slave address 192.168.1.100 starting at index 100. It puts the data into the NOE address starting at %MW5 using a write size of 1 word.		
Gateway/Bridge Device	To allow slower TCP/IP network devices (i.e., gateways and bridges) to be compatible with the I/O Scanner: ● Select the check box to enable this feature. Defines a new bit, and sets it to high (1). ● Deselect the check box to disable this feature (default). Defines a new bit, and sets it to zero (0).	
Description	Additional information	
*Master refers to the client PLC that makes the request. **Slave refers to the server from which data is read or to which data is written.		

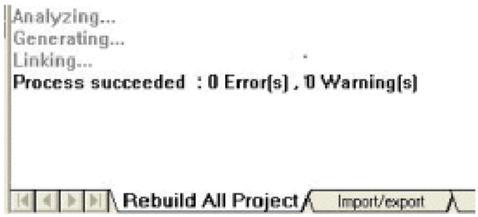
NOTE: For more information, refer to the Contextual Menu for Copy/Cut/Paste topic *(see page 205)*.

NOTE: For more information, refer to the I/O Scanning with Multiple Lines topic *(see page 207)*.

Building and Downloading the Configuration Program

Building the Program

Next, you need to build the whole program before downloading it to the PLC. To do this, select **Build\Rebuild All Project** in the toolbar at the bottom of the screen (shown below). If it is successful, a **Process succeeded** message will appear at the program’s completion.

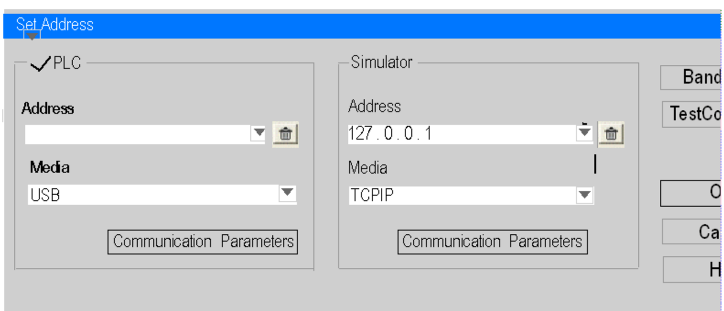


Connection Options

In order to run the configuration program it must first be downloaded to the PLC. Prior to downloading the program, the PLC must be connected to the PC containing the Unity Pro software. The connection can be accomplished using a communication network such as Ethernet, USB, Modbus, or Modbus Plus cabling. We describe both the USB and Modbus cable setups in the following examples.

Connecting the PC to the PLC with a USB Cable

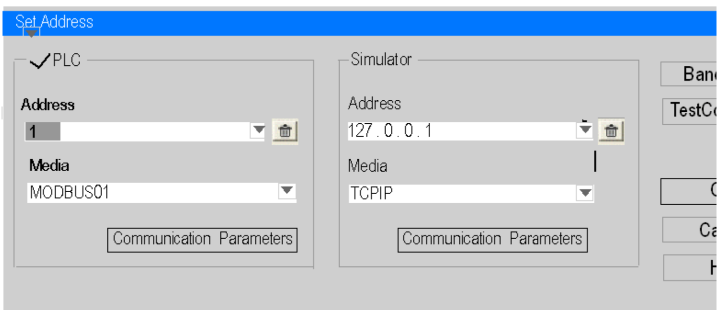
Proceed as follows to connect the PC to the PLC with USB.

Step	Action
1	Ensure that the Quantum system is powered up.
2	Connect the PLC to the PC with a USB cable.
3	On the PC, click the Unity Pro **PLC\Set Address** tab to bring up the Set Address dialog box. 

Step	Action
4	Select **USB** in the PLC Media box.
5	Leave the PLC Address field blank.
6	Click **OK**.
7	Proceed to Downloading and Running the Configuration Program *(see page 378)*.

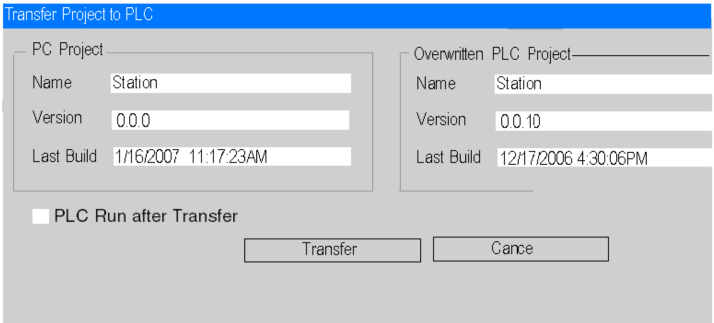
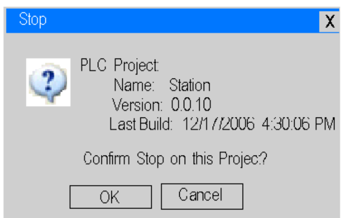
Connecting the PC to the PLC with a Modbus Cable

Proceed as follows to connect the PC to the PLC with Modbus.

Step	Action
1	Ensure that the Quantum system is powered up.
2	Connect the PLC to the PC with a Modbus cable.
3	On the PC, click the Unity Pro **PLC\Set Address** tab to bring up the Set Address menu. 
4	Select **MODBUS01** in the PLC Media field.
5	Enter the Modbus address in the PLC Address field (the default address is 1). **Note**: You can also use the keypad on the 140 CPU 651 50 module to enter or change the Modbus address.
6	Click **OK**.
7	Proceed to Downloading and Running the Configuration Program (below).

Downloading and Running the Configuration Program

Once the PC and the PLC are physically connected (above steps), the I/O scanning program can be downloaded to the PLC.

<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>1</td><td>On the PC, select PLC \Connect on the Unity Pro screen.</td></tr>
<tr><td>2</td><td>Click the PLC\Transfer Project to PLC tab to bring up the Transfer Project to PLC dialog box.

</td></tr>
<tr><td>3</td><td>Click the Transfer button to download the program to the PLC.</td></tr>
<tr><td>4</td><td>When the confirm screen appears, click OK.

</td></tr>
<tr><td>5</td><td>Click Run on the Unity Pro upper toolbar to start the program.</td></tr>
</table>

Diagnosing the Ethernet Module Using the Web Server

Introduction

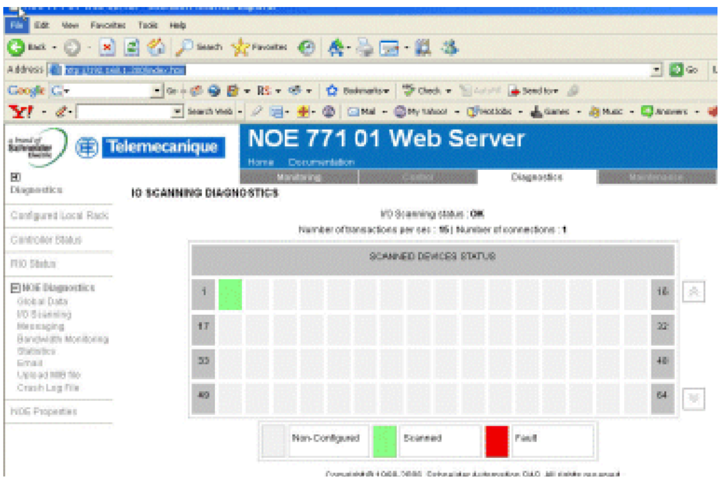
Quantum Ethernet modules have an embedded web server that provides web pages to diagnose the Ethernet module services, such as statistics, I/O scanning, Messages, global data, etc. You can access an Ethernet module's web pages by entering the IP address of the module in the web browser. No password is required to display the home page. From the NOE home page, you can access the Monitoring, Diagnostics, and Setup pages.

You can also use the module's LED display to diagnose problems. For information regarding the module's LED display refer to the *Quantum NOE 771xx Ethernet Modules User Manual*.

Accessing the Ethernet Module's Web Page

To access the NOE 771 01 module's web page, proceed as follows:

Step	Action
1	At the PC, start a Web browser such as Internet Explorer.
2	Enter the NOE 771 01's currently assigned IP address in the Address field of the browser to bring up the NOE's home page. 
3	Click the **Diagnostics** tab.
4	Enter a user name and password (the default is USER for both). **Note**: Check with your system administrator to see if the user name and password have been changed.

Step	Action
5	Click **OK** to bring up the NOE diagnostic web page. 
6	Click the **I/O Scanning** link on the left-hand side of the screen to access the I/O scanning diagnostics web page. 

Glossary

A

ACK

Acknowledgement

ARP

Address Resolution Protocol. A network layer protocol used to determine the physical address that corresponds to the IP address for a host on the network. ARP is a sub-protocol that operates under TCP/IP.

B

BOOTP

BOOTstrap Protocol. A protocol used at power-up in order to get an IP address that is provided by a BOOTP server and is based on the module's MAC address.

D

default gateway

The IP address of the network or host to which all packets addressed to an unknown network or host are sent. The default gateway is typically a router or other device.

DNS

Domain Name System. A protocol within TCP/IP used to find IP addresses based on host names

F

FactoryCast

An embedded Web server that the user customizes, permitting user access to controller diagnostics and Ethernet configuration.

firewall

A gateway that controls access to a network or an application.

framing types

Two common framing types are Ethernet II and IEEE 802.3.

FTP

File Transfer Protocol. The protocol (over TCP) used to read or write a file into a remote station (the FTP server side).

H

HTTP

A domain name given to a specific computer on a network and used to address that computer.

I

IP Address

Internet Protocol Address. A 32-bit address assigned to hosts using TCP/IP.

M

MAC Address

Media Access Control address. The hardware address of a device. A MAC address is assigned to an Ethernet TCP/IP module in the factory.

MIB

Management Information Base. Database that holds the configuration of a SNMP enabled device.

N

NTP

Network Time Protocol. A protocol used to synchronize the time of a client or server to the time of another server or referenced source like a satellite receiver.

P

PLC

Programmable Logic Controller

S

SMTP

Simple Mail Transfer Protocol. A common protocol used to transfer e-mail messages.

SNMP

Simple Network Management Protocol

socket

The association of a port with an IP address, serving as an identification of sender or recipient.

stack

The software code that implements the protocol being used. In the case of the NOE modules it is TCP/IP.

STP

Shielded Twisted Pair. A type of cabling consisting of several strands of wire surrounded by foil shielding, twisted together.

subnet mask

A bit mask used to identify or determine which bits in an IP address correspond to the network address and which bits correspond to the subnet portions of the address. The subnet mask is the network address plus the bits reserved for identifying the subnetwork.

T

TCP

Transmission Control Protocol.

Transparent Factory

Transparent Factory is a Schneider Electric initiative to bring Internet technologies to the factory floor to "information enable" Schneider Products to provide easy "transparent" access to plant operational data over open networks with open tools. Interfacing with products from other manufacturers for similar access, customers can expect improved methods for monitoring and controlling factory processes at reduced costs.

U

URL

Uniform Resource Locator. The network address of a file.

UTP

Unshielded Twisted Pair. A type of cabling consisting of insulated cable strands that are twisted together in pairs.

Index

T

U

W