

GE
Automation & Controls
Programmable Control Products

VersaMax* System Ethernet Network Interface Unit User Manual

GFK-1860C
November 2017



For Public Disclosure

Warnings, Cautions, and Notes as Used in this Publication

GFL-002



Warning

Warning notices are used in this publication to emphasize that hazardous voltages, currents, temperatures, or other conditions that could cause personal injury exist in this equipment or may be associated with its use.

In situations where inattention could cause either personal injury or damage to equipment, a Warning notice is used.



Caution

Caution notices are used where equipment might be damaged if care is not taken.

Note: Notes merely call attention to information that is especially significant to understanding and operating the equipment.

These instructions do not purport to cover all details or variations in equipment, nor to provide for every possible contingency to be met during installation, operation, and maintenance. The information is supplied for informational purposes only, and GE makes no warranty as to the accuracy of the information included herein. Changes, modifications, and/or improvements to equipment and specifications are made periodically and these changes may or may not be reflected herein. It is understood that GE may make changes, modifications, or improvements to the equipment referenced herein or to the document itself at any time. This document is intended for trained personnel familiar with the GE products referenced herein.

GE may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not provide any license whatsoever to any of these patents.

GE PROVIDES THE FOLLOWING DOCUMENT AND THE INFORMATION INCLUDED THEREIN AS-IS AND WITHOUT WARRANTY OF ANY KIND, EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY IMPLIED STATUTORY WARRANTY OF MERCHANTABILITY OR FITNESS FOR PARTICULAR PURPOSE.

* indicates a trademark of General Electric Company and/or its subsidiaries.
All other trademarks are the property of their respective owners.

©Copyright 2014-2017 General Electric Company.
All Rights Reserved

If you purchased this product through an Authorized Channel Partner, please contact the seller directly.

General Contact Information

Online technical support and GlobalCare	www.geautomation.com/support
Additional information	www.geautomation.com
Solution Provider	solutionprovider.ip@ge.com

Technical Support

If you have technical problems that cannot be resolved with the information in this manual, please contact us by telephone or email, or on the web at www.geautomation.com/support

Americas

Phone	1-800-433-2682
International Americas Direct Dial	1-780-420-2010 (if toll free 800 option is unavailable)
Customer Care Email	digitalsupport@ge.com
Primary language of support	English

Europe, the Middle East, and Africa

Phone	+800-1-433-2682
EMEA Direct Dial	+ 420-296-183-331 (if toll free 800 option is unavailable or if dialing from a mobile telephone)
Customer Care Email	digitalsupport.emea@ge.com
Primary languages of support	English, French, German, Italian, Czech, Spanish

Asia Pacific

Phone	+86-400-820-8208
	+86-21-3877-7006 (India, Indonesia, and Pakistan)
Customer Care Email	digitalsupport.apac@ge.com
Primary languages of support	Chinese, Japanese, English

Table of Contents

VersaMax* System Ethernet Network Interface Unit GFK-1860C

Table of Contents.....		i
Table of Figures.....		v
Chapter 1	Introduction, Description, and Specifications	1
1.1	Revisions in this Manual	2
1.2	Related Documentation	3
1.3	Hardware Implementation Distinctions	4
1.3.1	Transitioning from the Old to the New Hardware	6
1.4	Product Overview	7
1.5	Hardware Features	10
1.6	Ethernet NIU (ENIU) Specifications	12
1.6.1	Compatible Power Supplies	12
1.7	Configuration Software.....	13
1.8	Network Interface Unit Operation	14
1.8.1	Modbus Mode.....	14
1.8.2	EGD Mode	14
1.8.3	Maximum Number of I/O for ENIU	14
1.8.4	“Default” and “Hold Last State” Configuration Options.....	15
1.9	Modbus Operation	16
1.9.1	Operation at Power-up.....	16
1.9.2	Operation if Ethernet Cable is Disconnected, Ethernet Connection is Lost, or Communication Error Occurs	16
1.9.3	Reading Values from the ENIU after Power-up but Before a Write has occurred.....	16
1.9.4	Reading Values from the ENIU after an Upset, but Before a Write has occurred.	16
1.9.5	Maximum Data Transfer per Message in Modbus Mode	17
1.10	EGD Operation	18
1.10.1	Operation at Power-up.....	18
1.10.2	Operation on Upset.....	18
1.10.3	How to Write to all Discrete and Analog Outputs in a Single Message.....	18
1.10.4	Maximum Data Transfer per Message in EGD Mode	18
Chapter 2	Installation	19
2.1	Installation Instructions	20
2.1.1	Mounting Instructions.....	20

2.1.2	Installing an Ethernet Network Interface Unit.....	21
2.1.3	µUSB Port	30
2.1.4	CE Mark Installation Requirements	30
2.2	Communicating via the µUSB Port	31
2.2.1	Installing the USB Port Driver	33
2.3	Setting the Network IP Address and Other Network Settings.....	34
2.3.1	Setting the Network IP Address via the Serial Console.....	34
2.3.2	Setting the Network IP Address via the ENIU Rotary Switches	35
2.4	Firmware Updates for the ENIU Module.....	38
2.4.1	Firmware Updates for the ENIU via the µUSB Port.....	38
2.4.2	Firmware Updates for the ENIU via the Ethernet Port Using FTP	41
2.5	Firmware Updates for Modules in the Remote Node	43
2.5.1	Indirect Firmware Upgrade via the µUSB Port.....	43
2.5.2	Indirect Firmware Upgrade via the Expansion Port	45
Chapter 3	Configuring an Ethernet NIU and I/O Station	49
3.1	Using Autoconfiguration or Programmer Configuration	50
3.1.1	Autoconfiguration (Modbus Mode Only).....	50
3.1.2	Software Configuration	50
3.2	Configuring “Racks” and “Slots”	51
3.2.1	Configuring I/O References	52
3.3	Software Configuration of the Ethernet NIU and I/O Station.....	53
3.3.1	Notes on Using the Configuration Software	53
3.3.2	Basic Steps of Software Configuration.....	53
3.3.3	Configuring ENIU Network Parameters	54
3.3.4	Selecting EGD or Modbus.....	55
3.3.5	Software Configuration: Load, Store, Verify, Clear	56
3.3.6	Creating a New Communications Setup.....	56
3.3.7	Modifying the ENIU Communications Settings.....	57
3.3.8	Storing a Configuration to the Ethernet NIU	57
3.3.9	Loading a Configuration from the ENIU to the Programmer	57
3.3.10	Comparing Configurations in the Programmer and ENIU	58
3.3.11	Clearing a Software Configuration from the ENIU	58
3.4	Autoconfiguration of the Ethernet NIU and I/O Station	59
3.4.1	Autoconfiguration Sequence.....	59
3.4.2	Autoconfiguration Assigns Reference Addresses.....	60
3.4.3	Adding I/O modules to an Autoconfigured I/O Station.....	60
3.4.4	Replacing I/O modules in an Autoconfigured I/O Station	60
3.4.5	Clearing an Autoconfiguration	60
3.4.6	Hot Inserting I/O Modules	60
3.4.7	Autoconfiguring an I/O Station with Expansion Racks.....	61

3.5	Configuring the ENIU Produced Exchange	62
3.5.1	Phase 1 – Planning the Exchange	62
3.5.2	Phase 2 – Configuring the PLC CPU to Receive the ENIU Produced Exchange	64
3.6	Configuring the ENIU Consumed Exchange	67
3.6.1	Phase 1 – Planning the Exchange	67
3.6.2	Phase 2 - Configuring the CPU to Send to the ENIU's Consumed Exchange	69
3.6.3	Blank Memory Mapping Worksheet	72
3.7	EGD Exchange Status and Control Bytes for Fault Handling	73
3.7.1	Control Bytes for Consumed Exchange	73
3.7.2	Status Bytes for ENIU's Produced Exchange	74
Chapter 4	Modbus	77
4.1	Modbus Protocol	78
4.1.1	Modbus Connections	78
4.1.2	Modbus Port	78
4.1.3	Modbus Conformance Classes	78
4.1.4	Modbus Message Format	78
4.1.5	Modbus Byte Order	78
4.1.6	Modbus Bit Order	78
4.2	Modbus Tables	79
4.2.1	Modbus Register/Input Register Table	79
4.2.2	Modbus Input Discrete Table	79
4.2.3	Modbus Coil Table	79
4.3	Supported Function Codes	80
4.3.1	Read Multiple Registers	80
4.3.2	Write Multiple Registers	81
4.3.3	Read Coils	82
4.3.4	Read Input Discretes	83
4.3.5	Read Input Registers	84
4.3.6	Write Coil	85
4.3.7	Write Single Register	86
4.3.8	Read Exception Status	87
Chapter 5	Ethernet Global Data	89
5.1	EGD Overview	90
5.1.1	Exchange Parameters	90
5.2	EGD Protocol	91
5.2.1	Number of EGD Exchanges	91
5.2.2	EGD Port	91
5.2.3	EGD Destination Addresses	91

5.3	EGD Exchange Definition	92
5.3.1	Consumed Exchange	92
5.3.2	Produced Exchange	93
Chapter 6	Troubleshooting	95
6.1	Checking Status and Operation with the ENIU's LEDs.....	96
6.1.1	LED Power-up Sequence	96
6.1.2	LED Descriptions	96
6.1.3	Troubleshooting Using LED Indicator Lights	98
6.2	ENIU Fault Table	102
6.2.1	Modbus Mode	102
6.2.2	EGD Mode	102
6.3	Using FTP to Obtain Text Files	104
6.3.1	FTP for Version Information.....	104
6.3.2	FTP for Network Status	104
6.4	EGD Troubleshooting	105
6.5	Determining the MAC Address of the ENIU.....	106
6.5.1	Technique using Serial Console	106
6.5.2	Technique using Flashing LED Code.....	106
6.6	Reading the Stored IP Address of the ENIU.....	108
6.6.1	Technique using Serial Console	108
6.6.2	Technique using Flashing LED Code.....	108
Chapter 7	The VersaMax™ Family of Products	109
7.1	Product Family Overview.....	109
Appendix A	Glossary of Terms	111
Appendix B	IP and MAC Addresses.....	113
B-1	IP Addresses	113
B-2	Network Classes	114
B-3	Routers and Gateways.....	115
B-4	Subnets	116
B-4.1	Subnet Masks.....	116
B-4.2	Subnet Example.....	117
B-5	MAC Addresses	118
Appendix C	Number System Conversion Table.....	119

Table of Figures

Figure 1: IC200EBI001-MH or later (New Hardware Implementation June 2016).....	4
Figure 2: IC200EBI001-LG or earlier (Old Hardware Implementation Obsoleted 2016).....	4
Figure 3: ENIUs Attached to Ethernet Network.....	7
Figure 4: ENIU Controlling I/O Station with Expansion Racks.....	7
Figure 5: ENIU Controlling I/O Station with Booster Power Supply.....	8
Figure 6: DIN Rail Mounted ENIU.....	8
Figure 7: RJ-45 Connector and Shield Ground Lug Located on Underside (new hardware).....	9
Figure 8: IC200EBI001 Features (New Hardware).....	10
Figure 9: IC200EBI001 Features (Old Hardware).....	11
Figure 10: Location of Panel Mount Hole.....	20
Figure 11: Securing the Module.....	20
Figure 12: Clearances Required.....	21
Figure 13: Position Module Latch over Din Rail Edge.....	22
Figure 14: Pivot into Position.....	22
Figure 15: Latch shown in Unlocked Position.....	23
Figure 16: Align with Tabs.....	23
Figure 17: Move Latch to Locked Position.....	23
Figure 18: Rotate Latch to Unlocked Position.....	23
Figure 19: Disengage Power Supply Module from Carrier.....	23
Figure 20: Left- and Right-Side Connector Covers.....	24
Figure 21: Mounting Additional Modules.....	24
Figure 22: Ethernet Cable Interconnections.....	25
Figure 23: Expansion Transmitter Module Located to left of ENIU.....	26
Figure 24: Observe LEDs on Expansion Transmitter Module.....	26
Figure 25: Rotary Switch Indicating Expansion Rack ID.....	27
Figure 26: LED Behavior on Expansion Receiver Module.....	27
Figure 27: VersaMax Multi-Rack Expansion.....	28
Figure 28: RS-485 Inter-Rack Connection Signals & Pinout.....	28
Figure 29: Single-Ended Connection CPU or ENIU to ERM.....	29
Figure 30: Single-Ended Inter-Rack Connection Cable Signals & Pinout.....	29
Figure 31: Open Hinged Door.....	31
Figure 32: Access to µUSB Port with Hinged Door Open.....	31
Figure 33: Port Settings.....	32
Figure 34: Installing USB Port Driver using WinLoader.....	33
Figure 35: Display of ENIU Parameters on Serial Console.....	34
Figure 36: Changing IP Address via Serial Console.....	34
Figure 37: ENIU Rotary Switches (Old Hardware).....	35
Figure 38: MS-DOS Display of Internet Address and Physical Address.....	36
Figure 39: MS-DOS Connect Failed (Expected).....	37
Figure 40: Install ENIU Device Management.....	38
Figure 41: Driver Installation Complete Notification.....	38
Figure 42: Assign COM port to ENIU.....	39
Figure 43: Rack/Slot Selection in WinLoader.....	43
Figure 44: Rack/Slot Selection in WinLoader.....	45

Figure 45: Slot Notation Convention for VersaMax I/O Station.....	51
Figure 46: VersaMax I/O Station with 1 Expansion Rack (Single-Ended Cable).....	51
Figure 47: VersaMax Multi-Rack Expansion Notation.....	52
Figure 48: PC to ENIU Connection.....	56
Figure 49: Set up Communications Parameters for ENIU.....	56
Figure 50: Booster Power Supply Effect on Slot Numbering.....	59
Figure 51: ENIU Memory Types.....	60
Figure 52: ENIU Rack Configuration Example.....	62
Figure 53: ENIU Produced Exchange Example.....	62
Figure 54: Configure Parameters Menu - CPU.....	64
Figure 55: Consumed Exchanges Tab of CPU Configuration.....	64
Figure 56: Configure Consumed Exchange Number.....	65
Figure 57: CPU's Consumed Exchange Table Configured.....	65
Figure 58: Range Table Configured.....	66
Figure 59: ENIU Example Rack Configuration.....	67
Figure 60: ENIU Consumed Exchange Example.....	67
Figure 61: Produced Exchanges Tab of CPU Configuration.....	69
Figure 62: Configure Produced Exchange Number.....	69
Figure 63: CPU's Produced Exchange Configured.....	70
Figure 64: CPU's Produced Exchange Table Configured.....	71
Figure 65: Format of Consumed Data Exchange.....	92
Figure 66: Mapping of Consumed Data to Slot Locations of Output Modules.....	92
Figure 67: Format of Produced Data Exchange.....	93
Figure 68: Mapping of Produced Data to Slot Locations of Input Modules.....	93
Figure 69: Faceplate of Module (New Hardware Version).....	96
Figure 70: IP Address Formats for Network Classes A, B & C.....	113
Figure 71: Connecting Two Networks with a Router.....	115
Figure 72: Network Configuration Using a Subnet Mask.....	117

Chapter 1 Introduction, Description, and Specifications

This manual describes installation and operation of the VersaMax™ Ethernet Network Interface Unit (ENIU), Catalog Number IC200EBI001.

This manual covers the following topics:

Chapter 1:	Introduction, Description, and Specifications.
Chapter 2:	Installation.
Chapter 3:	Configuring an Ethernet NIU and I/O Station.
Chapter 4:	Modbus.
Chapter 5:	Ethernet Global Data.
Chapter 6:	Troubleshooting.
Chapter 7:	The VersaMax™ Family of Products.
Appendix A:	Glossary of Terms.
Appendix B:	IP and MAC Addresses.
Appendix C:	Number System Conversion Table.

1.1 Revisions in this Manual

Note: A given feature may not be implemented on all VersaMax Ethernet interfaces. To determine whether a feature is available on a given model and firmware version, please refer to the Important Product Information (IPI) document provided with the product.

Rev	Date	Description
C	Nov 2017	<ul style="list-style-type: none">• New hardware: adds USB port, changes LEDs, eliminates rotary switches (see Section 1.3) changes power ratings. (Section 1.6) and relocates RJ-45 connection to underside (Figure 7).• Added alternating sections for procedures that are different with old and new hardware implementations: 2.3.1 & 2.3.2, 2.4.1 & 2.4.2, 2.5.1 & 2.5.2.• Removed Appendix D (this information is presented and kept up-to-date in IPIs)• Removed portions of Chapter 7 that duplicate data available in GFK-1504.
B	Sep 2014	General update.
A	Jun 2001	Original Issue.

1.2 Related Documentation

Doc #	Title
GFK-1503	<i>VersaMax PLC User's Manual</i>
GFK-1504	<i>VersaMax Modules, Power Supplies, and Carriers User's Manual</i>
GFK-1533	<i>VersaMax DeviceNet Communications Modules User's Manual</i>
GFK-1534	<i>VersaMax Profibus Communications Modules User's Manual</i>
GFK-1535	<i>VersaMax Genius NIU User's Manual</i>
GFK-1670	<i>VersaPro Programming Software User's Guide,</i>
GFK-1847	<i>Remote I/O Manager User's Guide</i>

In addition to these manuals, datasheets and Important Product Information documents describe individual modules and product revisions. The most recent VersaMax documentation is available online on the Support website www.geautomation.com/support.

1.3 Hardware Implementation Distinctions

The IC200EBI001 product has evolved over time, resulting in two hardware implementations. The distinctions are discussed in this section.

- New hardware (Figure 1) was introduced in June 2016: catalog number IC200EBI001-MH, where “M” (or later) represents the new hardware content. The primary distinguishing feature is the presence of a micro USB Port (close to bottom edge, to the left of the GE logo, behind a hinged door). There was no USB Port on the older hardware. The RJ-45 port is also relocated to the underside of the module.
- Old hardware (Figure 2) was discontinued in June 2016: catalog number IC200EBI001-LG or earlier, where letter “A” thru “L” represent the old hardware content. The primary distinguishing features are three rotary switches located behind a transparent plastic door, neither of which is present in the new hardware.
- See table on the following page for a full list of distinctions.
- Both hardware implementations are covered in this manual. Be sure to consult the appropriate sections for the hardware used in your application.
- Transitioning from the older hardware version to the newer is also discussed in section 1.3.1.



Figure 1: IC200EBI001-MH or later (New Hardware Implementation June 2016)



Figure 2: IC200EBI001-LG or earlier (Old Hardware Implementation Obsolete 2016)

Item	New Hardware (IC200EBI001-MH and later)	Old Hardware (IC200EBI001-LG and earlier)
1	The left side of the new hardware is flush with the power supply front surface when mounted. The overall depth dimension is the same for both old and new hardware.	The old hardware is compact, but requires that a power supply be plugged in on top. With the power supply plugged in, the overall depth dimension is the same for both old and new hardware.
2	The new hardware profile brings the LEDs forward, making them easier to view in the equipment panel.	Viewed from outside the equipment panel, the LEDs are recessed in the old hardware implementation when compared to the new hardware.
3	The new LED panel incorporates some extra / reserved LEDs which were not part of the old hardware.	Reserved LEDs are not part of the LED panel in the old hardware.
4	The RJ-45 connector is located on the underside of the new module. This is consistent with other designs and provides for improved strain relief on the Ethernet cable.	In the old hardware implementation, the RJ-45 connector is located on the front panel, bottom left.
5	The RJ-45 connector in the new hardware implementation is electrically shielded and provides a grounding lug for the cable shield. This provides superior immunity to noise on the Ethernet network.	The RJ-45 connector in the old hardware implementation is not electrically shielded and does not provide a connection point for the cable shield. This compromises any shielding within the Ethernet cable.
6	A micro USB port is present in the new hardware. It is located behind the hinged door to the left of the GE logo. This port is used for performing firmware upgrades of the ENIU, using the WinLoader tool.	There is no μ USB port on the old hardware. It does not support WinLoader. Firmware upgrade of the ENIU itself is accomplished using an FTP session with the Host over Ethernet. This method works only with the old hardware.
7	The WinLoader tool is also used to perform indirect firmware upgrades of the intelligent I/O modules sitting in the local or expansion racks controlled by the ENIU module. Connect the PC hosting WinLoader to the Micro USB port via a standard micro-USB cable (order separately). Note: With the EBI001 new hardware implementation installed in the rack, the pass-through serial port of the VersaMax ETM no longer supports firmware upgrades.	Indirect firmware upgrades of the intelligent I/O modules located in the local or expansion racks controlled by the ENIU module is performed by connecting a PC running WinLoader via cable IC200CBL002 to the expansion port located on the left side of the ENIU module. This method works only with the old hardware. Note: With the EBI001 old hardware implementation installed in the rack, the pass-through serial port of the ETM can also be used to perform firmware upgrades.
8	There are no rotary switches on the faceplate of the new hardware. The equivalent functionality is provided via the serial console which connects via the micro USB port.	The old hardware presented three rotary switches, which were used to set the default IP address of the device.
9	The default IP address is 192.168.0.10.	The default IP address is set to 195.0.0.X, where 'X' is set by the user via a combination of rotary switches. This method is supported by the old hardware only.
10	Use the serial console to change the network portion of the IP address.	Changing the network portion of the IP address is not supported by the old hardware implementation.
11	The IP and MAC address information of the ENIU can be obtained from the welcome banner of the serial console. This method is supported by the new hardware only.	Obtaining IP and MAC Address identification is supported using a unique combination of rotary switches and blinking of LEDs using different patterns. This method works only with the old hardware.
12	The new hardware does not support setting of a temporary IP address	Setting of the temporary IP address using Telnet or using <i>Set Temporary IP Address</i> utility from within the Programmer (PME) is supported by the old hardware.
13	MAINT and COMM LEDs, located behind the hinged door, indicate the activity of the μ USB port.	This feature is not supported by the old hardware.
14	A pushbutton located behind the hinged door is used to put the module in maintenance mode.	This feature is not supported by the old hardware.
15	The OK LED now blinks to indicate the status of power-up diagnostics. Refer to Section 6.1.2.	This feature is not supported by the old hardware.
16	FAULT LED blinks RED color when an invalid MAC address has been detected. Refer to Section 6.1.2.	This feature is not supported by the old hardware.
17	During firmware update all the LEDs except Power LED will blink in unison at the rate of 1Hz. Refer to Section 6.1.2.	This feature is not supported by the old hardware.

1.3.1 *Transitioning from the Old to the New Hardware*

When replacing older version of ENIU with a later version, as discussed in Section 1.3 above,

- 1) The new hardware is designed to be a drop-in-replacement for the old hardware.
- 2) The interfacing software at the supervising controller will observe no difference.
- 3) Prior to usage, configure the ENIU using its default IP address 192.168.0.10. Otherwise, the user must first change the IP address of the ENIU using the serial console and then configure it. For more details on how to change the IP address of ENIU, please refer to Section 2.3.1.

1.4 Product Overview

The Ethernet Network Interface Unit (IC200EBI001) operates as either a Modbus TCP server or an Ethernet Global Data (EGD) station on an Ethernet network, providing I/O data, status, and diagnostic data, usually to a supervising controller (Figure 3).

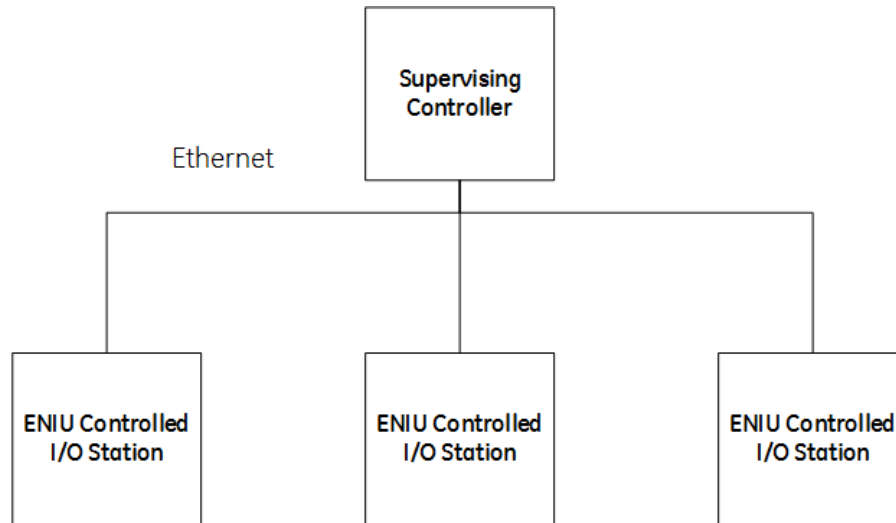


Figure 3: ENIUs Attached to Ethernet Network

The ENIU interfaces VersaMax I/O modules to an Ethernet network.

The ENIU may be configured to run either Modbus or EGD. It cannot run both protocols at the same time.

- In Modbus mode, up to 2 Modbus TCP connections can be made to the ENIU at one time.
- In EGD mode, the ENIU has exactly one produced and one consumed EGD exchange.

The ENIU and its expansion racks form an I/O station (Figure 4) capable of handling up to 64 I/O modules (8 racks maximum, containing up to 8 modules per rack).

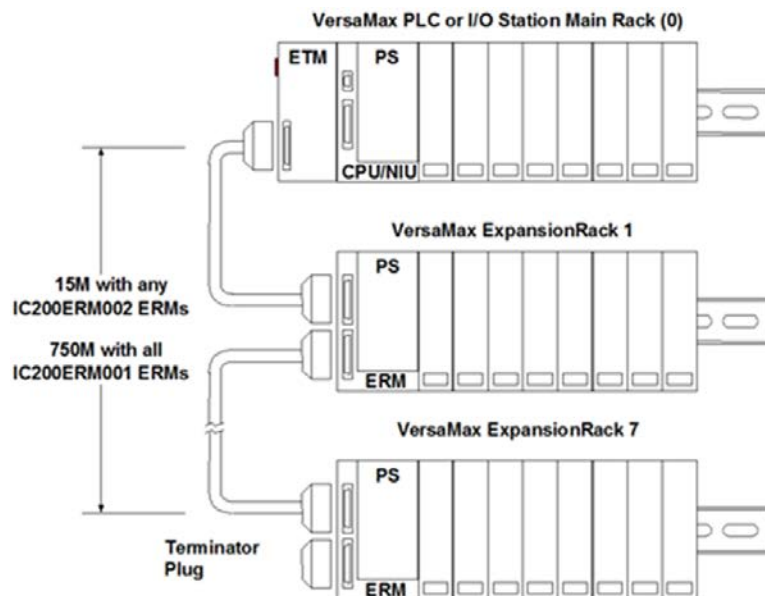


Figure 4: ENIU Controlling I/O Station with Expansion Racks

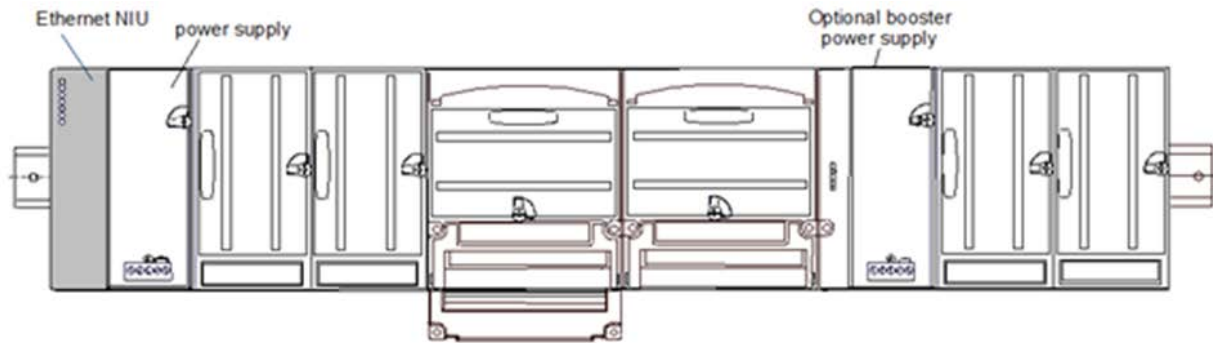


Figure 5: ENIU Controlling I/O Station with Booster Power Supply

Figure 5 shows an ENIU Module controlling a single-rack I/O Station with a Booster Power Supply present. The memory space of each ENIU supports I/O data consisting of up to 256 bytes of discrete input, 256 bytes of discrete output, 128 words of analog input, and 128 words of analog output data. The maximum I/O that can be supported by an ENIU station is limited by the combination of I/O modules that are installed in the backplanes which the ENIU controls (Figure 4).

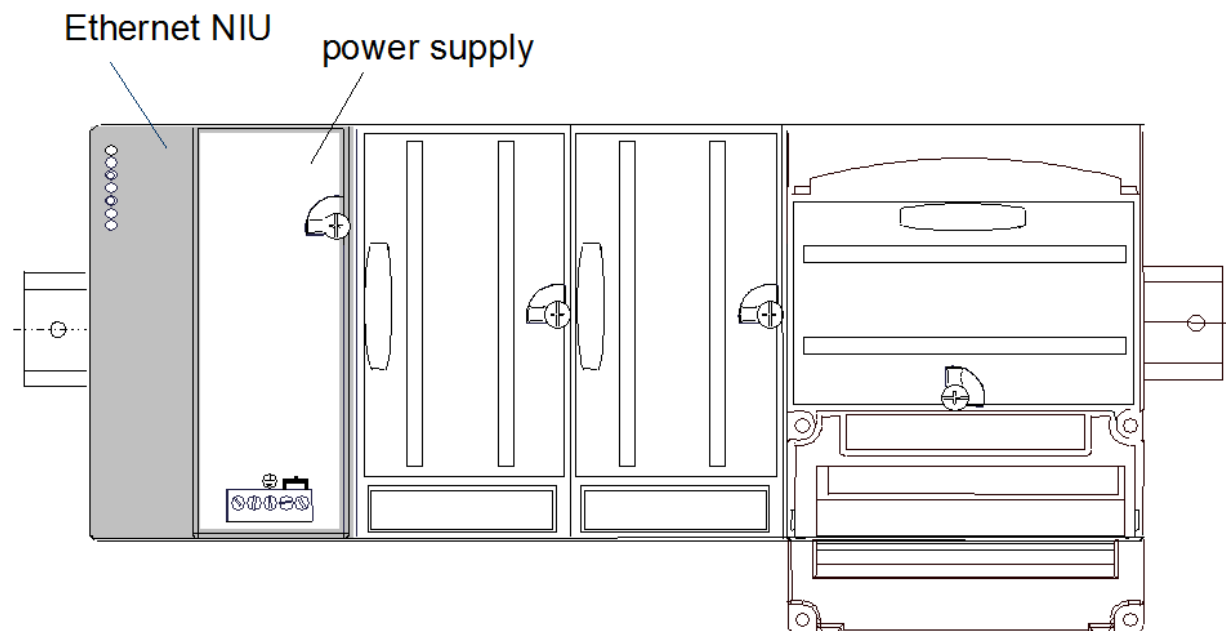


Figure 6: DIN Rail Mounted ENIU

The Ethernet Network Interface Unit installs on a 35mm x 7.5mm conductive DIN-rail. A VersaMax power supply module¹ mounts directly to the right side of the ENIU module itself (Figure 6). LEDs at top left of the ENIU indicate the presence of power and show the operating mode and status of the ENIU.

¹ Compatible power supply modules are listed in Section 1.6.1.

The RJ-45 connector is used to connect the ENIU to the Ethernet network. As noted in Section 1.3, the RJ-45 connector is not electrically shielded in the old hardware set, but is electrically shielded in the new hardware set. In addition, the new hardware implementation (Figure 7) provides a ground connection to which the shield wire of the Ethernet cable is attached.

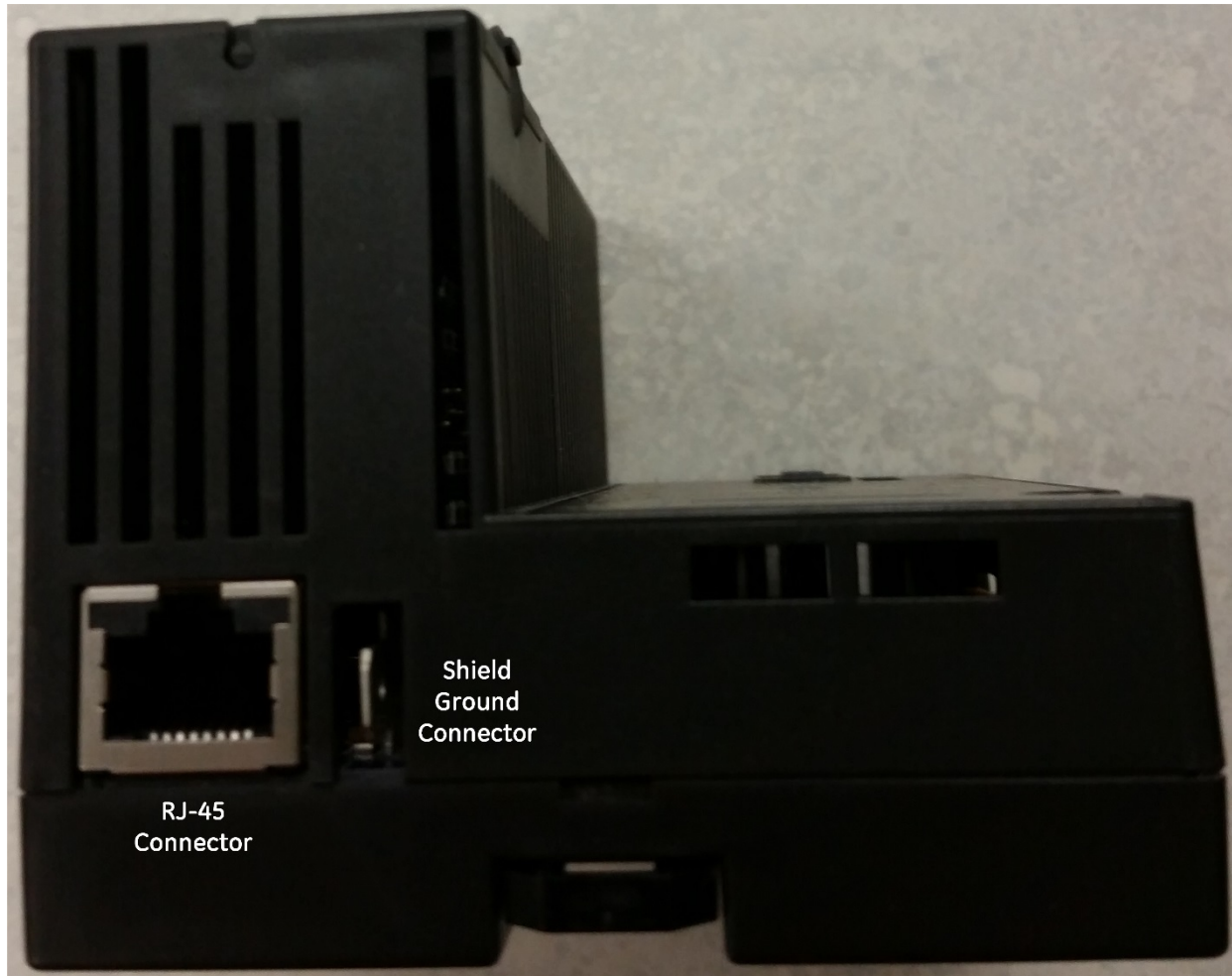


Figure 7: RJ-45 Connector and Shield Ground Lug Located on Underside (new hardware)

1.5 Hardware Features

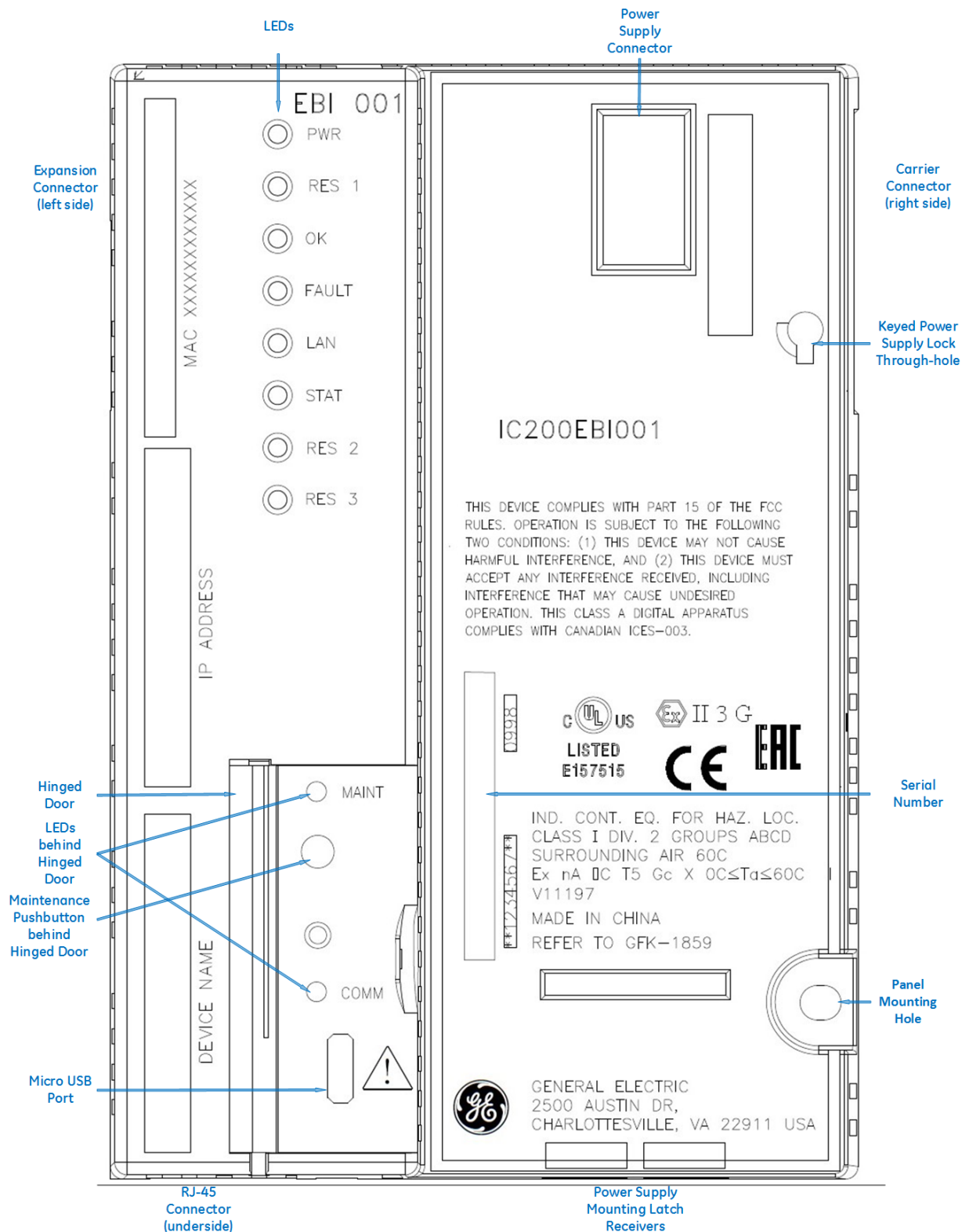


Figure 8: IC200EBI001 Features (New Hardware)

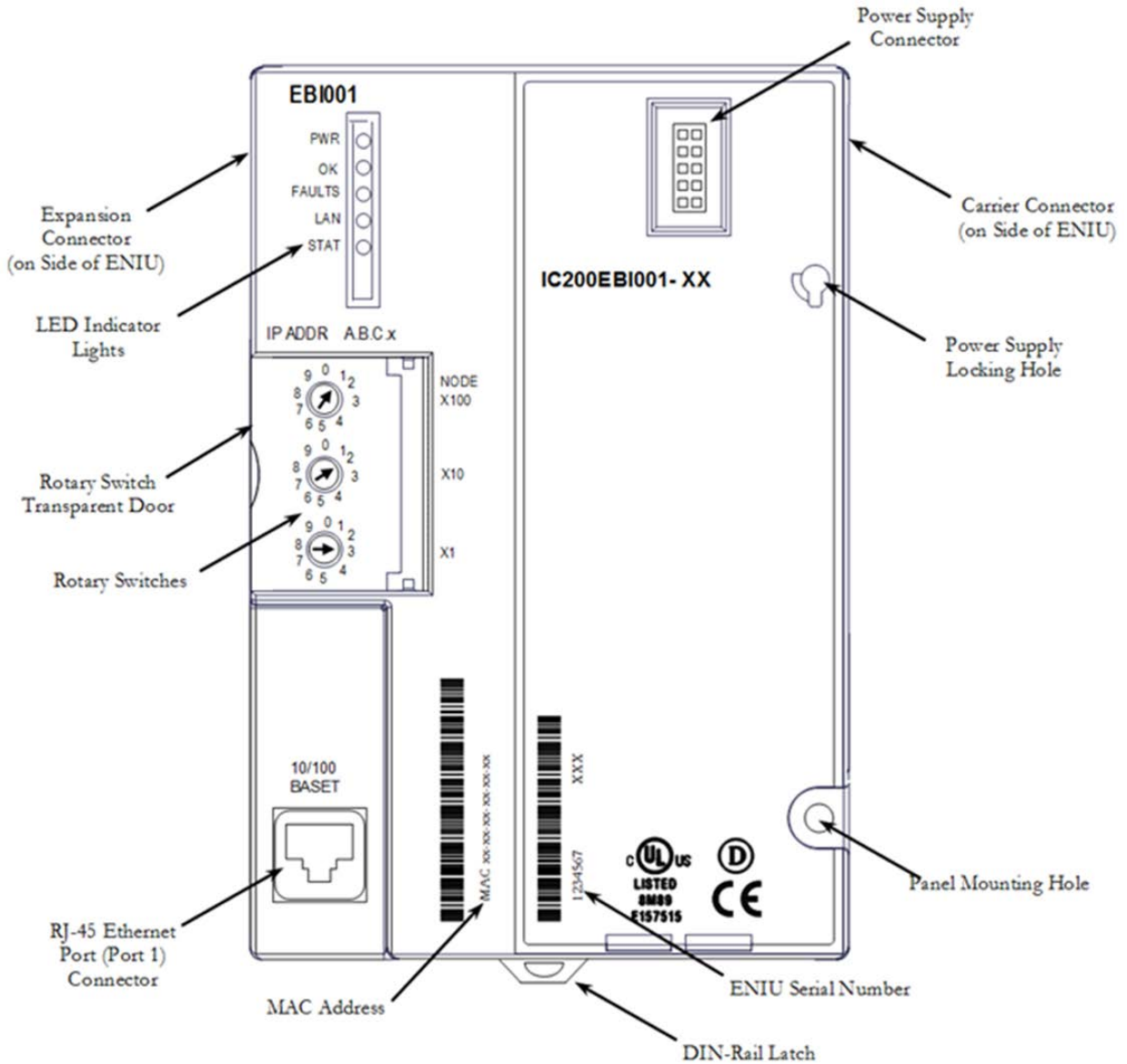


Figure 9: IC200EBI001 Features (Old Hardware)

1.6 Ethernet NIU (ENIU) Specifications

Number of racks (Max)	8 per station.
Number of modules (Max)	8 per rack for a total of 64 per station.
I/O data (Max)	1024 bytes maximum. Any combination of: %I: 2048 points (= 256 bytes) %Q: 2048 points (= 256 bytes) %AI: 128 channels (= 256 bytes) %AQ: 128 channels (= 256 bytes)
Fault table data	32 Faults (128 bytes)
Ethernet network IP address	Any valid Class A, B or C address.
Ethernet network data rate	10/100Mbps auto-detect.
Ethernet Duplex	Full-/Half-Duplex auto-detect
Indicators (5) Refer to Section 6.1.2.	PWR LED to indicate power OK LED to indicate health of the ENIU FAULTS LED to indicate presence of faults LAN LED to indicate traffic on the Ethernet network STAT LED to indicate presence of a Modbus connection or that EGD is consuming. New Hardware provides additional three LEDs (RES1, RES2, RES3) which are non-functional at this time.
Indicators Behind Hinged Door Refer to Section 6.1.2.	New Hardware provides 2 LEDs located behind hinged door. MAINT and COMM LEDs indicate the activity of the µUSB port.
Power Consumption ¹	New Hardware: 600 mA drawn from +5Vdc rail. Old Hardware: 175mA drawn from +5Vdc rail plus 425mA drawn from +3.3Vdc rail.

Note: In Proficy Machine Edition (PME), the Power Consumption value of ENIU is still displayed as: +5Vdc at 175mA; +3.3Vdc at 425mA, which corresponds to the old hardware implementation. For the new hardware implementation, the correct specification is +5Vdc at 600mA. Adjust calculations accordingly.

1.6.1 Compatible Power Supplies

The following VersaMax power supplies are compatible with both hardware versions of the VersaMax ENIU:

24Vdc Expanded 3.3V Power Supply	IC200PWR002
120/240Vac Power Supply	IC200PWR101
120/240Vac Expanded 3.3V Power Supply	IC200PWR102
12Vdc Expanded 3.3V Power Supply	IC200PWR202
Power Supply Booster Carrier	IC200PWB001

The Power Supply unit employed in an ENIU rack supplies power to all modules in that rack, including the ENIU module itself. Power supply units are described in the *VersaMax Modules, Power Supplies, and Carriers User's Manual*, GFK-1504.

Note: The **IC200PWR001** power supply does **not** have sufficient power capacity to support a typical ENIU system.

Note: The VersaMax power supply connected to the Ethernet ENIU and any other power supplies that reside in a booster carrier must be connected to the same external power source.

1.7 Configuration Software

There are three choices of configuration software for the Ethernet NIU:

- **VersaPro 2.0** (or later). Allows the user to
 - a) configure all VersaMax products and
 - b) program PLC ladder logic.
- **Remote I/O Manager 2.0** (or later). Allows the user to configure all VersaMax I/O products, but does not support PLC ladder logic programming. If you will be using GE I/O products with a third-party CPU, you can use this software to configure the I/O products. This software is a stand-alone version of the VersaPro 2.0 (or later version) configuration tool.
- **Proficy Machine Edition** Release 9.0 (or later).

Note: The Remote I/O Manager comes bundled with a serial programmer cable if ordered as catalog number IC641CFG110; however, neither hardware implementation of the Ethernet NIU supports this serial cable. The IC641CFG110 package was created for the convenience of Genius and Profibus NIU users, since those two products require the serial cable for software configuration.

The Remote I/O Manager software runs on a computer equipped with Windows 95/98, Windows NT 4.0, or Windows 2000. Note that VersaPro 1.1 and the Remote I/O Manager software cannot be installed on the same computer. If VersaPro 1.1 is present, you will be prompted to un-install it during the Remote I/O Manager installation. If both VersaPro and the Remote I/O Manager software are required, you must upgrade to VersaPro version 1.5 (or later), which includes a built-in version of the Remote I/O Manager. However, VersaPro 2.0 or later is required to support high density analog modules.

1.8 Network Interface Unit Operation



Warning

Be sure you understand the operation of outputs at power up and with communications upsets for the mode (Default, Hold Last State) that you are using. Failure to heed this warning could result in unexpected operation, possibly leading to injury to personnel and damage to equipment.

Note: The following topics should be understood before starting an ENIU setup:

1.8.1 Modbus Mode

- Operation of discrete outputs and analog outputs upon first communication
- How much data can be exchanged in a single message
- Choosing between “Default” and “Hold Last State” configuration
- Operation at power-up
- Operation if the Ethernet Cable is disconnected, the Ethernet connection is lost, or a communication error occurs

These topics are covered in Section 1.9, *Modbus Operation*.

1.8.2 EGD Mode

- The ordering of data in an EGD packet
- Operation if the Ethernet Cable is disconnected, the Ethernet connection is lost, or a communication error occurs

These topics are covered in Section 1.10, *EGD Operation*.

1.8.3 Maximum Number of I/O for ENIU

Maximum No. of I/O Modules per Rack	8
Maximum No. of I/O Racks	8 (1 local + 7 expansion)
Maximum No. of Discrete I/O Points ²	2048 (8 x 8 x 32)
Maximum No. of Analog I/O Channels ³	256 (8 x 8 x 4) – which equates to 128 in and 128 out.

Note: A full system of all %AI and all %AQ will not fit in the %AI or %AQ table of the ENIU, each of which is limited to 128 words.

² Currently, the maximum discrete point count for a VersaMax module is 32.

³ Starting with release 1.1, the ENIU supports high density analog modules.

1.8.4 "Default" and "Hold Last State" Configuration Options

The output mode of each Output Module under the control of the ENIU must be configured individually using the configuration software. The two output mode choices are

- "Default" and
- "Hold Last State".

Select one of these two options for each module in the ENIU I/O Station.

The following two tables show Output Table and Real Output states/values for both of these configuration choices; one table is for power-up and the other for recovery from upsets.

- The Output Table is stored in the internal memory of the ENIU module.
- Real Outputs are the values presented on the output terminals of the corresponding module. The Output Table is segmented into two parts: Discrete Outputs and Analog Outputs.

The Power-up Sequence Table

The ENIU Output Table is stored in volatile memory. Therefore, all Output Table values are initially zero upon power-up. Configuration settings, including the table of default values for each output module, are stored in non-volatile Flash memory.

Power-up Sequence				
ENIU Condition	Configured for "Default"		Configured for "Hold Last State"	
	Output Table	Real Outputs	Output Table	Real Outputs
Power-up	Zeros	Default table	Zeros	Zeros
First Write to segment of Output Table after Power-up (Modbus only)	Zeros plus written values	Output table	Zeros plus written values	Output table
Operational (Modbus only)	Zeros plus sum of all writes	Output table	Zeros plus sum of all writes	Output table
After first communication (EGD only)	Values from EGD exchange	Output table	Values from EGD exchange	Output table

The After-Upset Sequence Table

This table shows the sequence if power remains on but the connection is upset (by such things as disconnection or communication error) and then restored.

Sequence After Ethernet Connection Loss, Cable Removal, or Communication Error				
ENIU Condition	Configured for "Default"		Configured for "Hold Last State"	
	Output Table	Real Outputs	Output Table	Real Outputs
After a disconnect, cable removal, or communications error	Last values in Output table	Default table	Last values in Output table	Output table
First write to segment of Output Table after recovery (Modbus only)	Last values in Output table plus written value	Output table	Last values in Output table plus written values	Output table
Operational (Modbus only)	Last values in Output table plus sum of all writes	Output table	Last values in Output table plus sum of all writes	Output table
After first communication (EGD only)	Values from EGD exchange	Output table	Values from EGD exchange	Output table

Note: In Modbus mode, after a disconnect, cable removal, or communications error, and before the first Write message is implemented, any reads of the ENIU (which returns values from the Output Table) may not be the same as the values of the Real Outputs for those modules configured for "Default."

1.9 Modbus Operation

1.9.1 Operation at Power-up

Upon power-up, the ENIU sends values for discrete outputs and analog outputs to each output module based on the Default /Hold Last State selection configured for the corresponding module. The output modules will continue to use these values until the first Modbus output message is written into the corresponding segment of the ENIU Output Table.

Upon receiving a Modbus output message, the ENIU will first update the corresponding segment of its output table, then subsequently, the ENIU scans the entire output table to the output modules.

1.9.2 Operation if Ethernet Cable is Disconnected, Ethernet Connection is Lost, or Communication Error Occurs

If the Ethernet cable is disconnected, the Ethernet connection is lost, or a communication error occurs, this is considered to be a communications upset.

Following an upset, the discrete and analog outputs will either default or hold last state, depending on the corresponding module configuration. The output modules will continue using these values until the first Modbus output message is sent to the ENIU. Upon receiving this output message, the ENIU will update its output table, then subsequently, the ENIU scans the entire output table to the output modules.

1.9.3 Reading Values from the ENIU after Power-up but Before a Write has occurred.

If, after power-up, an Output Table Read is performed before a Write is performed in Modbus mode, the values returned for modules configured for Default may not be the same as the Real output values (which will have been set from the Default table values). The Output Table and Real Outputs will remain at these initial values until updated by the first Write.

1.9.4 Reading Values from the ENIU after an Upset, but Before a Write has occurred.

This applies to the situation where power continues to be applied to the ENIU following the occurrence of a communications upset in Modbus mode. If, after the communications upset, an Output Table Read is attempted before a Write is performed, the Output Table values returned may not be the same as the Real Output values for any discrete output modules configured to Default.

Warning



For both Discrete and Analog Output or Mixed modules, if the ENIU loses power but the Output or Mixed modules do not lose power, the Hold Last State parameter does not work as might be expected. Upon ENIU power loss, outputs configured for Hold Last State go to the Last State value on modules that still have power. When power is restored to the ENIU, outputs go to the Default value. When communications are reestablished and the ENIU receives a Write message, outputs go to the values in the Output Table of the ENIU.

1.9.5 Maximum Data Transfer per Message in Modbus Mode

Reading Inputs (Modbus Mode)

The ENIU only responds to direct requests from clients. All values, both input and output, can be requested. Messages sent from the ENIU to the client are limited to 250 bytes of data per request, per the Modbus TCP specification. Read requests are for contiguous values.

Writing Outputs (Modbus Mode)

The maximum overall length of output message data sent to the ENIU from the client is 200 bytes per message, per the Modbus TCP specification. As cautioned below, the first message to the ENIU after power-up or after a communication upset should contain all discrete and analog outputs actively used in the I/O map. This can be accomplished by using a Write Multiple Registers command that writes to bytes that overlap both the discrete and analog tables of the ENIU.



Caution

It is very important that the first output message contain all discrete and analog outputs used in the I/O map; otherwise, *spurious outputs* will be produced in the time between the first message releasing the output tables to the modules and the arrival of the second message.



Caution

In Modbus mode, reading the output tables (%Q and %AQ) may not return the actual state of the outputs if defaults are active.

1.10 EGD Operation

1.10.1 Operation at Power-up

Default is used until the first communication, then the exchange values are used.

1.10.2 Operation on Upset

Default or Hold Last State used. On first communication, exchange values are used.

1.10.3 How to Write to all Discrete and Analog Outputs in a Single Message

The segments for Discrete Outputs and Analog Outputs are contiguous in memory. By using only the higher indexed Discrete Outputs and the lower indexed Analog Output references, a contiguous block of memory that contains all the outputs and will fit in a single Write can be achieved. The following table provides an example of how this can be accomplished.

No. of Discrete	No. of Analog	%Q Range	%AI Range	Write Start Address	Length (Bytes)
256	32	%Q1793-2048	AQ1-32	368	96

1.10.4 Maximum Data Transfer per Message in EGD Mode

Reading Inputs

The ENIU has one produced exchange. It contains all of the discrete (%I) and analog (%AI) input module data configured in the ENIU.

The maximum configurable discrete input module data is 2048 bits or 256 bytes. The maximum configurable analog input data is 128 words or 256 bytes. The combined maximum total is therefore 512 bytes. There is an additional 4 bytes of status information at the beginning of the data in the exchange for a total of 516 total bytes allowed in one produced exchange.

Writing Outputs

The ENIU has one consumed exchange. It contains all of the discrete (%Q) and analog (%AQ) output module data configured in the ENIU.

The maximum configurable discrete output module data is 2048 bits or 256 bytes. The maximum configurable analog output data is 128 words or 256 bytes. The combined maximum total is therefore 512 bytes. There is an additional 4 bytes of control information at the beginning of the data in the exchange for a total of 516 total bytes allowed in one consumed exchange.

Chapter 2 *Installation*

This chapter gives basic installation instructions. For more information, please refer to the *VersaMax Modules, Power Supplies, and Carriers Manual*, GFK-1504.

This chapter is divided into the following sections:

Section 2.1: *Installation Instructions*

Instructions for mounting (and removing) the various components, wiring the system, and powering up the system.

Section 2.2, *Communicating via* .

Section 2.3, *Setting the Network IP Address*

Section 2.4: *Firmware Updates for the ENIU Module*

Section 2.5: *Firmware Updates for Modules in the Remote Node*

2.1 Installation Instructions

2.1.1 Mounting Instructions

Both old and new hardware mount in the same fashion.

All VersaMax modules and carriers in the same PLC “rack” must be installed on the same section of 7.5mm x 35mm DIN-rail, which must be electrically grounded to provide EMC protection. The rail must have a conductive (unpainted) corrosion-resistant finish. DIN-rails compliant with DIN EN50032 are preferred.

For vibration resistance, the DIN-rail should be installed on a panel using screws spaced approximately 5.24cm (6 inches) apart. DIN-rail clamps (available as part number IC200ACC313) can also be installed at both ends of the station to lock the modules in position.

Panel-Mounting

For maximum resistance to mechanical vibration and shock, the DIN-rail-mounted module must also be installed on a panel. Using the module as a template, mark the location of the module’s panel-mount hole on the panel. Drill and tap an M3.5 (6-32) hole in the panel at the marked location. Install the module using an M3.5 (#6) screw in the panel-mount hole.

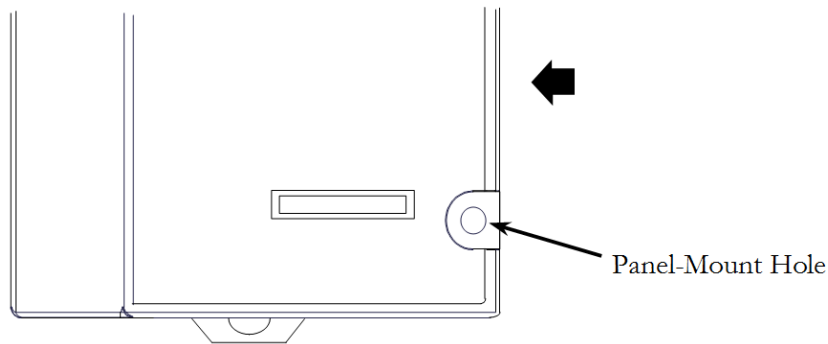


Figure 10: Location of Panel Mount Hole

Note1: Tolerances on all dimensions are +/-0.13mm (0.005in) non-cumulative.

Note2: 1.1-1.4Nm (10-12 in/lbs) of torque should be applied to M3.5 (#6-32) steel screw threaded into material containing internal threads and having a minimum thickness of 2.4mm (0.093in). See Figure 11 below.

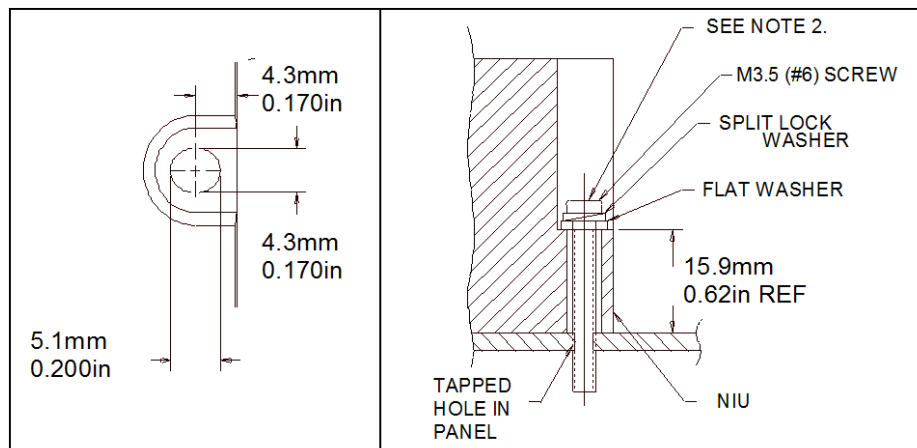


Figure 11: Securing the Module

2.1.2 Installing an Ethernet Network Interface Unit

If the I/O Station will have more than one expansion rack, or one expansion rack that uses an Isolated Expansion Receiver Module (IC200ERM001) as its interface to the expansion bus, an Expansion Transmitter Module must be installed to the left of the ENIU. The Expansion Transmitter Module must be installed on the same section of DIN-rail as the rest of the modules in the main “rack” (rack 0). See the instructions for installing expansion equipment on pages 26 through 29.

Clearance Required for the ENIU

Maintain a clearance of 2 inches (5.1cm) above and below the equipment and 1 inch (2.54cm) to the left. Additional clearance requirements are shown below.

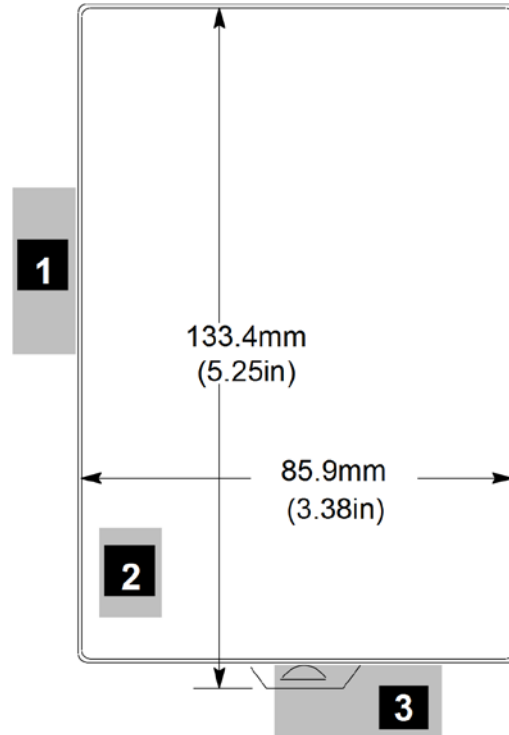


Figure 12: Clearances Required

(The numbered items below refer to the numbered areas in Figure 12 above.)

- 1) Allow sufficient clearance for opening ENIU USB socket door and plugging the USB cable.
- 2) Allow adequate clearance for the Ethernet cable, which connects to the RJ-45 jack in area 2. Note that the new hardware location of the RJ-45 connector is underneath (Figure 7) whereas the old hardware provided an RJ-45 connector on the front face (Figure 2).
- 3) Allow adequate space for power supply wiring. The power supply mounts on the front of the ENIU on the right-hand side. Also, allow sufficient space in front of the ENIU for the depth of the power supply.

Installing the ENIU on the DIN-rail

1. Turn off power to the system
2. Place the stationary top latch of the ENIU over the top edge of the DIN-rail.

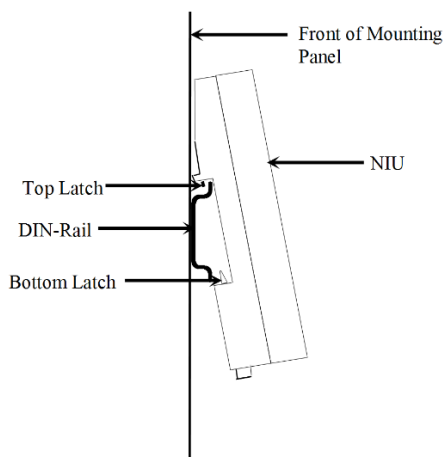


Figure 13: Position Module Latch over Din Rail Edge

3. Pivot the ENIU downward until the ENIU's spring-loaded bottom latch snaps around the bottom edge of the DIN-rail.

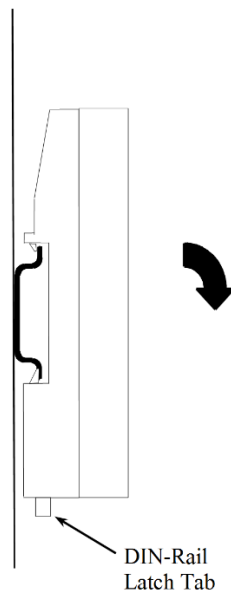


Figure 14: Pivot into Position

Removing the ENIU from the DIN-rail

- 1) Turn off power to the power supply.
- 2) If the ENIU is attached to the panel with a screw, remove the power supply module and then remove the panel-mount screw.
- 3) Slide the ENIU to the left along the DIN-rail away from the other modules until its carrier connector disengages from the carrier on its right. (If the ENIU will not slide freely along the DIN-rail, pull downward on the DIN-rail latch tab with a small screwdriver while sliding the ENIU.)
- 4) Use a small flathead screwdriver to pull the DIN-rail latch tab downward, then pull the bottom of the ENIU forward until it clears the bottom edge of the DIN-rail. Finally, lift the ENIU off the top edge of the DIN-rail.

Installing the Power Supply on the ENIU

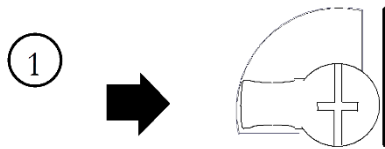


Figure 15: Latch shown in Unlocked Position

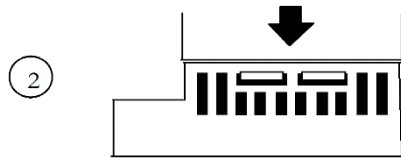


Figure 16: Align with Tabs

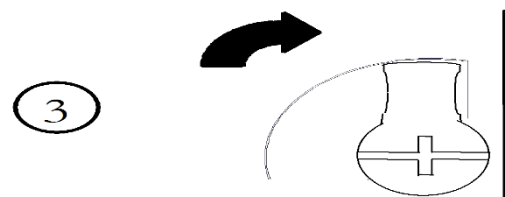


Figure 17: Move Latch to Locked Position

- 1) The power supply module installs directly on top of the ENIU. The latch on the power supply must be in the unlocked position as shown in Figure 15.
- 2) Align the connectors and the latch post and press the power supply module down firmly, until the two tabs on the bottom of the power supply click into place. Be sure the tabs are fully inserted in the holes in the bottom edge of the ENIU.
- 3) Turn the latch to the locked position (clockwise) to secure the power supply to the top of the ENIU module (Figure 17).

Removing the Power Supply from the ENIU

Exercise care when working around operating equipment. Devices may become very hot and could cause injury.

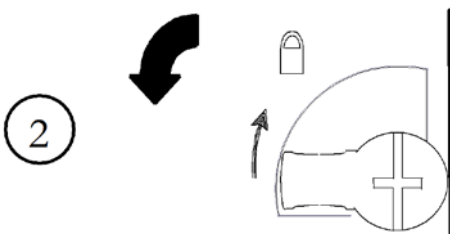


Figure 18: Rotate Latch to Unlocked Position

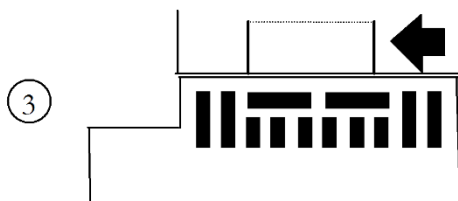


Figure 19: Disengage Power Supply Module from Carrier

- 1) Remove power.
- 2) Turn the latch to the unlocked position (counter clockwise) as illustrated (Figure 18).
- 3) Press the flexible panel on the lower edge of the power supply to disengage the tabs on the power supply from the holes in the carrier.
- 4) Pull the power supply straight off.

Installing Additional Modules

Before joining carriers to the ENIU, remove the connector cover on the right-hand side of the ENIU (Figure 20). Do not discard this cover; you will need to install it on the last carrier. It protects the connector pins from damage and from ESD during handling and use.

Do not remove the connector cover on the left-hand side (Figure 20).

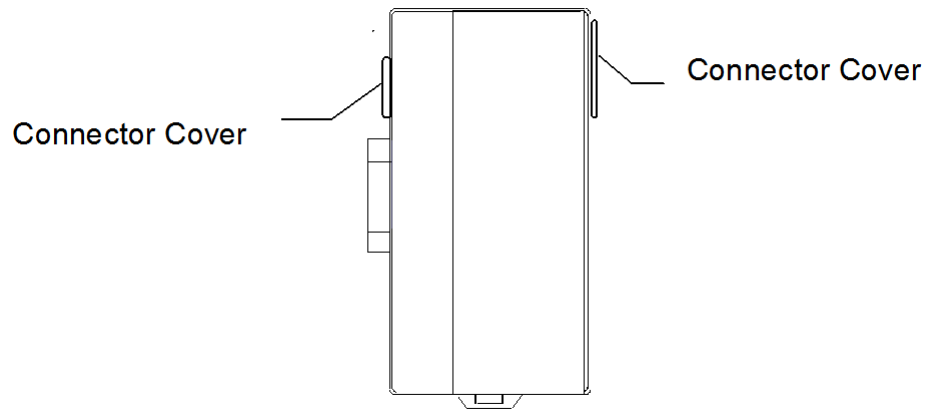


Figure 20: Left- and Right-Side Connector Covers

Install additional modules by mounting modules on their carriers and sliding them along the DIN-rail to fully engage the connectors in the sides of the carriers (Figure 21).

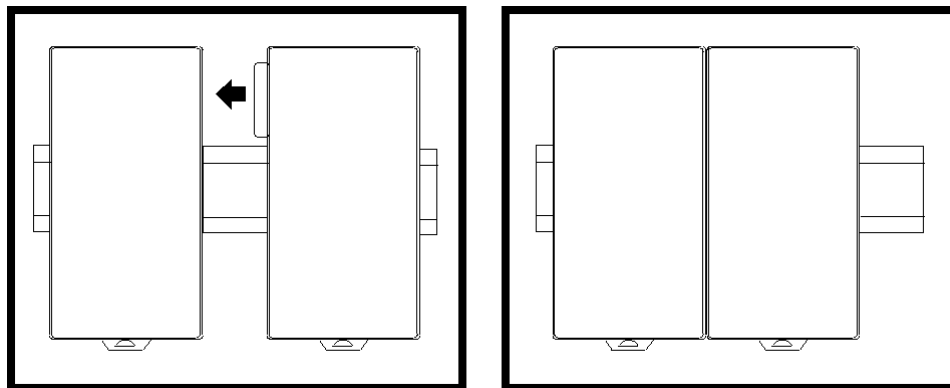


Figure 21: Mounting Additional Modules

Installing the Ethernet Cable

The 10Base-T / 100Base-TX RJ-45 port on the Ethernet NIU connects directly to a network without an external transceiver. Connect the port to an external 10Base-T / 100Base-TX hub or switch using a twisted pair cable. Category 5 cable is required for 100Base-TX operation. 10Base-T / 100Base-TX cables are readily available from commercial distributors. GE recommends purchasing rather than making cables. Cables must meet the applicable IEEE 802.3 or 802.3u standard, noted in the table below.

The Ethernet NIU automatically senses whether it is connected to a 10BaseT or 100BaseTX network, and whether communications are half-duplex or full duplex.

Note: GE recommends use of Category 5 cable even if using 10Base-T for the following reasons: (1) it is a higher quality cable, (2) it will support a later system upgrade to 100Base-TX, and (3) cable cost is low compared to installation labor cost.

Unshielded Twisted Pair (UTP) Ethernet Cables				
Cable Category	Rating	Max. Drop Length	Standard	Suitable for:
Category 5	100 Mbps.	100 Meters	IEEE 802.3u	100Base-TX 10Base-T
Category 4	20 Mbps.	100 Meters	IEEE 802.3	10Base-T
Category 3	16 Mbps.	100 Meters	IEEE 802.3	10Base-T

Network Connection

Connection of the Ethernet Interface to a 10Base-T or 100Base-TX network is shown below (each cable drop can be up to 100 meters long):

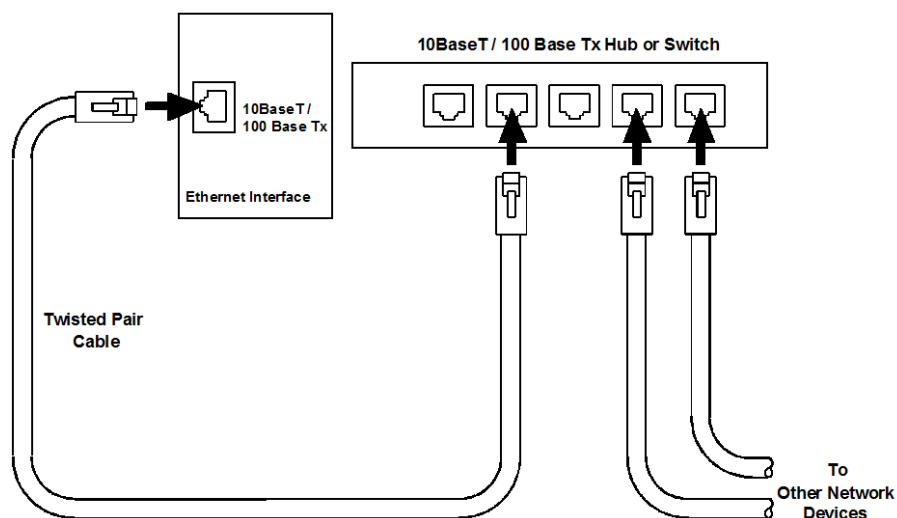


Figure 22: Ethernet Cable Interconnections

Installing an Expansion Transmitter Module

If the I/O Station will have more than one expansion rack, or one expansion rack that uses an Isolated Expansion Receiver Module (IC200ERM001) as its interface to the expansion bus, an Expansion Transmitter Module must be installed to the left of the ENIU. The Expansion Transmitter Module must be installed on the same section of DIN-rail as the rest of the modules in the main “rack” (rack 0).

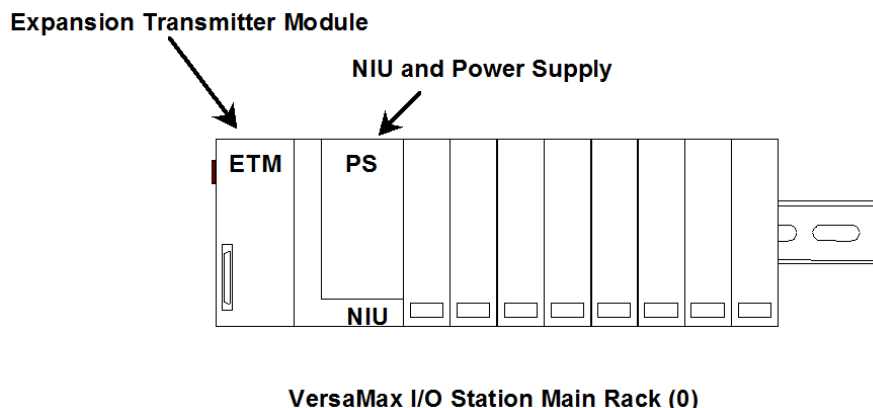


Figure 23: Expansion Transmitter Module Located to left of ENIU

- 1) Make sure rack power is off.
- 2) Attach the Expansion Transmitter to DIN-rail to the left of the ENIU position.
- 3) Install the ENIU as instructed. Connect the modules and press them together until the connectors are mated.
- 4) After completing any additional system installation steps, apply power and observe the LEDs on the module (Figure 24).

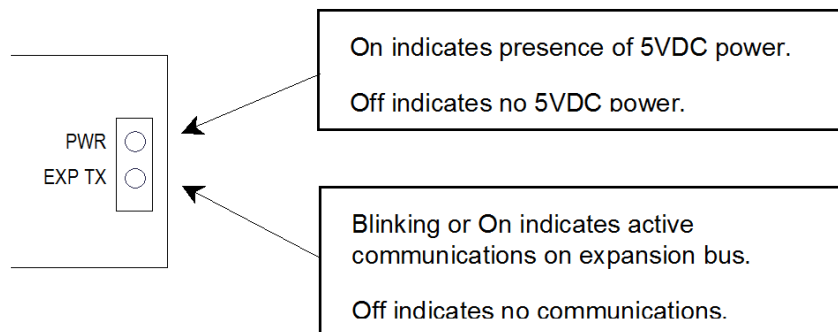


Figure 24: Observe LEDs on Expansion Transmitter Module

Removing an Expansion Transmitter Module

- 1) Make sure rack power is off.
- 2) Slide module on DIN-rail away from the ENIU in the main rack.
- 3) Using a small screwdriver, pull down on the tab on the bottom of the module and lift the module off the DIN-rail.

Installing an Expansion Receiver Module

An Expansion Receiver Module (IC200ERM001 or 002) must be installed in the leftmost slot of each VersaMax expansion rack.

- 1) Insert the label inside the access door at the upper left corner of the module.
- 2) Attach the module to the DIN-rail at the left end of the expansion rack.
- 3) Select the expansion rack ID (1 to 7) using the rotary switch under the access door at upper left corner of the module.

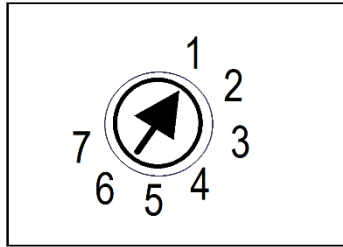


Figure 25: Rotary Switch Indicating Expansion Rack ID

- 4) Install the Power Supply module on top of the Expansion Receiver.
- 5) Attach the cables. If the system includes an Expansion Transmitter Module, attach the terminator plug to the EXP2 port on the last Expansion Receiver Module.
- 6) After completing any additional system installation steps, apply power and observe the module LEDs.

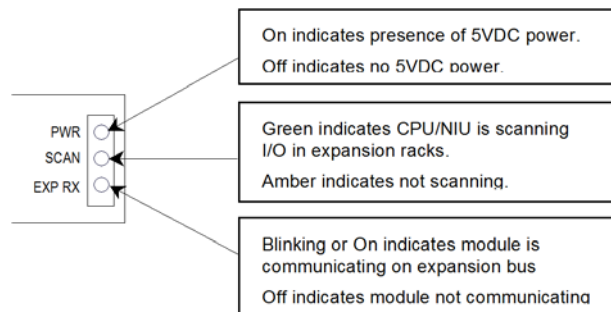


Figure 26: LED Behavior on Expansion Receiver Module

Removing an Expansion Receiver Module

- 1) Make sure rack power is off.
- 2) Separate the Power Supply module from the Expansion Receiver Module.
- 3) Slide the Expansion Receiver Module on DIN-rail away from the other modules.
- 4) Using a small screwdriver, pull down on the tab on the bottom of the module and lift the module off the DIN-rail.

Expansion Rack Power Sources

Power for module operation comes from the Power Supply installed on the Expansion Receiver Module. If the expansion rack includes any Power Supply Booster Carrier and additional rack Power Supply, it must be tied to the same source as the Power Supply on the Expansion Receiver Module.

Connecting the Expansion Cable: RS-485 Differential

For a multiple-rack expansion system, connect the cable from the expansion port on the Expansion Transmitter (ETM) to the Expansion Receivers (ERM) as shown in Figure 27 below. If all the Expansion Receivers are the Isolated type (IC200ERM001), the maximum overall cable length is 750 meters. If the expansion bus includes any non-isolated Expansion Receivers (IC200ERM002), the maximum overall cable length is 15 meters.

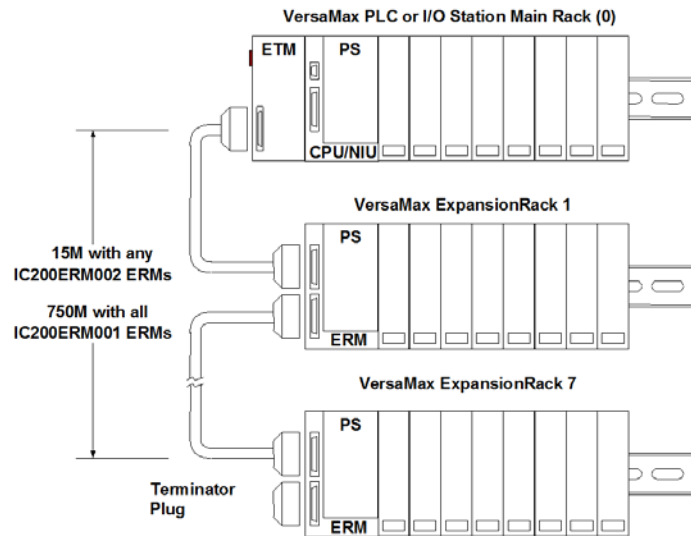


Figure 27: VersaMax Multi-Rack Expansion

Install the Terminator Plug (supplied with the Expansion Transmitter module) into the lower port on the last Expansion Receiver. Spare Terminator Plugs can be purchased separately as part number IC200ACC201 (Qty 2).

RS-485 Differential Inter-Rack Connection (IC200CBL601, 602, 615)

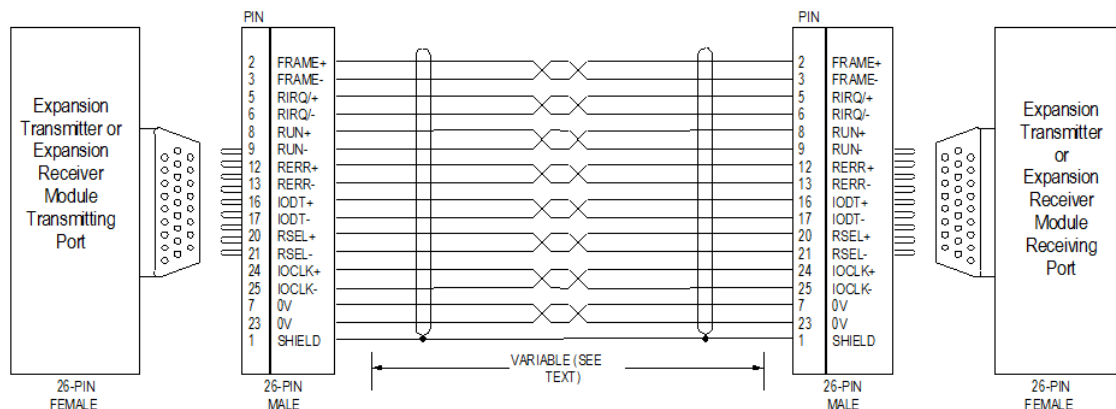


Figure 28: RS-485 Inter-Rack Connection Signals & Pinout

Building a Custom Expansion Cable

Custom expansion cables can be built using Connector Kit IC200ACC202, Crimper AMP 90800-1, and Belden 8138, Manhattan/CDT M2483, Alpha 3498C, or equivalent AWG #24 (0.22mm²) cable.

Connecting the Expansion Cable: Single-ended

For a system with one non-isolated expansion rack (IC200ERM002) and no Expansion Transmitter, connect the expansion cable from the serial port on the VersaMax ENIU to the Expansion Receiver (ERM) as shown below. The maximum cable length is one meter. Cables cannot be fabricated for this type of installation; cable IC200CBL600 must be ordered separately.

Note: No Terminator Plug is needed in a single-ended installation; however, it will not impede system operation if installed.

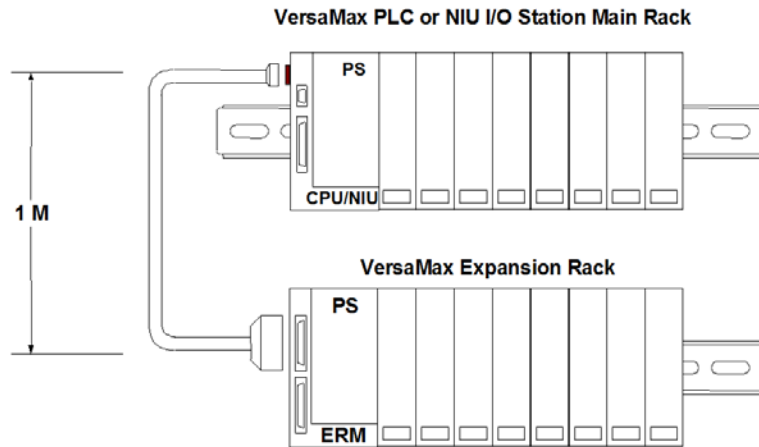


Figure 29: Single-Ended Connection CPU or ENIU to ERM

Single-Ended Inter-Rack Connection Cable (IC200CBL600)

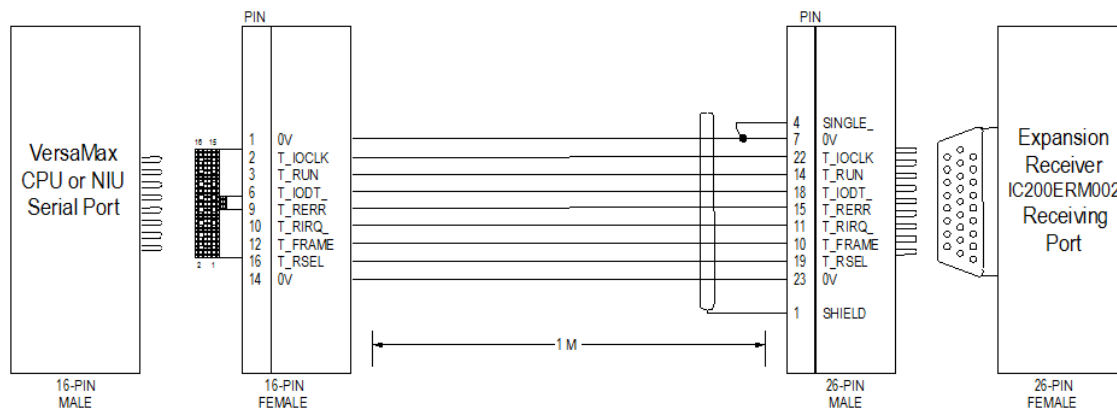


Figure 30: Single-Ended Inter-Rack Connection Cable Signals & Pinout

Power Sources for Single-Ended Expansion Rack Systems

When operating the system in single-ended mode, the power supplies for the main rack and expansion rack must be fed from the same main power source. The main rack and expansion racks cannot be switched ON and OFF separately; either both must be ON or both must be OFF for proper operation.

Power for module operation comes from the Power Supply installed on the Expansion Receiver Module. If the expansion rack includes any Power Supply Booster Carrier and additional rack Power Supply, it must be tied to the same source as the Power Supply on the Expansion Receiver Module.

2.1.3 μ USB Port

This section applies to new hardware implementation only.

The micro USB (μ USB) port, located behind the hinged door on the front of the ENIU, can be used to connect a computer (section 2.2) to update firmware in the ENIU module itself (section 2.4.1), and to update firmware in any modules located in the I/O Station that support indirect firmware updates (section 2.5.1). It may also be used to set the IP address of the ENIU Ethernet port (section 2.3.1).

The USB port accepts a standard USB cable (USB Micro B Male to USB Type A Male). This cable is not included.

Note: The USB port is for firmware upgrades and for setting the IP address only. It is not intended for permanent connection.

Note: With the EBI001 new hardware implementation installed in the rack, the pass-through serial port of the VersaMax ETM no longer supports firmware upgrades.

2.1.4 CE Mark Installation Requirements

The following requirements for surge, electrostatic discharge (ESD), and fast transient burst (FTB) protection must be met for applications that require CE Mark listing:

- The VersaMax I/O Station is considered to be open equipment and should therefore be installed in an enclosure (IP54).
- This equipment is intended for use in typical industrial environments that utilize anti-static materials such as concrete or wood flooring. If the equipment is used in an environment that contains static material, such as carpets, personnel should discharge themselves by touching a safely grounded surface before accessing the equipment.
- If the AC mains are used to provide power for I/O, these lines should be suppressed prior to distribution to the I/O so that immunity levels for the I/O are not exceeded. Suppression for the AC I/O power can be made using line-rated MOVs that are connected line-to-line, as well as line-to-ground. A good high-frequency ground connection must be made to the line-to-ground MOVs.
- AC or DC power sources less than 50V are assumed to be derived locally from the AC mains. The length of the wires between these power sources and the PLC should be less than a maximum of approximately 10 meters.
- Installation must be indoors with primary facility surge protection on the incoming AC power lines.

2.2 Communicating via the μ USB Port

This section applies only to the new hardware version (IC200EBI001-MH or later).

Before using the μ USB Port, refer to Section 2.2.1.

- The IP Address of the ENIU can be set using the μ USB Port. See Section 2.3.1.
- Firmware updates may also be accomplished via the μ USB Port. See Sections 2.4 and 2.5.

Before attempting communications via the μ USB Port, connect the required cable between your PC and the ENIU, as described below.



Figure 31: Open Hinged Door

Use your fingers, or a flathead screwdriver, to lever the tab on the hinged door of the ENIU module, as shown in Figure 31. Once open, you will have access to the micro USB port (Figure 32). Connect a standard micro USB serial cable between your PC and the micro USB port of the ENIU.



Figure 32: Access to μ USB Port with Hinged Door Open

Set up the Serial communications port of your PC to match the ENIU μ USB serial port set-up, as shown in Figure 33.

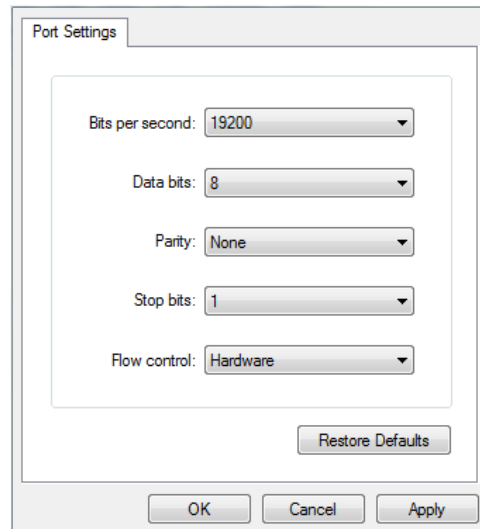


Figure 33: Port Settings

2.2.1 Installing the USB Port Driver

The μ USB port is used for firmware updates and setting the IP address. USB driver files are provided as part of upgrade packages compatible with the VersaMax ENIU.

With the provided installation files accessible on either a local or network drive, connect the USB port on your computer to the micro USB port on the VersaMax ENIU. When requested, direct the installation to the proper location of the installation files.

The serial port name is COM followed by the next available number in the range 1 to 256. After the computer assigns a COM port number to the ENIU micro USB port, that computer uses the same COM port number each time it connects to that particular ENIU module.

During firmware upgrades, WinLoader will display the COM port name followed by the serial number of the VersaMax ENIU which is displayed as shown in Figure 8.

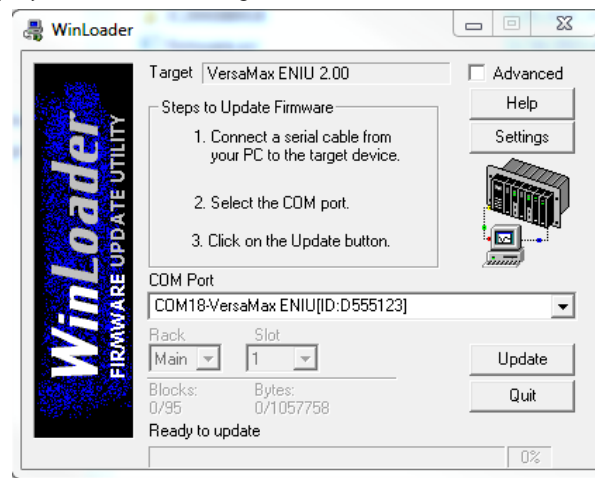


Figure 34: Installing USB Port Driver using WinLoader

In this example (Figure 34), WinLoader shows that COM18 is a VersaMax ENIU with serial number D555123.

Note: When connecting the USB cable to the VersaMax ENIU, you may receive a warning for installing a driver that has not passed Windows Logo testing. Because each VersaMax ENIU has a different serial number, the operating system recognizes each driver installation as different.

2.3 Setting the Network IP Address and Other Network Settings

2.3.1 Setting the Network IP Address via the Serial Console

A serial console is a connection over a serial port that allows a person access to a computer or network device. Using MS Windows®, the default program is Hyper Terminal. Other utilities may be substituted.

Steps to activate the serial console and set the IP Address (or any other Network Setting) of the ENIU.

- 1) Activate the serial console on your PC with parameters as shown in Figure 33.
- 2) Open the hinged door on the ENIU module to reveal a pushbutton and two LEDs.
- 3) Connect your PC to the μ USB port of the ENIU, as discussed in Section 2.2.
- 4) Briefly depress the pushbutton (located just below the MAINT LED). This forces the ENIU to enter serial console mode. In this mode, the ENIU permits configuration of the IP address, subnet mask and default gateway address interactively. If inactive for more than 3 minutes the ENIU will exit this mode and resume normal mode.
- 5) Once the console is communicating with the PC, the MAINT LED turns green.
- 6) The serial console displays information of the ENIU (current IP parameters, MAC Address, Version information), as shown in Figure 35.
- 7) Follow the on-screen instructions to change the modifiable parameters, such as IP Address.

```

Welcome to Ethernet NIU Serial Console.
Copyright (c) 2015-2016. All rights reserved.

Catalogue Number: IC200EBI001-MH
Serial Number: D555123
Date Code Data: 25-MAY-2016
Primary Version: 1.00 (22A1)
Bootloader Version: 1.10 (19A1)
FPGA Version: 1.04 (14C1)
IP Address = 10.40.0.20
Subnet Mask = 255.255.255.0
Gateway = 0.0.0.0
Mac Address = <<00:09:91:46:8D:CE>>

Press 'Enter' to edit IP settings or 'ESC' to exit

```

Figure 35: Display of ENIU Parameters on Serial Console

- 8) Once the IP address is set successfully, the confirmation message is displayed, as shown in Figure 36. For the IP Address change to be effective, power cycle the ENIU or re-download the ENIU configuration.

```

Welcome to Ethernet NIU Serial Console.
Copyright (c) 2015-2016. All rights reserved.

Catalogue Number: IC200EBI001-MH
Serial Number: D555123
Date Code Data: 25-MAY-2016
Primary Version: 1.00 (22A1)
Bootloader Version: 1.10 (19A1)
FPGA Version: 1.04 (14C1)
IP Address = 10.10.0.43
Subnet Mask = 255.0.0.0
Gateway = 0.0.0.0
Mac Address = <<00:09:91:46:8D:CE>>

Press 'Enter' to edit IP settings or 'ESC' to exit
Please enter the IP Address and press 'Enter'
10.10.0.45
Please enter the Subnet mask and press 'Enter'
255.0.0.0
Please enter the Gateway and press 'Enter'
0.0.0.0
.
.
Configuring please wait....
IP Configuration successfully done.
For changes to be effective, power-cycle/re-download configuration.
Exiting..... Bye.
-

```

Figure 36: Changing IP Address via Serial Console

Note: For IC200EBI001-MH or later revisions, the user cannot set the IP address using PME Set Temporary IP Address tool.

2.3.2 Setting the Network IP Address via the ENIU Rotary Switches

This section applies to the old hardware implementation only. Refer to Sections 2.2 and 2.3.1 for equivalent information when using the new hardware implementation.

The IP Address of the ENIU needs to be permanently set by storing a configuration to the ENIU from VersaPro or the Remote I/O config tool. This operation is conducted over Ethernet.

The ENIU first needs to be assigned a temporary IP address to permit the storing of the permanent IP Address.

There are two methods for setting a temporary IP Address

- Use the rotary switches on the ENIU (this only works if the ENIU does not have a permanent IP address stored to it (a new ENIU).
- Create an entry in the ARP table of your PC that sets an IP address for the MAC Address of the ENIU, then Telnet to Port 1 which causes the ENIU to temporarily use the IP address. A utility to do this is available on the GE Support website (www.geautomation.com/support).

Note: If your network will be connected to another network, you must select an IP address that will be compatible with the other network. In that case, you must obtain a unique IP address for the Ethernet ENIU from your system administrator.

Initially Setting the Network IP Address using the Rotary Switches

This operation only works if no IP address has previously been set.

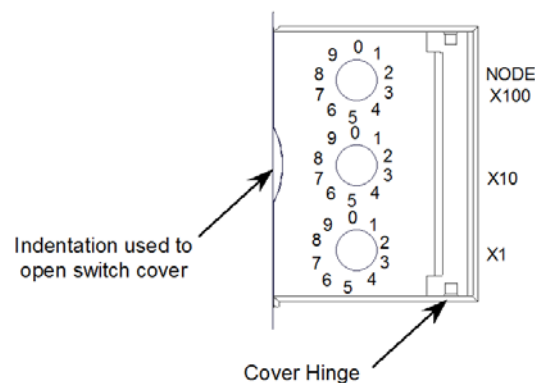


Figure 37: ENIU Rotary Switches (Old Hardware)

- 1) Open the clear protective hinged cover by pulling forward at the indentation in the left side of the ENIU (see Figure 37 above).
- 2) Use a 2.44mm (3/32in) flathead screwdriver to adjust the rotary switches. **Always cycle power after changing the switch settings.**
- 3) The rotary switches, marked Node X100, X10 and X1 select the hundreds, tens, and units digits of the last octet of the network IP address (BCD notation). Select any valid address in the range 1-254. The full IP address will be 195.0.0.X, with the X being set by the rotary switches. For example, setting
 X100 to 2
 X10 to 3
 X1 to 4
 Results in a temporary IP Address of 195.0.0.234 being selected.

At this point, the subnet mask defaults to 255.255.255.0 and there is no configured gateway IP address. So, to communicate to the ENIU (which now has an IP address of 195.0.0.x), an Ethernet device or the programmer must also have a unique IP address in the range of 195.0.0.X, which places it in the same subnet as the ENIU. See Chapter 3 for details on setting up communications parameters and storing the configuration to the ENIU. Once a software configuration that contains a valid IP address (0.0.0.0 should not be used) is stored to the ENIU, the IP address setting on the rotary switches is ignored.

Forcing a Temporary Network IP Address

This section applies to the old hardware implementation only.

If there are no Modbus TCP connections, EGD consumption is not active and the programmer is not connected, the IP address can be temporarily forced to any valid IP address. Forcing the IP address will place the ENIU at the specified address with a default subnet mask matching the class of the forced IP address and no configured gateway IP address.

In the following procedure, an example is given to illustrate the commands used. The example shows the use of the Windows Telnet and ARP (Address Resolution Protocol) utility programs from a Windows command prompt to assign an IP address of 3.16.27.5 to an ENIU that has a MAC address of 08-00-19-01-48-64.

Note: MAC address is in hex characters.

Note that if your computer is connected to the same network (subnet) as the ENIU, they both must have the same network identification (net id) to be able to communicate (refer to Appendix B for further information). The computer's IP address in this example is 3.16.88.139. To force a specific IP address, use the following procedure:

- 1) Determine the Media Access Control (MAC) address of the ENIU. The ENIU's MAC Address is laser marked on the plastic case of the ENIU and is of the form 080019xxxxxx. If the MAC address cannot be read from the plastic case, see the section 6.5, *Determining the MAC Address of the ENIU* for an alternate method.
- 2) On the programmer computer, we will use the Windows ARP utility program (ARP.exe) to set the IP address.
- 3) Start the MS-DOS command prompt from the Windows Start/Programs menu.
- 4) Determine the IP address of your computer. You can use two possible methods:

use the following commands at the command prompt: winipcfg in Windows 95/98/ME or ipconfig in Windows NT/2000 or

look it up in the Windows Start/Settings/Control Panel/Network dialog box under Protocol (TCP/IP) Properties.

The configuration of your computer system will determine which of these methods will work successfully for you.

- 5) .At the MS-DOS prompt, type in the following, then press the Enter key:

arp -s (IP address you want to assign to ENIU) (MAC address of the ENIU) (IP address of your computer)

For this example, we will use the following:

arp -s 3.16.27.5 08-00-19-01-48-64 3.16.88.139

Note: You will not see any reply on the screen (see Figure 38).

- 6) To verify that the ARP table entry was accepted, type the following, then press the Enter key:

arp -a

An entry matching the desired ENIU IP address ("Internet Address") and MAC address ("Physical Address") should be displayed as in Figure 38, with the "Type" listed as 'static'.

```

Microsoft(R) Windows 95
(C)Copyright Microsoft Corp 1981-1996.
C:\WINDOWS>arp -s 3.16.27.5 08-00-19-01-48-64 3.16.88.139
C:\WINDOWS>arp -a
Interface: 3.16.88.139
Internet Address      Physical Address      Type
3.16.27.5             08-00-19-01-48-64    static
C:\WINDOWS>

```

Figure 38: MS-DOS Display of Internet Address and Physical Address

- 7) Next, create a Telnet connection to the ENIU (Port 1) by typing in the following command at the computer's MS-DOS prompt, then pressing the Enter key:

telnet (IP Address) 1

For this example, the command is **telnet 3.16.27.5 1**

A Telnet window will appear. After several seconds, a "Connect Failed!" dialog box will appear (see Figure 39); regardless, the ENIU will change its IP address to the one designated in the Telnet command. (If it takes more than 15 seconds for the Connect Failed box to appear, the Telnet command probably didn't work.)

Note: Do not wait too long to execute the Telnet command because the ARP table static entry will time out in 2 minutes and the Telnet command will not work.

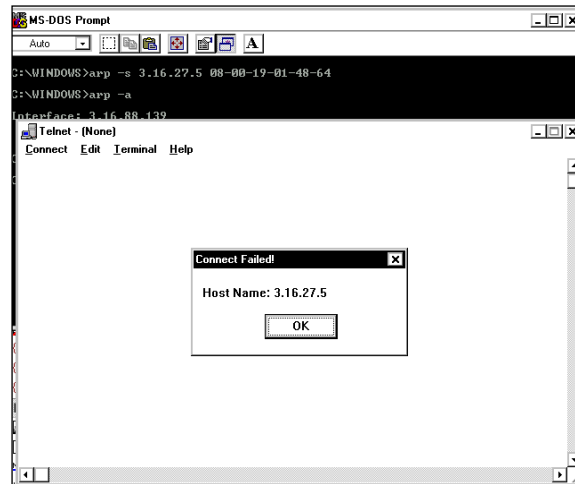


Figure 39: MS-DOS Connect Failed (Expected)

At this point, the forced IP address is **temporary** and will be lost if the ENIU is power-cycled. The following steps show how to make this IP address permanent.

- 8) Configure the network settings of the ENIU to the final desired values using the VersaPro or Remote I/O Manager configuration software. See the configuration software documentation or on-line help for detailed instructions.
- 9) Store the configuration to the ENIU, using the forced IP address in your communications settings. The stored settings will be permanent. See Chapter 3 for details on communications settings and instructions on storing a configuration.

Note: The forced IP address is lost after every power reset of the ENIU. Be sure to store the ENIU configuration parameters using the configuration software as detailed above to make the IP address permanent.

Note: Static ARP table entries are kept in your computer's ARP table, but dynamic entries are automatically cleared after about two minutes. If you wish to remove the static listing once the ENIU has been configured with the configuration program, type "arp -d [IP_ADDRESS]" from the command prompt. Note that all ARP table entries are removed automatically when your computer is shut down.

2.4 Firmware Updates for the ENIU Module

2.4.1 Firmware Updates for the ENIU via the μ USB Port

This section applies to the new hardware implementation only.

Firmware updates using the WinLoader utility are conducted as follows.

Step-By-Step Instructions

1. Open the hinged door on the ENIU module and connect your PC via the micro USB cable, as discussed in Section 2.2.
2. Extract the ZIP file contents of the ENIU firmware upgrade kit to any desired folder on your PC.
3. If the USB driver for this particular VersaMax ENIU (as defined by its serial number) has been installed on your PC, skip to step 7. If not, continue with the following instructions to install new hardware.

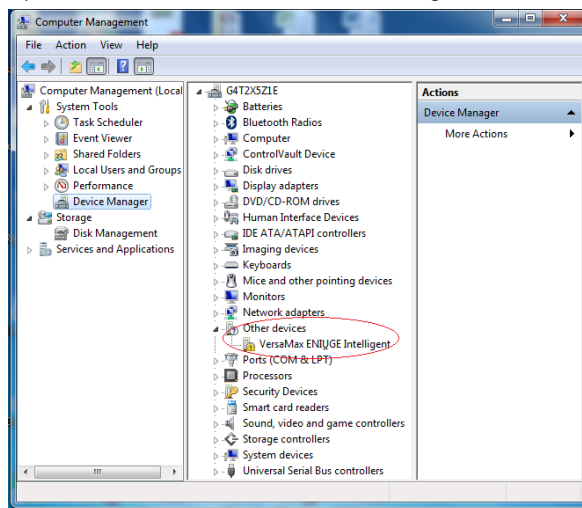


Figure 40: Install ENIU Device Management

4. Extract the ZIP file contents of USB_DRIVER_ENIU to a desired folder on your PC and Click on Start. Open Device Manager under Computer Management. Right Click on *VersaMax ENIUGE Intelligent* and select *Update Driver Software*. Click *Browse*, then select the folder to which the Zip file was extracted; locate the USB_DRIVER_ENIU file. Click *Next*

If a warning stating that driver is not digitally signed is presented at this point, ignore the warning and go ahead with installation. When installation is complete, the following dialog will be presented Click *Close* to complete the installation.

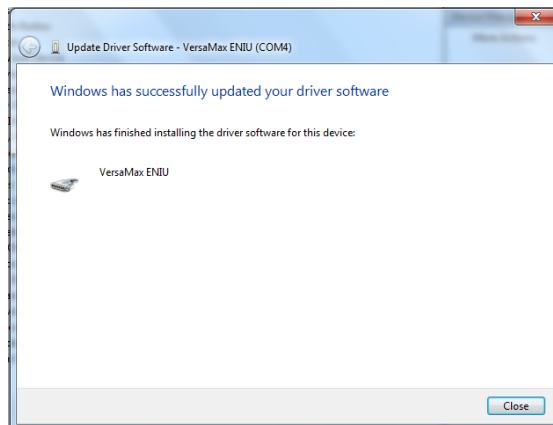


Figure 41: Driver Installation Complete Notification

5. At this point, VersaMax ENIU has been assigned a serial port resource on this PC using the next available COM identifier from 1 to 256. It should be displayed in the Device Manager under the Ports (COM & LPT) listing.
6. If there was a failure to install the driver, the ENIU will be displayed as an Unknown device in the Device Manager and the process can be retried by unplugging the USB cable and reconnecting the ENIU or by updating the driver of the Unknown item in the Device Manager.
7. During firmware upgrades, WinLoader will display the COM port name followed by the serial number of the ENIU as marked on the unit (refer to Figure 8). In the example below, WinLoader shows that serial port COM4 is assigned to a VersaMax ENIU with serial number 99999999.

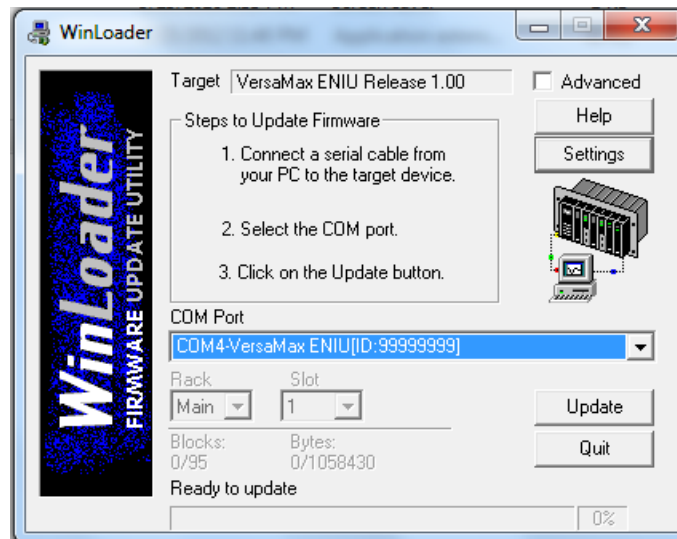


Figure 42: Assign COM port to ENIU

8. Start the WinLoader application using either of these methods:
 - a. Find the WinLoader application by using My Computer or Windows Explorer to view the directory where you extracted or copied the upgrade files. Then double-click the WinLoader icon; or
 - b. Execute WinLoader by clicking the Windows *Start* button and then the *Run...* icon. In the Run dialog, browse to the directory into which you extracted or copied the upgrade files, and click on WinLoader or WinLoader.exe. Then click OK in the Browse and Run dialogs.
9. Read the WinLoader license agreement. If you agree to the terms, click the *I agree* radio button on the License Agreement dialog box and then click the *Next* button. If you do not agree, you will not be able to proceed with the upgrade.
10. If the micro USB cable is properly connected to the USB port on your PC, the installed virtual COM port driver will be listed in the WinLoader COM Port select box.
11. Start the firmware upgrade by clicking the WinLoader *Update* button. While the upgrade is in progress,
 - a. the Ethernet port on the target ENIU is not operational
 - b. the OK, FAULT, LAN, and STAT LEDs on the target ENIU module will blink GREEN LEDs in unison, and
 - c. a status bar on the WinLoader dialog will show the current percentage of completion.
12. When the firmware upgrade has completed successfully, an Update Status dialog box will indicate a successful status and query whether there are additional ENIU modules to be upgraded. Click *No* to close the dialog box and exit WinLoader. If you need to upgrade additional ENIU modules, you will need to connect the micro USB cable to each in turn, and continue from step 3.
13. If the upgrade was not successful, a Firmware Update Status dialog box will indicate that the "Firmware Update Failed". Refer to the *Common Causes of Failure* section below to determine how to correct the situation.

14. When you have successfully completed the firmware upgrade:
 - a. Disconnect the micro USB cable from the ENIU and close the hinged door;
 - b. Power cycle ENIU, then verify that the target ENIU module's OK LED is ON after power-up initialization, indicating correct module operation.
15. When the firmware update has been completed and once the VersaMax ENIU has been rebooted, its previous Ethernet connections are re-established and control resumes.

The VersaMax ENIU enters firmware update mode when commanded to do so from the WinLoader update utility (step 11), or if a firmware component is corrupt or invalid.

If the VersaMax ENIU has experienced a fatal error, the module goes to an LED blink-code error condition and does not communicate with WinLoader, causing WinLoader to return a timeout failure indication. The module must be restarted or power-cycled before reattempting the update.

Note: Recommendation: remove the Ethernet cable while doing a firmware upgrade.

Common Causes of Failure

PC COM Port Already In Use

INDICATION: In WinLoader, the COM port which is already in use will appear in disabled (greyed out) in the COM Port list box.

REMEDY: Look for other applications that may be using the port and try to close them, which causes the COM port to appear as enabled in WinLoader. Start the upgrade procedure again from step 6. If problem still exists, try to reconnect the USB cable or use a different USB port on the PC.

ENIU COM Port Not Listed by WinLoader

INDICATION: In WinLoader, the ENIU COM port with correct serial number is not listed.

REMEDY: Go to Windows Device Manager. Verify there is no yellow exclamation mark on the ENIU device indicating the ENIU driver is installed successfully. If there is a yellow exclamation mark, resolve this issue by re-installing the USB driver.

Wrong PC COM Port Selected or ENIU Powered Off

INDICATION: A WinLoader Firmware Update Status dialog box will report that the "Firmware Update Failed" and that WinLoader "Cannot connect to the target device".

REMEDY: Verify if the serial number shown in WinLoader is the same as that marked on ENIU and follow the instructions in the WinLoader dialog. Click the OK button on the dialog box, and start the upgrade again from step 6.

Firmware Update Process has been Aborted from WinLoader

INDICATION: A WinLoader Firmware Update Status dialog box will report that the "The Firmware update process has been aborted by the user".

REMEDY: Restart the upgrade process again from step 6.

Restarting an Interrupted Firmware Upgrade

If the upgrade process has been interrupted for any reason (for example, a power failure or accidental cable disconnection) a WinLoader Firmware Update Status dialog box will report that the "Firmware Update Failed" and that WinLoader is "Unable to store firmware to target". Restart the firmware upgrade process again.

2.4.2 Firmware Updates for the ENIU via the Ethernet Port Using FTP

This section applies to the old hardware implementation only.

The computer used to perform this upgrade must have an Ethernet interface and be connected to the same Ethernet network as the target ENIU.

Step-By-Step Instructions

1. If your upgrade kit is a ZIP file, extract the ZIP file contents to any directory on your PC. If your upgrade kit contains a diskette, copy the files from the diskette to any directory on your PC. It is recommended that a new or temporary directory be used to contain the contents of the ZIP file or the diskette.
2. On the PC open a Command Prompt Window and if running Windows 95/98/ME enter the command *winipcfg*. If you are running Windows NT/2000/XP/Vista, enter the command *ipconfig* to determine the current IP Address and Subnet Mask of the Ethernet interface(s) of the computer.
3. If you know the IP address of the ENIU and it is compatible with the settings for the Ethernet interface of the computer then go to Step 12. Note on a local Ethernet segment a compatible IP Address will have the Network ID portion of the IP Address (that part that is identified by the Subnet Mask) equal between the PC interface and the ENIU settings.
4. If the ENIU IP Address is known but not compatible with the PC Ethernet interface settings decide if it is possible to change the PC Ethernet interface settings to establish a compatible IP Address. If that is possible make the changes to the PC's Ethernet interface settings and go to Step 12.
5. At this point it is necessary to force a temporary IP Address into the ENIU. Refer to *Forcing a Temporary Network IP Address* on page 36.
6. Pick an IP Address that is compatible with the current network settings of the PC's Ethernet interface that is connected to the network with the ENIU. If necessary have a network administrator assign an IP Address to use for the ENIU – this will typically be necessary if trying to upgrade a ENIU that is connected to a large network along with many other devices.
7. Enter the command *ping* followed with the proposed IP Address to verify that no device on the network is currently assigned that address. The expected response to the ping command at this point is four repeats of the message *Request timed out*.
8. Make note of the MAC Address of the ENIU. This is a 12-digit hexadecimal number etched into the plastic of the ENIU alongside the equivalent barcode. For example, the text MAC 00099100ABCD would be etched into the plastic next to the barcode. Refer to Figure 9 for a picture of the placement of the MAC address and other features of the ENIU.
9. Enter the command *arp -s* followed by the desired IP Address of the ENIU, the MAC Address of the ENIU, and the IP Address of the PC's Ethernet interface that is connected to the network with the ENIU. Example: *arp -s 10.10.1.20 00-09-91-00-AB-CD 10.10.1.2* - note the dashes between the bytes of the MAC Address are required by the arp program. In this example, the IP Address 10.10.1.20 is the proposed address for the ENIU and 10.10.1.2 is the PC Ethernet interface's address.
10. Enter the command *telnet* followed by the desired ENIU IP Address and the port number 1. Example: *telnet 10.10.1.20 1* - this command will time out but the ENIU will now have the selected temporary IP Address.
11. Configure a programmer software package to communicate with the ENIU using the temporary IP Address configured in step 10. Use the programmer to upload the hardware configuration out of the ENIU. Save a note of the IP Address present in the configuration then change the configuration to have the selected temporary IP Address. Download the changed configuration to the ENIU – this will make the IP address permanent. After the firmware upgrade is done, use the programming software to change the IP Address back to the original IP Address, if needed, to restore the ENIU settings.
12. Enter the command *ping* followed by the ENIU's IP Address to verify communication from the PC to the ENIU. If the response to the ping is a timed out response go back to step 5.
13. Enter the command *ftp* followed by the IP Address of the ENIU. Example: *ftp 10.10.1.20*

14. When prompted for the username by the ftp program enter *erase* to cause the ENIU to erase the current firmware in the ENIU. Example: User (10.10.1.20:(none)): *erase* - at this point the LEDs on the ENIU will all turn on for a few seconds then the OK and Faults LEDs will start blinking together to indicate that the ENIU is ready to receive the new firmware. Also note the ftp program will have terminated with a Connection closed by remote host message.
15. Make sure the new firmware file, *eniu.bin*, is in the current FTP command prompt directory then enter the command *ftp* followed by the IP Address of the ENIU again. Example: *ftp 10.10.1.20*
16. When prompted for the username by the ftp program just hit enter with no text. If the ftp program does not accept just an enter keystroke then enter the username (*none*) being sure to include the parenthesis.
17. Enter the ftp command *put eniu.bin* to start sending the new firmware to the ENIU. Do not cycle power while this transfer is ongoing. This transfer may take several minutes to complete.
18. When the ftp transfer completes enter the ftp command *bye*. Do not cycle power before the ENIU completes its reset sequence following this ftp command. (The reset occurs a few seconds after the *bye* command is executed. If the *bye* command is not executed, the reset occurs several minutes after the ftp transfer completes.) The LEDs on the ENIU will stop blinking when the reset sequence is has completed.
19. The configuration data previously stored in the ENIU will still be present after the ENIU completes its reset sequence in step 18 but if the IP Address was changed in step 11 be sure use the programmer software package to change the IP Address back to the original address, which was recorded in step 11.
20. If you have a factory upgrade kit with a label, remove power from the ENIU, place the "UPDATED" label on the surface of the module in a position where it does not cover any vent holes, and restore power to the ENIU.

Common Causes of Failure

Incompatible IP Addresses

INDICATION: No responses to the ping command at step 12.

REMEDY: Have a network administrator assign an IP Address for use in the upgrade that is compatible with the IP Address of the PC. Repeat the procedure from step 7.

Trying to force ENIU IP Address when not on local network segment

INDICATION: Failure to connect with programmer software at step 11.

REMEDY: Move either the computer or the ENIU to a different connection such that both the PC and the ENIU are on the same local segment of the network. Repeat the procedure from step 7.

File Not Found

INDICATION: Unable to transfer the *eniu.bin* file with the ftp program.

REMEDY: Make sure the new firmware *eniu.bin* file is present in the FTP working directory on the PC.

Restarting an Interrupted Firmware Upgrade

If the upgrade process is interrupted for any reason (for example, a power failure or accidental cable disconnection), you can restart it by powering the ENIU off and on, and then starting the upgrade process again from step 12.

2.5 Firmware Updates for Modules in the Remote Node

2.5.1 Indirect Firmware Upgrade via the μ USB Port

This section applies to the new hardware implementation only. The old hardware implementation does not support the μ USB port.

Firmware updates using the WinLoader utility are conducted via the μ USB port. See Section 2.2, *Communicating via the μ USB Port*. Once the standard micro USB cable is in place, the firmware update process may commence.

The target module to be updated is identified by its rack and slot location in the VersaMax ENIU I/O Station.

Step-By-Step Instructions

1. If your upgrade kit is a ZIP file, extract the ZIP file contents to any directory on your PC. If your upgrade kit contains a diskette, copy the files from the diskette to any directory on your PC.
2. Close any PC applications (including PLC programming applications) that may be using the PC COM port to which the standard micro USB cable is connected.
3. Disconnect the network cable connector from the ENIU RJ-45 port (refer to Figure 8).
4. Start the WinLoader application on your PC using either of these methods:
 - a. Find the WinLoader application by using My Computer or Windows Explorer to view the directory into which you extracted or copied the upgrade files. Then double-click the WinLoader icon; or
 - b. Execute WinLoader by clicking the Windows Start button and then the Run... icon. In the Run dialog, browse to the directory into which you extracted or copied the upgrade files, and click on WinLoader or WinLoader.exe. Then click OK in the Browse and Run dialogs.
5. Read the WinLoader license agreement. If you agree to the terms, click the "I agree" radio button on the License Agreement dialog box and then click the Next button. If you do not agree, you will not be able to proceed with the upgrade.
6. If the standard micro USB cable is not connected to the COM1 port on your PC, select the correct port via the WinLoader COM Port select box.
7. Select the rack and slot location of the target module (Figure 43) by clicking the Rack/Slot drop-down box.
 - a. Expand the rack where the target module is installed (by clicking its "+" control) to show the slot list for that rack.
 - b. Select the slot number where the target module is installed by clicking the corresponding check box.
 - c. Do not check the select box for the rack.

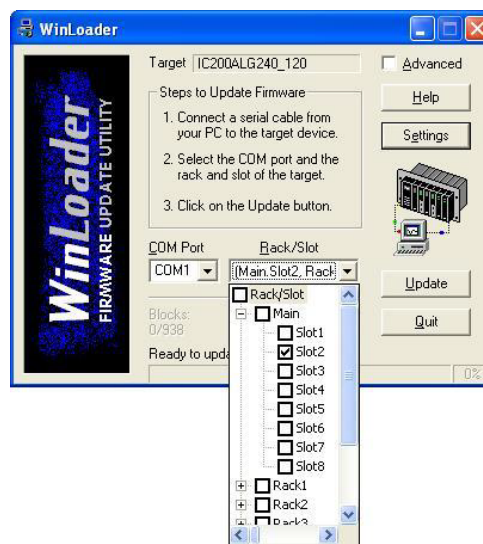


Figure 43: Rack/Slot Selection in WinLoader

The illustration in Figure 43 shows that Slot 2 in the Main rack has been selected. Do not select more than one slot – VersaMax ENIU modules do not support multiple upgrade targets in one session. Note that the text displayed in the Target box will depend on the target module and its firmware version.

If multiple modules within a rack require firmware updates, select the individual slots sequentially and perform a firmware upgrade on each module in this manner.

8. Start the firmware upgrade by clicking the WinLoader Update button. While the upgrade is in progress, the OK LED on the target module will blink, and a status bar on the WinLoader dialog will show the current percentage of completion.
9. When the firmware upgrade has completed successfully, a Firmware Upgrade Summary dialog box will indicate success. Click OK to close the dialog box.
10. If you have more modules of the same type to upgrade, return to step 8 for modules in the same I/O station or to step 3 for modules in a different I/O station (skipping steps 5 and 6). If you do not have more modules to upgrade, click the WinLoader Quit button.
11. If the upgrade was not successful, a Firmware Update Status dialog box will indicate that the “Firmware Update Failed”. See the *Common Causes of Failure* section below to determine how to correct the situation.
12. When you have successfully completed all the firmware upgrades in the remote I/O station, disconnect the standard micro USB cable from the ENIU and close the hinged door.
13. If you have a factory upgrade kit with a label, affix the “UPDATED” label on the side of the module in a location where it does not cover any vent holes.
14. Verify that the OK LED of the target module is ON, indicating correct module operation. In the unlikely event that the OK LED is blinking, cycle the power source for the corresponding rack off and on.
15. Reconnect the network cable connector to the ENIU.
16. Verify that the ENIU network LEDs indicate correct network operation.
17. You have completed the upgrade procedure.

While the firmware upgrade to the IO modules is in progress, Ethernet communication ceases. Ethernet communications resume automatically once the ENIU is power cycled (which is automatic after the firmware upgrade).

During the I/O module firmware upgrade process, LED blinks in a pattern described in the LED section. The Ethernet communication is blocked during the update. The module’s Ethernet ports remain non-operational.

When the firmware update on a module has completed, previous connections are re-established and control resumes.

Note: Remove Ethernet Cable while doing the firmware upgrade.

Note: With the EBI001 new hardware implementation installed in the rack, the pass-through serial port of the VersaMax ETM no longer supports firmware upgrades.

2.5.2 Indirect Firmware Upgrade via the Expansion Port

This section applies to old hardware implementation only.

An indirect firmware upgrade of an intelligent I/O module located in a local or expansion rack controlled by the ENIU module is performed by connecting a PC running WinLoader to the expansion port located on the left side of the ENIU module via cable IC200CBL002.

Step-By-Step Instructions

1. If your upgrade kit is a ZIP file, extract the ZIP file contents to any directory on your PC. If your upgrade kit contains a diskette, copy the files from the diskette to any directory on your PC.
2. Close any PC applications (including PLC programming applications) that may be using the PC COM port to which IC200CBL002 cable will be connected.
3. Disconnect the network cable connector from the ENIU RJ-45 port (refer to Figure 9).
4. Connect the firmware upgrade cable between a 9-pin serial port on your PC or converter and the 16-pin expansion connector on the ENIU. This connector is located behind a flexible cover on the left edge of the ENIU (refer to Figure 2). Remove and save the cover. Be careful to center the cable connector on the 16-pin header. Seat the connector by pushing it gently onto the header pins. Avoid bending the pins.
5. Start the WinLoader application on your PC using either of these methods:
 - a. Find the WinLoader application by using My Computer or Windows Explorer to view the directory into which you extracted or copied the upgrade files. Then double-click the WinLoader icon; or
 - b. Execute WinLoader by clicking the Windows Start button and then the Run... icon. In the Run dialog, browse to the directory into which you extracted or copied the upgrade files, and click on WinLoader or WinLoader.exe. Then click OK in the Browse and Run dialogs.
6. Read the WinLoader license agreement. If you agree to the terms, click the "I agree" radio button on the License Agreement dialog box and then click the Next button. If you do not agree, you will not be able to proceed with the upgrade.
7. If the IC200CBL002 cable is not connected to the COM1 port on your PC, select the correct port via the WinLoader COM Port select box.
8. Select the rack and slot location of the target module (Figure 44) by clicking the Rack/Slot drop-down box.
 - a. Expand the rack where the target module is installed (by clicking its "+" control) to show the slot list for that rack.
 - b. Select the slot number where the target module is installed by clicking the corresponding check box.
 - c. Do not check the select box for the rack.

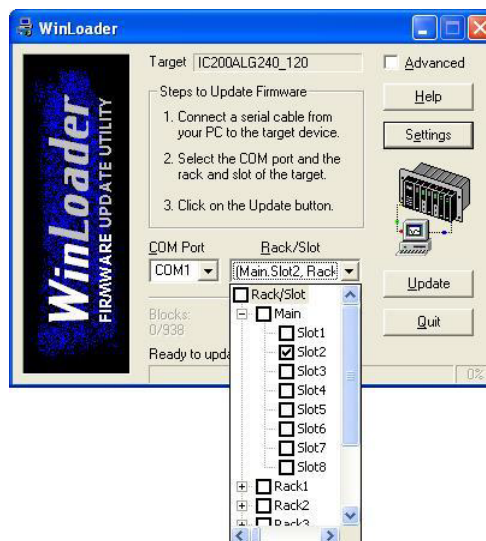


Figure 44: Rack/Slot Selection in WinLoader

The illustration in Figure 44 shows that Slot 2 in the Main rack has been selected. Do not select more than one slot – VersaMax ENIU modules do not support multiple upgrade targets in one session. Note that the text displayed in the Target box will depend on the target module and its firmware version.

If multiple modules within a rack require firmware updates, select the individual slots sequentially and perform a firmware upgrade on each module in this manner.

9. Start the firmware upgrade by clicking the WinLoader Update button. While the upgrade is in progress, the OK LED on the target module will blink, and a status bar on the WinLoader dialog will show the current percentage of completion.
10. When the firmware upgrade has completed successfully, a Firmware Upgrade Summary dialog box will indicate success. Click OK to close the dialog box.
11. If you have more modules of the same type to upgrade, return to step 8 for modules in the same I/O station or to step 3 for modules in a different I/O station (skipping steps 5 and 6). If you do not have more modules to upgrade, click the WinLoader Quit button.
12. If the upgrade was not successful, a Firmware Update Status dialog box will indicate that the “Firmware Update Failed”. See the *Common Causes of Failure* section below to determine how to correct the situation.
13. When you have successfully completed all the firmware upgrades in the remote I/O station, disconnect the upgrade cable from the ENIU and replace the flexible connector cover.
14. If you have a factory upgrade kit with a label, affix the “UPDATED” label on the side of the module in a location where it does not cover any vent holes.
15. Verify that the OK LED of the target module is ON, indicating correct module operation. In the unlikely event that the OK LED is blinking, cycle the power source for the corresponding rack off and on.
16. Reconnect the network cable connector to the ENIU.
17. Verify that the ENIU network LEDs indicate correct network operation.
18. You have completed the upgrade procedure.

Note: With the EBI001 old hardware implementation installed in the rack, the pass-through serial port of the ETM can also be used to perform firmware upgrades.

Common Causes of Failure

PC COM Port Already in Use

INDICATION: A WinLoader Firmware Update Status dialog box will report that the "Firmware Update Failed" and that the selected COM port is either busy or does not exist.

REMEDY: If you selected a port that is not present on your PC, select a different one that is present. If the port actually exists, look for other applications (including PLC programming applications) that may be using the port and close them. Verify that the serial cable is connected to correct COM port on the PC, click the Cancel Upgrade button on the Firmware Update Status dialog box, click OK on the Firmware Upgrade Summary box, select a different COM port if necessary, and start the upgrade again from step 9.

Wrong PC COM Port Selected or CPU/ENIU Powered Off

INDICATION: A WinLoader Firmware Update Status dialog box will report that the "Firmware Update Failed" and that WinLoader cannot connect to the target device.

REMEDY: Verify that you selected the COM port where the upgrade cable is connected. Then follow the instructions in the WinLoader dialog. Then click the Cancel Upgrade button on the Firmware Update Status dialog box, click OK on the Firmware Upgrade Summary box, select a different COM port if necessary, and start the upgrade again from step 9.

Wrong Module Rack/Slot Location

INDICATION: A WinLoader Firmware Update Status dialog box will report that the "Firmware Update Failed" and display either a "Request not supported by target" or "The selected firmware files are not meant for this target device" message.

REMEDY: Click the Cancel Upgrade button on the Firmware Update Status dialog box, click OK on the Firmware Upgrade Summary box, and then verify that the correct rack/slot location for the module is displayed in the Rack/Slot box of the WinLoader dialog. If the location is correct, cycle the power source for the module's rack off and on and then start the upgrade again from step 9. If the location is incorrect, return to step 8 and select the correct location.

Restarting an Interrupted Firmware Upgrade

If the upgrade process is interrupted for any reason (for example, a power failure or accidental cable disconnection), you can restart it by cycling power to the rack in which the target module is located and then starting the upgrade process again from step 9.

Chapter 3 *Configuring an Ethernet NIU and I/O Station*

This chapter explains how an Ethernet NIU and the modules in an I/O Station can be configured. Configuration determines certain characteristics of module operation and also establishes the program references to be used by each module in the system.

This chapter describes:

- *Using Autoconfiguration or Programmer Configuration*
The Ethernet NIU and I/O Station can be either autoconfigured or configured from a programmer using the Remote I/O Manager configuration software.
- *Configuring “Racks” and “Slots”*
Even though a VersaMax I/O Station does not have a module rack, both autoconfiguration and software configuration use the traditional convention of “racks” and “slots” to identify module locations.
- *Software Configuration of the Ethernet NIU and I/O Station*
Software configuration provides greater flexibility than autoconfiguration in setting up an I/O Station. Software configuration is done using VersaPro version 1.5 (or later) or the Remote I/O Manager configuration software. (See relevant Important Product Information (IPI) documents for a version vs. feature table.)
- *Autoconfiguration of the Ethernet NIU and I/O Station*
Autoconfiguration provides a default configuration for the ENIU in Modbus mode and I/O Station and does not require the use of a programmer. I/O modules that have software-configurable features always use their default settings when autoconfigured.
Autoconfiguration using EGD mode is not supported.
- *Configuring the ENIU Produced Exchange, Configuring the ENIU Consumed Exchange, and EGD Exchange Status and Control Bytes for Fault Handling*
These sections use examples to show how to configure consumed and produced exchange data including status and control data.

3.1 Using Autoconfiguration or Programmer Configuration

The Ethernet NIU and I/O Station can be either

- 1) autoconfigured in Modbus mode, or
- 2) configured using the programming software tools discussed in Section 1.7 (applies to both Modbus and EGD mode).

The choice of which configuration method to use depends on the requirements of the application.

3.1.1 Autoconfiguration (Modbus Mode Only)

Autoconfiguration is done by the ENIU itself. It provides a default configuration for the ENIU and I/O Station and does not require the use of a programmer. If there is not a stored configuration already present at powerup, the ENIU sees which modules are installed and automatically creates a configuration for the I/O Station. I/O modules that have software-configurable features can only use their default settings when the I/O Station is autoconfigured. Under autoconfiguration, discrete output data defaults to 0 while analog output data holds its last state. Under Autoconfiguration EGD functionality is disabled. Autoconfiguration is described later in this chapter.

An autoconfiguration can be used *as is* for Modbus mode if the default settings for software-configurable features are acceptable. For EGD mode, and for Modbus mode where default settings are not acceptable, auto configuration can be used as a starting point, and the default configuration can be loaded to VersaPro or Remote I/O Manager and modified.

3.1.2 Software Configuration

Using the configuration software makes it possible to reassign I/O table addresses and to configure many I/O module features. The configuration software runs on a computer that connects to the ENIU via Ethernet.

The configuration software can be used to:

- Create a customized configuration
- Store (write) a configuration to the ENIU
- Load (read) an existing configuration or autoconfiguration from an ENIU
- Compare the configuration in an ENIU with a configuration file stored in the programmer
- Clear a configuration that was previously stored to the ENIU
- Enable EGD functionality

The ENIU retains a software configuration across power cycles. Storing a configuration disables autoconfiguration, so the ENIU will not overwrite the configuration during subsequent startups.

However, clearing a configuration with the programmer does cause a new autoconfiguration to be generated. In that case, autoconfiguration is enabled until a configuration is stored from the programmer again.

Software configuration is summarized later in this chapter. Instructions for installing and using the configuration software are found in the *Remote I/O Manager Software User's Guide*, GFK-1847 or the *VersaPro Programming Software User's Guide*, GFK-1670.

3.2 Configuring "Racks" and "Slots"

Even though a VersaMax I/O Station does not have a physical rack, both autoconfiguration and software configuration use the traditional convention of "racks" and "slots" to identify module locations (Figure 45). Each logical rack consists of the ENIU or an Expansion Receiver module plus up to 8 additional I/O and option modules mounted on the same DIN-rail. Each I/O or option module occupies a "slot". The module next to the ENIU or Expansion Receiver module is in slot 1. Booster power supplies do not count as occupying slots.

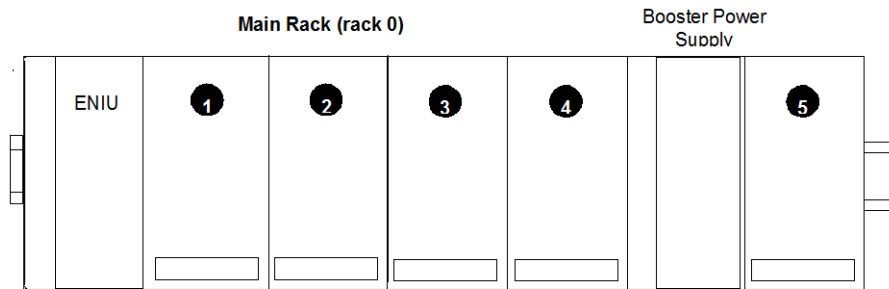


Figure 45: Slot Notation Convention for VersaMax I/O Station

The main rack (containing the ENIU) is always called rack 0.

In an I/O Station that has one expansion rack attached to the expansion bus by a non-isolated Expansion Transmitter Module (IC200ERM002), the expansion rack must be configured as rack 1 (Figure 46).

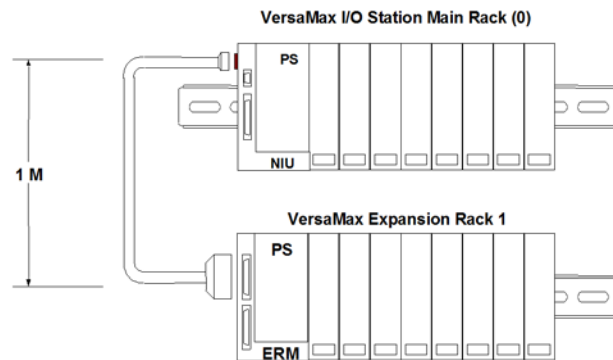


Figure 46: VersaMax I/O Station with 1 Expansion Rack (Single-Ended Cable)

In an I/O Station with an Expansion Transmitter Module (IC200BTM001) and up to seven expansion “racks”, each with an Expansion Receiver Module (IC200ERM001 or IC200ERM002), the additional racks are configured as rack 1 through rack 7 (Figure 47).

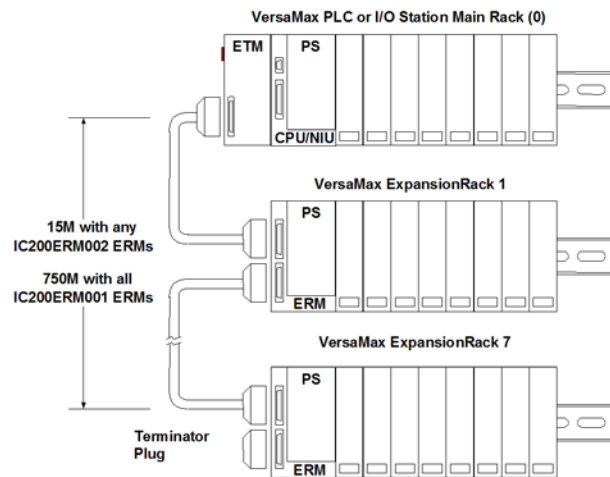


Figure 47: VersaMax Multi-Rack Expansion Notation

3.2.1 Configuring I/O References

As I/O modules are added to the configuration, the configuration software keeps a running total of input/output memory. If the modules added consume more than the maximum memory available, the configuration software displays (1) the reference address of the module that caused the error and (2) an error message.

The I/O Station, including all expansion racks, can include up to 1024 total bytes of data. You can change the I/O references assigned to a module when configuring that module.

3.3 Software Configuration of the Ethernet NIU and I/O Station

Software configuration provides greater flexibility than autoconfiguration in setting up an I/O Station. Use one of the configuration packages listed in Section 1.7.

3.3.1 Notes on Using the Configuration Software

- The same Remote I/O Manager software can configure different types of VersaMax NIUs and all supported IO modules.
- Empty slots are allowed in an ENIU software configuration (unlike an autoconfiguration). However, if empty slots are later filled when using EGD mode, the EGD exchange on the controller will need to be updated.
- The I/O Station cannot include the following communication modules: IC200BEM002 (Profibus Slave module) and IC200BEM103 (DeviceNet Master module).
- The reference addresses assigned to modules in the I/O Station can be edited. Addresses do not need to be consecutive.

3.3.2 Basic Steps of Software Configuration

The Remote I/O Manager software provides a simple default configuration that you edit to match the actual system modules. The default configuration consists of a power supply (PWR001 – see Note below) and an NUI. Carriers and modules are then added in the same sequence as the hardware installation.

Note: The default PWR001 power supply is not powerful enough for use in an ENIU system. Use one of the higher-rated VersaMax power supplies such as the IC200PWR002.

The basic configuration steps are listed below.

- 1) Configure the expansion rack system (local, local single rack, multiple remote rack). This automatically adds the appropriate types of expansion modules to the racks.
- 2) Configure the power supply type and any booster power supplies and carriers.
- 3) Configure the ENIU. This includes changing the NIU type, if necessary, and assigning its parameters as described on the next page.
- 4) Configure the expansion modules if the system has expansion racks.
- 5) Add module carriers and define wiring assignments.
- 6) Place modules on carriers and select their parameters. Configurable parameters of I/O modules are described in the *VersaMax Modules, Power Supplies, and Carriers User's Manual*, GFK-1504.
- 7) Save the configuration file so that it can be stored to the ENIU.

For step-by-step instructions, please refer to the *Remote I/O Manager Software User's Guide*, GFK-1847.

3.3.3 Configuring ENIU Network Parameters

ENIU Network configuration establishes the basic operating characteristics of the Network Interface Unit.

Set the desired IP address, Subnet mask and Gateway prior to connecting the ENIU to the network, using one of the methods discussed in Section 2.3.1 (for new hardware implementation) or Section 2.3.2 (for old hardware implementation). Remember to cycle power after setting the IP Address via the serial console. Once a configuration is stored, the network settings of the ENIU immediately operate using those in the stored configuration.

If your ENIU will be communicating with devices on other networks, the parameters in the following table must be set appropriately. These values should be assigned by the person in charge of your network (the network administrator). Refer to Appendix B for more detailed information about IP Addressing and network setup.

Feature	Description	Config. Default	Choices
IP Address	The IP Address is the unique address of the Ethernet interface as a node on the network.	192.168.0.10	A valid Class A, B, or C address
Subnet Mask	Subnet mask of the ENIU used to identify the section of the overall network the ENIU is on.	255.255.255.0	A valid dotted-notation mask.
Gateway IP Address	IP address of the default gateway (router) device to be used when the ENIU is unable to locate the desired remote device on the local sub-network.	192.168.0.1	A valid Class A, B, or C address in the same subnet as the ENIU.

Note: As all ENIUs contain same default IP address parameters when shipped from the factory. User must ensure each has a unique IP address set before placing it on a network.

Note: The old hardware implementation does not use the defaults listed above. See Section 2.3.2.

3.3.4 Selecting EGD or Modbus

Configuring Modbus Parameters

There are no parameters that are specific to the Modbus protocol. There are, however, three EGD parameters that must be set correctly to disable EGD and enable Modbus. The following table shows how to enable Modbus:

Tab	Feature	Values ⁴
Produced Exchange	Exchange Type	IP Address
Produced Exchange	Consumer IP Address	0.0.0.0
Consumed Exchange	Producer ID	0.0.0.0

If any of these parameters are not set, the ENIU assumes EGD is trying to be configured and will verify the remaining EGD parameters.

Configuring EGD Parameters

To use EGD on the ENIU, the EGD configuration parameters must be set correctly. The following tables show these parameters and their acceptable values.

Produced Exchange Parameters			
Feature	Description	Config. Default	Choices
Exchange ID	Uniquely identifies a particular exchange on a specific producing node.	1	Cannot be changed from default of 1
Exchange Type	Determines whether the exchange will be produced directly to an IP address or to a group address (multicast).	IP Address	IP Address or Group ID
Consumer IP Address	The destination IP address of the exchange. The Consumer IP Address field will only be present if the Exchange Type is IP Address	0.0.0.0	0.0.0.0 to 255.255.255.255 0.0.0.0 is only valid in Modbus mode
Group ID	The destination group of the exchange. The Group ID field will only be present if the Exchange Type is Group ID	1	1 To 32
Producer Period (in ms)	The period at which the exchange is produced on the network	200	5 To 3,600,000

Consumed Exchange Parameters			
Feature	Description	Config. Default	Choices
Exchange ID	Uniquely identifies a particular exchange on a specific producing node	0	1 To 16383
Producer ID	Uniquely identifies the producer of an exchange on a given network	0.0.0.0	0.0.0.0 to 255.255.255.254 0.0.0.0 is only valid in Modbus mode
Group ID	The group IP address the exchange is to be consumed from	0	0 to 32 0 Means no group
Consumed Period (in ms)	The period at which the exchange is to be consumed	200	5 To 3,600,000
Update Timeout (in ms)	The time allowed for the consumption of an exchange before it is deemed inactive. Upon timeout, outputs go to their Default or Hold Last State values.	2000	10 To 3,600,000

⁴ These are the default settings for these parameters

3.3.5 Software Configuration: Load, Store, Verify, Clear

To transfer and check the contents of a configuration, use the Load/Store/Verify functions from the Tools menu. A configuration file must be saved in the programmer (computer) before using the load/store/verify functions.

The computer connects to the Ethernet NIU via the Ethernet network.

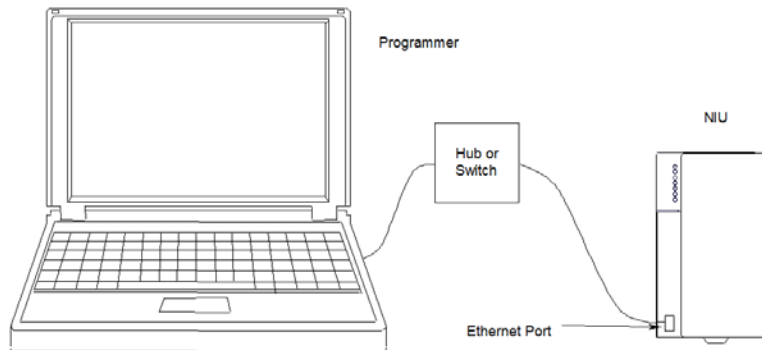


Figure 48: PC to ENIU Connection

To use the Load/Store/Verify functions, the programmer must be able to communicate with the ENIU. The configuration software has a set of communications parameters that need to be correctly set for communicating with the Ethernet NIU, as discussed in the following section.

3.3.6 Creating a New Communications Setup

In VersaPro or Remote I/O Manager, create a communications setup for the Ethernet NIU using the ENIU's IP address. Use these steps:

- 1) Choose Communications Setup from the Tools menu. The Communication Configuration Utility dialog box will appear (Figure 49).
- 2) Click the New button and create your new setup. Configure the following parameters:
 - Enter a Device Name. In the example below, the Device Name of the newly-created setup is "EthernetNIU":
 - Device Model as *VersaMax* (Optional)
 - Default Port as *ENET*
 - IP address as the address of your ENIU (3.16.27.5 in the example)
 - Leave the other parameters blank

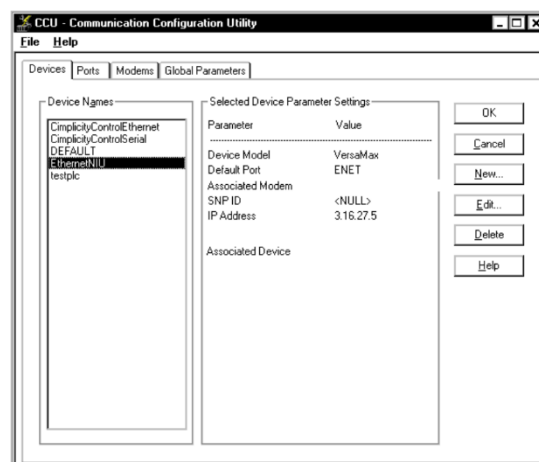


Figure 49: Set up Communications Parameters for ENIU

3.3.7 Modifying the ENIU Communications Settings

If your communications parameter settings are not compatible with the target network, you can change them in the Communications Configuration Utility dialog box, shown in Figure 49. Click the Device Name of your ENIU communications setup to select it (it is called EthernetNIU in Figure 49). Click the Edit button to make changes. Make sure that:

- The IP address is set to the ENIU's current IP address
- The Default Port is ENET
- The Device Model is VersaMax (optional)

Note: The programmer must be able to communicate with the ENIU at the current ENIU settings. This means that if the IP address is set to 192.168.0.10, the Subnet Mask is at 255.255.255.0, and the Gateway IP address is 192.168.0.1, then the programmer must be at a different IP address in order to communicate with the ENIU.

3.3.8 Storing a Configuration to the Ethernet NIU

After completing a configuration in the programmer (and saving it in the programmer), the configuration must be *Stored* to the Ethernet NIU. In the Tools menu, select Load/Store/Verify and click on Store. Since you cannot store a configuration to the ENIU while it is communicating with a controller, you must take the controller off-line to perform the store.

Storing a software configuration disables autoconfiguration, so the ENIU will not overwrite a software configuration with an autoconfiguration during subsequent startups.

If any mismatched, missing, or extra modules are detected, the ENIU turns on its FAULTS LED and will autoconfigure upon a subsequent power cycle for either of the following conditions:

- If the ENIU rejects the configuration due to corrupted or unacceptable data
- If the Ethernet network is disconnected during the store operation

Please see Chapter 6 for information on troubleshooting.

3.3.9 Loading a Configuration from the ENIU to the Programmer

The programming software can *Load* a previously stored configuration from the Ethernet NIU back to the programmer. In the Tools menu, select Load/Store/Verify and click Load.

Note that the following modules share hardware module IDs:

IC200MDL650 loads as IC200MDL636
 IC200MDL750 loads as IC200MDL742
 IC200MDL331 loads as IC200MDL329
 IC200MDD844 loads as IC200MDD842
 IC200MDL141 loads as IC200MDL140

If an Autoconfiguration containing any of the above modules is loaded, the software may display an incorrect catalog number and description for them. Edit any incorrect modules using the programmer before storing the configuration back to the ENIU. Once this has been done, you will be able to load the configuration properly.

Note: User must not perform Load Configuration operation from ENIU immediately after a Clear Configuration or Software Configuration download operation. (ENIU may take 40-60 seconds (based on setup) to configure/clear/auto configure).

3.3.10 Comparing Configurations in the Programmer and ENIU

Use the *Verify* function to compare a configuration file in the programmer with a configuration that was previously stored to the Ethernet NIU. In the Tools menu, select Load/Store/Verify and click Verify.

3.3.11 Clearing a Software Configuration from the ENIU

Use the *Clear* function to remove a previously stored configuration from the ENIU. Clearing a software configuration causes a new autoconfiguration to be generated. For a successful autoconfiguration, users have to wait for 40 – 60 seconds after the clear operation and then power cycle the ENIU. Autoconfiguration remains enabled until a software configuration is stored from the programmer again. A clear function will not erase the network settings of the ENIU. After a *Clear* function, the ENIU will remain at the same IP address with the same subnet mask and gateway IP address.

Note: Clearing the configuration will erase EGD parameter information and cause the ENIU to change to Modbus mode.

3.4 Autoconfiguration of the Ethernet NIU and I/O Station

Autoconfiguration is done by the ENIU itself. It provides a default configuration for the ENIU and I/O Station and does not require the use of a programmer. I/O modules that have software-configurable features always use their default settings when autoconfigured. Under autoconfiguration, discrete output data defaults to 0 while analog output data holds its last state.

When no previous autoconfiguration exists, the ENIU automatically reads, at power-up, the default configuration of the modules installed in the system.

Once this autoconfiguration is complete as described below, the ENIU retains this configuration until it is either (1) cleared or (2) powered up with a changed I/O module configuration, in which case, a new autoconfiguration will be generated that reflects the changes.

Autoconfiguration disables EGD functionality and sets the ENIU to Modbus mode.

3.4.1 Autoconfiguration Sequence

Each module is considered to occupy a “slot”. The position adjacent to the ENIU is slot #1. Booster power supplies do not count as occupying slots.

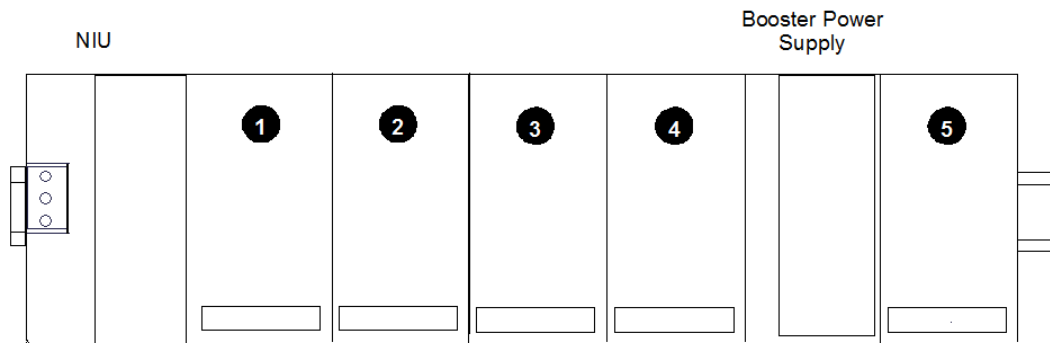


Figure 50: Booster Power Supply Effect on Slot Numbering

Autoconfiguration starts at slot 1 of rack 0 (the main rack) and continues in the same order the modules occupy in the I/O Station.

Autoconfiguration stops at the first empty slot or faulted module. For example, if there are modules in slots 1, 2, 3, 5 and 6 but slot 4 is empty, the modules in slots 5 and 6 are not autoconfigured. The ENIU reports *Extra I/O Module* faults.

Note: For the autoconfiguration process to work as expected, any additional power supplies in the I/O Station must be powered up at “exactly” the same time as or before the main power supply.

3.4.2 Autoconfiguration Assigns Reference Addresses

The ENIU stores data internally as discrete input bits, discrete output bits, analog input words, and analog output words.

The NIU Data Memories	
%I	<i>discrete input bits</i>
%AI	<i>analog input words</i>
%Q	<i>discrete output bits</i>
%AQ	<i>analog output</i>

Figure 51: ENIU Memory Types

During autoconfiguration, the ENIU automatically looks at the modules installed in the I/O Station and assigns them to addresses in this internal I/O map. Reference addresses are assigned in ascending order. For modules that have multiple data types (for example, mixed I/O modules), each data type is assigned reference addresses individually. Chapter 4 contains related information on memory mapping and the Modbus Function Codes used to access ENIU memory.

Modules that have software-configurable features use their default settings when autoconfigured. Under autoconfiguration, discrete output data defaults to 0 while analog output data holds its last state. I/O module features are described in the *VersaMax Modules, Power Supplies, and Carriers Manual*, GFK-1504.

3.4.3 Adding I/O modules to an Autoconfigured I/O Station

If additional I/O modules are added to an existing I/O Station, they do not become part of the autoconfiguration until the ENIU is power-cycled.

3.4.4 Replacing I/O modules in an Autoconfigured I/O Station

If I/O modules need to be replaced in an existing I/O Station, they do not become part of the autoconfiguration until the autoconfiguration is cleared and freshly configured.

3.4.5 Clearing an Autoconfiguration

To clear an existing autoconfiguration, power down the ENIU, disconnect the ENIU from the first I/O module, and power up the ENIU. The autoconfiguration in the ENIU is then cleared. (An existing software configuration can only be cleared using the programmer, as described previously in this chapter.)

3.4.6 Hot Inserting I/O Modules

It is possible to hot insert I/O modules into an I/O Station. If the module being replaced already exists in the configuration, no other action is necessary to make the module operable. Otherwise, the ENIU will report a system configuration mismatch, and the user will need to store an updated configuration to the ENIU in order to make the newly inserted module operational.

3.4.7 **Autoconfiguring an I/O Station with Expansion Racks**

- The Expansion Receiver modules must have their rack ID selection dials set correctly.
- Any available rack number can be used for a new expansion rack but they must all be unique (no duplicate rack numbers). It is best to assign expansion rack numbers from lowest (1) to highest (7) as they are installed.
- If a new expansion rack is added in the future, it should be assigned a rack number that is higher than the racks that are already installed. If a new expansion rack with a lower rack number is added and the system is then auto-configured, the racks numbered higher than the new rack number have their I/O reference addresses shifted in the reference tables. Any existing program logic using those references would need to be adjusted to use the new references.
- When autoconfiguring an I/O Station with expansion racks, either all racks must be powered from the same source or the expansion racks must be powered up before the main rack.
- To add another expansion rack to the I/O Station, the I/O Station must be powered down. After adding the expansion rack, power up the I/O Station. It will then autoconfigure.
- To force autoconfiguration for expansion racks, first power down the ENIU. Remove the transmitter module from the ENIU or remove the expansion cable at the transmitter. Power up the ENIU and let it autoconfigure. Power the ENIU down again, reattach the transmitter or cable, and power up the ENIU again.

3.5 Configuring the ENIU Produced Exchange

The ENIU Produced Exchange contains input module and fault status data. Even if your ENIU system does not use any input modules, you must configure a Produced Exchange for the transmission of fault status data.

3.5.1 Phase 1 – Planning the Exchange

This phase should be done first. Planning the exchange will save time and mistakes during the configuration phase. A worksheet should be filled out and kept as part of your system documentation. Blank worksheets are provided at the end of this section.

ENIU Rack Configuration Example

In this example, the ENIU rack is configured as illustrated below.

ENIU	Rack 0 Module 1	Rack 0 Module 2	Rack 0 Module 3	Rack 0 Module 4	Rack 0 Module 5	Rack 0 Module 6
	%Q33-48	%I33-48	%Q17-32	%I17-32	%AQ5-8	%AI5-8

Figure 52: ENIU Rack Configuration Example

ENIU Produced Exchange Details

An ENIU can only have one Produced Exchange. It automatically configures its own Produced Exchange, which contains only input (%I and %AI) data; the produced exchange is automatically generated from the hardware configuration stored in the ENIU. The ENIU's Produced Exchange data will be sent in the same order that the input modules are configured in the rack, from left to right. For this example (Figure 53), 16 bytes of data will be sent in the following order:

- The first four bytes contain Status data.
- The next two bytes contains data from Module 2 (%I0033 - %I0048)
- The next word contains data from Module 4 (%I0017 - %I0032)
- The final eight bytes contain data from Module 6 (%AI0005 - %AI0008) Note that each analog input produces two bytes of data.

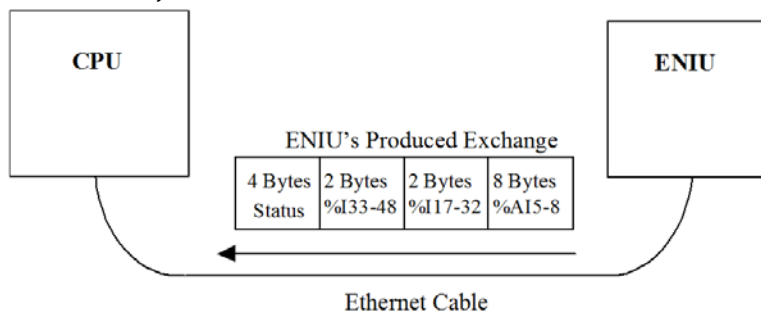


Figure 53: ENIU Produced Exchange Example

ENIU Produced Exchange Memory Mapping Worksheet Example

This example worksheet (Section 3.5.2) shows the mapping of CPU memory to ENIU memory. Note that the organization of memory addresses in the ENIU's Produced Exchange is based on the ENIU rack's physical configuration. In automatically configuring its Produced Exchange, the ENIU uses its stored configuration, starting with Slot 1 (from left to right). As it encounters input addresses, it adds them to the exchange in the order in which it encounters them. Therefore, the ENIU's Produced Exchange is not configurable in any way by the user. For example, the first input module the ENIU encounters when scanning its configuration from left to right is the module in slot 2; therefore, the first byte of data the ENIU sends after the Status data will be from the input module in slot 2.

Note that from the CPU's standpoint this is a Consumed Exchange. Addresses in CPU memory must be configured by the user. The first byte of an exchange is called byte 0. All discrete addresses used in an exchange must be byte-aligned.

ENIU Produced Exchange Memory Mapping Worksheet				
Produced by: ENIU			Consumed by: CPU	
Exchange Byte No.	CPU Memory (assigned by user)	Data Type	ENIU Rack Location	ENIU Memory (assigned automatically by ENIU)
0-3	%R0050 - %R0051	Status	N/A	N/A
4-5	%I0033 - %I0048	Discrete	Rack 0, Slot 2	%I0033 - %I0048
6-7	%I0017 - %I0032	Discrete	Rack 0, Slot 4	%I0017 - %I0032
8-15	%AI0005 - %AI0008	Analog	Slot 6, inputs 1-4	%AI0005 - %AI0008

Note: To avoid confusion, GE recommends that ENIU memory addresses be mapped to the same addresses in CPU memory; however, this is not a requirement (any applicable CPU memory addresses can be used).

3.5.2 Phase 2 – Configuring the PLC CPU to Receive the ENIU Produced Exchange

This section uses the data from the planning phase in the example shown in the previous section. This example uses an IC693CPU364 (which is a combination CPU/Ethernet Interface module) as the master device in the system that will be exchanging EGD data with the ENIU. The screens in the following figures were copied from VersaPro Release 2.00.

- To begin, open the Hardware Configuration File in VersaPro
- Configure the CPU364 assigning all applicable CPU and Ethernet Interface settings.
- Next, right-click the CPU364 module. A short cut menu will appear as shown in Figure 54.

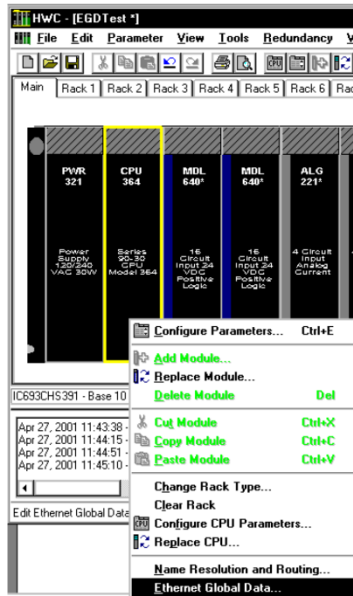


Figure 54: Configure Parameters Menu - CPU

- On the short cut menu, choose Ethernet Global Data. The Ethernet Global Data configuration window will appear.
- Select the Consumed Exchanges tab (Figure 55) since the ENIU's Produced exchange becomes the CPU364's Consumed exchange.

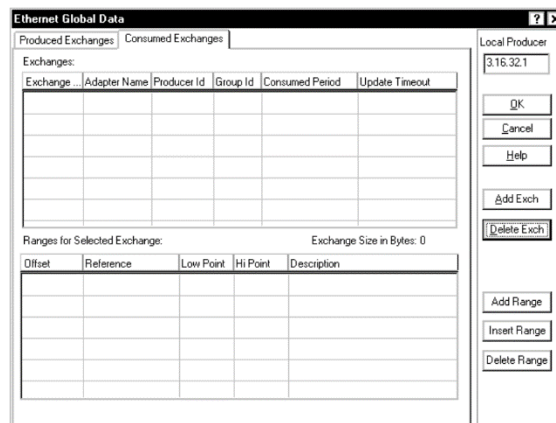


Figure 55: Consumed Exchanges Tab of CPU Configuration

- Note that the ENIU only supports one Produced exchange and one Consumed exchange. Begin configuring the CPU's Consumed exchange (to receive the ENIU's Produced exchange) by clicking the Add Exch. button.

- In the Exchange field, enter 1 for the Exchange Number (Figure 56).

IP Address of Local Producer (CPU364)

Exchange Table

Range Table

Figure 56: Configure Consumed Exchange Number

- 1) Configure the following Consumed Exchange fields:

- **Exchange:** Enter the exchange number. The ENIU only produces one exchange, and it must be set to a value of 1.
- **Adapter Name:** This is the configured Adapter Name of the CPU364. It was configured previously in the CPU configuration window.
- **Producer ID:** The IP address of the ENIU (3.16.32.2 for this example).
- **Group ID:** Since the exchange is only being sent to the CPU364 and not to a group, this is set to 0.
- **Consumed Period:** This field sets how frequently this exchange is produced. The default is 200ms. This is probably too slow for many applications. We will set it to 30ms for this example. The lower the value, the more frequently the exchanges will occur; however, this also increases network traffic. (Whether the network traffic increase is an issue depends on whether other devices besides the CPU364 and ENIU are using this network. If these are the only two devices on the network, then it's not an issue.)
- **Update Timeout:** Should be set to a value at least twice the value of the ENIU producer exchange period value. If the CPU364's Ethernet Interface does not receive an exchange from the ENIU within the time set in this parameter, it will declare the occurrence of a refresh error.

Once configured, the Exchange Table should look similar to Figure 57.

Exchange ...	Adapter Name	Producer Id	Group Id	Consumed Period	Update Timeout
1	EGD1	3.16.32.2	0	30	70

Figure 57: CPU's Consumed Exchange Table Configured

- 2) Next, configure the Range Table for the CPU364's consumer memory addresses that will be used to store the ENIU's produced data. The data you entered in the ENIU Produced Exchange Memory Mapping Worksheet will be used to configure the Range Table.
 - Set the Status reference address first (in the first row of the Range Table). You can fill in the Reference field by clicking it and selecting the memory type from the drop-down list.
 - Click the Low Point field and enter a starting address number for this range. The Hi Point field will be filled in by the software based on the Reference and Low Point values.
 - Enter a Description to help you identify this data range.
 - Click the Add Range button. This will active a row in the Range Table.
 - Click the Reference field and choose a memory type (such as %R, %AI, %I, etc.) from the drop-down menu.
 - Click the Low Point field and enter a start address number for this range. Then click the High Point field and enter an end address number for this range.
 - Click the Description field and enter a description to help identify this data range.
 - Continue to add ranges until the exchange is fully configured. For this example, the data from the example worksheet is used.

When finished, the Ethernet Global Data window looks like Figure 58.

The screenshot shows the 'Ethernet Global Data' window with the 'Consumed Exchanges' tab selected. It displays a table of exchanges and a detailed 'Ranges for Selected Exchange' table.

Exchange	Adapter Name	Producer Id	Group Id	Consumed Period	Update Timeout
1	EGD1	3.16.32.2	0	30	70

Offset	Reference	Low Point	Hi Point	Description
Status	%R	501	501	Status: This is where the PLC is to
Time Stamp	NOT USED			Time Stamp: Optional place for the
0.0	%R	50	51	ENIU Status Data
4.0	%I	33	48	ENIU Rack 0, Slot 2
6.0	%I	17	32	ENIU Rack 0, Slot 4
8.0	%AI	5	8	ENIU Rack 0, Slot 6

A Memory Range to be passed in the exchange

Figure 58: Range Table Configured

Note that the Status word in the first row of the Range Table is for PLC exchange status use and is not part of the exchange.

3.6 Configuring the ENIU Consumed Exchange

The ENIU Consumed Exchange contains output module and fault command data from the CPU. Even if your ENIU system does not use any output modules, you must configure a Consumed Exchange for the transmission of fault command data.

3.6.1 Phase 1 – Planning the Exchange

This phase should be done first. Planning the exchange will save time and mistakes during the configuration phase. A worksheet should be filled out and kept as part of your system documentation. Blank worksheets are provided at the end of this section

ENIU Rack Configuration Example

In this example, the ENIU rack is configured per Figure 59.

ENIU	Rack 0 Slot 1	Rack 0 Slot 2	Rack 0 Slot 3	Rack 0 Slot 4	Rack 0 Slot 5	Rack 0 Slot 6
	%Q33-48	%I33-48	%Q17-32	%I17-32	%AQ5-8	%AI5-8

Figure 59: ENIU Example Rack Configuration

ENIU Consumed Exchange Example Overview

The ENIU Consumed Exchange contains only output (%Q and %AQ) data. The exchange data will be automatically written to the output modules in the order in which the output modules are configured in the ENIU rack, from left to right; no user configuration is required at the ENIU end. For this example, the 16 bytes of data consumed will be written in the following order:

- The first four bytes contain Control data that instruct the ENIU how to act on fault data (if faults are present).
- The next two bytes will be written to %Q0033 - %Q0048 (from module in Slot 1)
- The next two bytes will be written to %Q0017 - %Q0032 (from module in Slot 3)
- The final eight bytes will be written to %AQ0005 - %AQ0008 (from module in Slot 5). Note that each analog input requires two bytes in the exchange.

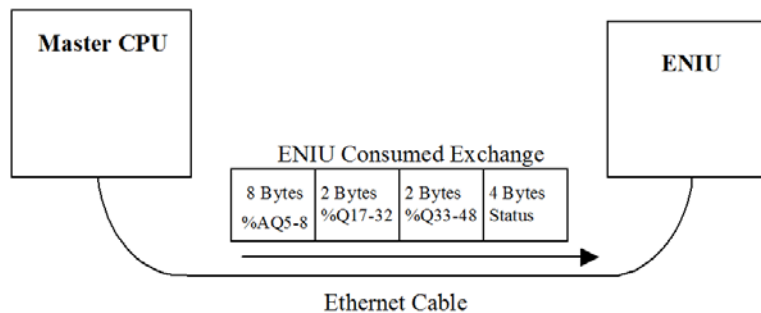


Figure 60: ENIU Consumed Exchange Example

ENIU Consumed Exchange Example Memory Mapping Worksheet

This example worksheet shows the mapping of CPU memory to ENIU memory. Note that the memory addresses in the ENIU are based on the ENIU rack's physical configuration. In automatically writing consumed data to its output modules, the ENIU scans its rack configuration starting with Slot 1, and writes the Consumed Exchange data it received to the output modules in the order in which it encounters them. Therefore, the ENIU exchange addresses are not selectable by the user. For example, the first output module encountered when scanning from left to right is the module in slot 1; therefore, the first byte of data received (other than Control data), will be written to the module in slot 1.

Note that from the CPU's standpoint this is a Produced Exchange. CPU addresses must be configured by the user. The first byte of an exchange is designated byte 0. All discrete addresses used in an exchange must be byte-aligned.

ENIU Consumed Exchange Memory Mapping Worksheet				
Produced by: CPU			Consumed by: ENIU	
Exchange Byte No.	CPU Memory (assigned by user)	Data Type	ENIU Rack Location	ENIU Memory (assigned automatically by ENIU)
0-3	%R0200 - %R0201	Status	N/A	N/A
4-5	%Q0033 - %Q0048	Discrete	Slot 1	%Q0033 - %Q0048
6-7	%Q0017 - %Q0032	Discrete	Slot 3	%Q0017 - %Q0032
8-15	%AQ0005 - %AQ0008	Analog	Slot 5, outputs 1-4	%AQ0005 - %AQ0008

Note: To avoid confusion, GE recommends that ENIU memory addresses be mapped to the same addresses in CPU memory; however, this is not a requirement (any applicable CPU memory addresses can be used).

3.6.2 Phase 2 - Configuring the CPU to Send to the ENIU's Consumed Exchange

The CPU364's Produced Exchange (which is the ENIU's Consumed Exchange) is configured in a similar manner as the CPU364's Consumed Exchange, described in the previous section, "Configuring the ENIU's Produced Exchange."

- To begin, access the Ethernet Global Data configuration window, as in the previous example. Select the Produced Exchanges tab (Figure 61).

Exchange	Adapter Name	Cons Type	Cons Addr...	Send Type	Prod Period	Reply Rate

Offset	Reference	Low Point	Hi Point	Description

Local Producer Identifier: A unique number (in IP Address Format) to identify the PLC

Figure 61: Produced Exchanges Tab of CPU Configuration

- Note that the ENIU only supports one Produced Exchange and one Consumed Exchange. Begin configuring the Produced Exchange by clicking the Add Exch button.
- In the Exchange field, you must enter 1 for the Exchange number, as shown in Figure 62.

Exchange	Adapter Name	Cons Type	Cons Addr...	Send Type	Prod Period	Reply Rate
1	EGD1	Group ID		Always	200	0

Offset	Reference	Low Point	Hi Point	Description
Status	1	1	16	Status: This is where the PLC is to put

Exchange Identifier: Enter a unique number to identify the exchange

Figure 62: Configure Produced Exchange Number

- 1) Configure the following Produced Exchange fields:
 - **Adapter Name:** This field is based on the Adapter Name you already configured for the Ethernet Interface (in the CPU364 configuration window).
 - **Cons Type:** This field specifies the consumer type. Three choices are offered: Group ID, IP Address, or Name. For this example, the choice is IP Address, since only the ENIU is to receive this exchange. We will use the ENIU's IP address in the Cons Addr field to identify it as the consumer of the exchange.
 - **Cons Addr:** Here we will enter the ENIU's IP address; this is based on our choice for the Cons Type field.
 - **Send Type:** This field is not currently configurable. It is "permanently" set to Always.
 - **Prod Period:** This field sets how frequently this exchange is produced. The default is 200 ms. This is probably too slow for many applications. We will set it to 30 ms for this example, which is the same value we set for the Consumed Period in the Consumed Exchange in the previous section. Setting these to the same value is recommended for best efficiency.

Note: The **Prod. Period** parameter sets the network production time and is independent of I/O scan time. For example, if you process changes at a 20ms rate and you produce at 10ms, you may still not see every change if the I/O is scanning at a 25ms. rate. I/O scan is affected by number and type of I/O modules, TAN configuration, and network loading.

- **Reply Rate:** This field is not currently used. It is "permanently" set to 0.

Once configured, the Exchange Table should look similar to Figure 63.

Exchange ...	Adapter Name	Cons Type	Cons Addr...	Send Type	Prod Period	Reply Rate
1	EGD1	IP Address	3.16.32.2	Always	30	0

Figure 63: CPU's Produced Exchange Configured

- 2) Next, configure the Producer's memory addresses that will be the source of the data shared in the exchange.
- Set the Status reference address first (in the first row of the Range Table). You can fill in the Reference field by clicking it and selecting the memory type from the drop-down list.
 - Click the Low Point field and enter a starting address number for this range. The Hi Point field will be filled in by the software based on the Reference and Low Point values.
 - Enter a Description to help you remember what the data range is used for.
 - Click the Add Range button. This will active a row in the Range Table.
 - Click the Reference field and choose a memory type (such as %R, %AI, %I, etc.) from the drop-down menu.
 - Click the Low Point field and enter a start address number for this range. Then click the High Point field and enter an end address number for this range.
 - Click the Description field and enter a description to help identify this data range.
 - Continue to add ranges until the exchange is fully configured. For this example, the data shown in the example worksheet is used.

When finished, the Ethernet Global Data window looks like Figure 64.

The screenshot shows the 'Ethernet Global Data' window with the 'Produced Exchanges' tab selected. The 'Local Producer' field is set to '3.16.32.1'. The 'Exchanges' table has one entry: Exchange 1, Adapter Name EGD1, Cons Type IP Address, Cons Addr 3.16.32.2, Send Type Always, Prod Period 30, and Reply Rate 0. The 'Ranges for Selected Exchange' table shows five ranges for Exchange 1, with an 'Exchange Size in Bytes' of 16. The ranges are: Status (Offset 0.0, Reference %R, Low Point 500, Hi Point 500, Description 'Status: PLC Internal Status Word'), 0.0 (Offset 0.0, Reference %R, Low Point 200, Hi Point 201, Description 'ENIU Status Command Words'), 4.0 (Offset 4.0, Reference %Q, Low Point 33, Hi Point 48, Description 'To ENIU slot 1'), 6.0 (Offset 6.0, Reference %Q, Low Point 17, Hi Point 32, Description 'To ENIU slot 3'), and 8.0 (Offset 8.0, Reference %AQ, Low Point 5, Hi Point 8, Description 'To ENIU slot 5, outputs 1-4'). The 'Add Range' button is highlighted.

Exchange ...	Adapter Name	Cons Type	Cons Addr...	Send Type	Prod Period	Reply Rate
1	EGD1	IP Address	3.16.32.2	Always	30	0

Offset	Reference	Low Point	Hi Point	Description
Status	%R	500	500	Status: PLC Internal Status Word
0.0	%R	200	201	ENIU Status Command Words
4.0	%Q	33	48	To ENIU slot 1
6.0	%Q	17	32	To ENIU slot 3
8.0	%AQ	5	8	To ENIU slot 5, outputs 1-4

Figure 64: CPU's Produced Exchange Table Configured

Note that the Status word in the first row of the Range Table is for PLC internal status use and is not part of the exchange.

3.6.3 Blank Memory Mapping Worksheet

Make copies of this blank worksheet and use it to configure your ENIU exchanges. Use multiple worksheets to fit all required exchanges.

[illegible]

3.7 EGD Exchange Status and Control Bytes for Fault Handling

Both ENIU exchanges have four bytes dedicated to Status or Control data. These four bytes are the first data sent in the exchanges. This section describes how to configure this portion of the exchanges. In the exchanges produced by the ENIU, the status bytes contain fault status information from the ENIU. In the exchanges consumed by the ENIU, the status bytes contain Control commands from the master device.

3.7.1 Control Bytes for Consumed Exchange

For the ENIU's Consumed Exchange, only Control Byte 1 is active; the other three bytes are always set to 0. Control Byte 1 is used to acknowledge and clear ENIU faults.

Control Byte 1

7	6	5	4	3	2	1	0
CLR	Reserved (always 0)						ACK

Bit(s)	Value	Meaning
0	0 or 1	Fault acknowledged command. When this bit changes from 0 to 1, the ENIU updates the ENIU status data to contain the next fault. If there is no next fault, the ENIU clears the ENIU status data.
1-6	always 0	Reserved (always 0)
7	0 or 1	Clear all faults command. When this bit changes from 0 to 1, the ENIU clears its internal fault table. It also turns off its FAULT LED unless a new fault is immediately logged or an existing fault condition continues to exist.

Control Byte 2

7	6	5	4	3	2	1	0
Reserved (always 0)							

Control Byte 3

7	6	5	4	3	2	1	0
Reserved (always 0)							

Control Byte 4

7	6	5	4	3	2	1	0
Reserved (always 0)							

Consumed Exchange Control Byte Examples

Value of Control Byte 1	Description
0	No faults acknowledged (NACK). The ENIU takes no resulting action.
Changes from 0 to 1	Indicates that the last fault sent by the ENIU is acknowledged (ACK) by the master. As a result, the ENIU sends the next fault in its fault table. If no additional faults exist, the ENIU sends the "NO FAULT" indication. Note that the ENIU does not clear any faults due to an acknowledgement; use the Clear Fault Table (128) command to clear the fault table.
128	Commands the ENIU to clear its fault table.
Changes from 1 to 0	No fault is acknowledged by the master. If the ENIU sent a fault in its previous exchange, it will send the same fault again. If the ENIU has sent all of its faults and they have all been acknowledged, it will send a NO FAULT status in its next exchange.

3.7.2 Status Bytes for ENIU's Produced Exchange

The first four bytes of the ENIU's produced exchange are used to report fault status information from the ENIU to the master device. The first three bytes refer to the location of the fault. The fourth byte contains the status code. A table of fault status codes can be found in Chapter 6. If a fault exists, bit 7 of Status Byte 0 will be set to 1 (value = 128). If no fault exists, bit 7 of Status Byte 0 will be set to 0. If the ENIU is consuming exchanges, bit 6 of Status Byte 0 will be set to 0; if not consuming, bit 6 will be set to 1.

Status Byte 0

7	6	5	4	3	2	1	0
FLT	NC	Reserved (all bits=0)			Rack Number (0-7)		

Bit(s)	Value	Meaning
0-2	0-7	The physical "rack" location of the faulted I/O module. The value 0 refers to the ENIU main rack.
3-5	0	Reserved
6	0 or 1	0 = Normal operation – consuming. 1 = Not consuming (NC)
7	0 or 1	0 = no fault data present. The remaining fields in bytes 0-3 may be ignored. 1 = a fault is present. The remaining fields in bytes 0-3 provide the rack, slot, point, and fault code of the fault.

Status Byte 1

7	6	5	4	3	2	1	0
Reserved (all bits=0)				Slot Number (0-8)			

Bit(s)	Value	Meaning
0-3	0-8	The "slot" location of the faulted I/O module. The value 0 refers to the ENIU itself.
4-7	0	Reserved

Status Byte 2

7	6	5	4	3	2	1	0
Point Number (0-63)							

Bit(s)	Value	Meaning
0-7	0, 1-63	If no faults exist (Bytes 1 and 2 = 0), this byte will default to 0. If faults exist (Bytes 1 and 2 indicate faulted rack and slot location) and this byte = 0, it indicates a module-level fault on the module indicated by Bytes 1 and 2. If this byte equals a value between 1 and 63 inclusive, it indicates the faulted point number on the module indicated by Bytes 1 and 2.

Status Byte 3

7	6	5	4	3	2	1	0
Fault code (0-63)							

Bit(s)	Value	Meaning
0-7	0-63	See Chapter 6 for a list of the fault codes.

Produced Exchange Control Byte Examples

The following examples show the first four bytes (Status bytes) of the ENIU produced exchange and the interpretation of the values.

Binary	Decimal	Meaning
1000000 00000100 00000001 00000101	128,4,1,5	Module in Rack 0, Slot 4 has a blown fuse fault (fault code 5) on point 1
10000001 00000110 00000010 00001011	129,6,2,11	Module in Rack 1, Slot 6 has a high alarm fault (fault code 11) on point 2
01000000 00000000 00000000 00000000	64,0,0,0	ENIU is not consuming
00000000 00000000 00000000 00000000	0,0,0,0	There are no faults

Fault Handling Example (User Programming Required)

In the example below, the ENIU detects three faults after it sends its second produced exchange.

Master Exchanges 1, 2, and 3	Master's Control Byte is 0 since it has not received any fault data from the ENIU.
ENIU Exchanges 1 and 2	ENIU detects no faults, so sends a NO FAULT status.
ENIU Exchange 3	ENIU has detected three faults: it sends the data for Fault #1.
Master Exchange 4	The Master changes its Control Byte status from 0 to 1, acknowledging receipt of the ENIU's Fault #1 data.
ENIU Exchange 4	In response, ENIU sends Fault #2 data in its Status bytes.
Master Exchange 5	In response, the Master resets its Control Byte to logic 0.
ENIU Exchange 5	The Master didn't acknowledge, so ENIU sends Fault #2 data again.
Master Exchange 6	The Master acknowledges by setting its Control Byte to logic 1.
ENIU Exchange 6	In response to this acknowledgement, the ENIU sends Fault #3 data.
Master Exchange 7	In response, the Master resets its Control Byte to logic 0.
ENIU Exchange 7	The Master didn't acknowledge, so ENIU sends Fault #3 data again.
Master Exchange 8	The Master acknowledges by setting its Control Byte to logic 1.
ENIU Exchange 8	ENIU recognizes acknowledgement, but has no more faults, so sends the NO FAULT status, and continues to do so until another fault is detected.
Master Exchange 9	Master sends the Clear Fault Table command (128), which causes the ENIU to clear its fault table and turn off its FAULT LED.

Master			ENIU		
Produced Exchange No	Control Data		Status Data	Produced Exchange No	FAULT LED
1	NACK (0)	→			OFF
		←	NO FAULT	1	OFF
2	NACK (0)	→			OFF
		←	NO FAULT	2	OFF
3	NACK (0)	→			ON
		←	Fault #1 Data	3	ON
4	ACK (1)	→			ON
		←	Fault #2 Data	4	ON
5	NACK (0)	→			ON
		←	Fault #2 Data	5	ON
6	ACK (1)	→			ON
		←	Fault #3 Data	6	ON
7	NACK (0)	→			ON
		←	Fault #3 Data	7	ON
8	ACK (1)	→			ON
		←	NO FAULT	8	ON
9	CLEAR (128)	→			OFF

Chapter 4 Modbus

This chapter describes the implementation of Modbus/TCP communications on the Ethernet NIU.

- Modbus Protocol
 - Modbus Connections
 - Modbus Port
 - Modbus Conformance Classes
 - Modbus Message Format
- Modbus Tables
 - Register/Input Register Table
 - Input Discrete Table
 - Coil Table
- Supported Function Codes
 - Read Multiple Registers
 - Write Multiple Registers
 - Read Coils
 - Read Input Discretes
 - Read Input Registers
 - Write Coil
 - Write Single Register
 - Read Exception Status

4.1 Modbus Protocol

The Ethernet NIU supports a Modbus/TCP server with the following features:

4.1.1 Modbus Connections

The Ethernet NIU supports up to 2 simultaneous connections. This allows for the fast re-establishment of a connection. If a Modbus connection is lost, an immediate attempt at reconnecting by the client will be successful.

4.1.2 Modbus Port

Modbus communication on the Ethernet NIU is supported on the Modbus industry standard port 502.

4.1.3 Modbus Conformance Classes

The Ethernet NIU supports Modbus Conformance classes 0 and 1.

4.1.4 Modbus Message Format

The Modbus/TCP protocol has a specific message format as follows:

BYTE 0 – 1	Transaction Identifier – unique ID generated by the client.
BYTE 2 – 3	Protocol Identifier = 0
BYTE 4	Length Field (upper byte) = 0 (all msgs < 256)
BYTE 5	Length Field (lower byte) = number of bytes following
BYTE 6	Unit Identifier
BYTE 7	Modbus Function Code
BYTE 8 on	Data as needed

Note: The 'CRC-16' or 'LRC' check fields normally associated with Modbus are not needed in Modbus/TCP since the TCP/IP and link layer checksum mechanisms are used to verify packet delivery.

4.1.5 Modbus Byte Order

MODBUS uses a 'big-endian' representation for addresses and data items. This means that when a numerical quantity larger than a single byte is transmitted (as in a word or double-word), the MOST significant byte is sent first. So, for example:

The quantity 0x1234 is transmitted in the order 0x12 0x34

The quantity 0x12345678 is transmitted in the order 0x12 0x34 0x56 0x78

4.1.6 Modbus Bit Order

If a series of bits is read as a register, such as %I1 to %I16, the highest numbered bit (%I16 in this example) is the least significant, and the lowest numbered bit (%I1 in this example) is the most significant.

4.2 Modbus Tables

The Modbus protocol's reference table definition is different from the internal structure of the Ethernet NIU tables. Modbus refers to a Register, Input Register, Input Discrete and Coil table; the Ethernet NIU refers to a Discrete Input (%I), Discrete Output (%Q), Analog Input (%AI), Analog Output (%AQ) and Fault Table. The following table shows how each Modbus table has been mapped to the ENIU tables. Note that everything in this table refers to physical memory inside the ENIU. In effect, ENIU memory has been given Modbus names. For example, if we issue a Read Inputs Discrete command to read inputs in the Modbus Input Discrete table, we are actually reading from the ENIU's %I internal table, which is mapped to the Modbus Input Discrete table.

Modbus Register Table	Modbus Input Register Table	Modbus Input Discrete Table	Modbus Coil Table	ENIU Internal Tables
0 – 127 (16-bit words)	0 – 127 (16-bit words)	0 – 2047 (bits)	---	%I1 – 2048 (bits)
128 – 255 (16-bit words)	128 – 255 (16-bit words)	---	---	%AI1 – 128 (16-bit words)
256 – 383 (16-bit words)	256 – 383 (16-bit words)	---	0 – 2047 (bits)	%Q1 – 2048 (bits)
384 – 511 (16-bit words)	384 – 511 (16-bit words)	---	---	%AQ1 – 128 (16-bit words)
1024 – 1087 (16-bit words)	1024 – 1087 (16-bit words)	---	---	Fault Table (32 Faults x two 16-bit words per fault)

4.2.1 Modbus Register/Input Register Table

The Ethernet NIU makes no distinction between the Modbus Register and Modbus Input Register tables. The Modbus Register and Modbus Input Register tables are identically mapped to all four ENIU I/O tables as well as the Fault table.

Applicable Functions

Read Multiple Registers	All ENIU tables can be read with this function.
Write Multiple Registers	Only the %Q and %AQ memory types may be written to with this function.
Read Input Registers	All ENIU tables can be read with this function.
Write Single Register	Only the %Q, %AQ, and Fault tables may be written to with this function.

4.2.2 Modbus Input Discrete Table

The Modbus Input Discrete table is mapped exclusively to the ENIU Discrete Input (%I) table.

Applicable Functions

Read Input Discretes

4.2.3 Modbus Coil Table

The Modbus Coil table is mapped exclusively to the ENIU Discrete Output (%Q) table.

Applicable Functions

Read Coils
Write Coil

4.3 Supported Function Codes

The Function Codes are defined for the Modbus memory mapping, so to determine what area of ENIU memory is affected, refer to Section 4.2, which maps the Modbus table names to their corresponding ENIU table names.

The Ethernet NIU supports the following Modbus function codes:

- Read Multiple Registers (Function Code 3)
- Write Multiple Registers (Function Code 16)
- Read Coils (Function Code 1)
- Read Input Discretes (Function Code 2)
- Read Input Registers (Function Code 4)
- Write Coil (Function Code 5)
- Write Single Register (Function Code 6)
- Read Exception Status (Function Code 7)

Note: The following function request and response message descriptions start with the Modbus Function Code (byte 0 is actually byte 7 of the Modbus message format). See Section 4.1.4, *Modbus Message Format*.

4.3.1 Read Multiple Registers

This command reads from 1 to 125 16-bit words from the Modbus Register table. Any part of the Modbus Register table can be read from using this function. When reading from the fault table, however, the entire fault table must be read.

The Read Multiple Registers request is in the following form:

BYTE 0	Function Code = 3
BYTE 1-2	Register Table Offset
BYTE 3-4	Word Count (1-125)

The Read Multiple Registers response is in the following form:

BYTE 0	Function Code = 3
BYTE 1	Byte Count of response ($B = 2 \times \text{word count of request}$)
BYTE 2 - (B + 1)	Register Values

If the request accesses an invalid offset or it contains an invalid length, the response is an Exception Response in the following form:

BYTE 0	Function Code = 0x83
BYTE 1	Exception Code = 2

Read Multiple Registers Examples:

Register Table Offset = 0 and word count = 2 returns %I1-32.

Register Table Offset = 383 and word count = 2 returns %Q2033-2048 and %AQ1.

Register Table Offset = 1024 and word count = 64 returns the fault table.

Any combination of Register Table Offset and Word Count that accesses an offset > 511 and < 1024 produces an Exception Response.

Trying to read the fault table and giving a Register Table Offset > 1024 or a Word Count <> 64 produces an Exception Response.

4.3.2 Write Multiple Registers

This command writes from 1 to 100 16-bit words to the Modbus Register table. Only the part of the Modbus Register table mapped to the %Q and %AQ I/O tables may be written to using this function.

The Write Multiple Registers request is in the following form:

BYTE 0	Function Code = 0x10
BYTE 1-2	Register Table Offset
BYTE 3-4	Word Count(1-100)
BYTE 5	Byte Count (B = 2 x word count)
BYTE 6 – (B + 5)	Register Values

The Write Multiple Registers response is in the following form:

BYTE 0	Function Code = 0x10
BYTE 1-2	Register Table Offset (same as request)
BYTE 3-4	Word Count (same as request)

If the request accesses an invalid offset or it contains an invalid length, the response is an Exception Response in the following form:

BYTE 0	Function Code = 0x90
BYTE 1	Exception Code = 2

Write Multiple Registers Examples:

Register Table Offset = 256 and Word Count = 2 writes the Register Values into %Q1-32.

Register Table Offset = 383 and Word Count = 2 writes the Register Values into %Q2033-2048 and %AQ1.

Any combination of Register Table Offset and Word Count that accesses an offset < 256 or > 511 produces an Exception Response.

4.3.3 Read Coils

This command reads from 1 to 2000 bits from the Modbus Coil table.

The Read Coils request is in the following form:

BYTE 0	Function Code = 1
BYTE 1-2	Coil Table Offset
BYTE 3-4	Bit Count (1-2000)

The Read Coils response is in the following form:

BYTE 0	Function Code = 1
BYTE 1	Byte count of response; $B = (\text{bit count of request} + 7)/8$
BYTE 2 – (B + 1)	Bit Values (least significant bit is first coil)

If the request accesses an invalid offset or it contains an invalid length, the response is an Exception Response in the following form:

BYTE 0	Function Code = 0x81
BYTE 1	Exception Code = 2

Read Coils Examples:

Coil Table Offset = 0 and Bit Count = 1 returns coil %Q1.

Coil Table Offset = 0 and Bit Count = 2000 returns coil values %Q1-2000.

Coil Table Offset = 4 and Bit Count = 13 returns coil values %Q5-17

Any combination of Coil Table Offset and Bit Count that accesses an offset > 2047 produces an Exception Response.

4.3.4 Read Input Discretes

This command reads from 1 to 2000 bits from the Modbus Input Discrete table.

The Read Input Discretes request is in the following form:

BYTE 0	Function Code = 2
BYTE 1-2	Input Discrete Table Offset
BYTE 3-4	Bit Count (1-2000)

The Read Coils response is in the following form:

BYTE 0	Function Code = 2
BYTE 1	Byte count of response ($B = (\text{bit count of request} + 7)/8$)
BYTE 2 – (B + 1)	Bit Values (least significant bit is first coil)

If the request accesses an invalid offset or it contains an invalid length, the response is an Exception Response in the following form:

BYTE 0	Function Code = 0x82
BYTE 1	Exception Code = 2

Read Input Discretes Examples:

Input Discrete Table Offset = 0 and Bit Count = 1 returns input discrete %I1.

Input Discrete Table Offset = 0 and Bit Count = 2000 returns input discrete values %I1-2000.

Input Discrete Table Offset = 4 and Bit Count = 13 returns input discrete values %Q5-17

Any combination of Input Discrete Table Offset and Bit Count that accesses an offset > 2047 produces an Exception Response.

4.3.5 Read Input Registers

This command reads from 1 to 125 16-bit words from the Modbus Register table. This command is handled exactly as the Read Multiple Registers command.

The Read Input Registers request is in the following form:

BYTE 0	Function Code = 4
BYTE 1-2	Register Table Offset
BYTE 3-4	Word Count (1-125)

The Read Coils response is in the following form:

BYTE 0	Function Code = 4
BYTE 1	Byte Count of response ($B = 2 \times \text{word count of request}$)
BYTE 2 – (B + 1)	Register Values

If the request accesses an invalid offset or it contains an invalid length, the response is an Exception Response in the following form:

BYTE 0	Function Code = 0x84
BYTE 1	Exception Code = 2

Read Input Registers Examples:

See Read Multiple Registers section for examples.

4.3.6 Write Coil

This command writes 1 bit to the Modbus Coil table.

The Write Coil request is in the following form:

BYTE 0	Function Code = 5
BYTE 1-2	Coil Table Offset
BYTE 3	= 0xFF to turn coil ON, = 0 to turn coil OFF
BYTE 4	= 0

The Write Coil response is in the following form:

BYTE 0	Function Code = 5
BYTE 1-2	Coil Table Offset (same as request)
BYTE 3	= 0xFF to turn coil ON, = 0 to turn coil OFF (same as request)
BYTE 4	= 0

If the request accesses an invalid offset, the response is an Exception Response in the following form:

BYTE 0	Function Code = 0x85
BYTE 1	Exception Code = 2

Write Coil Examples:

Coil Table Offset = 0 and value = 0xFF turns coil %Q1 ON.

Coil Table Offset = 0 and value = 0 turns coil %Q1 OFF.

Any Coil Table Offset > 2047 produces an Exception Response.

4.3.7 Write Single Register

This command writes one 16-bit word to the Modbus Register table. Only the part of the Modbus Register table mapped to the %Q and %AQ I/O tables or the first word of the Fault table may be written to using this function.

The Write Single Register request is in the following form:

BYTE 0	Function Code = 6
BYTE 1-2	Register Table Offset
BYTE 3-4	Register Value

The Write Single Register response is in the following form:

BYTE 0	Function Code = 6
BYTE 1-2	Register Table Offset (same as request)
BYTE 3-4	Register Value (same as request)

If the request accesses an invalid offset, the response is an Exception Response in the following form:

BYTE 0	Function Code = 0x86
BYTE 1	Exception Code = 2

Write Single Register Examples:

Register Table Offset = 256 writes the register value into %Q1-32.

Register Table Offset = 384 writes the register value into %AQ1.

Register Table Offset = 1024 and register value = 0 clears the Fault table.

Any Register Table Offset < 256 or (> 511 and < 1024) or > 1024 produces an Exception Response.

4.3.8 Read Exception Status

This command reads one 8-bit status of the Ethernet NIU.

The Read Exception Status request is in the following form:

BYTE 0	Function Code = 7
--------	-------------------

The Read Exception Status response is in the following form:

BYTE 0	Function Code = 7
BYTE 1	Exception Status

Exception Status Data Format

7	6	5	4	3	2	1	0
Unused							Fault Present

Chapter 5 Ethernet Global Data

This chapter describes the implementation of Ethernet Global Data (EGD) communications on the Ethernet NIU.

- EGD Protocol
- EGD Exchange Definition

5.1 EGD Overview

Ethernet Global Data is a GE protocol that provides efficient connectionless periodic data transfer over an Ethernet network. It operates over the industry standard User Datagram Protocol (UDP). The UDP protocol works at the ISO Transport layer. It supports fast, efficient communications because it is connectionless and is not acknowledged.



Caution

Since Ethernet Global Data (EGD) communications is connectionless and is not acknowledged, error-checking and interlocking circuitry must be designed into the application to ensure the safety of personnel and equipment in the event that EGD data is lost. Failure to heed this warning could result in injury to personnel and damage to equipment.

In EGD communications, a device (called a **producer**) shares a portion of its memory contents periodically with one or more other devices (called **consumers**). This sharing of memory between devices is called an **exchange**.

5.1.1 Exchange Parameters

When an exchange is configured, several key pieces of information must be included:

- Exchange ID. The unique identification of the exchange configuration
- Producer ID. The unique identification of the producing (sending) device
- The area of memory in the producing device to be exchanged
- How frequently the exchange will be produced
- Consumer ID. The unique identification of the consuming (receiving) device or devices
- The area of memory where the received data is to be stored for each consuming device

Note that a device can be configured to be both a producer and a consumer of exchanges.

5.2 EGD Protocol

The Ethernet NIU supports Ethernet Global Data communication with the following features:

5.2.1 Number of EGD Exchanges

The Ethernet NIU supports one EGD consumed data exchange and one EGD produced data exchange. The consumed data exchange is comprised of ENIU control data and the discrete and analog module output data being sent to the ENIU. The produced data exchanged is comprised of ENIU status data and the discrete and analog module input data being sent from the ENIU.

5.2.2 EGD Port

EGD communication on the Ethernet NIU is supported on the EGD defined UDP Data Port 18246 (4746H).

5.2.3 EGD Destination Addresses

The Ethernet NIU allows EGD exchanges to be sent to or received from a single destination address (IP Unicast addressing), a group of addresses (IP Multicast addressing), or all EGD nodes (IP Broadcast addressing).

Note: Unicast addressing is recommended unless there are special requirements.

The following table shows the defined Multicast addresses for transmission of data exchanges to a group of nodes:

Parameter	IP Address
Group 1	224.0.7.1
Group 2	224.0.7.2
:	:
:	:
Group 32	224.0.7.32

5.3 EGD Exchange Definition

As stated earlier, the Ethernet NIU supports two EGD exchanges: one consumed, and one produced. The ENIU consumed exchange will be sent from a master to the ENIU, and the ENIU produced exchange will be sent from the ENIU to a master.

5.3.1 Consumed Exchange

The ENIU consumes one exchange containing all of the output data for discrete output (Q) and analog output (AQ) areas configured in the ENIU's network I/O map. The data is received in the same sequence as that which the modules occupy in the I/O Station itself. If a single module receives both discrete and analog output data, its discrete data is located before its analog data. The maximum overall length of this output data is 512 bytes. An additional 4 bytes at the start of the data are used by the master for control operations. The total maximum length of the data portion of the exchange is 516 bytes.

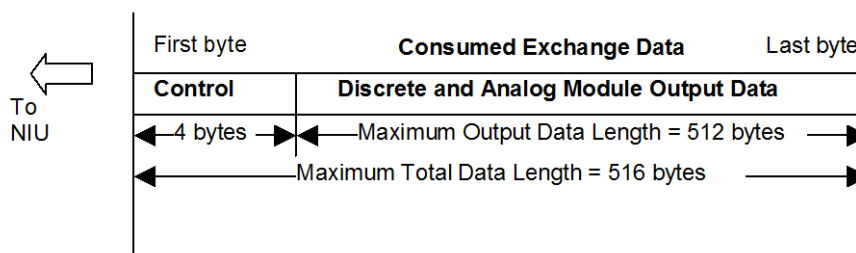


Figure 65: Format of Consumed Data Exchange

The following is an example of how the byte ordering is arranged in an EGD consumed exchange. Please note that the order is based on module order not reference (memory address) order.

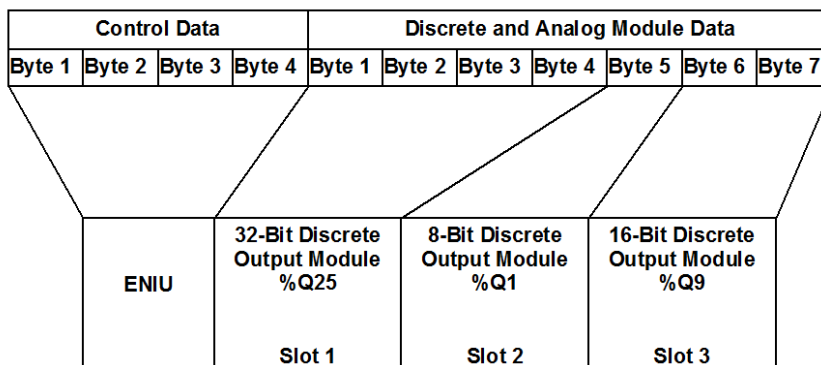


Figure 66: Mapping of Consumed Data to Slot Locations of Output Modules

Note: If a discrete output module does not map to all bits in the assigned byte(s), unused bits will be padded with zeros. For instance, a 5-point output module will always have three zeroes in its most significant bit locations.

5.3.2 Produced Exchange

The ENIU produces one exchange containing all of the input data for the configured discrete input (I) and analog input (AI) areas configured in the ENIU's network I/O map. The data is sent in the same sequence as that which the modules occupy in the I/O Station itself. If a single module provides both discrete and analog input data, its discrete data is placed before its analog data. The maximum overall length of this input data is 512 bytes. An additional 4 bytes at the start of the data are used by the master for control operations. The total maximum length of the data portion of the exchange is 516 bytes.

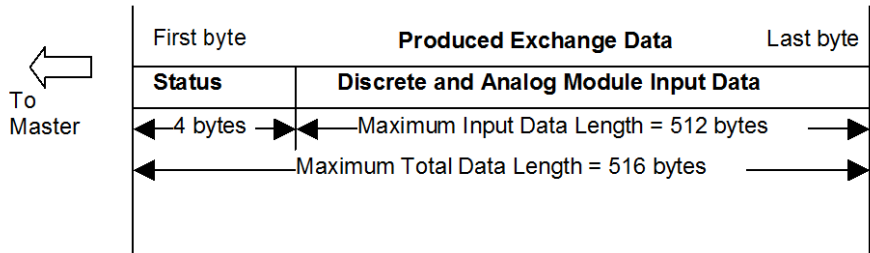


Figure 67: Format of Produced Data Exchange

The following is an example of how the byte ordering is arranged in an EGD produced exchange. Please note that the order is based on module order and not on reference (memory address) type or order. As implied in the following figure, each analog channel required two bytes of memory.

Status Data				Discrete and Analog Module Exchange Data										
Byte 1	Byte 2	Byte 3	Byte 4	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Byte 9	Byte 10	Byte 11
ENIU				4-Channel Analog Input Module %AI5				16-Bit Discrete Input Module %I9				8-Bit Discrete Input Module %I1		
				Slot 1				Slot 2				Slot 3		

Figure 68: Mapping of Produced Data to Slot Locations of Input Modules

Note: If a discrete output module does not map to all bits in the assigned byte(s), unused bits will be padded with zeros. For instance, a 5-point output module will always have three zeroes in its most significant bit locations.

Note: If an empty slot is left in the I/O (carrier present, but no I/O module), it will not affect the EGD exchanges. If a module is added later in the empty slot and configured, the ENIU exchange(s) will change to add the new module in the appropriate place in the data. The controller EGD exchange(s) will need to be changed to match.

Chapter 6 Troubleshooting

This chapter discusses the following topics:

- **Checking Status and Operation with the ENIU's LEDs:** The ENIU has five active LED indicators on its front panel that indicate both normal and fault conditions. This section discusses how to interpret these LED indicators. (The new hardware has three inactive LEDs (RES1, RES2 & RES3) that have no impact on the following discussion).
- **ENIU Fault Table:** The ENIU stores up to 32 fault codes in its Fault Table. This section describes how to read and interpret the Fault Table codes.
- **Using FTP to Obtain Text Files.**
- **EGD Troubleshooting:** Checking common EGD communication problems.
- **Determining the MAC Address of the ENIU:** The ENIU's MAC address is printed on the front cover of the ENIU. However, if it becomes illegible or you suspect it is incorrect, you can determine it using one of the methods described.
- **Reading the Stored IP Address of the ENIU:** If the ENIU's stored IP address is not known, you can determine it using the method described.

6.1 Checking Status and Operation with the ENIU's LEDs

When power is applied to the ENIU, you can verify status and operation by checking the module LEDs.

6.1.1 LED Power-up Sequence

Normally, the PWR LED will turn on immediately once power is applied. If it fails to turn on, refer to Section 6.1.3.

During a self-test, which takes about 5 to 20 seconds, the OK LED blinks. After completion of self-test, the PWR LED should be ON steady.

If the ENIU has an active network connection, the LAN LED should begin flashing (or could be ON steady if communications traffic is heavy) at that time also.

6.1.2 LED Descriptions



Figure 69: Faceplate of Module (New Hardware Version)

LED	Color	State	Indication
PWR	Green	Solid	Power applied
	---	Off	No power applied / Defective Module
RES 1	Unused		Present in new hardware only (inactive).
OK	Green	Solid	OK
	---	Off	Not OK
	Amber	Blink Pattern ¹	Fatal Error (Blink Trap)
	Green	Blinking ¹	Powering up
FAULTS	Amber	Solid	Detected a fault within itself or backplane module
	Amber	Blinking	Encountered a fatal error
	Red	Blinking	Invalid MAC address. (Indicates error if any MAC address is missing or bad)
	---	Off	No faults detected
LAN	Green	Solid	Link connected 100 Mbps
	Green	Blinking	Activity (this port) 100 Mbps
	Amber	Solid	Link connected 10 Mbps
	Amber	Blinking	Activity (this port) 10 Mbps
	---	OFF	No link
STAT	Green	Solid	Master Connected (Modbus TCP) / Receiving exchanges (EGD)
	---	Off	No Master Connected (Modbus TCP) / Not Receiving exchanges (EGD)
	Amber	Blinking	At 1/sec rate indicates an IP address problem / Duplicate IP has been detected on the network.
RES 2	Unused		Present in new hardware only (inactive).
RES 3	Unused		Present in new hardware only (inactive).
LEDs behind door			
MAINT	Green	Solid	Serial Console Active
	---	Off	Port Disabled
COMM	Green	Blinking	USB (serial) port activity
	---	Off	No Activity on port

Blink Pattern on fatal error / exception

The ENIU OK LED will flash a repeating fatal error code and other LEDs (FAULTS, LAN & STAT) indicate start and stop of the blink pattern.

For example: (error code 2-5) two blinks, pause, five blinks, all LEDs blink, repeat. All LEDs will blink in AMBER.

Blink Pattern during Firmware Upgrade

The ENIU OK, FAULTS, LAN & STAT LED blink in unison with GREEN color at the rate of 1Hz. This indicates the ENIU has entered the firmware upgrade mode.

6.1.3 Troubleshooting Using LED Indicator Lights

PWR LED

ON

The ENIU derives power from the power supply that is connected via its faceplate power connector (Figure 8 or Figure 9). If power is applied to the power supply and the connector is properly inserted, the PWR LED on the ENIU should be ON.

Note: Compatible power supplies are documented in Section 1.6.1.

OFF

If this LED is OFF, check the following

- Check that the input power to the power supply is present.
- If input power is present, try substituting a known good power supply.
- If the power supply checks OK, the problem may be a defective ENIU, or a problem with one of the modules connected to the ENIU. Turn off input power to the power supply, and then separate the ENIU from all of the connected modules. Now try powering up the ENIU again. If the PWR LED is still OFF, the problem is in the ENIU. If the PWR LED is ON, the problem is in one of the other modules. Reconnect them one-by-one to determine which one is “killing” the power supply.

OK LED

ON

Normally, upon power-up, the ENIU will conduct a series of self-tests. If these tests are passed, the ENIU turns on its OK LED. This LED should normally come on within a few seconds after power is applied.

OFF

If this LED is OFF, it means the ENIU did not pass its power-up self-tests and has an internal problem.

FLASHING

During the power-up sequence, while the ENIU is getting ready, this LED will be blinking. Once the ENIU is ready for operation, OK LED remains ON.

FAULTS LED

OFF

This LED should be OFF for normal operation.

ON Steady or Flashing

If this LED is not OFF, check the following

- If **ON steady**, the ENIU has detected a non-fatal fault. The fault may be related to the ENIU or to a connected module. Check the ENIU fault table to determine which fault or faults are present. See the section "ENIU Fault Table" for details.
- If **FLASHING** with amber (yellow) color, a fatal fault is present. The ENIU will be disabled by the presence of this fatal fault. The flashing will be a repeating 2,5 pattern (flashes twice, pauses, flashes five times, pauses, repeats). This can indicate either a software error (very unlikely) or, more commonly, a configuration error. A configuration error occurs if a configuration being stored from the programmer is rejected as unusable (see next topic "Causes for Configuration Store Fault"), or if a network disconnect occurs during a configuration store. There are a number of rules that are checked on a configuration. It is suggested that if a user is having problems with a store being rejected, autoconfigure the ENIU into a temporary folder and alter one setting at a time until the problem is isolated.

Note: When a store is rejected or a disconnect occurs during a store, a power cycle is required to restart the ENIU and it will autoconfigure after the power cycle.

Causes for Configuration Store Fault

The following rules are likely causes of a configuration being rejected:

General Configuration Problems

Subnet mask of 0.0.0.0 or 255.255.255.255

Gateway IP address of 0 in subnet or broadcast address (unless 0.0.0.0)

Gateway IP not in subnet (unless 0.0.0.0)

EGD Configuration Problems

If (1) the produced exchange type is changed from the default IP Address, or (2) the produced IP is changed from the default of 0.0.0.0, or (3) the consumed IP is changed from the default of 0.0.0.0, then it is assumed the device is to run EGD instead of Modbus TCP, and the following rules are checked during configuration store:

If produced exchange type is IP, the produced IP address must not be 0.0.0.0

If produced exchange type is Group, group ID must be between 1 and 32 inclusive

Consumed exchange cannot be 0.0.0.0 or 255.255.255.255.

Consumed group ID cannot be above 32

Note: If EGD is configured, both consumption and production must be configured, even if one side is not used. It is suggested that the time periods be adjusted for the unused side to minimize network traffic. However, even if one side is "not used," its exchange still contains four bytes of Fault or Control Status data.

Example: If the EGD ENIU system is used only for outputs (Consumed exchanges), the Produced exchange must be configured for the stored configuration to be accepted, but the production period can be set at the default of 200 ms.

LAN LED

Flashing or ON Steady

This LED will be flashing or appear to be ON steady to indicate the ENIU is communicating over the network.

OFF

If this LED is OFF, it indicates that the ENIU is not communicating over the network. Check the following

- Determine if there are any other operational devices connected to the network. Check the STAT LED (see “STAT LED” below).
- Ensure that the Ethernet cable is connected firmly in the ENIU’s RJ-45 connector. Also, this cable may be defective; try substituting another cable. Also, check the other end of the cable to ensure that the device it connects to (hub, switch, etc.) is connected and powered up.
- There may be a problem with some other network device or in the network configuration (ensure that the ENIU’s IP address and MAC address were configured correctly in the applicable network devices).
- The ENIU’s IP address may be invalid. Check to see if the STAT LED is flashing (see the “STAT LED” section below).
- Even if the ENIU has a valid IP address, it may not be compatible with the network. Ensure that the ENIU’s netid matches the netid of the network/subnetwork it is installed on. Netid information can be found in Appendix B.

STAT LED (Modbus Mode)

ON

This LED will be ON with a green color to indicate the presence of at least one Modbus TCP Master on the network.

OFF

If OFF, it indicates that no Modbus TCP Master device has been detected on the network.

Flashing

If this LED is flashing an amber color at a one per second rate, it indicates an invalid ENIU IP address. Check the following

- You may have set the ENIU to an IP address that is the same as the base address of a subnet (such as 3.16.32.0).
- You may have set the ENIU to an IP address that is the same as the broadcast address of a subnet (such as 3.16.32.255)
- ENIU may have a duplicate IP address in the network.
- ENIU may have Invalid MAC or Invalid IP address or Subnet Mask or Gateway.
- If this blink persists, you may have to remove this from network and set the IP address of desire using Serial IP console or programmer.

STAT LED (EGD Mode)

ON

This LED will be ON with a green color to indicate that the ENIU is consuming the EGD exchange and controlling its outputs.

OFF

If OFF, this LED indicates that the ENIU is not consuming EGD exchanges and that its outputs have reverted to their configured states of Hold Last State or Default.

Flashing

If this LED is flashing an amber color at a one per second rate, it indicates an invalid ENIU IP address. Check the following

- You may have set the ENIU to an IP address that is the same as the base address of a subnet (such as 3.16.32.0).
- You may have set the ENIU to an IP address that is the same as the broadcast address of a subnet (such as 3.16.32.255)
- ENIU may have a duplicate IP address in the network.
- ENIU may have Invalid MAC or Invalid IP address or Subnet Mask or Gateway.
- If this this blink persists, you may have to remove this from network and set the IP address of desire using Serial IP console or programmer.

6.2 ENIU Fault Table

Fault Table Data Format

The ENIU's internal fault table, which can store up to 32 fault codes, can be accessed by a client application using either Modbus or EGD. This internal fault table operates as a First-In-First-Out (FIFO) stack. When fault 33 occurs, fault 1 is dropped from the table. These faults can include both faults provided by the I/O modules and diagnostic information provided by the ENIU itself. Faults may be read as an extension to the normal register table. A client application may request all faults or clear all faults by sending the applicable message to the ENIU.

Each fault is represented by 2 words, starting from reference 1024 in the register table. Whenever a fault is present in the fault table, the FAULTS LED will be on (solid).

All fault table faults are informational only and do not halt operation of the ENIU. The presence of one or more fault table faults will cause the ENIU's Fault LED to turn ON steady. If a fatal fault occurs, the Fault LED will blink ON and OFF.

6.2.1 Modbus Mode

Reading the ENIU Fault Table

The entire fault table can be read using a *Read Multiple Registers* command, starting at the beginning of the fault table (1024), with a length 64 registers. No partial read access of the fault table is permitted. Empty fault table entries are all zeros. Note that faults are shifted so that the most recent fault is always located at offset 1024.

Clearing the ENIU Fault Table

When desired, the client application may clear all faults in the table by using a *Write Single Register* command to write a zero to the first register (1024) of the fault table. No other write to the fault table by a client is permitted.

Fault Table Entries

Each fault is two words in length and is formatted as follows:

Fault Entry (2 Words)		
Word No.	Upper byte	Lower byte
Word 0	Rack Number (0-7)	Slot Number (0-15)
Word 1	Module I/O Point Number (0-63)	Fault Code (see table below)

6.2.2 EGD Mode

The top fault in the FIFO fault stack is sent in the status message. The top fault or all faults can be cleared by sending an ACK or clear in the control section of a consumed exchange. Refer to Section 3.7, *EGD Exchange Status and Control Bytes for Fault Handling* for details and examples on configuring and handling fault codes in EGD mode.

Fault Table Codes for Modbus and EGD Modes

Fault Table Codes	
Fault Description	Fault Number
UNKNOWN_FAULT	0
CORRUPTED_CONFIGURATION_FAULT	1
UNSUPPORTED_FEATURE_FAULT	2
CONFIG_MISMATCH_FAULT	4
FUSE_BLOWN_FAULT	5
LOSS_OF_IO_MODULE_FAULT	6
ADDITION_OF_IO_MODULE_FAULT	7
EXTRA_IO_MODULE_FAULT	8
LOSS_OF_USER_POWER_FAULT	9
OPEN_WIRE_FAULT	10
HIGH_ALARM_FAULT	11
LOW_ALARM_FAULT	12
OVERRANGE_FAULT	13
UNDERRANGE_FAULT	14
SHORT_CIRCUIT_FAULT	15
NONVOLATILE_STORE_FAULT	16
LOSS_OF_NON_IO_MODULE_FAULT	17
ADDITION_OF_NON_IO_MODULE_FAULT	18
INSUFFICIENT_CONFIG_MEMORY_FAULT	19
MODULE_NOT_CONFIGURED_FAULT	20
INPUT_POINT_FAULT	21
WIRING_FAULT	22
THERMISTOR_FAULT	23
A_TO_D_CONVERTOR_FAULT	24
MAIL_QUEUE_FULL	25
MAIL_LOSS_FAULT	26
MODULE_IN_BOOT_MODE	27
LOSS_OF_RACK_FAULT	28
ADDITION_OF_RACK_FAULT	29
RACK_NOT_CONFIGURED_FAULT	30
LOSS_OF_TAN_XMIT_FAULT	31
ADDITION_OF_TAN_XMIT_FAULT	32
EXTRA_TAN_XMIT_FAULT	33
TAN_SPEED_CHANGE_FAULT	34
LOSS_OF_MOD_UNSUPP_FEATURE	35

6.3 Using FTP to Obtain Text Files

6.3.1 FTP for Version Information

To obtain version information such as version of firmware, Boot firmware and FPGA and EGD production & consumption length etc.

- 1) Start at an MS-DOS prompt.
- 2) If using a networked computer, change to a directory where you have write privileges.
- 3) FTP to the ENIU.
- 4) Log into the ENIU as **programmer** (all lower case)
- 5) Enter the word **password** (all lower case) as the password
- 6) Type **get version.txt**. This copies the text file **version.txt** into the current directory of your personal computer. This file contains version of firmware, Boot firmware and FPGA etc.
- 7) When the operation completes, enter the word **bye** to terminate the FTP session.

Version information of ENIU can also be obtained from the serial console. The version of firmware, Boot firmware and FPGA are available on the banner information of the serial console (shown in Figure 35). The serial console is supported by the new hardware implementation only.

6.3.2 FTP for Network Status

This feature is operational with the old hardware, but is not supported with the new hardware implementation.

To obtain network status information such as number of packets transmitted and received, number of collisions, etc.

- 1) Start at an MS-DOS prompt.
- 2) If using a networked computer, change to a directory where you have write privileges.
- 3) FTP to the ENIU.
- 4) Log into the ENIU as **programmer** (all lower case)
- 5) Enter the word **password** (all lower case) as the password
- 6) Type **get netstat.txt**. This copies the text file **netstat.txt** into the current directory of your personal computer. This file contains formatted Ethernet statistics.
- 7) When the operation completes, enter the word **bye** to terminate the FTP session.

6.4 EGD Troubleshooting

The following are common EGD communications problems and solutions.

Wrong network setup. Check exchange configuration on both the ENIU and master device. If using a GE Fanuc CPU, try pinging the ENIU from the Station Manager. Obtain network status from the ENIU (see previous section “Using FTP to Obtain Network Status and Version Information”) and see if UDP packets are being sent and received by the ENIU at appropriate rate for the exchange speed.

Wrong exchange size. EGD exchange length must match expected length. Obtain version information from ENIU (see previous section “Using FTP to Obtain Network Status and Version Information”) to see length of produced and consumed exchange ENIU is expecting. Make sure four bytes were added to the start of the exchanges for Status and Control data.

Data Going to the wrong Place. If the exchanges are transferring data, but the data seems to be going to the wrong places, verify the exchange definitions in configuration and check the modules in the ENIU starting with Rack 0, Slot 1 onward until the discrepancy is found.

6.5 Determining the MAC Address of the ENIU

A MAC address is a unique six-byte number (written as six 2-digit hex numbers separated by dashes, such as 08-00-19-01-24-B6). The MAC address uniquely identifies each Ethernet hardware device. These numbers are assigned by manufacturers and are often not configurable by users.

The MAC address of each ENIU is printed on the front of the module. However, if it becomes illegible or appears to be incorrect, use one of the following techniques (Section 6.5.1 or 6.5.2) to recover the MAC address.

6.5.1 Technique using Serial Console

This section applies to the new hardware implementation only.

Check for ENIU network information using the serial console. The MAC Address, IP Address, Subnet Mask and Gateway are available on the banner information of the serial console (Figure 35).

6.5.2 Technique using Flashing LED Code

This section applies to the old hardware implementation only.

To determine the MAC address of the ENIU before it is on the network, set the rotary switches (refer to Section 2.3.2) to 901, then power cycle the ENIU.

The OK, Fault, LAN, and Status LEDs will flash twelve patterns (two for each byte) spelling out the MAC address digit by digit. This follows binary encoding with the exception of the use of the green Status LED to represent zero. The following table contains the coding for these patterns.

Use the worksheet on the next page to record the twelve LED patterns, then return to the following table to decode them. The numbers in parentheses represent the binary value of the condition, and Logic 1 conditions are shown in bold type for improved readability.

LED Decoding Table				
Hex Value	OK LED	Fault LED	LAN LED	Status LED
0	Off (0)	Off (0)	Off (0)	Green (0)
1	Off (0)	Off (0)	Off (0)	Amber (1)
2	Off (0)	Off (0)	On (1)	Off (0)
3	Off (0)	Off (0)	On (1)	Amber (1)
4	Off (0)	On (1)	Off (0)	Off (0)
5	Off (0)	On (1)	Off (0)	Amber (1)
6	Off (0)	On (1)	On (1)	Off (0)
7	Off (0)	On (1)	On (1)	Amber (1)
8	On (1)	Off (0)	Off (0)	Off (0)
9	On (1)	Off (0)	Off (0)	Amber (1)
A	On (1)	Off (0)	On (1)	Off (0)
B	On (1)	Off (0)	On (1)	Amber (1)
C	On (1)	On (1)	Off (0)	Off (0)
D	On (1)	On (1)	Off (0)	Amber (1)
E	On (1)	On (1)	On (1)	Off (0)
F	On (1)	On (1)	On (1)	Amber (1)

Note: All GE ENIU MAC addresses start with 08-00-19.

Worksheet for Determining the MAC Address

If the rotary switches are set to 901 when the Ethernet ENIU powers up, it displays its MAC address in a series of 12 patterns on the OK, FAULT, LAN, and STATUS LED lights. Each pattern will be separated by a brief “no-number” condition, in which all four LEDs are off. As each of the 12 patterns is displayed, record them in the table below. Then, decode the binary patterns into hexadecimal numbers using either (1) the “Weight” column values in the table below, or (2) the “LED Decoding Table” from the previous page. An additional “Value” row is provided below the table for you to write in the decoded numbers. Use the following guidelines for recording the patterns:

- 1) For the OK, FAULT, and LAN LEDs, record one of these two states: 1 if On or leave blank if Off (or, you may wish to record zeros if Off, instead of blanks).
- 2) For the STATUS LED, record one of these three states: 1 if On with an amber (yellow) color, 0 if On with a green color, or leave blank if Off. The “On with a green color” state of this LED is used to distinguish the number zero from a no-number indication.

Note: Since GE MAC addresses always begin with 08-00-19, these first six digits have been filled in already.

LED ↓	1	2	3	4	5	6	7	8	9	10	11	12	Weight
OK		1				1							8
FAULT													4
LAN													2
STATUS	0		0	0	1	1							1

Value	0	8	0	0	1	9							N/A
-------	---	---	---	---	---	---	--	--	--	--	--	--	-----

Example

LED ↓	1	2	3	4	5	6	7	8	9	10	11	12	Weight
OK		1				1					1		8
FAULT										1		1	4
LAN									1		1	1	2
STATUS	0		0	0	1	1	0	1			1		1

Value	0	8	0	0	1	9	0	1	2	4	B	6	N/A
-------	---	---	---	---	---	---	---	---	---	---	---	---	-----

Thus, the MAC address in this example is 08-00-19-01-24-B6

6.6 Reading the Stored IP Address of the ENIU

6.6.1 Technique using Serial Console

This section applies to the new hardware implementation only.

Check for ENIU network information using the serial console. The MAC Address, IP Address, Subnet Mask and Gateway are available on the banner information of the serial console (Figure 35).

6.6.2 Technique using Flashing LED Code

This section applies to the old hardware implementation only.

If the stored IP address of an ENIU is lost or forgotten, you can cause it to be displayed on the front panel LEDs of the ENIU by setting the rotary switches (refer to Section 2.3.2) to 900, then power cycling the ENIU.

The OK, Fault, LAN, and Status LEDs will flash eight patterns (two for each byte) spelling out the IP address digit by digit. As each of the eight patterns is displayed, record them in the table below. Then, decode the binary patterns into hexadecimal (hex) numbers using the Weight column values or the "LED Decoding Table." Two additional (Value) rows are provided below the table for you to write in the decoded hex numbers and equivalent decimal numbers. Use the following guidelines for recording the patterns:

- For the OK, FAULT, and LAN LEDs, record 1 if On or 0 if Off.
- For the STATUS LED, record one of these three conditions: 1 if On with an amber (yellow) color, 0 if On with a green color, or 0 if Off.

If no IP address has been stored, the LEDs will indicate the default address of 195.0.0.0.

LED ↓	1	2	3	4	5	6	7	8	Weight
OK									8
FAULT									4
LAN									2
STATUS									1

Hex. Values									N/A
Decimal Values									N/A

IP Address Example

In the following example, the bit patterns have been recorded and then converted to the two-byte hex number, then to an equivalent decimal number:

LED ↓	1	2	3	4	5	6	7	8	Weight
OK						1			8
FAULT								1	4
LAN		1				1			2
STATUS	0	1	1	0	1	1	0	1	1

Hex. Values	0	3	1	0	1	B	0	5	N/A
Decimal Values	3		16		27		5		N/A

Thus, the decoded IP address is 3.16.27.5.

Chapter 7 The VersaMax™ Family of Products

7.1 Product Family Overview

The VersaMax family of products provides distributed I/O that spans PLC and PC-based architectures. Designed for industrial and commercial automation, VersaMax I/O provides a common, flexible I/O structure for local and remote control applications. The VersaMax PLC provides big-PLC power with a full range of I/O and option modules. VersaMax I/O Stations with Network Interface Modules make it possible to add VersaMax I/O to other types of networks. VersaMax meets UL, CUL, CE, Class1 Zone 2 and Class I Division 2 requirements.

The 70-mm depth and small footprint of VersaMax I/O enables easy, convenient mounting as well as space-saving benefits. Modules can accommodate up to 32 points of I/O each.

VersaMax products feature DIN-rail mounting with up to eight I/O and option modules per “rack” and up to 8 racks per VersaMax PLC or VersaMax I/O Station system. Expansion racks can be located up to 750 meters from the main VersaMax PLC or VersaMax I/O Station rack. Expansion racks can include any VersaMax I/O, option, or communications module.

VersaMax provides automatic addressing that can eliminate traditional configuration and the need for hand-held devices. Multiple field wiring termination options provide support for 2-, 3-, and 4-wire devices.

For faster equipment repair and shorter Mean-Time-To-Repair, the hot insertion feature enables addition and replacement of I/O modules while a machine or process is running and without affecting field wiring.

For more information about modules and system installation instructions, refer to the *VersaMax Modules, Power Supplies and Carriers User's Manual*, GFK-1504.

Appendix A Glossary of Terms

ARP	<p>Address Resolution Protocol. A system used to determine the MAC address of a device whose IP address is known. In practice, an ARP request message is sent to a particular IP address, requesting the device's MAC address. The device's reply, containing its MAC address, is called an ARP response message.</p> <p>It is also a computer utility program (such as ARP.EXE in Windows 95) that can be used in conjunction with Telnet software to assign IP addresses to devices.</p>
Byte	A group of eight bits. Also called by the data communications term, octet .
Carrier	A device that provides mounting, backplane communications, and field wiring connections for VersaMax modules. Carriers, in turn, mount to DIN-rails.
DIN-rail	A metal rail measuring a standard 7.5mm x 35mm that serves as a mounting device for various products, such as the VersaMax products, which usually snap on or off the DIN-rail. DIN-rails should be compliant, preferably, with the DIN EN50032 standard.
EGD	Ethernet Global Data. A proprietary GE protocol that provides efficient connectionless periodic data transfer over an Ethernet network. It operates over the industry standard User Datagram Protocol (UDP).
ENIU	Ethernet Network Interface Unit
ERM	Expansion Receiver Module. A VersaMax module that mounts in an expansion rack. It is used to interface the expansion rack to the expansion bus.
ETM	Expansion Transmitter Module. A VersaMax module that mounts in an I/O station. It is used to interface the I/O station to the expansion bus.
Gateway	See Router
Host	See Node
Hostid	The portion of an IP Address that identifies the host. Each host on the same network must have a unique hostid .
I/O	Input/Output
IP	Internet Protocol
IP Address	A unique address number assigned to a network device. It has a network portion, called the netid , and a host portion called the hostid . According to the current standard, it is a 32-bit value. To make it more readable to humans, the 32 binary bits are often divided into four eight-bit segments ("octets"); then each segment is converted to a decimal number (in the range of 0-255), and each decimal number is separated by dots. For example, 192.2.1.123. This type of notation is called "dotted decimal" notation. In practice, a block of IP address numbers is assigned to an organization, which then assigns unique numbers from within its assigned number block to devices that it manufactures or implements.
LAN	Local Area Network
MAC	Media Access Control.

MAC Address	A unique address number assigned to a network device. According to the current standard, it is a 48-bit value, divided into twelve four-bit segments, with each segment converted to a hexadecimal number, and each pair of hexadecimal number separated by a dash. For example, 08-00-19-01-24-B6. In practice, an organization is assigned a unique 24-bit initial number sequence (GE's sequence is 08-00-19). The organization uses its assigned sequence with 24 additional bits to assign a unique number to the network devices it produces. MAC Addresses conform to the 802.3 MAC Layer Standard.
Modbus™	A serial communications protocol, originally developed by Modicon, that has come into widespread use worldwide. Often called "Modbus RTU", or just "RTU." Modbus is now a trademark of Gould, Inc.
Netid	The portion of a host's IP Address that identifies what network the host is located on. Each host on a given network/subnet must have the same netid . A router, which serves as a host on two or more networks, has a different IP address for each network it is connected to, with each IP address being based upon the netid for its network.
NIC	Network Interface Card
NIU	Network Interface Unit
Node	A device connected to a network. Also called a "host." Examples of Ethernet nodes are (1) a personal computer with an Ethernet NIC, (2) a PLC with an Ethernet interface module, or (3) an Ethernet router. Each node must have its own unique IP Address.
Octet	A group of eight bits. Octet is the data communications term for byte .
PLC	Programmable Logic Controller
Router	A device that routes messages between/among two or more networks. A router uses a message's destination IP Address to determine what network to route the message to. A router has a different IP Address for each network it connects to (see also, Netid). A router is also known by the term "gateway."
Subnet	Shortened version of the term subnetwork. It is the most basic or lowest level of network. Subnets are formed by subdividing an organization's assigned Network ID using a subnet mask . This subdividing involves reassigning bits from the host portion to the network portion of the address.
Subnet mask	A 32-bit binary number, assigned to each network or subnetwork, that is used in conjunction with IP addresses. It allows an organization to use their assigned IP addresses, which otherwise would be limited to one network, to form additional internal subnetwork addresses.
TCP	Transmission Control Protocol
UDP	User Datagram Protocol. UDP is an ISO transport layer protocol that is very efficient because it is connectionless and uses a minimum of error checking.
UTP	Unshielded Twisted Pair.
VersaMax	A GE Automation product series, consisting of PLC and I/O devices.
VersaPro	A GE Automation programming and configuration software product.

Appendix B IP and MAC Addresses

This appendix gives an overview of IP addresses, routers (gateways), subnet masks and MAC addresses.

B-1 IP Addresses

Each TCP/IP *node* on a network must have a unique *IP address*. The *Ethernet NIU* is such a node, as is a Personal Computer running TCP/IP. There may be other nodes on the network that are not involved with communications to the PLCs, but no matter what their function, each TCP/IP node must have its own *IP address*. It is the unique *IP address* that identifies each node on the network (or system of connected networks). (Note that Internet terminology often uses the term “host” to identify a node on a network.)

The *IP address* is 32 bits long and has a *netid* part and a *hostid* part. Each network is a Class A, Class B, or Class C network. The class of a network determines how the IP Address bits are apportioned between the *netid* and *hostid* parts:

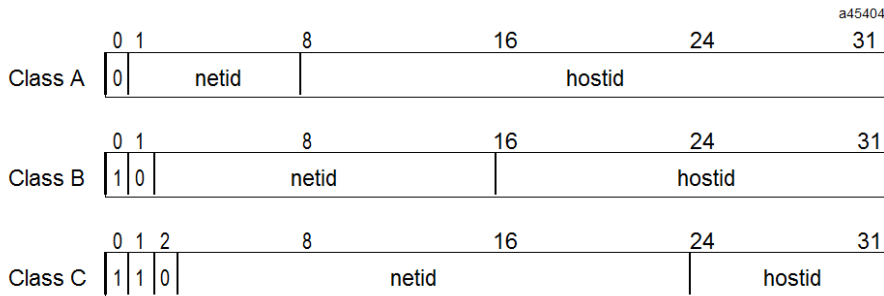


Figure 70: IP Address Formats for Network Classes A, B & C

Each node on the same local network must have an *IP address* of the same class and each must have the same *netid*. Each node on the same network must have a different *hostid* thus giving it a unique *IP address*. IP addresses are written as four decimal integers (0–255) separated by periods (called “dotted-decimal”) where each integer gives the value of one byte of the IP address. For example, the 32-bit IP address:

00000011 00000000 00000000 00000001

is written in dotted-decimal format as:

3.0.0.1

B-2 Network Classes

As shown in the previous section, there are three major network classes, A, B, and C. You can distinguish the class of an IP address from its first integer if it's written in dotted-decimal format, or from the leading bit or bits (called "Class ID Bits Patterns" in the following table) if it's written in binary format. Using the leading bits method, we see that any IP address (in binary format) that starts with a zero is a Class A address, any IP address starting with 10 is a Class B address, and any starting with 110 is a Class C address. To illustrate this, the following example IP address is shown in both formats:

Dotted-decimal format: 191.12.3.77

Binary format: 10111111 00001100 00000011 01001101

In the table below, we see from the "Range of First Integer" column, that the first of the four integers, 191, indicates this to be a Class B address. Additionally, we see from the "Class ID Bit Patterns" column that the leading bits 10 also indicate that this is a Class B address.

Be aware that utilizing these leading bits (which are in the netid portion of the IP address) for the Class ID leaves less bits for network identification. Compare the "Size of Netid" column with the "Remaining Netid Bits" column in the table, which reflects the netid size after the class ID bits are subtracted.

The following table reveals how many total network numbers are available. As shown, the total number is $126 + 16,382 + 2,097,150$ which equals 2,113,658 possible networks for the entire world. That seemed like a large number 20 years ago, but due to the enormous growth of the Internet, there is a danger of running out of network addresses. This assessment is based upon the current 32-bit standard known as IP Version 4. A new IP address standard, IP Version 6, will, when implemented, address this problem by increasing IP address length to 128 bits.

Class	Size of Netid	Size of Hostid	Range of First Integer	Class ID Bit Patterns	Remaining Netid Bits	Max. Networks per Class	Max. no. of Hosts per Network	Default Subnet Mask
A	8 bits	24 bits	0 – 127	0	7	126	16,777,216	255.0.0.0
B	16 bits	16 bits	128 – 191	10	14	16,382	65,536	255.255.0.0
C	24 bits	8 bits	192 – 223	110	21	2,097,150	254	255.255.255.0
Other	N/A	N/A	224 – 255	N/A	N/A	N/A	N/A	N/A

B-3 Routers and Gateways

Routers (also known as gateways) connect individual physical networks into a system of networks. When a node needs to communicate with a node on another network, a router transfers the data between the two networks. Each router port must have an appropriate IP address for the network (or subnetwork) to which it connects.

The following figure shows Router R1 connecting Network 1 with Network 2. Note that each router port has its own IP address (the Network 1 port is 128.1.0.2, and the Network 2 port is 128.2.0.3). A router port's IP address must contain the netid of the network it connects to in keeping with the rule that all nodes on a given network must have the same netid.

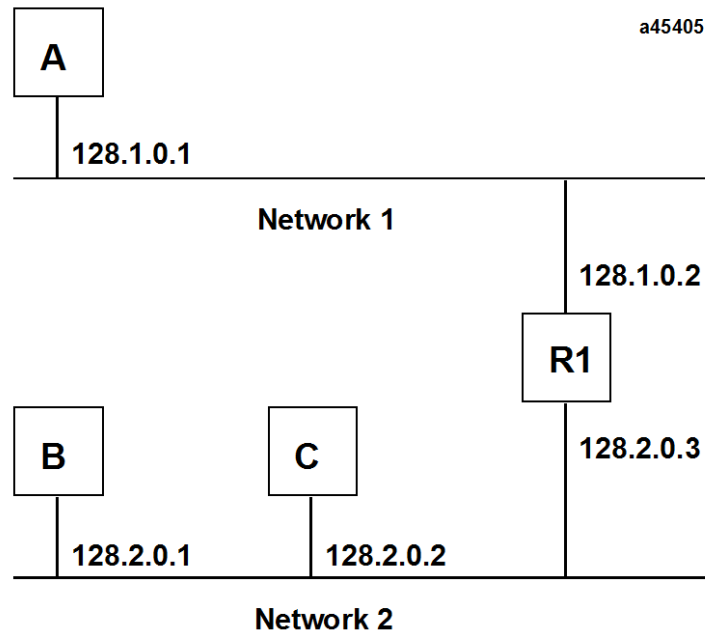


Figure 71: Connecting Two Networks with a Router

When host B communicates with host C, it knows from C's IP address that C is on the same network. In an Ethernet environment, B can then resolve C's IP address to a MAC address (via ARP) and communicate with C directly.

When host B communicates with host A, it knows from A's IP address that A is on another network (the *netids* are different). In order to send data to A, B must have the IP address of the router connecting the two networks. In this example, the router's IP address on Network 2 is 128.2.0.3.

B-4 Subnets

When an organization is assigned a block of IP addresses, it receives one network ID (designated by the *netid* portion of the IP address) and a block of host IDs. The *netid* portion of the assignment is fixed; it cannot be changed. All IP addresses that an organization assigns from its block have the same *netid*, so they are all assigned to one basic network. At the time this arrangement was conceived, it was not foreseen that organizations might want or need to divide a single network into many subnetworks, so no provision was originally made for this. Later, because the need for more networks arose, the subnet arrangement was devised. Since the *netid* portion of an IP address is fixed, it was necessary to use a portion of the *hostid* to create subnetwork IDs.

So, subnet addressing is an extension of the IP address scheme that allows a site to use a single *netid* for multiple networks (subnets). Routing outside the site continues as usual by dividing the IP address into a *netid* and a *hostid* based on the Network Class (A, B, or C) definition. However, inside a site, the IP address is rearranged into custom *netid* and *hostid* portions by a *subnet mask*, discussed next.

B-4.1 Subnet Masks

A subnet mask is a 32-bit binary number that is assigned to a network to indicate how many of the *hostid* bits will be reallocated for use as subnet ID bits. Each class of network, A, B, or C, has a default subnet mask. A default subnet mask does not create subnet ID capability. You must create a custom subnet mask to be able to create subnets. Let's use a Class B network address as an example. The normal bit allocation of a class B IP address is that the first 16 bits are the *netid* and the last 16 bits are the *hostid*. The following is the default subnet mask for a class B network:

Default Class B subnet mask: 11111111 11111111 00000000 00000000

In a subnet mask, the ones indicate how many bits are in the *netid* and the zeros indicate how many bits are in the *hostid*. So the default Class B subnet mask shown (16 one-bits and 16 zero-bits) does not change the allocation of the bits in a Class B IP address, which means we cannot create any subnets if we have a default subnet mask. We must create a "custom" subnet mask in order to create subnets. This consists of taking the default subnet mask and replacing some zeros with ones, starting with the left-most zero. For example, let's change the default subnet mask above by changing the first two zeros to ones. Here is the resulting subnet mask:

Custom Class B subnet mask: 11111111 11111111 11000000 00000000

The two extra *netid* bits, taken from the *hostid* bits, can be used to designate subnets. These two bits have four possible combinations:

00 01 10 11

Therefore, we might assume that we could designate four subnets with these two bits. However, the current subnet standard states that the all-zeros and all-ones combinations are not allowed to be used as subnet IDs. Therefore, our two bits only allow us to name two subnets. (Note that the two bits are the two most significant bits in their byte, so they have values of 128 and 64.) If we needed more than two subnets, we would have to reallocate more *hostid* bits to the subnet ID by changing more of the zeros to ones in the subnet mask. The possible number of subnets for a given number of reallocated *hostid* bits is found by this formula:

Subnets = $2^N - 2$ where N = the number of reallocated *hostid* bits

So, using this formula, we see that 2 bits gives 2 subnets, 3 bits gives 6 subnets, 4 bits gives 14 subnets, etc. Note that by reallocating *hostid* bits for use in the subnet ID, there are fewer host bits left to use for assigning host IDs. Therefore, depending on your application, you may need to carefully balance the number of subnets you create against the number of hosts required to ensure that you have sufficient of each.

B-4.2 Subnet Example

Lets create a pair of subnets from Network 2 (a Class B network) in the previous figure. As we discussed in the last section, using the following custom subnet mask would add two additional *netid* bits, allowing us to create two subnetworks:

11111111 11111111 11000000 00000000

(In dotted-decimal format, our subnet mask looks like this: 255.255.192.0)

The new configuration for our network system would be as shown in the following figure. Notice that Network 2 has been subdivided into subnetworks 2.1 and 2.2. Hosts on these subnets still use the basic netid of 128.2 as the first numbers in their IP addresses, but note the additional subnet numbers added to the addresses:

128.2.64.x and 128.2.128.x

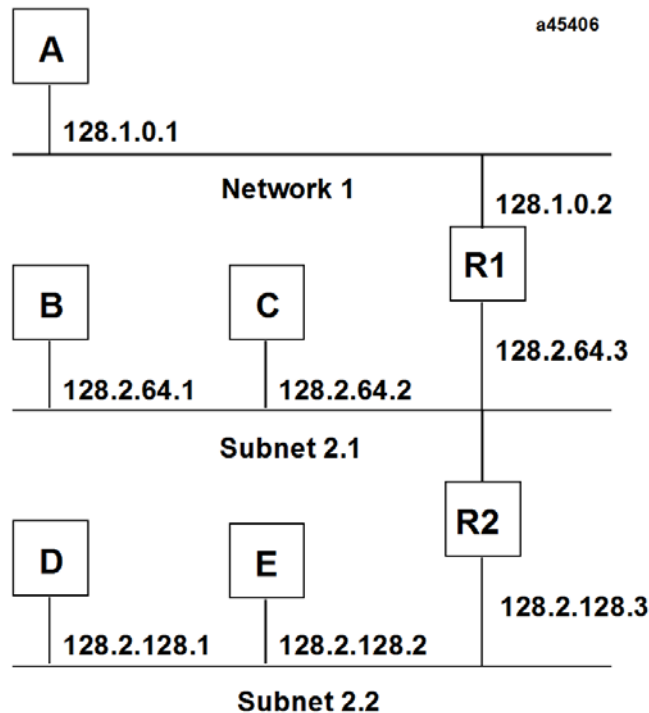


Figure 72: Network Configuration Using a Subnet Mask

In our new arrangement, shown above, Network 2 has been subdivided into two subnets and Hosts D and E have been added. Router R2 connects Subnet 2.1 with Subnet 2.2. Hosts D and E will use Router R2 to communicate with hosts not on Subnet 2.2. Hosts B and C will use Router R1 to communicate with hosts not on Subnet 2.1. When B is communicating with D, R1 (the configured Router for B) will route the data from B to D through Router R2.

In practice, a network's routers (gateways) use the subnet masks (each subnetwork has one) in conjunction with a message's destination IP Address (which is located at the beginning of each message) to determine which subnet to route the message to.

B-5 MAC Addresses

Each byte of the MAC Address is an 8-bit binary number. Thus, the 12-digit hex address is really a 48-bit binary number. A typical MAC Address, 08-00-19-00-53-12, is represented as a binary number in the following example:

Byte#	1		2		3		4		5		6	
Hex	0	8	0	0	1	9	0	0	5	3	1	2
Binary	000	100	000	000	000	100	000	000	010	001	000	001
	0	0	0	0	1	1	0	0	1	1	1	0

Another characteristic that is important, especially for multi-vendor networks, is the order of address-bit transmission on the physical medium. MAC Addresses are transmitted in ascending byte order, with the least significant bit of each byte transmitted first.

The example above is shown including bit transmission order as follows:

Byte#	1		2		3		4		5		6	
Hex	0	8	0	0	1	9	0	0	5	3	1	2
Binary	000	100	000	000	000	100	000	000	010	001	000	001
	0	0	0	0	1	1	0	0	1	1	1	0
Bit Order	876	432		9								
	5	1										

LSB of the MAC Address—
first bit transmitted

MSB of the MAC Address—
last bit transmitted

Appendix C Number System Conversion Table

A hexadecimal digit represents a decimal value from 0 to 15, as follows:

0...9,
A (for 10 decimal),
B (for 11 decimal),
.
.
F (for 15 decimal).

Use the following table to convert binary-to-decimal-to-hex:

Decimal	Binary	Hexadecimal
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

To convert a binary value that has more than four bits into a hexadecimal value,

- 1) first separate the binary number into groups of four bits. (If the number of binary bits is not an even multiple of four, add leading zeroes to make it an even multiple.)
- 2) Each group of four binary bits represents a single hexadecimal number.
- 3) Convert the pattern using the table above

For example, let's convert the binary number 1011110000 to the hexadecimal number 2F0.

- First separate the binary number into groups of four bits (add two leading zeroes): 0010 1111 0000.
- Then use the above conversion table to find the hexadecimal value for each group:
0010 = 2,
1111 = F,
0000 = 0.

GE Automation and Controls
Information Centers

Headquarters:

1-800-433-2682 or 1-434-978-5100

Global regional phone numbers
are available on our web site

www.geautomation.com

Copyright © 2014-2017 General Electric Company.

All Rights Reserved

*Trademark of General Electric Company.

All other brands or names are property
of their respective holders.

Additional Resources

For more information, please
visit our web site:

www.geautomation.com



GFK-1860C