

ARTISAN®
TECHNOLOGY GROUP

Your **definitive** source for quality pre-owned equipment.

Artisan Technology Group
(217) 352-9330 | sales@artisantg.com | artisantg.com

Full-service, independent repair center
with experienced engineers and technicians on staff.

We buy your excess, underutilized, and idle equipment
along with credit for buybacks and trade-ins.

Custom engineering
so your equipment works exactly as you specify.

- Critical and expedited services
- In stock / Ready-to-ship
- Leasing / Rentals / Demos
- ITAR-certified secure asset solutions

Expert team | Trust guarantee | 100% satisfaction

All trademarks, brand names, and brands appearing herein are the property of their respective owners.

Visit our website - *Click HERE*

AUTOMATIONDIRECT.com

Ethernet Communications Modules

Manual Number HX–ECOM–M

WARNING

Thank you for purchasing automation equipment from ***Automationdirect.com***™, doing business as, **AutomationDirect**. We want your new ***Direct***LOGIC™ automation equipment to operate safely. Anyone who installs or uses this equipment should read this publication (and any other relevant publications) before installing or operating the equipment.

To minimize the risk of potential safety problems, you should follow all applicable local and national codes that regulate the installation and operation of your equipment. These codes vary from area to area and usually change with time. It is your responsibility to determine which codes should be followed, and to verify that the equipment, installation, and operation are in compliance with the latest revision of these codes.

At a minimum, you should follow all applicable sections of the National Fire Code, National Electrical Code, and the codes of the National Electrical Manufacturer's Association (NEMA). There may be local regulatory or government offices that can also help determine which codes and standards are necessary for safe installation and operation.

Equipment damage or serious injury to personnel can result from the failure to follow all applicable codes and standards. We do not guarantee the products described in this publication are suitable for your particular application, nor do we assume any responsibility for your product design, installation, or operation.

Our products are not fault–tolerant and are not designed, manufactured or intended for use or resale as on–line control equipment in hazardous environments requiring fail–safe performance, such as in the operation of nuclear facilities, aircraft navigation or communication systems, air traffic control, direct life support machines, or weapons systems, in which the failure of the product could lead directly to death, personal injury, or severe physical or environmental damage ("High Risk Activities"). **AutomationDirect** *specifically disclaims any expressed or implied warranty of fitness for High Risk Activities.*

For additional warranty and safety information, see the Terms and Conditions section of our Desk Reference. If you have any questions concerning the installation or operation of this equipment, or if you need additional information, please call us at 770–844–4200.

This publication is based on information that was available at the time it was printed. At **AutomationDirect** we constantly strive to improve our products and services, so we reserve the right to make changes to the products and/or publications at any time without notice and without any obligation. This publication may also discuss features that may not be available in certain revisions of the product.

Trademarks

This publication may contain references to products produced and/or offered by other companies. The product and company names may be trademarked and are the sole property of their respective owners. **AutomationDirect** disclaims any proprietary interest in the marks and names of others.

Copyright 2006, *Automationdirect.com*™ Incorporated
All Rights Reserved

No part of this manual shall be copied, reproduced, or transmitted in any way without the prior, written consent of ***Automationdirect.com***™ Incorporated. **AutomationDirect** retains the exclusive rights to all information included in this document.

AVERTISSEMENT

Nous vous remercions d'avoir acheté l'équipement d'automatisation de **Automationdirect.com™**, en faisant des affaires comme, **AutomationDirect**. Nous tenons à ce que votre nouvel équipement d'automatisation ***Direct***LOGIC™ fonctionne en toute sécurité. Toute personne qui installe ou utilise cet équipement doit lire la présente publication (et toutes les autres publications pertinentes) avant de l'installer ou de l'utiliser.

Afin de réduire au minimum le risque d'éventuels problèmes de sécurité, vous devez respecter tous les codes locaux et nationaux applicables régissant l'installation et le fonctionnement de votre équipement. Ces codes diffèrent d'une région à l'autre et, habituellement, évoluent au fil du temps. Il vous incombe de déterminer les codes à respecter et de vous assurer que l'équipement, l'installation et le fonctionnement sont conformes aux exigences de la version la plus récente de ces codes.

Vous devez, à tout le moins, respecter toutes les sections applicables du Code national de prévention des incendies, du Code national de l'électricité et des codes de la National Electrical Manufacturer's Association (NEMA). Des organismes de réglementation ou des services gouvernementaux locaux peuvent également vous aider à déterminer les codes ainsi que les normes à respecter pour assurer une installation et un fonctionnement sûrs.

L'omission de respecter la totalité des codes et des normes applicables peut entraîner des dommages à l'équipement ou causer de graves blessures au personnel. Nous ne garantissons pas que les produits décrits dans cette publication conviennent à votre application particulière et nous n'assumons aucune responsabilité à l'égard de la conception, de l'installation ou du fonctionnement de votre produit.

Nos produits ne sont pas insensibles aux défaillances et ne sont ni conçus ni fabriqués pour l'utilisation ou la revente en tant qu'équipement de commande en ligne dans des environnements dangereux nécessitant une sécurité absolue, par exemple, l'exploitation d'installations nucléaires, les systèmes de navigation aérienne ou de communication, le contrôle de la circulation aérienne, les équipements de survie ou les systèmes d'armes, pour lesquels la défaillance du produit peut provoquer la mort, des blessures corporelles ou de graves dommages matériels ou environnementaux ("activités à risque élevé"). La société **AutomationDirect** nie toute garantie expresse ou implicite d'aptitude à l'emploi en ce qui a trait aux activités à risque élevé.

Pour des renseignements additionnels touchant la garantie et la sécurité, veuillez consulter la section Modalités et conditions de notre documentation. Si vous avez des questions au sujet de l'installation ou du fonctionnement de cet équipement, ou encore si vous avez besoin de renseignements supplémentaires, n'hésitez pas à nous téléphoner au 770–844–4200.

Cette publication s'appuie sur l'information qui était disponible au moment de l'impression. À la société **AutomationDirect**, nous nous efforçons constamment d'améliorer nos produits et services. C'est pourquoi nous nous réservons le droit d'apporter des modifications aux produits ou aux publications en tout temps, sans préavis ni quelque obligation que ce soit. La présente publication peut aussi porter sur des caractéristiques susceptibles de ne pas être offertes dans certaines versions révisées du produit.

Marques de commerce

La présente publication peut contenir des références à des produits fabriqués ou offerts par d'autres entreprises. Les désignations des produits et des entreprises peuvent être des marques de commerce et appartiennent exclusivement à leurs propriétaires respectifs. **AutomationDirect™** nie tout intérêt dans les autres marques et désignations.

Copyright 2006, Automationdirect.com™ Incorporated

Tous droits réservés

Nulle partie de ce manuel ne doit être copiée, reproduite ou transmise de quelque façon que ce soit sans le consentement préalable écrit de la société **Automationdirect.com™** Incorporated. **AutomationDirect** conserve les droits exclusifs à l'égard de tous les renseignements contenus dans le présent document.

Manual Revisions

If you contact us in reference to this manual, be sure and include the revision number.

Title: Ethernet Communication Modules
Manual Number: HX–ECOM–M

Issue	Date	Description of Changes
Original	4/98	Original Issue
1st Edition, Rev A	8/02	Added DL250–1 and DL260 references Removed DL250 references Minor changes (Note: DL250 has the same functionality as the DL250–1 except for local expansion capability.)
2nd Edition	6/03	Added H0–ECOM module, changed manual part number
2nd Edition, Rev A	6/03	Corrected Special Purpose Communication Relays
3rd Edition	11/04	Added H2–ECOM100 module Updates for NetEdit3
3rd Edition, Rev A	04/05	Added H0–ECOM100 module
3rd Edition, Rev B	04/06	Added H4–ECOM100 module

Table of Contents

Chapter 1: Introduction

Manual Overview **1–2**
- The Purpose of this Manual 1–2
- Other Reference Materials 1–2
- Who Should Read this Manual 1–2
- Quality Technical Manuals and Technical Support 1–2

Conventions Used **1–3**

ECOM Module Introduction **1–4**
- ECOM Communication Possibilities 1–4
- Your Network PC 1–5

Frequently Asked Questions **1–6**

Chapter 2: Setup and Installation

ECOM Network Identifiers **2–2**
- Module ID 2–3
- Name 2–3
- IP Address 2–4
- Ethernet (MAC) Address 2–4
- Using Multiple Network Identifiers 2–4

Setting the Module ID with the DIP Switch **2–5**
- H0 / H2 Series ECOM DIP Switch 2–5
- The H4 Series ECOM DIP Switch 2–6

Inserting the ECOM Module in the PLC Base **2–7**
- H0 Series ECOM Module Installation 2–7
- DL205 Slot Choices 2–7
- H2 Series ECOM Module Installation 2–8
- DL405 Slot Choices 2–9
- H4 Series ECOM Module Installation 2–10

ECOM Network Layouts **2–10**

Network Cabling **2–12**
- ECOM Supports Two Standards 2–12
- 10/100BaseT Networks 2–12
- 10/100BaseT Connections 2–13
- UTP Cable 2–13
- 10BaseFL Connections 2–13
- Fiber Optic Cable 2–13
- Fiber Optic Module ST Connector 2–13

Maximum Cable Length **2–14**

Maximum Number of ECOM Modules on the Network **2–15**

Chapter 3: Configuring ECOMs Using NetEdit3

NetEdit3 Software 3–2
Installing NetEdit3 3–2
Launching NetEdit3 3–3
The NetEdit3 Screen 3–3
Adding Network Protocol Support to the NetEdit3 PC 3–4
Using NetEdit3 3–5
Ethernet Communication Protocol 3–5
Ethernet Address 3–6
Module Type, IP Address and ID 3–6
Module Info> General Information 3–7
Module Info>Ethernet Stats 3–7
ECOM Settings 3–7
ECOM Settings>Configuration>General 3–8
ECOM Settings>Configuration>Advanced 3–9
ECOM Settings>Configuration>Peer to Peer 3–10
ECOM Settings>Utils>Test CPU Access 3–12
ECOM Settings>Firmware 3–12
FileMenu>Live Update 3–13
F / B / C Columns 3–13

Chapter 4: RLL Programming for Communications

PLC-to-PLC Communications 4–2
How RLL is Used for Communications 4–2
Network Instructions 4–3
Read (RX) and Write (WX) Instructions 4–3
Building the Read (RX) or Write (WX) Routine 4–3
The First LD Instruction 4–3
The Second LD Instruction 4–4
The LDA Instruction 4–4
Write (WX) Instruction 4–5
Addressing the Different Memory Types 4–6
Bit Memory 4–6
Word Memory and Aliases 4–6
DL05 CPU 4–7
DL06 CPU 4–7
D2–240 CPU 4–8
D2–250–1 CPU 4–8
D2–260 CPU 4–9
D4–430 CPU 4–9
D4–440 CPU 4–10
D4–450 CPU 4–10
Special Relays for Communications 4–11
Program with One Read Instruction 4–13
Program for the Initiating PLC 4–13
Program for the Responding PLC 4–13
Rung 1 4–14
Rung 2 4–14
Rung 3 4–14

Rung 4 4–14
Program for the Responding PLC 4–14
Example Program with One Write Instruction 4–15
Program for the Initiating PLC 4–15
Program for the Responding PLC 4–15
Rung 1 4–16
Rung 2 4–16
Rung 3 4–16
Rung 4 4–16
Program for the Responding PLC 4–16
Integrating Multiple Read and Write Instructions 4–17
Interlocking Relays 4–17
First RX/WX Instruction 4–18
Second RX/WX Instruction 4–19
Third RX/WX Instruction 4–19
Returning to the First RX/WX Instruction 4–19
Shift Register 4–20
Store If Equal 4–20
First RX/WX Instruction 4–21
Second RX/WX Instruction 4–21
Third RX/WX Instruction 4–21

Chapter 5: MODBUS TCP/IP for H0/H2/H4–ECOM100

MODBUS TCP/IP 5–2
Client / Server Model 5–2
Protocol Description 5–2
Supported MODBUS Function Codes 5–4
Network Server (slave) Operation 5–5
MODBUS Function Codes Supported 5–5
Determining the MODBUS Address 5–5
If Your Host Software or Client Requires the Data Type and Address 5–6
Example 1: V2100 5–11
Example 2: Y20 5–11
Example 3: T10 Current Value 5–11
Example 4: C54 5–11
If the Host Software or Client Requires an Address ONLY 5–12
Example 1: V2100 5–14
Example 2: Y20 5–14
Example 3: C54 5–14
Network Client (master) Operation 5–15
MODBUS Function Codes Supported 5–15
PLC Memory Supported for Client Operation 5–16
Example 1: Calculating Word PLC Address 5–17
Example 2: Calculating Discrete Input PLC Address 5–17
Building the Read (RX) or Write (WX) Routine 5–18
Step 1: Identify ECOM Slot Location and Server Node # 5–18
Step 2: Load Number of Bytes to Transfer 5–19
Step 3: Specify Master Memory Area 5–19
Step 4: Specify Slave Memory Area 5–19
Communications from a Ladder Program 5–20

Multiple Read and Write Interlocks 5–20
H0/H2–ECOM100 System Memory 5–21

Chapter 6: H0/H2/H4–ECOM100 DHCP & HTML Configuration

H0/H2/H4–ECOM100 DHCP 6–2
DHCP Issues 6–2
Disabling DHCP and Assigning a Static IP Address 6–2
Using HTML Configuration 6–3
Connecting to the H0/H2/H4–ECOM100 6–3
H0/H2/H4–ECOM100 Client Peer to Peer Configuration 6–5

Chapter 7: Maintenance and Troubleshooting

Isolating a Communication Problem 7–2
Diagnostic Tools and Techniques 7–2
Troubleshooting Chart 7–2
ECOM Module Diagnostic LEDs 7–4
H0 Series Indicators 7–4
H24–ECOM–(F) Indicators 7–5
H2/H4–ECOM100 Indicators 7–6
Using NetEdit3 for Troubleshooting 7–7
Select a Module 7–7
Module Information 7–7
Change Protocol 7–8
Ethernet Stats 7–8
RX/WX Settings 7–9
Record the 7–9
Module Settings 7–9
Replacing the ECOM Module 7–10
Diagnosing Network Cable Problems 7–11

Appendix A: General Specifications

General Specifications A–2
Ethernet Standards A–4

1

Introduction

In This Chapter. . . .

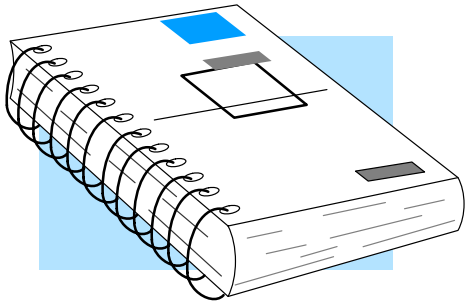
- — Manual Overview
- — ECOM Module Introduction
- — Frequently Asked Questions

Manual Overview

The Purpose of this Manual

This manual describes how to use the **Ethernet Communication (ECOM) Modules.** You will find information about:

- Setting up the ECOM module
- Network layouts
- PC-to-PLC communications
- PLC-to-PLC communications
- RLL programming examples
- Maintenance and troubleshooting



Other Reference Materials

Other ***Direct***LOGIC™ manuals may be useful for your application.

User Manuals

- DL05 User Manual part number D0–USER–M
- DL06 User Manual part number D0–06USER–M
- DL205 User Manual part number D2–USER–M
- DL405 User Manual part number D4–USER–M
- ***Direct***SOFT User Manual (with part number PC–PGMSW v2.3 or later)
- KEP***Direct*** for PLCs (with part number DA–KEPPLC–M)

Who Should Read this Manual

If you need a high-speed communications link between your ***Direct***LOGIC PLC and PCs or other ***Direct***LOGIC PLCs and you understand the basics of installing and programming PLCs, this is the right manual for you. This manual gives you the information you need to set up and install a communication link to an ECOM module.

Quality Technical Manuals and Technical Support

We strive to make our manuals the best in the industry. We rely on your feedback to let us know if we are reaching our goal. If you cannot find the solution to your particular application, or, if for any reason you need additional assistance, please call us at **770–844–4200.** Our technical support group is glad to work with you in answering your questions. They are **available weekdays from 9:00 a.m. to 6:00 p.m. Eastern Time**. You can also contact us on the **web** at:

http://www.automationdirect.com

If you have a comment or question about any of our products, services, or manuals please fill out and return the ‘Suggestions’ card that came with this manual.

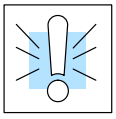
Conventions Used



When you see the “light bulb” icon in the left–hand margin, the paragraph to its immediate right will give you a **special tip**.
The word **TIP:** in boldface will mark the beginning of the text.



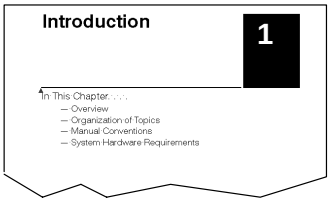
When you see the “notepad” icon in the left–hand margin, the paragraph to its immediate right will be a **special note**.
The word **NOTE:** in boldface will mark the beginning of the text.



When you see the “exclamation mark” icon in the left–hand margin, the paragraph to its immediate right will be a **warning**. This information could prevent injury, loss of property, or even death (in extreme cases).
The word **WARNING:** in boldface will mark the beginning of the text.

Key Topics for Each Chapter

The beginning of each chapter will list the key topics that can be found in that chapter.



ECOM Module Introduction

Several Ethernet Communication (**ECOM**) modules are currently available for use with DL05/06, DL205 and DL405 PLC systems. The ECOM modules are shown below. These modules provide a low-cost, high-speed Ethernet link for PLC systems. The modules are easy to set up and install on 10/100BaseT (twisted pair, copper wire) or 10BaseFL (fiber optic) Ethernet networks.

LEDs on the face of each module give vital information about the status of the module and the communication link. The 10/100BaseT modules use standard RJ45 modular connectors, and the 10BaseFL modules use ST style bayonet connectors.

DL05/06 ECOM Modules	DL205 ECOM Modules	DL405 ECOM Modules
H0–ECOM **H0–ECOM100**	**H2–ECOM** **H2–ECOM100** **H2–ECOM–F**	**H4–ECOM** **H4–ECOM100** **H4–ECOM–F**



H0/H2/H4–ECOM100 modules shown

NOTE: If you are using a D0–06LCD in your DL06 PLC, the date code for the display unit must be 032A_ or later to be compatible with the H0 series ECOM modules.

ECOM Communication Possibilities

You can use the ECOM modules to share data between two or more ***Direct***LOGIC PLCs or between ***Direct***LOGIC PLCs and personal computers. The H0/H2/H4–ECOM100 additionally allows client/server communications with other Ethernet devices using the MODBUS TCP/IP protocol. Communication between PLCs/MODBUS TCP/IP devices is accomplished by using the Read/Write (RX/WX) instructions which are available in the ***Direct***SOFT Programming Software Users Manual. Chapters 4and 5 explain the use of the RX/WX instructions.

You can also use a personal computer running ***Direct***SOFT Programming Software to program your PLCs over the Ethernet network. It is just like programming through the programming port on the CPU, but with the convenience of doing it from a single location.

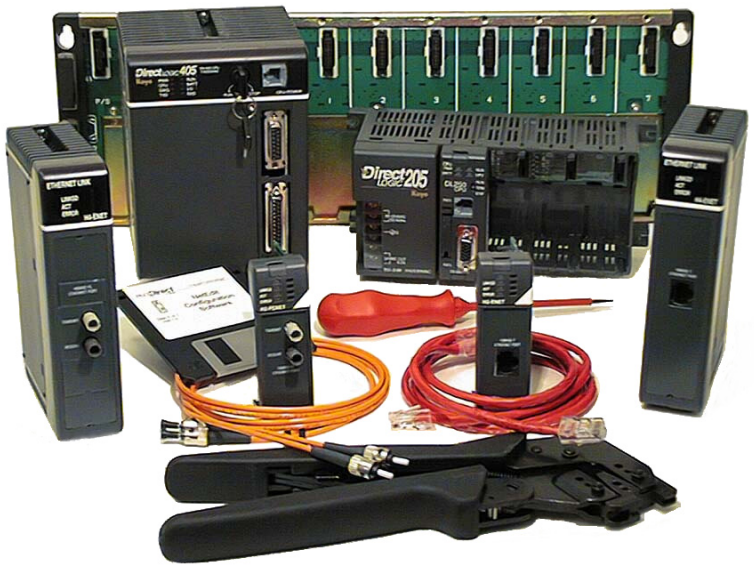
Use catalog number PC–DSOFT5 to program the complete ***Direct***LOGIC family of PLCs (the DL105 and DL305 series are the only PLCs that cannot be programmed over Ethernet). Chapter 2 will indicate the CPUs which can be used with the ECOM modules.

NOTE: We recommend using a dedicated network for your PLC control applications. For more information see Chapter 2, Setup and Installation.

PCs running our KEP***Direct*** for PLCs software can establish Ethernet links for exchanging information with ***Direct***LOGIC 05/06/205/405 PLCs.

Your Network PC

You can use a personal computer equipped with a 10/100BaseT or 10BaseFL network adapter card and NetEdit3 software to configure the ECOM module over the network. You can also use NetEdit3 for troubleshooting certain communication problems. The NetEdit3 utility is included with this manual and is available for download at **http://www.automationdirect.com**.



Frequently Asked Questions

Q. How can I speed up my ECOM communications?

A. Try shortening the scan time of the PLC (the PLC allows only one ECOM transaction per scan).

Q. What causes “Task code error response” with extended E353 error?

A. Error is from the PLC and means “timeout in the background communications”. Error is due to a backplane communication problem (ECOM, DCM, etc.).

Q. Can an ECOM be configured to talk through a gateway?

A. Yes. You configure the gateway to reroute traffic to and from the ECOM via port 7070 (hex).

Q. When using a DL205 with an ECOM, is there a way to turn on the PLC outputs from a computer using a non–Windows operating system?

A. Host Engineering has what is called an Ethernet SDK (software developers kit) which can normally be downloaded (free) from their website. However, this will be useless for your system, but to accommodate your operating system, you can fill out a form located on the Host Engineering website and request the source code for the Ethernet SDK. This is necessary so that Host Engineering will know who is using their source code. The source code can then be recompiled to work on your system.

With the SDK recompiled for your system, you can use CCM (i.e. ***Direct***NET) protocol to turn on PLC outputs by simply knowing the memory types and ranges as required for syntax.

The request form and details about the SDK can be found on the Host Engineering homepage (www.hosteng.com). Once there, click on “EBC/ECOM/EDRV” under “S/W Developer Kits” in the left–hand column.

Q. What is the fastest way to get data from PLC to PLC?

A. Install ECOM in both PLCs and use the RX/WX commands is the quickest way to do this. It is much faster that using serial communication.

Q. Can the ECOM do a broadcast message to multiple slave devices?

A. No.

Q. Can the MAC address be changed?

A. The MAC address is burned into ROM on the module and is set just before it leaves the factory. There is no logical way for any protocol to change this address. Host Engineering’s range for Ethernet (MAC) addresses is 00.E0.62.xx.xx.xx.

2

Setup and Installation

In This Chapter. . . .

- ECOM Network Identifiers
- Setting the Module ID with the DIP Switch
- Inserting the ECOM Module in the PLC Base
- ECOM Network Layouts
- Network Cabling
- Maximum Cable Length
- Maximum Number of ECOM Modules on the Network

ECOM Network Identifiers

This section describes network identifiers that can be assigned to the ECOM module. Each module must be assigned at least one unique identifier to make it possible for PCs or other ECOMs to recognize it on the network. Four methods of identifying the ECOM module give it the flexibility to fit most networking schemes.

The four ECOM identifiers are:

- Module ID
- Name
- IP (Internet Protocol) Address
- Ethernet (MAC) Address

The first three are user-selectable. The last one is set at the factory. Each of the identifiers is discussed in this chapter. If you have more than a few ECOMs on your network, consider making a chart or spreadsheet of network IDs, as shown below:

Ethernet (MAC) Address	Module ID	Name	IP Address
00 E0 62 20 01 20	3	PumpStationTwo	192.168.100.005
00 E0 62 20 01 58	8	Effluent	255.255.255.255
00 E0 62 20 01 8D	17	BldgThree	192.168.100.001
00 E0 62 20 01 94	2	PumpStationOne	192.168.100.002
00 E0 62 20 01 DE	61		192.168.100.003
00 E0 62 20 01 F1	33		192.168.100.004
00 E0 62 20 01 FB	1	Control Room	255.255.255.255
00 E0 62 20 01 F0	5	Mixer	192.168.100.006

The decision about which type of identifier to use is an important one. Much of the decision depends on the requirements of your particular application. PC-to-PLC communications are generally better accommodated with one type of identifier while PLC-to-PLC communications require a different type. Ease of maintenance and troubleshooting also must be considered before deciding which type to use.

The identifiers are used to **link** your PC to your PLC or one PLC to another PLC. The flexible design of the module allows you to use different identifiers for different links to the same module. This is particularly important if you require both PC-to-PLC and PLC-to-PLC communications on the same network.

The following table summarizes Network Identifiers and their uses:

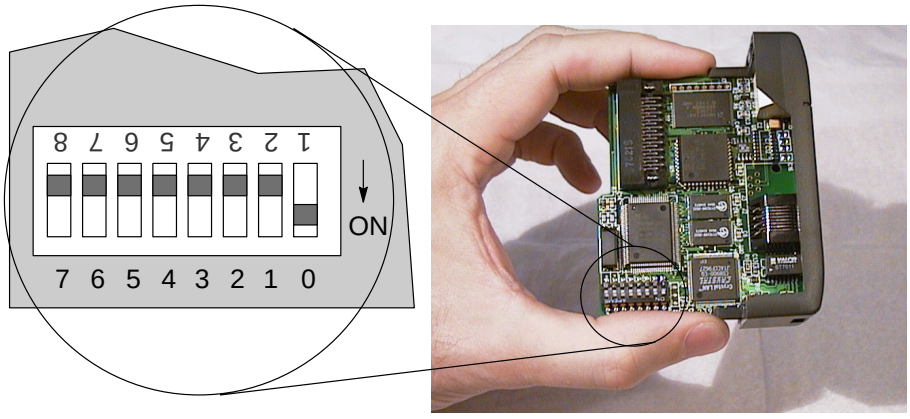
	How to Set	Format	Communication	Restrictions/Notes
Module ID	DIP Switch	Number 1-63	PLC-to-PLC or PC-to-PLC	Disables Module ID in NetEdit3
	NetEdit3	Number 1-90	PLC-to-PLC or PC-to-PLC	DIP Switch must be set to “0”
	NetEdit3	Number 1-999,999,999	PC-to-PLC Only	> 90 (Not for PLC-to-PLC)
Name	NetEdit3	32 Alphanumeric Characters	PC-to-PLC Only	HMI Software may have restrictions
IP Address	NetEdit3	4 Three-digit Numbers xxx.xxx.xxx.xxx (See Page 2-4)	PC-to-PLC; (PLC to PLC–Client/ Server using TCP/IP or MODBUS TCP/IP protocol)	See Your Network Administrator for IP address; (refer to pages 3–10 to 3–11, chapters 5 and 6)
Ethernet (MAC) Address	Set at Factory	12 Hex digits	PC-to-PLC Only	Factory-assigned, for IPX

Module ID

A Module ID is *required* for PLC-to-PLC communications, and it can be set either of two ways. You can assign the Module ID:

- using the DIP switches on the module.
- using the configuration tools in NetEdit3
- HTML Configuration (after IP address is assigned to module using NetEdit3; described in Chapter 5; H0/H2/H4–ECOM100 only)

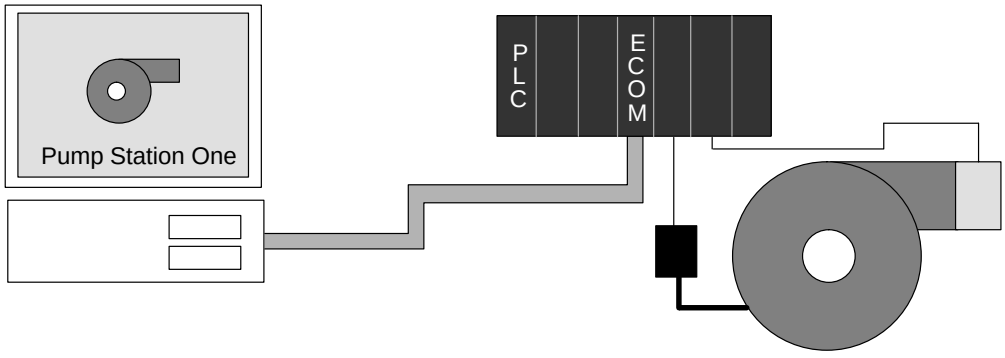
Use the DIP switch if you want the ability to install or change modules *without using a PC* to set the Module ID. Set the module's DIP switch, insert the module in the base, and connect the network cable. Your Module ID is set on powerup, and your ECOM is ready to communicate on the network. We step through setting the DIP switch on pages 2-5 and 2-6.



If you prefer to be able to set or change all Module IDs on your network from a single PC, use the tools in NetEdit3, discussed in chapter 3.

Name

A Name makes it easy to recognize the PLC by its function. An example of a Name is "PumpStationOne." The Name can be up to 32 alphanumeric characters in length. A Name can be assigned using NetEdit3.



PUMP STATION ONE

NOTE: Some HMI software products will not accept Names with numbers as the first character, spaces or certain other non-alphanumeric ASCII characters. Also, your HMI product may not accept Names longer than 16 characters. Consult your HMI product documentation about its naming conventions.

IP Address

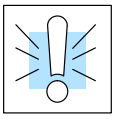
An IP Address can be assigned to the ECOM module if your network requires one. Usually, the IP Address is required in cases where PLCs are sharing the same network with PCs, and some of the PCs are carrying out functions unrelated to PLC control. Normally, a network administrator will assign an IP Address to each device on the network. If you have a separate dedicated network for your PLCs, you can use the Module ID or a Name for each communication link. You must use an IP Address, if you are using the UDP/IP or MODBUS TCP/IP protocol.
Use NetEdit3 to assign an IP address to the ECOM (refer to chapter 3).

The module ships from the factory with an IP Address of 0.0.0.0. This is not a usable IP Address for normal communications. It only serves as a default setting which can be changed using NetEdit3. The **valid settings** are **1 through 254**. You do not have to change the default IP Address unless you are using the IP Address to link to your ECOM module. The default setting does not cause conflicts with other network communications.

If you change the default IP Address for linking to other network devices, you must change all four "0" fields. If any field contains **the number 255 *and* other fields have been changed, the module will *not* be recognized on the network.**

Example

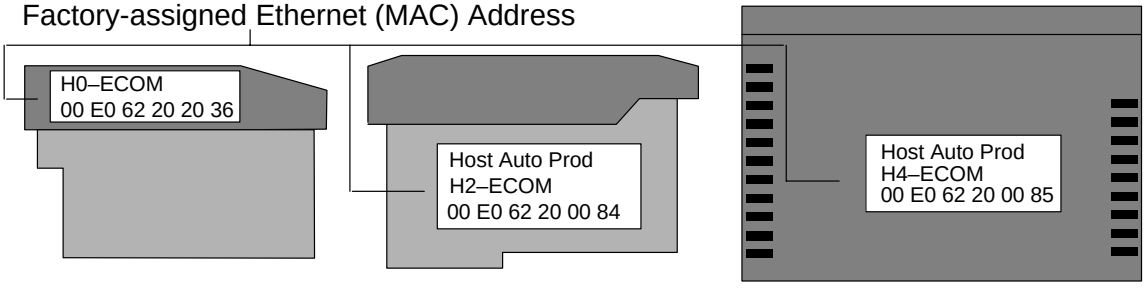
Client (PC/ECOM) Subnet Mask:	255.255.0.0	
Valid Client (PC/ECOM) IP Address:	192.168.50.2	**1–254**
Valid Server ECOM IP Address:	192.168.**55.5**	**Valid settings for Bold number fields**
Valid Server ECOM IP Address:	192.168.**70.15**	**(Do not duplicate)**



WARNING: It is extremely important not to have duplicate IP Addresses on your network. If you are using the IP Address to link the ECOM to **any** network devices (PCs or PLCs), the ECOM must have a unique number.

Ethernet (MAC) Address

A unique Ethernet (MAC) Address is assigned to each module at the factory and will not change. It is printed on a label attached to each ECOM module. The Ethernet (MAC) Address is recognized by NetEdit3. The Ethernet (MAC) Address is a twelve digit number with no deliberate relationship to your network or functional areas of your plant. It does not usually serve as a convenient and easily remembered identifier for your ECOM.



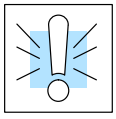
Using Multiple Network Identifiers

You can use the IP Address to satisfy network requirements, a Name for PCs running HMI software and the Module ID for PLCs to share data among themselves. Using one type of identifier does not limit your use of the other identifier types.

Setting the Module ID with the DIP Switch

You can use the DIP switch on the ECOM module to set the Module ID to a number from one to sixty-three. Each module on a given network must be assigned a **unique** Module ID if the Module ID is to be used for communications. Do not use Module ID "0" *for communications*.

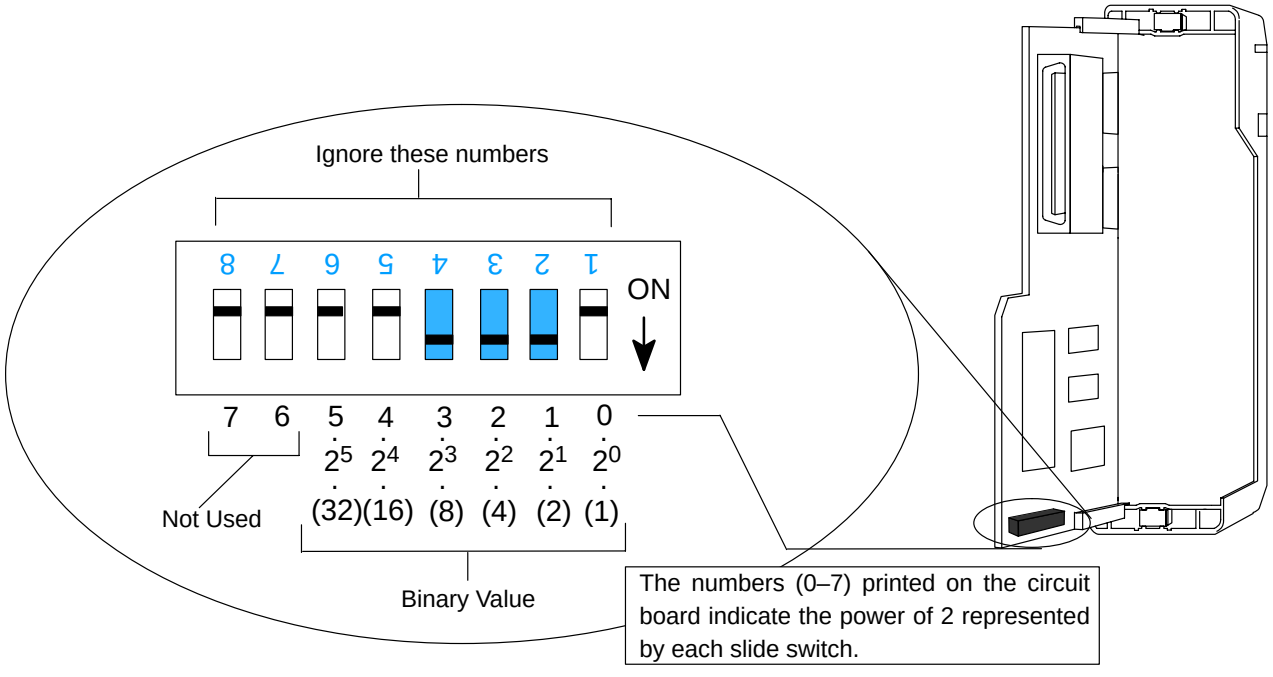
If the DIP switch is set to a number greater than 0, the software tools are disabled from setting the Module ID. The software tools will only allow changes to the Module ID if the DIP switch setting is 0 (zero, all switches OFF). The DIP switch settings are read at powerup. You *can use* the software tools to set the Name and IP Address even if you use the DIP switch for setting the Module ID.



WARNING: Using duplicate Module IDs on a single network will cause unreliable PLC-to-PLC communications.

H0 / H2 Series ECOM DIP Switch

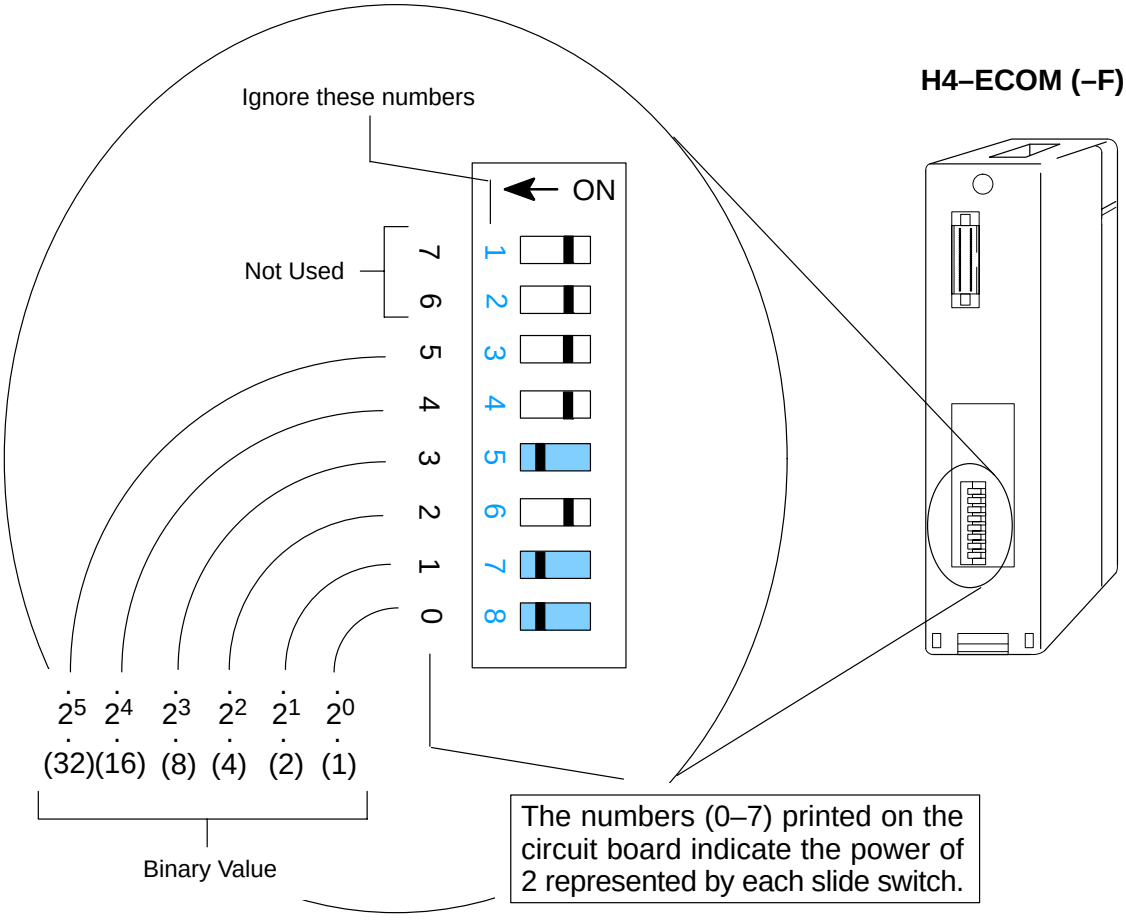
The H0 Series ECOM and H2 Series ECOM's DIP switches contain eight individual slide switches, but only six of these are active. Two are not used. Notice that the individual slide switches are labeled 1 through 8 on the body of the DIP switch (upside down in the figure below). You will also find that the printed circuit board is labeled 0 (zero) through 7. We use the labeling *on the printed circuit board* in describing how to set the switch. The numbers on the printed circuit board indicate the power of 2 represented by each slide switch. For example, switch 0 represents 2^0 (or 1), switch 1 is 2^1 (or 2), switch 2 is 2^2 (or 4), and so on. The figure below shows the binary value of each switch in parentheses ( ).



The Module ID equals the *sum* of the binary values of the slide switches set in the ON position. For example, if you set slide switches 1, 2, and 3 to the ON position, the Module ID will be 14. This is found by adding 8+4+2=14. The maximum value you can set on the DIP switch is 32+16+8+4+2+1=63. This is achieved by setting switches 0 through 5 to the ON position.

The H4 Series ECOM DIP Switch

The H4 series ECOM DIP switch contains eight individual slide switches, but only six of these are active. Two are not used. Notice that the individual slide switches are labeled 1 through 8 on the body of the DIP switch. You will also find that the printed circuit board is labeled 0 (zero) through 7 (as shown in the figure below). We use the labeling *on the printed circuit board* in describing how to set the switch. The numbers on the printed circuit board indicate the power of 2 represented by each slide switch. For example, switch 0 represents 2^0 (or 1), switch 1 is 2^1 (or 2), switch 2 is 2^2 (or 4), and so on. The figure below shows the binary value of each switch in parentheses ( ).



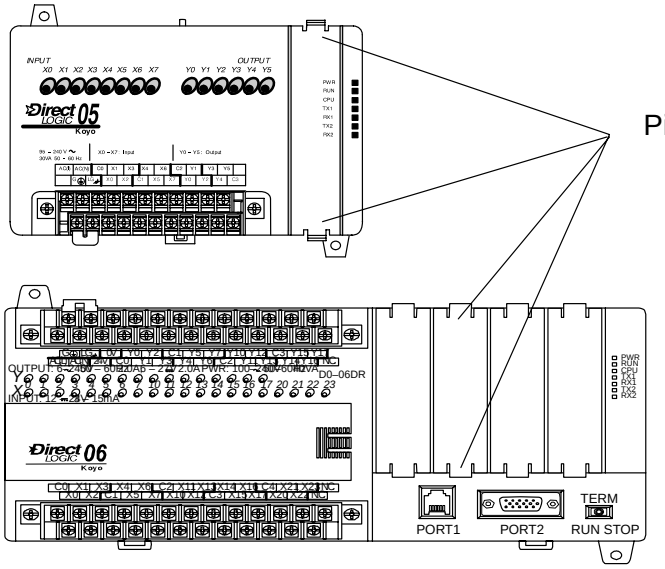
The Module ID equals the *sum* of the binary values of the slide switch set in the ON position. For example, if you set slide switches 0, 1, and 3 to the ON position, the Module ID will be 11. This is found by adding 8+2+1=11. The maximum value you can set on the DIP switch is 32+16+8+4+2+1=63. This is achieved by setting switches 0 through 5 to the ON position.

NOTE: When all the switches are set to OFF (Module ID = 0), the Module ID can be set using the software utilities in NetEdit and ***Direct***SOFT32. **Do not use Module ID "0" for normal communications.** It is okay to leave the Module ID set at zero if you are using the Name or IP Address for communications.

Inserting the ECOM Module in the PLC Base

H0 Series ECOM Module Installation

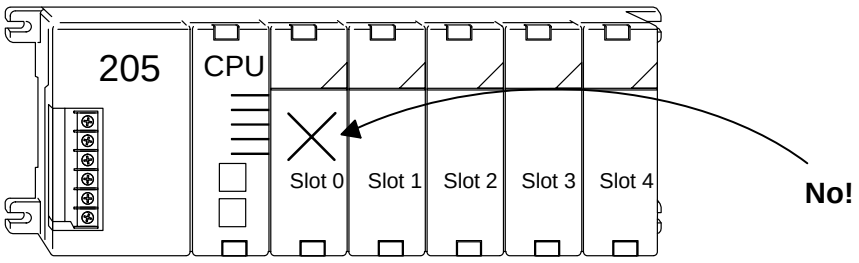
Before installing the option module in the DL05 option slot or any of the DL06 option slots, set the Module ID dip switch (if your application requires this) on the H0 Series ECOM module. The next step is to remove the protective option slot cover. Remove the cover by squeezing the pinch tabs and lifting the cover off.



Now, insert the module into the open slot on the DL05 or into any one of the four slots in the DL06. Locate the module so the printed information is oriented in the same direction as the markings on the PLC. Be careful to align the female connector on the printed circuit board of the module with the male connector on the PLC mother board. Press the module into the slot until the front of the module is flush with the front of the PLC. Check the DL06 power budget to be sure that it remains within the power supply limits before installing more modules.

DL205 Slot Choices

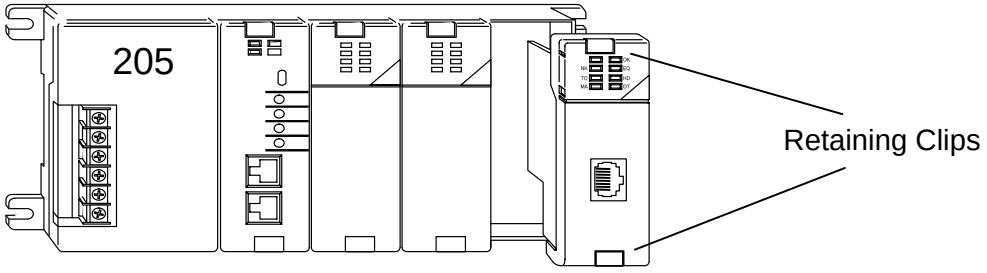
The DL205 system supports placement of the ECOM module in the **CPU-base *only***, not in local expansion bases or remote I/O bases. The number of usable slots depends on how many slots your base has. The module does not work in slot 0 of the DL205 series PLCs, the slot next to the CPU. The D2–240, D2–250–1 and D2–260 CPUs support the ECOM modules. The D2–230 CPU does not support the ECOM modules.



WARNING: Your system can be damaged if you install or remove system components before disconnecting the system power. To minimize the risk of equipment damage, electrical shock, or personal injury, always disconnect the system power before installing or removing any system component.

Module Type	CPU	CPU-Base	Usable Slots
H2–ECOM H2–ECOM100 H2–ECOM–F	DL240 DL250–1 DL260	D2–03B–1, D2–03BDC1–1, D2–03BDC–2	1
		D2–04B–1, D2–04BDC1–1, D2–04BDC–2	1, 2
		D2–06B–1, D2–06BDC1–1, D2–06BDC2–1	1, 2, 3, 4
		D2–09B–1, D2–09BDC1–1, D2–09BDC2–1	1, 2, 3, 4, 5, 6, 7

H2 Series ECOM Module Installation



To install the ECOM module, line up the module's printed circuit board with the grooves in the base and push the module until it is flush with face of the DL205 base power supply. If you feel more than moderate resistance when you push the module into the base, the circuit board may not be aligned with the grooves in the base. When the module is firmly seated in the slot, depress the top and bottom retaining clips to lock the module in place.

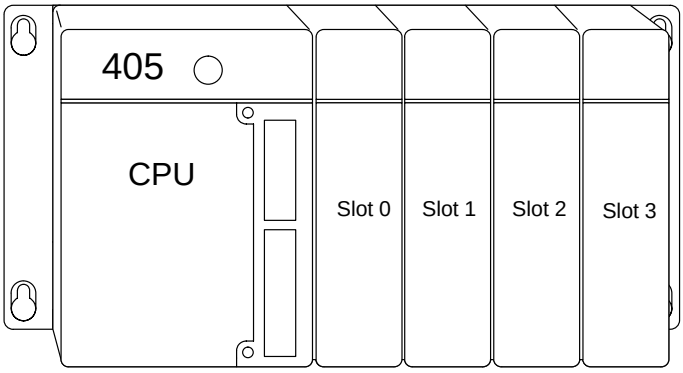


NOTE: When adding modules to your PLC always confirm that your **power budget** will accommodate the added module. See the User Manual for your PLC for more information about calculating the power budget. See Appendix A for the power consumption of the ECOM modules.

DL405 Slot Choices

For PLC systems with D4–430 and D4–440 CPUs, the ECOM modules can reside in any I/O slot but **only** in the CPU-base. The D4–450 CPU allows the installation of the ECOM module in the CPU-base or in **local expansion** bases.

If the ECOM module is used in a local expansion base, *all bases in the system* must be the "–1" type bases. The valid part numbers for these bases are D4–04B–1, D4–06B–1, and D4–08B–1. The "–1" on the end of the part number indicates that the base supports specialty modules including the ECOM. The "–1" bases can be connected as **local expansion** bases or **remote** bases. They are not the same thing. Remote bases **do not** support the ECOM modules.



WARNING: Your system can be damaged if you install or remove system components before disconnecting the system power. To minimize the risk of equipment damage, electrical shock, or personal injury, always disconnect the system power before installing or removing any system component.

Module Type	CPU	Base	Usable CPU-Base Slots	Usable Expansion Base Slots
H4–ECOM H4–ECOM100 H4–ECOM–F	D4–430/440	D4–04B, D4–04B–1	0, 1, 2, 3	N/A
		D4–06B, D4–06B–1	0, 1, 2, 3, 4, 5	N/A
		D4–08B, D4–08B–1	0, 1, 2, 3, 4, 5, 6, 7	N/A
	D4–450	D4–04B	0, 1, 2, 3	N/A
		D4–06B	0, 1, 2, 3, 4, 5	N/A
		D4–08B	0, 1, 2, 3, 4, 5, 6, 7	N/A
	D4–450	D4–04B–1	0, 1, 2, 3	0, 1, 2, 3*
		D4–06B–1	0, 1, 2, 3, 4, 5	0, 1, 2, 3, 4, 5*
		D4–08B–1	0, 1, 2, 3, 4, 5, 6, 7	0, 1, 2, 3, 4, 5, 6, 7*

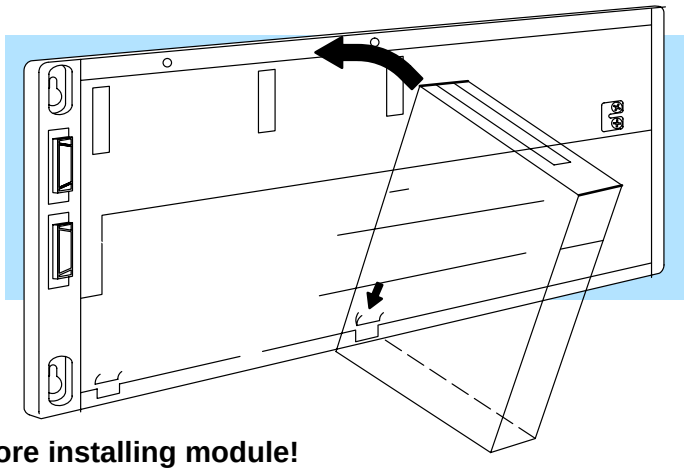
* You must use the "–1" base for the CPU-base and all local expansion bases.

NOTE: Before installing the ECOM module, confirm that your **power budget** will accommodate the added module. See the **DL205** or **DL405 User Manual** for your PLC for more information about calculating the power budget. See Appendix A for the power consumption of the ECOM modules.

H4 Series ECOM Module Installation

To insert the ECOM module in a DL405 base, place the bottom tab of the module into the notch at the bottom of the base. Pivot the module toward the base as shown below. Ensure that each module is tightly seated and secured with the captive screw at the top of the module.

DL405 Base



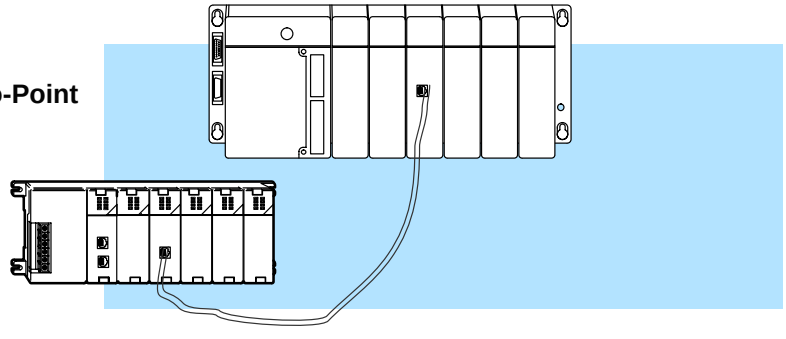
Disconnect power before installing module!

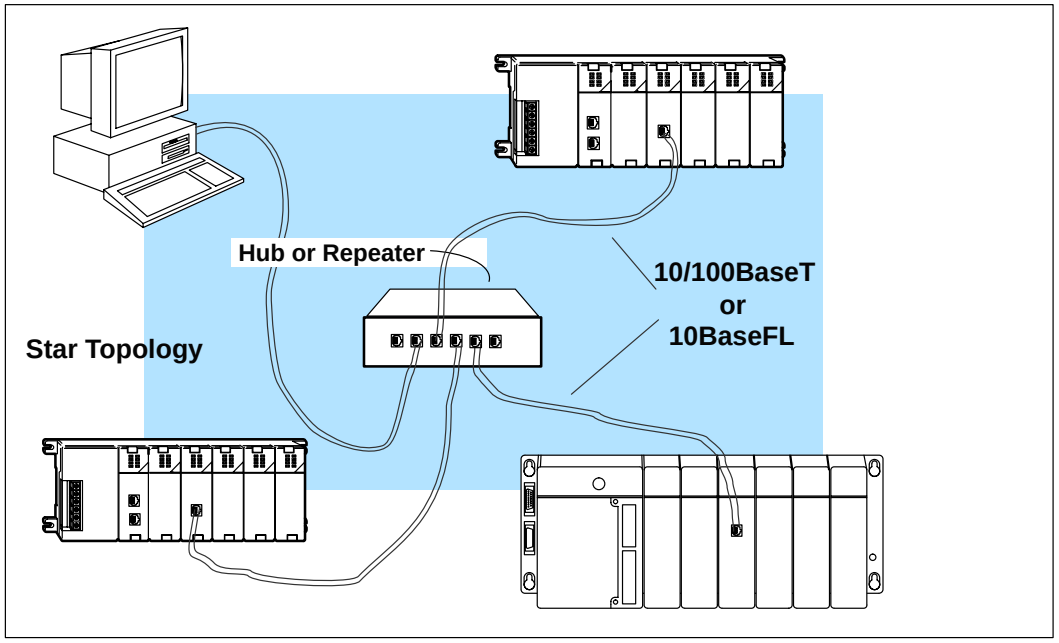
ECOM Network Layouts

The ECOM Ethernet network is a peer-to-peer network. Using Read (RX) or Write (WX) instructions, any PLC on the network can initiate communications with any other PLC on the network. A PC running our KEP***Direct*** for PLCs software can also initiate communications with any ECOM that is on the same network, but a PLC cannot initiate communication with the PC. A PLC cannot literally broadcast to all other PLCs at the same time, but a PLC can sequence through communication connections with each other PLC on the network, one at a time.

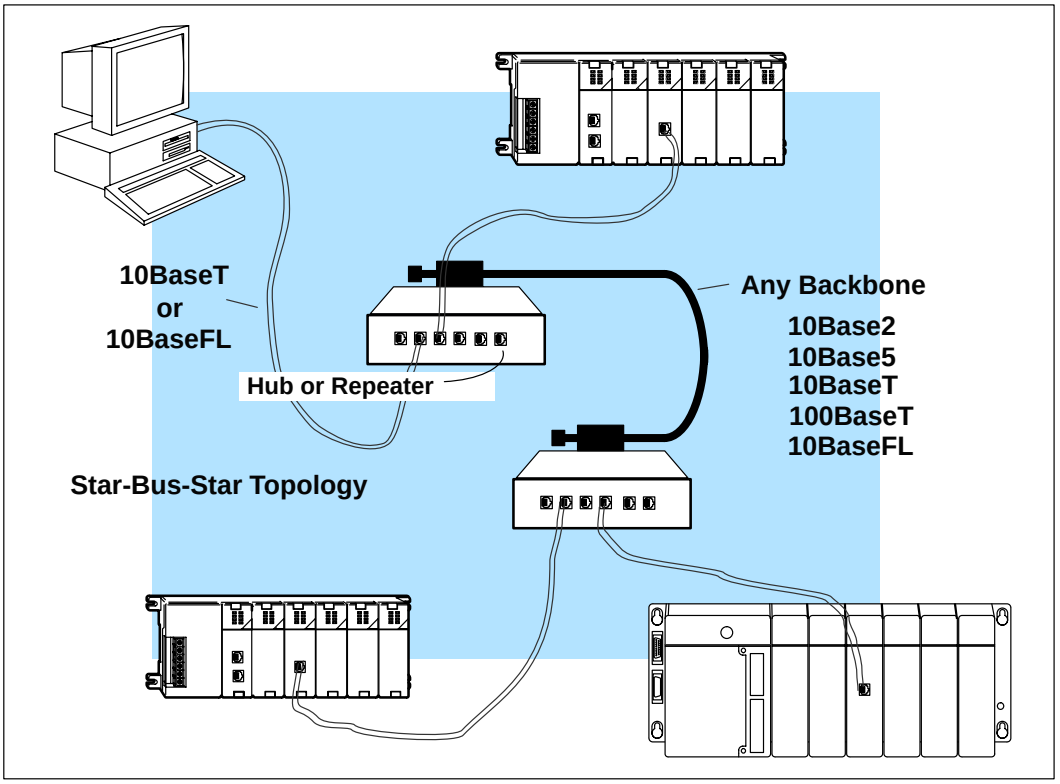
The ECOM products inherently support two network layouts: point-to-point and star. The point-to-point layout can be used to link together two PLCs or a PC and a PLC. A hub or repeater connects multiple networkable devices into a star topology. Multiple hubs or repeaters are used to modify the star topology so that it becomes a star-bus-star topology. See the figures below and on the next page.

Point-to-Point





Hubs or repeaters can connect together to make it possible to connect more devices to the network or to extend the range of the network.



NOTE: Hubs or repeaters often designate one port for *uplink* to another hub. This port may not be able to be used to connect to a PLC. If the uplink port is used to connect to another hub, it may disable the adjacent port. Use of the uplink port may require the use of a crossover cable.

Network Cabling

ECOM Supports Two Standards

Two types of ECOMs are available. One type supports the 10/100BaseT cabling standard, and the other supports the 10BaseFL connection standard. The 10/100BaseT standard uses twisted pairs of copper wire conductors, and the 10BaseFL standard is for fiber optic cabling.

10/100BaseT

Unshielded Twisted-Pair cable with **RJ45** connectors



10BaseFL

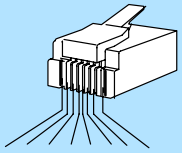
62.5/125 MMF fiber optics cable with **ST-style** connectors



10/100BaseT Networks

The cable used to connect a PLC (or PC) to a hub or repeater is called a **patch** (straight-through) **cable**. The cable used to connect together two PLCs, or a PC and a PLC or two hubs is a **crossover cable**. We recommend that you purchase cables pre-assembled with connectors for convenient and reliable networking.

10/100BaseT



1 2 3 4 5 6 7 8

8-pin RJ45 Connector (8P8C)

Patch (Straight–through) Cable

TD+	1	OR/WHT	OR/WHT	1	TD+
TD–	2	OR	OR	2	TD–
RD+	3	GRN/WHT	GRN/WHT	3	RD+
	4	BLU	BLU	4	
	5	BLU/WHT	BLU/WHT	5	
RD–	6	GRN	GRN	6	RD–
	7	BRN/WHT	BRN/WHT	7	
	8	BRN	BRN	8	
RJ45					RJ45

Crossover Cable

TD+	1	OR/WHT	GRN/WHT	1	TD+
TD–	2	OR	GRN	2	TD–
RD+	3	GRN/WHT	OR/WHT	3	RD+
	4	BLU	BLU	4	
	5	BLU/WHT	BLU/WHT	5	
RD–	6	GRN	OR	6	RD–
	7	BRN/WHT	BRN/WHT	7	
	8	BRN	BRN	8	
RJ45					RJ45

This diagram illustrates the standard wire positions in the RJ45 connector. We recommend all ECOM 10/100BaseT cables to be **Category 5**, UTP cable.

10/100BaseT Connections

Most 10/100BaseT hubs or repeaters use a **patch** (straight-through) cable for connecting the network devices (PLCs or PCs). For hub-to-hub connections a **crossover** type cable is commonly required. The figures on the previous page show pin assignments and insulation color codes for patch (straight-through) and crossover type Ethernet cables.

UTP Cable

The ECOM has an eight-pin modular port that accepts RJ45 type connectors. UTP (Unshielded Twisted-Pair) cable is rated according to its data-carrying ability (bandwidth) and is given a “category” number. We strongly recommend using a category 5 cable for all ECOM connections.

NOTE: See page 2–14 for 10/100BaseT distance limitations.

10BaseFL Connections

Each module has two bayonet ST-style connectors. The ST-style connector uses a quick release coupling which requires a quarter turn to engage or disengage. The connectors provide mechanical and optical alignment of fibers.

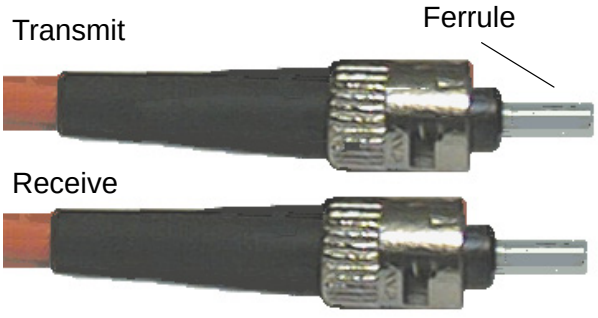
Each cable segment requires two strands of fiber: one to transmit data and one to receive data. The ST-style connectors are used to connect the H2–ECOM–F or H4–ECOM–F module to another H2–ECOM–F or H4–ECOM–F module or a fiber optic hub or repeater.

Fiber Optic Cable

The H2–ECOM–F and H4–ECOM–F modules accept 62.5/125 multimode fiber optic (MMF) cable. The glass core diameter is 62.5 micrometers and the glass cladding is 125 micrometers. The fiber optic cable is highly immune to noise and permits communications over much greater distances than 10BaseT.

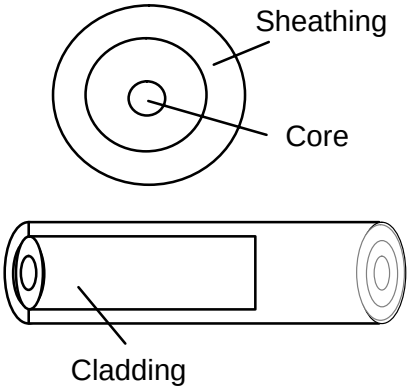
Fiber Optic Module ST Connector

Multimode Fiber Optic (MMF) Cable

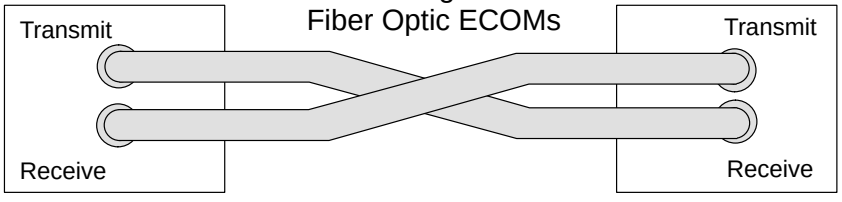


62.5/125 MMF cable with bayonet ST-style connectors

Fiber cross-section



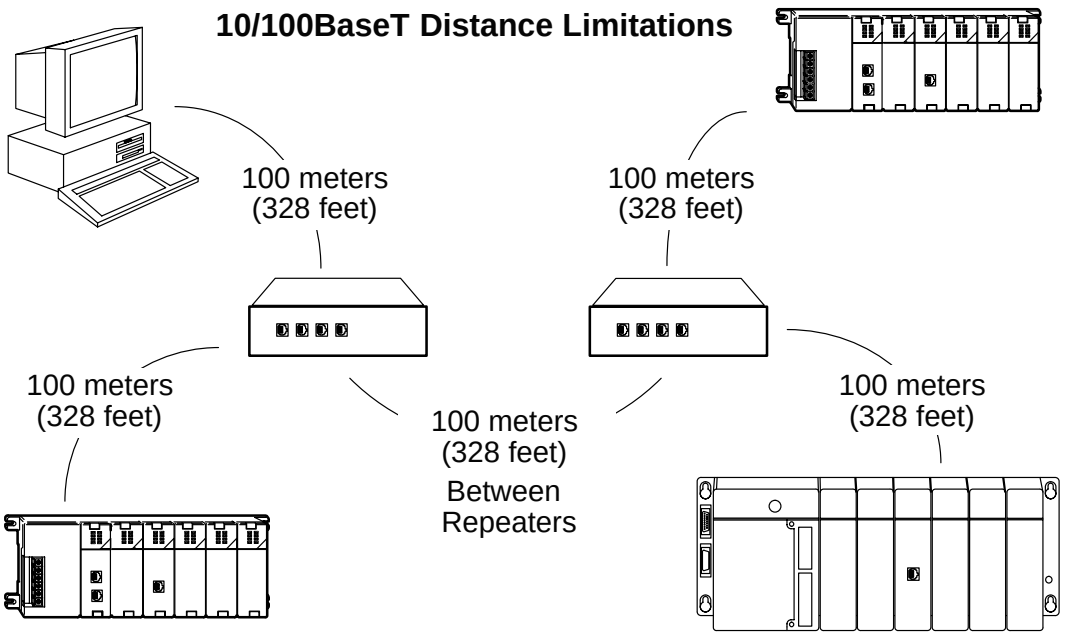
Connecting Two Fiber Optic ECOMs



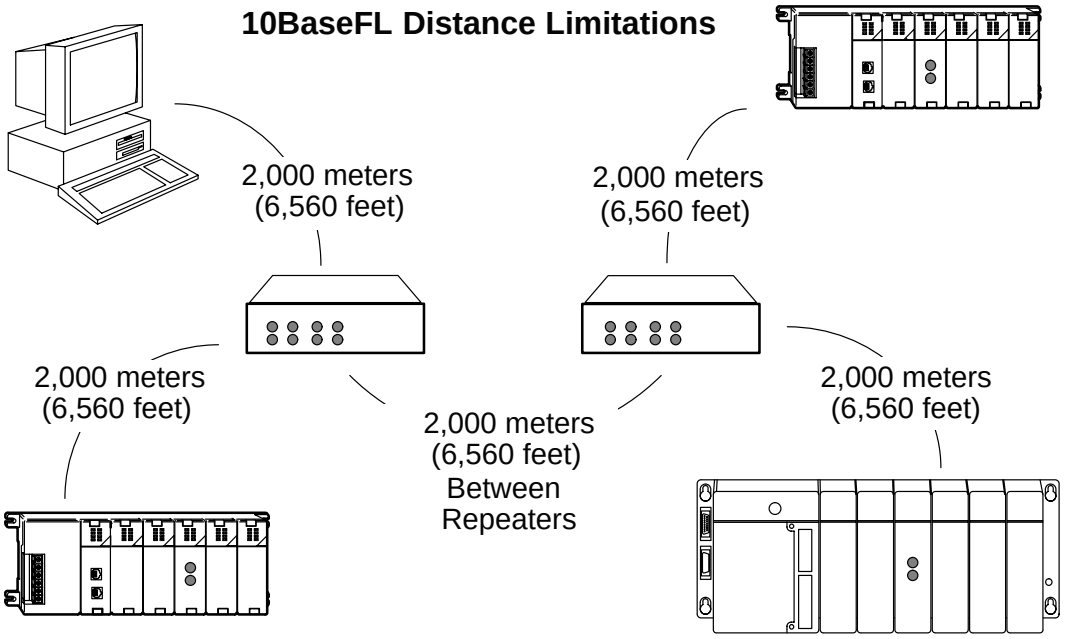
NOTE: See page 2–14 for 10BaseFL distance limitations.

Maximum Cable Length

The **maximum distance** per **10/100BaseT** cable segment is **100 meters** or **328 feet**. Repeaters extend the distance. Each cable segment attached to a repeater can be 100 meters. Two repeaters connected together extend the total range to 300 meters.



The **maximum distance** per **10BaseFL** cable segment is **2,000 meters** or **6,560 feet**. Repeaters extend the distance. Each cable segment attached to a repeater can be 2,000 meters. Two repeaters connected together extend the total range to 6,000 meters.



Maximum Number of ECOM Modules on the Network

The maximum number of nodes that can be connected to a 10/100BaseT or 10BaseFL network is a function of the topology used in constructing the network. Therefore, it is not possible to state an absolute maximum number of nodes that would apply in all cases.

The IEEE 802.3 specification defines the maximum node limit for an Ethernet *segment* in terms of the ability to detect and avoid data collisions. A “legal” network can have any number of devices provided that they can:

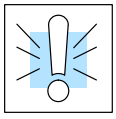
- detect all data collisions that may occur during the communication process *and*
- respond to these collisions appropriately.

You must take into consideration the network limitations imposed by all cabling and network devices. Consider the limitations imposed on your network if your network uses:

- a combination of cabling standards, such as 10/100 BaseT and 10Base2, or
- intermediate devices, such as switches or routers.

Each ECOM module can be assigned a Module ID ranging from 1 to 999,999,999. Theoretically, you could have this many Ethernet modules coexisting on a single network. Other network limitations would restrict the network size before reaching this limit. For the majority of network PLC applications there is practically no limit to the number of ECOM modules you can access from the NetEdit3, ***Direct***SOFT32 Programming Software or the KEP***Direc***t for PLCs software.

There *is* a node limit for PLC-to-PLC communications. The network Read and Write instructions performed by the initiating (master) PLC are only capable of accessing PLCs with Module IDs of 1 through 90. This effectively sets the maximum number of nodes available for PLC-to-PLC communications at 90.



WARNING: We recommend against connecting Ethernet modules to the same network that serves as your primary office network. While Ethernet networks can handle a very large number of data transmissions, and normally handle them very quickly, heavy Ethernet traffic can adversely affect the reliability and speed of the network.

3

Configuring ECOMs Using NetEdit3

In This Chapter. . . .

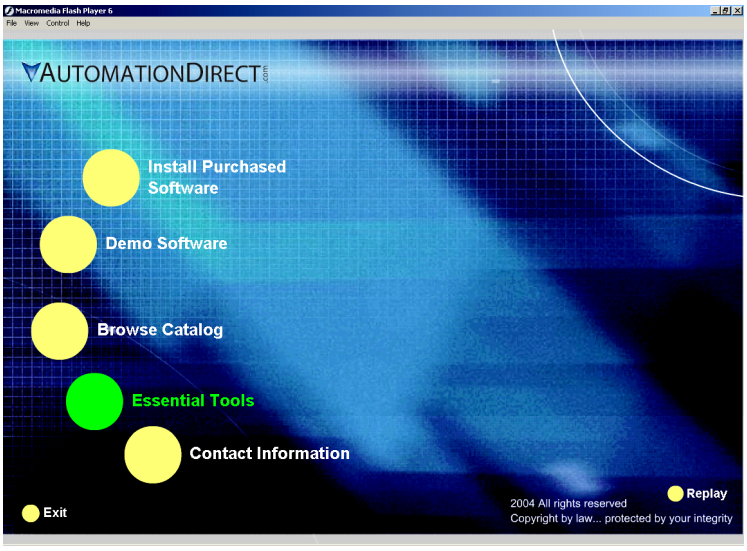
- NetEdit3 Software
- Using NetEdit3
- Locating the MAC Address Label

NetEdit3 Software

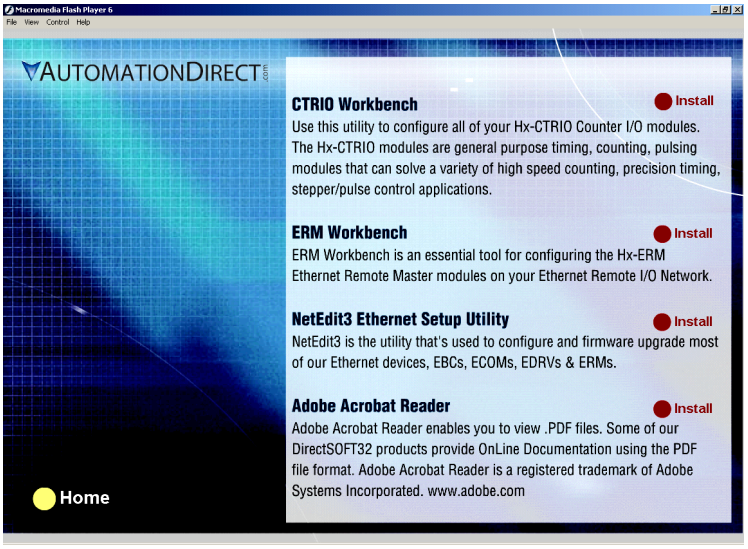
NetEdit3 is a software utility which can be used to set network identifiers (Module ID or IP Address), perform diagnostic and troubleshooting tasks and upgrade the firmware in the ECOM module if necessary. The H0/H2/H4–ECOM100 requires NetEdit 3.5 or later.

Installing NetEdit3

You can install NetEdit3 on Windows98/ME/2000/XP™ or Windows NT4™. NetEdit3 is included with this manual on the AutomationDirect Software Product Showcase CD (also available online at www.automationdirect.com). After inserting the CD into the drive, the following window will appear.



Click on the Essential Tools button. The following window will be displayed.

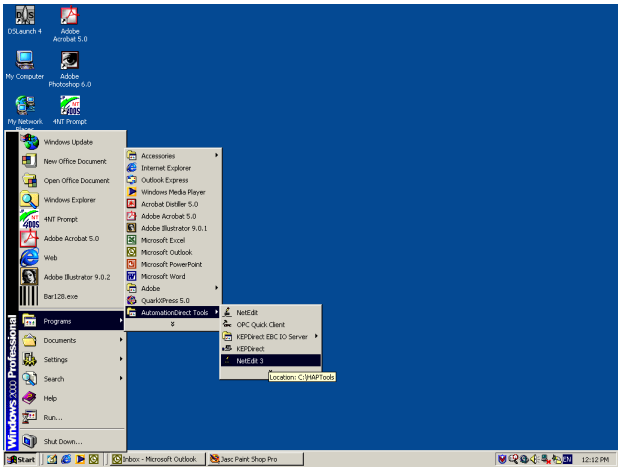


Click on Install NetEdit3. A series of windows will step you through the installation process. Fill in the necessary information as the installation wizard prompts through the install. In the Setup Type window, select Typical setup. This setup type is recommended for most users. The installation process places NetEdit3 in the C:\HAPTools directory (default).

Launching NetEdit3

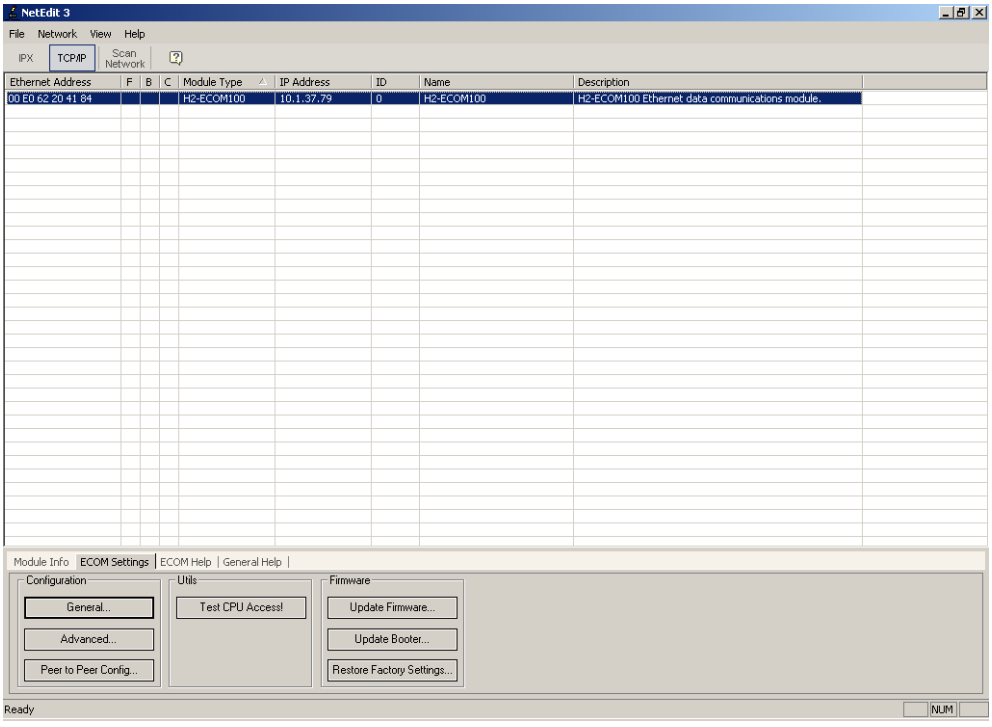
There are three methods to launch NetEdit3.
The three methods are:

- using the Windows Start menu Programs>AutomationDirect Tools> NetEdit3 as shown below
- launching ***Direct***Soft (if installed), from the programming window, select PLC>Tools>NetEdit3
- launching ***Direct***Soft (if installed), then select Utilities>NetEdit3



The NetEdit3 Screen

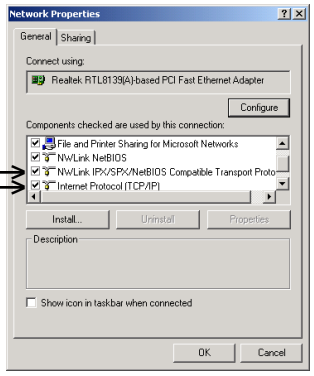
Starting NetEdit3 brings up the screen below. All NetEdit3 functions are accessed from this screen.



Adding Network Protocol Support to the NetEdit3 PC

You may have already set up your PC with selected networking protocols for Ethernet communications. If not, you will need to select the protocols now for communication with the Ethernet modules. We strongly recommend that you include the IPX protocol. For Windows 2000, go from My Computer on your Windows desktop to Control Panel. Double click on Network and Dial–up Connections, then double click on the desired Network Device to see the installed Protocols. If IPX is not listed among the protocols already loaded, add it now by clicking on the Install button. For Windows XP, go from Start>Settings>Control Panel. The steps are the same as Windows 2000 from this point.

Add the TCP/IP protocol if it is necessary for your application. The TCP/IP selection will give you support for the UDP/IP protocol. Also, add the IPX protocol if it is not already active.





NOTE: We strongly recommend you load IPX protocol on your PC for the connection from your PC to the ethernet modules. Use UDP/IP in your application, if required, but also add IPX to your list of active protocols. Having IPX loaded on your PC gives you a backup for troubleshooting communication problems.

Using NetEdit3

This section steps through the features and uses of NetEdit3. We will describe the individual segments of the NetEdit3 screen and the function of each.

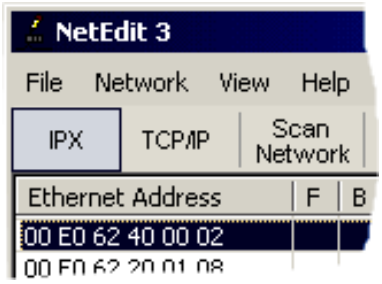
NOTE: Your PC-based Control software may be capable of configuring the EBC module. If so, please refer to the appropriate documentation for that software product to determine the best method to configure the EBC. Depending on which software you are using, it may not be necessary to use NetEdit3.

Ethernet Communication Protocol

In the upper left corner of the NetEdit3 screen, you will find buttons labeled *IPX* and *TCP/IP*. The ECOM module understands these protocols. Both protocols are *permanently resident* in the firmware of the module.

When you click on one of these buttons, you are selecting the protocol you want your PC to use to communicate with the ECOM module. You are not telling the module which protocol to use, because it is using both protocols all the time. IPX is a Novell standard in widespread use, and UDP/IP is a popular protocol supported by the TCP/IP suite of protocols in your PC.

The figure to the right shows the Protocol selection buttons in the upper left corner of the NetEdit3 screen. The choice you make here tells **your PC** which protocol to send to the ECOM to link NetEdit3 to the module.



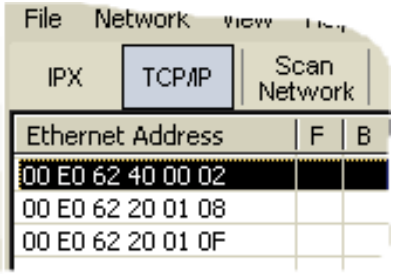
Some PC-based control software products may support only one of these protocols. Read the documentation for your software to be sure the protocol you select is supported.

Ethernet Address

The upper left section of the NetEdit3 screen displays the *Ethernet Address* of the modules currently on the network.

If modules are added or removed from the network, click on the *Scan Network* button to update the list. Notice that the MAC Address is the factory-assigned address that is on the permanent label on the module.

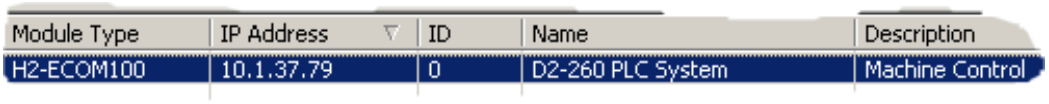
Select a specific module here by clicking on the MAC Address or by using the arrow keys. The selected module is highlighted.





NOTE: The Module window may list the MAC Addresses of devices not covered by this manual.

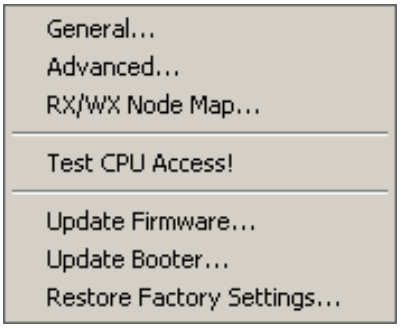
Module Type, IP Address and ID



The upper mid section of the NetEdit3 screen displays the *Module Type, IP Address,* module *ID, Name* and *Description* of the modules currently on the network.

A new ECOM will have an IP Address of 0.0.0.0, a Module ID of 0 (zero), and no Module Name or Description. To assign or change a module an IP address, ID, name or description refer to the EBC Settings>General Information description later in this section.

Right clicking on an ECOM module listed on the NetEdit3 screen will display the window to the right. This is an alternative to using the Module Info or ECOM settings tabs (shown below) to access the module's configuration settings. The settings are discussed later in this section.

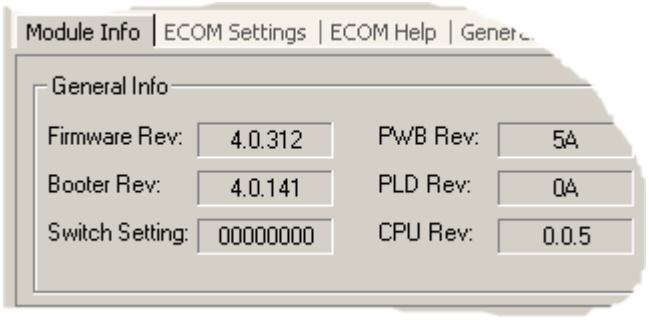




Module Info> General Information

When the Module Info tab is selected, the *General Info* box lists the selected module's Firmware Revision, Booter Revision, DIP Switch Setting, PWB Revision, PLD Revision and CPU Revision.

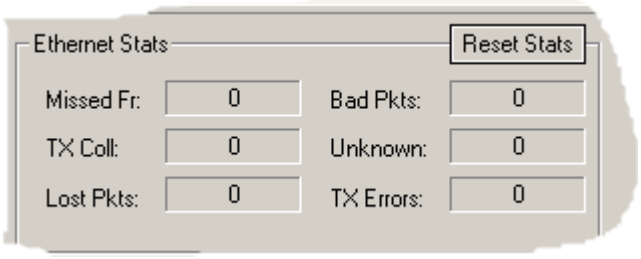
This box is in the lower left section of the NetEdit3 screen.



Module Info> Ethernet Stats

When the Module Info tab is selected, the *Ethernet Stats* box displays statistics related to the selected module's communication errors. Click on the Reset Stats button to reset all categories to 0 (zero).

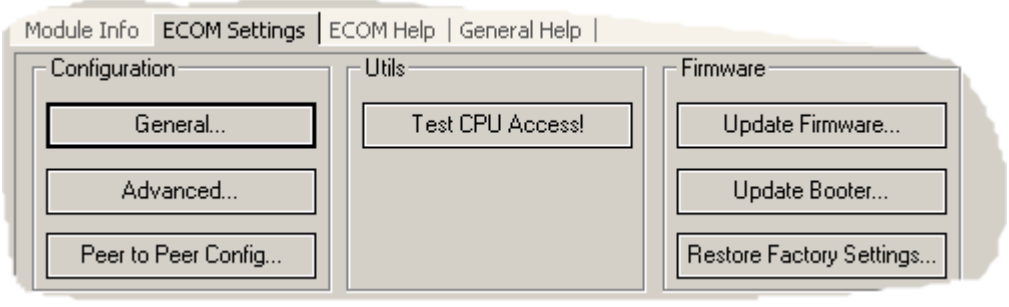
This box is in the lower middle section of the NetEdit3 screen.



ECOM Settings

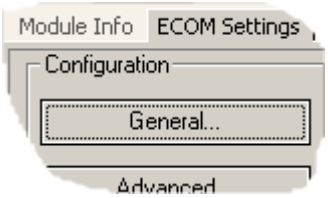
When the ECOM Settings tab is selected, the selected module's Configuration, Utilities and Firmware tools can be accessed.

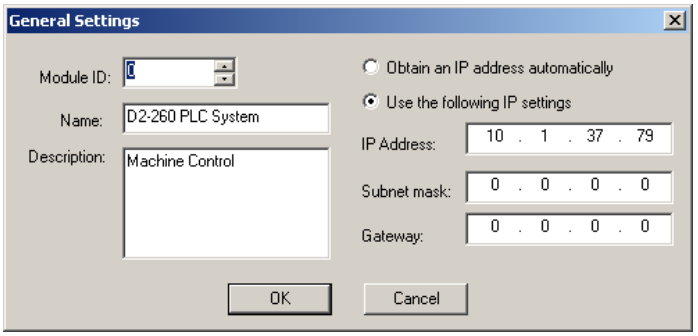
This box is in the lower middle section of the NetEdit3 screen.



ECOM Settings> Configuration> General

Clicking the General button in the ECOM Settings>Configuration box brings up the General Settings window below.





The General Settings box allows you to assign a **Module ID**. Module IDs must be unique for each ECOM, but they do not have to be in sequence. The module's DIP switches must be set to zero to allow NetEdit3 to set a Module ID. Do not use address zero for communications.

The **Name** field and **Description** field are optional.

To set an **IP Address**, highlight the number in each of the four boxes, and overwrite the number. Use the twelve-digit number assigned to the ECOM module by your network administrator. If you change the IP Address, do not use the number "255" in any field. Doing so will cause communication problems.

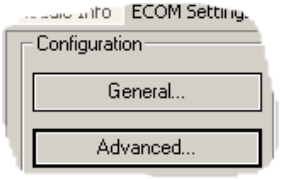
The OK button sends all the entries to the module's flash memory.

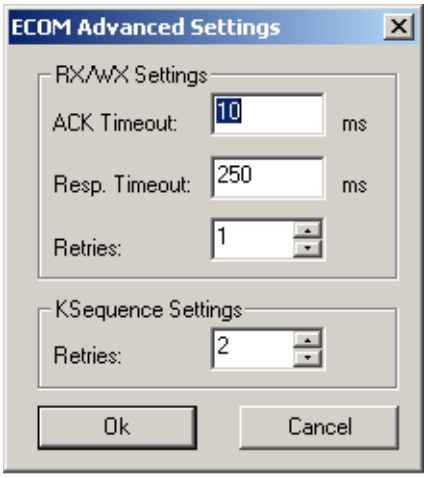
The following table summarizes NetEdit3's Network Identifiers and their uses:

	How to Set	Format	Communication	Restrictions/Notes
Module ID	DIP Switch	Number 1-63	PLC-to-PLC or PC-to-PLC	Disables Module ID in NetEdit3
	NetEdit3	Number 1-90	PLC-to-PLC or PC-to-PLC	DIP Switch must be set to "0"
	NetEdit3	Number 1-999,999,999	PC-to-PLC Only	> 90 (Not for PLC-to-PLC)
Name	NetEdit3	32 Alphanumeric Characters	PC-to-PLC Only	HMI Software may have restrictions
IP Address	NetEdit3	4 Three-digit Numbers xxx.xxx.xxx.xxx (See Page 2-4)	PC-to-PLC; (PLC to PLC–Client/ Server using TCP/IP or MODBUS TCP/IP protocol)	See Your Network Administrator for IP address; (refer to pages 3–10 to 3–11, chapters 5 and 6)
Ethernet (MAC) Address	Set at Factory	12 Hex digits	PC-to-PLC Only	Factory-assigned, for IPX

ECOM Settings> Configuration> Advanced

Clicking the Advanced button in the ECOM Settings>Configuration box brings up the ECOM Advanced Settings window below.





RX/WX – Settings box in this window provides a place to make changes that affect PLC-to-PLC communications.

ACK Timeout – sets the time limit for receiving the acknowledge (ACK) response to an RX or WX instruction. The ECOM sends a message across the LAN. The acknowledge response comes back directly from the ECOM module receiving the transmission. This timeout is the maximum for transmission and acknowledgement from ECOM to ECOM across the LAN. It is not dependent on the PLC scan time.

Resp. Timeout – sets the maximum time for the receiving PLC CPU to respond to the ECOM that initiated the communication. The instruction has travelled from the initiating PLC CPU to the initiating ECOM, across the Ethernet LAN to the receiving ECOM, then to the receiving PLC CPU, and back again to the initiating ECOM. Multiple PLC scans may be required to execute an RX/WX instruction so the Resp. Timeout should allow for multiple scans. Also, communication errors may result in retries which require more time. Your response timeout setting must accommodate retries as well.

The WX/RX **Retries** field shows the number of times the ECOM will retry transmissions to the network.

The **K–Sequence Retries** field shows the number of times the backplane transmission between ECOM and CPU is to be retried.

The OK button loads the communication settings in the ECOM module's flash memory.

ECOM Settings> Configuration> Peer to Peer

This function allows you to configure the client H0/H2/H4–ECOM(100) module to use an IP protocol packet to perform the RX/WX instruction, as opposed to a broadcast packet to locate the slave ECOM or MODBUS TCP/IP server.

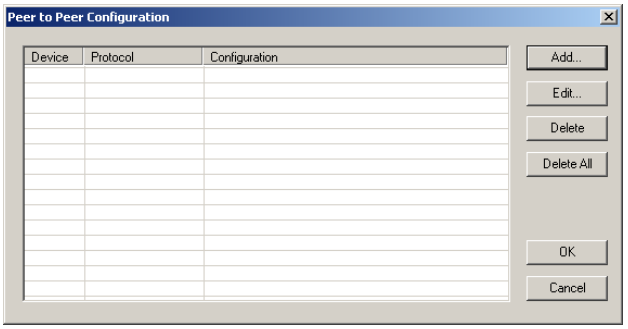
Chosing Peer to Peer: The following are reasons or conditions of when to configure the H0/H2/H4–ECOM(100) for peer to peer communications with a TCP/IP server device:

- To eliminate network broadcast traffic when the ECOM(100) is a client executing RX/WX network instructions.
- When an ECOM(100) is an “ECOM protocol client” and needs to access other server ECOMs through a router.
- When an ECOM100 needs to serve as Modbus TCP/IP client

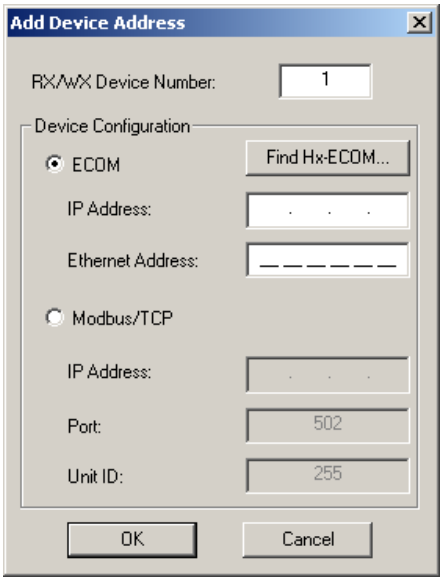
Clicking the Peer to Peer Config button in the ECOM Settings>Configuration box brings up the Peer to Peer Configuration window shown below. Any previous peer to peer configurations will be listed in the table.



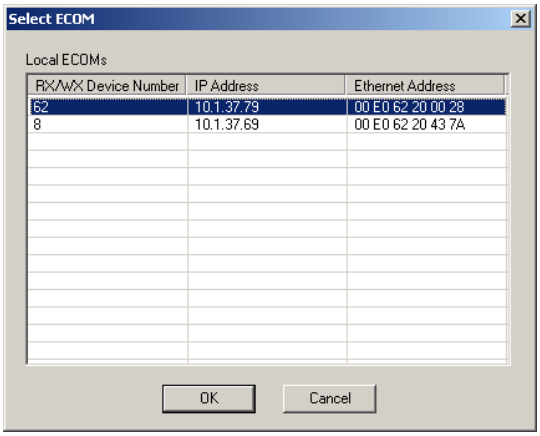
The RX and WX ladder logic instructions used by the PLC allow you specify a Node (Device) Number as the slave identifer in the range of 1–87. The Peer to Peer Configuration information is stored in the ECOM(100) module’s flash memory. This basically associates a Device Number (that is used by the RX/WX instructions) with a server’s IP address.



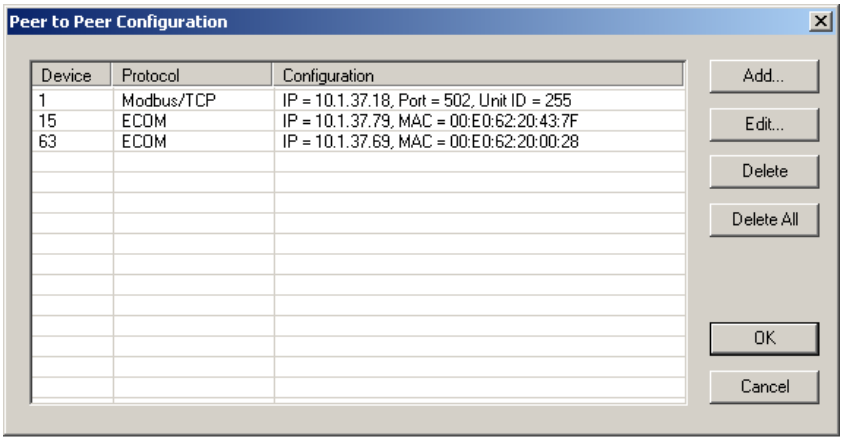
Clicking on the *Add* button in the Peer to Peer Configuration window will display the Add Device Address window shown to the right. Enter the necessary server’s network information (i.e. Device Number, server’s IP address and protocol). You can manually enter the information or you can click on the *Find Hx–ECOM* button to display all **local** ECOMs currently on the network. MODBUS TCP/IP device’s network information will have to be entered manually. Refer to page 6–6 for information on Port and Unit ID Numbers. Clicking the OK button adds the Device Configuration to the main peer to peer configuration screen. Be certain not to duplicate Device Numbers.



Clicking on the *Find Hx–ECOM* button in the Add Device Address window will display the Select ECOM window shown to the right. Highlight the server ECOM that you intend to communicate with peer to peer via the client ECOM(100) module. Then click the OK button. This will add the ECOM's Device Configuration information to the previous Add Device Address window.

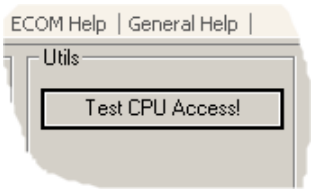


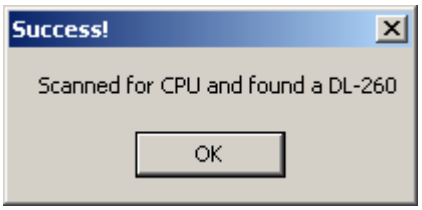
Once the module's network information is in the Add Device Address window (previous page), click the OK button to add the information to the main Peer to Peer Configuration window. Repeat the steps to add other ECOMs to the configuration. Clicking the OK button on the main screen window writes the module(s) network information to the ECOM(100)'s flash memory.



ECOM Settings> Utils>Test CPU Access

Clicking the Test CPU Access button in the ECOM Settings>Utilities box brings up the window shown below.

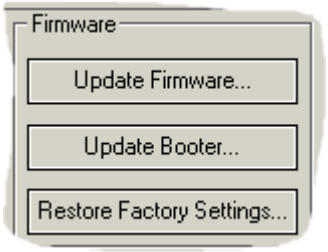




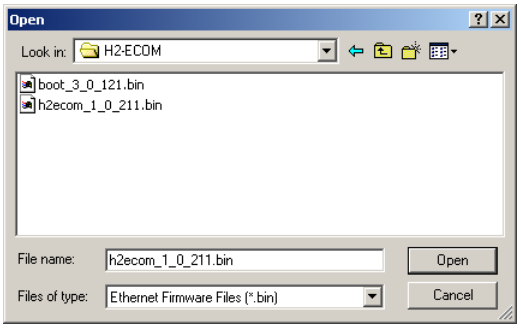
The ECOM has two distinct “sides”, the network side, which contains the network configuration settings, and the backplane side, which the ECOM uses to communicate with the PLC CPU. It is possible that only one side could be operating correctly, while the other side is not. The **Test CPU Access** function will test the connection from your PC to the ECOM module through the network side, across the backplane to the PLC CPU and back to your PC.

ECOM Settings> Firmware

The functions in the Firmware box are used to update the selected module’s Firmware and Booter versions. The Restore Factory Settings buttons resets the selected module’s IP address, ID, Name and Description to factory defaults. Refer the sections on the next page to determine if updates are necessary.

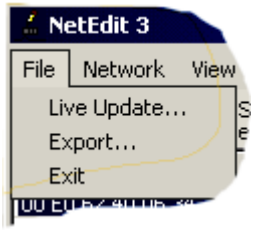


Clicking on the either of the Update buttons opens the appropriate ECOM folder within the Images folder, which is created during the install of NetEdit3. The Images folder is located in the same folder as NetEdit3.exe. Each module folder contains the module’s firmware and boot loader files. The next section discusses keeping the firmware files up to date.

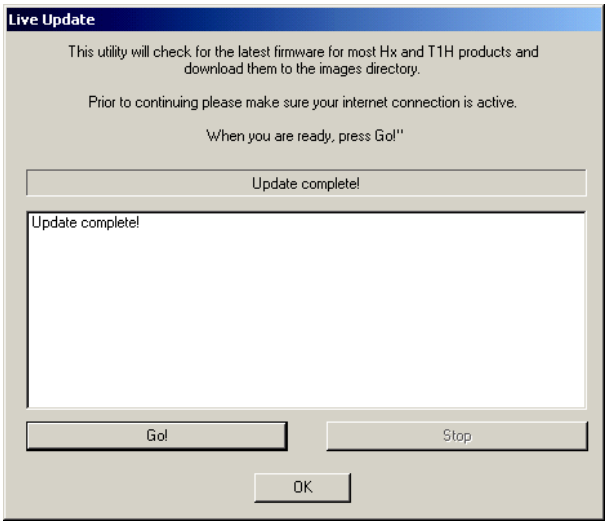


FileMenu>
Live Update

The Live Update will retrieve the latest firmware and boot loader files from the Host Engineering web site and place them in the NetEdit3 Images folder that was created during the install of NetEdit3. The feature requires that you have a functional Internet connection (dial–up or broadband). If the Images folder does not exist on your PC, it will be created as part of the retrieval process.

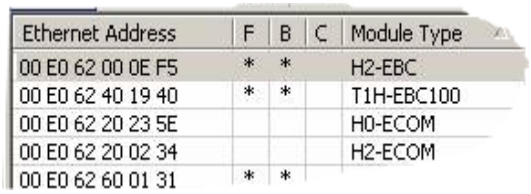


When you click the Go! button on the Live Update window, NetEdit3 will compare the version information of the files on the Host Engineering web site against the files you have locally on your PC, and it will download any newer files. Once this process is complete, NetEdit3 will rescan the devices on your network and refresh the “F” and “B” columns next to the listed devices.



F / B / C
Columns

The F, B, and C columns are provided to signify potential issues with devices on the network.



Ethernet Address	F	B	C	Module Type
00 E0 62 00 0E F5	*	*		H2-EBC
00 E0 62 40 19 40	*	*		T1H-EBC100
00 E0 62 20 23 5E				H0-ECOM
00 E0 62 20 02 34				H2-ECOM
00 E0 62 60 01 31	*	*		

The “F” column will display an asterick beside any device whose firmware is older than its firmware file in your Images folder.

The “B” column will display an asterick beside any device whose boot loader is older than its boot loader file in your Images folder.

The “C” column will display an asterick beside any device that has a configuration conflict with another device on the network. Duplicate module IDs (that are non–zero) and duplicate IP Addresses (that are not 255.255.255.255) will report as conflicts.

4

RLL Programming for Communications

In This Chapter. . . .

- PLC-to-PLC Communications
- How RLL is Used for Communications
- Network Instructions
- Addressing the Different Memory Types
- Special Relays for Communications
- Example Program with One Read Instruction
- Example Program with One Write Instruction
- Integrating Multiple Read and Write Instructions

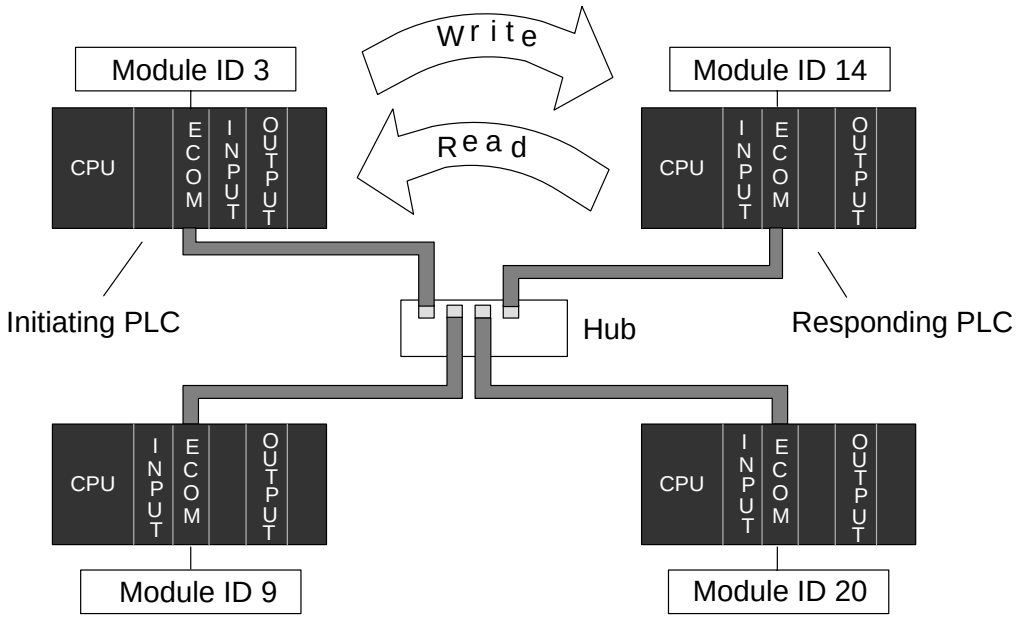
PLC-to-PLC Communications

This chapter steps you through the development of a Relay Ladder Logic (RLL) program to enable one PLC to communicate with another PLC. For the experienced programmer of ***Direct***LOGIC PLCs, the communication programs presented in this chapter will be simple to follow. If you have never programmed a ***Direct***LOGIC PLC, you may want to refer to the ***Direct***SOFT Programming Software User Manual and the User Manual for your PLC for additional information.

NOTE: The programs described in this chapter are not used for communication between a PC and a PLC. For PC-to-PLC communications, please see the product documentation for the PC software you are using. If you are using our DSData Server software, the manual you will need is the ***KEPD***irect for PLCs User Manual.

How RLL is Used for Communications

DirectSOFT Programming Software provides **Read** and **Write** instructions (RX/WX) for PLC-to-PLC communication over a network. The Read and Write instructions are part of the ladder logic program running in the CPU of the *initiating*, or master, PLC. These instructions tell the initiating CPU to send a message over the Ethernet network to a *responding*, or slave, PLC. The ECOM module is the connecting point to the network for each PLC. The initiating PLC's Read or Write communication finds its destination by the **Module ID** of the responding PLC's ECOM module. See Chapter 2 and 3 for information about assigning Module IDs.



In the figure above, the initiating PLC sends a Read or Write message to the responding PLC's ECOM module which is designated as "Module ID 14." The responding PLC processes the message. Any one of the PLCs could initiate communication with any one of the others.

Network Instructions

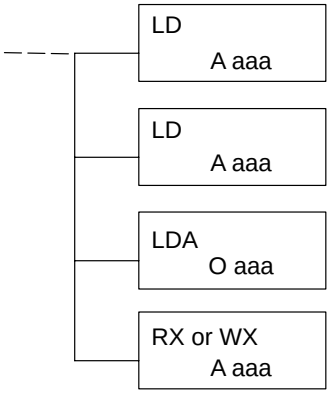
Read (RX) and Write (WX) Instructions

The Read (**RX**) and Write (**WX**) instructions are used by the initiating PLC to Read a block of data *from* another PLC or Write a block of data *to* another PLC. To perform their functions, the RX and WX *boxes* must be **preceded** in the ladder logic program by two Load instructions and one Load Address instruction.

The Load and Load Address instructions load communication parameters into the **accumulator** and the first and second level of the **accumulator stack**. The RX or WX instruction takes these parameters from the stack and the accumulator and prepares the data to be sent over the network. If you need to know more about the function of the accumulator and the accumulator stack, refer to the User Manual for your PLC.

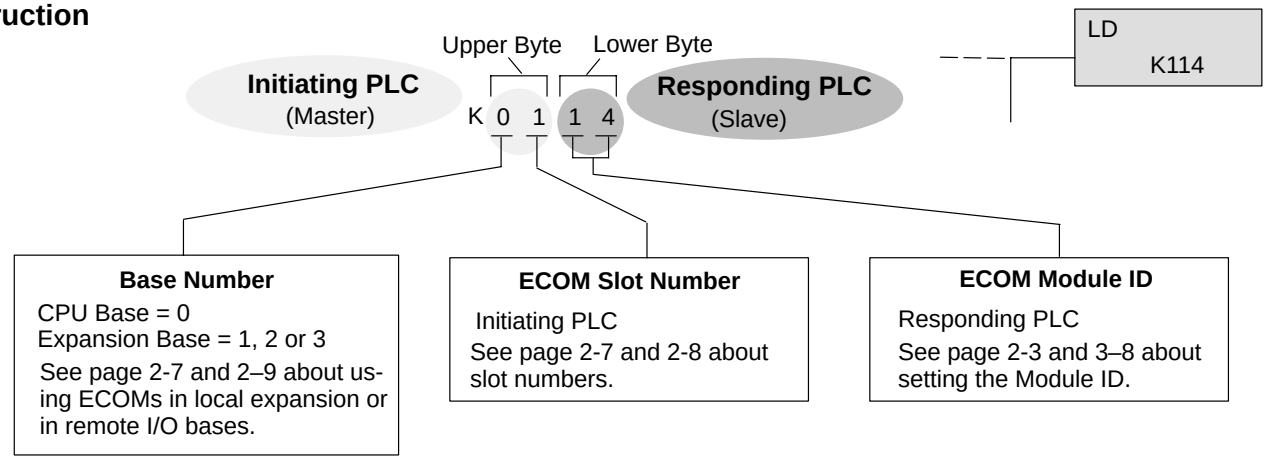
Building the Read (RX) or Write (WX) Routine

For network communications, you build the Read (RX) or Write (WX) instructions into a **routine** which requires the four instructions you see to the right. The function of each of these instructions is explained below or on the next page. They must be used in the sequence shown.



The first Load (LD) instruction accepts either a constant or a variable. Use a “K” to designate the number as a constant. Use a “V” if you are entering the address of a register. The contents of that register perform the same function as the constant shown below. For example, you could use V2000 in place of K0114. If the contents of V2000 is the number “114,” the function would be the same. Using a variable allows changing parameters while the program is running. It is recommended, however, to use a constant when possible.

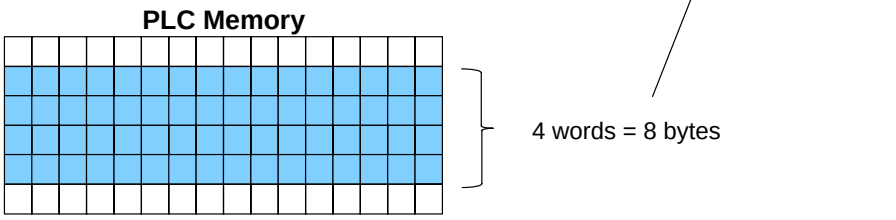
The First LD Instruction

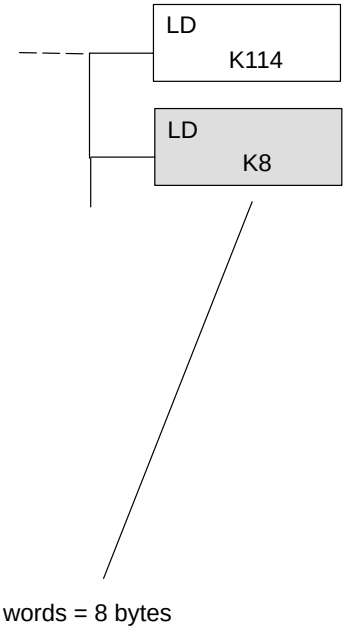


The Second LD Instruction

The second Load (LD) instruction determines the length of the data block to be transmitted during the Read or Write communication. This instruction will also accept two data types. Use a “K” to designate the number as a constant. Use a “V” if you are entering the address of a register.

For Word Memory data, you must use a multiple of two bytes between 2 and 128. For Bit Memory data, you can use any multiple of one byte between 1 and 128. For more information about addressing Word and Bit Memory, see page 4-6.



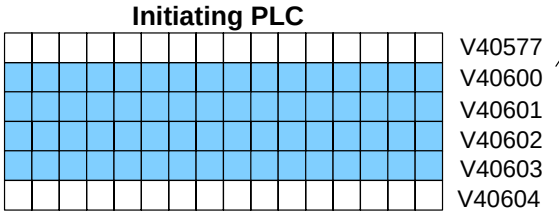


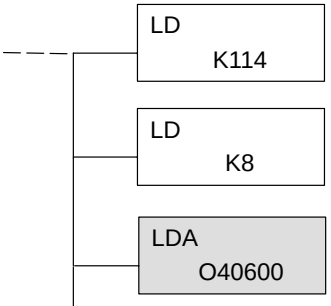
The LDA Instruction

The Load Address (LDA) instruction specifies the V-memory address of the *beginning* memory register in the **initiating**, or master, PLC. The data block to be transmitted will begin at this address and extend the number of bytes specified in the preceding LD instruction. The leading “O” indicates this is an octal number. Simply substitute the letter “O” for the “V” in the V-memory designation. For example, V40600 becomes O40600.

Read instructions copy the data block from the responding PLC memory into the initiating PLC memory.

Write instructions copy the data block from the initiating PLC memory into the responding PLC memory.

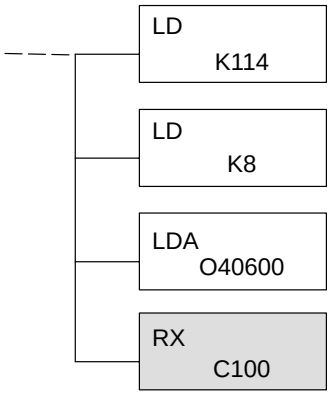


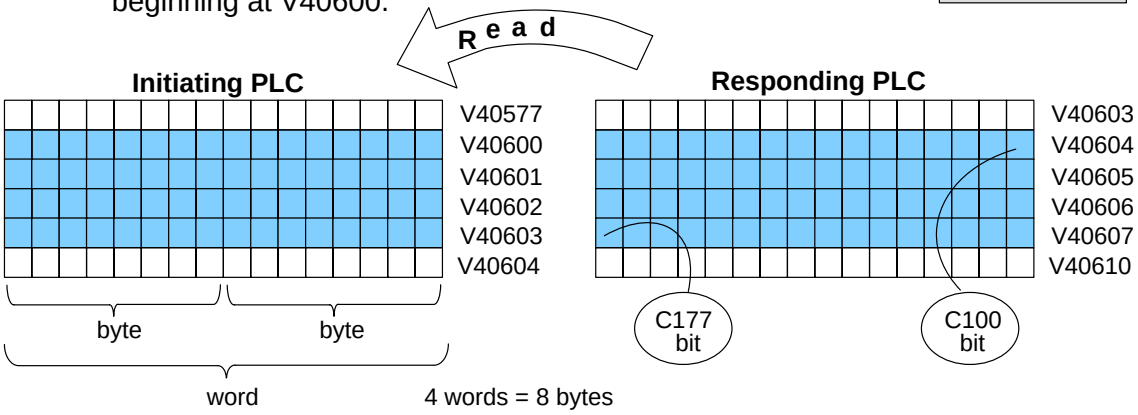


The Read (RX) instruction specifies the memory location to be *read from* the responding PLC.

A block of data is read that begins at the specified memory location and extends the number of bytes specified in the second LD instruction.

In this example, the eight byte block of data beginning at C100 and ending at C177 in the *responding*, or slave, PLC is read (copied) into the initiating PLC's memory beginning at V40600.



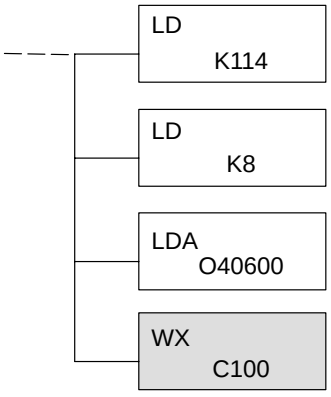


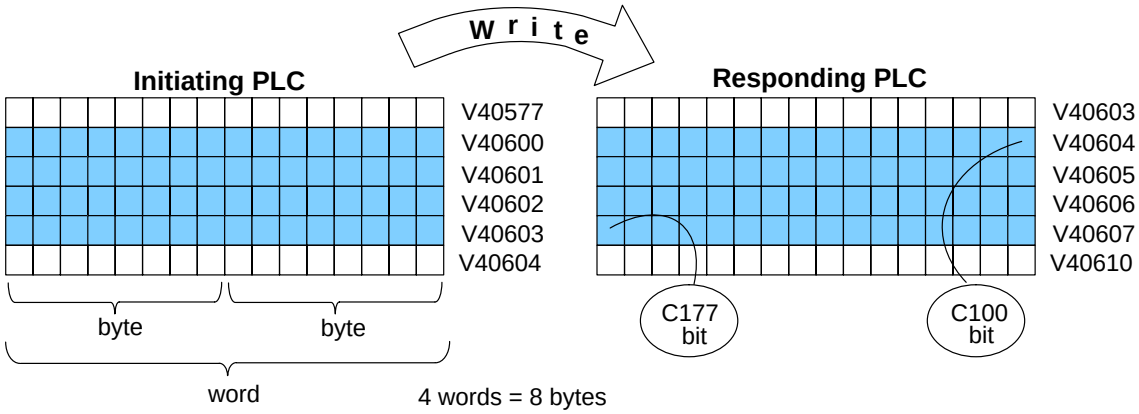
Write (WX) Instruction

The Write (WX) instruction specifies the memory location to be *written to* in the responding PLC.

A block of data is written that begins at the specified memory location and extends the number of bytes specified in the second LD instruction.

In the example, the 8-byte block of data beginning at V40600 and ending at V40603 in the *initiating*, or master, PLC is written (copied) into the responding PLC's memory beginning at C100 and ending at C177.





Addressing the Different Memory Types

Some data types are inherently 16 bits long, for example timer and counter current values. Other data types are 1 bit long, for example: discrete inputs and outputs. Word-length and bit-length data are mapped into Word Memory, also known as V-memory, which allows you to address *any* of the different memory types as 16-bit words.

Bit Memory

Bit memory can be addressed in Read and Write instructions by the name of the first bit of any byte. If your second LD instruction contains the constant K8, eight bytes will be transmitted. If you use C0 in your RX or WX instruction, you will transmit the eight bytes from C0 through C77.

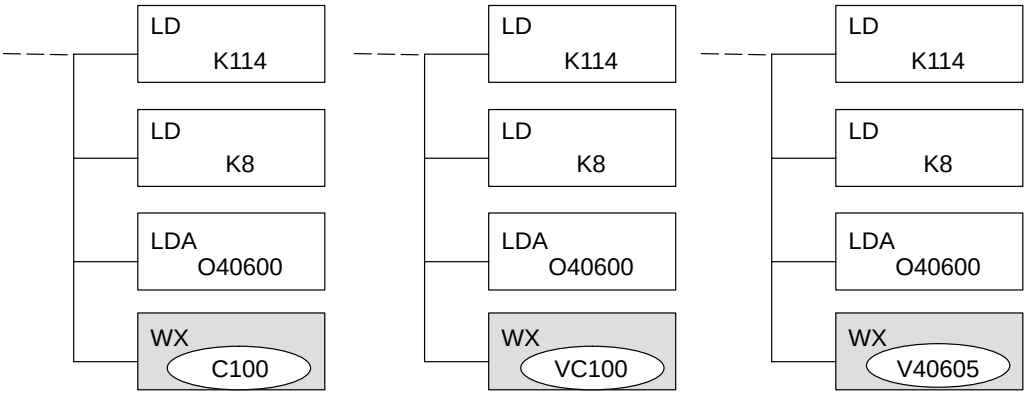
Word Memory and Aliases

In the example below, V40600 is the V-memory designation for the sixteen bits from C0 through C17. *Aliases* are a convenient substitute for V-memory designations, and can be used interchangeably in Read and Write instructions. VC0 is the alias for V40600. Either nomenclature addresses the same 16 bits.

The alias is simply the name of the first bit in a group of sixteen bits, with V added as a prefix. For example, VC0 represents the 16 bits beginning with C0. Word Memory, Bit Memory and Aliases all use the **octal** numbering system.

Word Memory Address V40600 = VC0 (Alias)	C17	C16	C15	C14	C13	C12	C11	C10	C7	C6	C5	C4	C3	C2	C1	C0
V40601 = VC20	C37	C36	C35	C34	C33	C32	C31	C30	C27	C26	C25	C24	C23	C22	C21	C20
V40602 = VC40	C57	C56	C55	C54	C53	C52	C51	C50	C47	C46	C45	C44	C43	C42	C41	C40
V40603 = VC60	C77	C76	C75	C74	C73	C72	C71	C70	C67	C66	C65	C64	C63	C62	C61	C60

The following Write routines are all equivalent. ***Direct***SOFT gives you the flexibility to identify the responding PLC's memory area in three different ways, as shown below.



*Direct*SOFT is Flexible

You can address the different data types by any available convention shown in the tables that follow. The largest block of data that can be sent in a single Read or Write operation is 128 bytes. The smallest block of data is one byte for Bit Memory types and two bytes, or one word for Word Memory types. The **octal** numbering system is used for all addresses in these tables.

DL05 CPU

DL05 CPU			
Data Types	**Bit Memory**	**Word Memory**	**Alias**
Timer Current Values	None	V0 – V177	TA0 – TA177
Counter Current Values	None	V1000 – V1177	CTA0 – CTA177
User Data Words	None	V1200 – V7377	None
Input Points (See note 1)	X0 – X377	V40400 – V40417	VX0 – VX360
Output Points (See note 1)	Y0 – Y377	V40500 – V40517	VY0 – VY360
Control Relays	C0 – C777	V40600 – V40677	VC0 – VC760
Special Relays	SP0 – SP777	V41200 – V41237	VSP0 – VSP760
Timer Status Bits	T0 – T177	V41100 – V41107	VT0 – VT160
Counter Status Bits	CT0 – CT177	V41140 – V41147	VCT0 – VCT160
Stages	S0 – S377	V41000 – V41017	VS0 – VS360

1 – The DL05 systems are limited to 8 discrete inputs and 6 discrete outputs with the present available hardware, but 256 point addresses exist.

DL06 CPU

DL06 CPU			
Data Types	**Bit Memory**	**Word Memory**	**Alias**
Timer Current Values	None	V0 – V377	TA0 – TA377
Counter Current Values	None	V1000 – V1177	CTA0 – CTA177
User Data Words	None	V400 – V677 V1200 – V7377 V10000 – V17777	None
Input Points (See note 1)	X0 – X777	V40400 – V40437	VX0 – VX760
Output Points (See note 1)	Y0 – Y777	V40500 – V40537	VY0 – VY760
Control Relays	C0 – C1777	V40600 – V40677	VC0 – VC1760
Special Relays	SP0 – SP777	V41200 – V41237	VSP0 – VSP760
Timer Status Bits	T0 – T377	V41100 – V41117	VT0 – VT160
Counter Status Bits	CT0 – CT177	V41140 – V41147	VCT0 – VCT160
Stages	S0 – S1777	V41000 – V41077	VS0 – VS1760
Remote I/O	GX0 – GX3777 GY0 – GY3777	V40000 – V40177 V40200 – V40377	VGX0 – VGX3760 VGY0 – VGY3760

1 – The DL06 systems are limited to 20 discrete inputs and 16 discrete outputs with the present available hardware, but 512 point addresses exist.



NOTE: The D2–230 CPU does not support the ECOM modules.

D2–240 CPU

D2–240 CPU			
Data Types	**Bit Memory**	**Word Memory**	**Alias**
Timer Current Values	None	V0 – V177	TA0 – TA177
Counter Current Values	None	V1000 – V1177	CTA0 – CTA177
User Data Words	None	V2000 – V3777 V4000 – V4377	None
Input Points	X0 – X477	V40400 – V40423	VX0 – VX460
Output Points	Y0 – Y477	V40500 – V40523	VY0 – VY460
Control Relays	C0 – C377	V40600 – V40617	VC0 – VC360
Special Relays	SP0 – SP137 SP540 – SP617	V41200 – V41205 V41226 – V41230	VSP0 – VSP120 VSP540 – VSP600
Timer Status Bits	T0 – T177	V41100 – V41107	VT0 – VT160
Counter Status Bits	CT0 – CT177	V41040 – V41147	VCT0 – VCT160
Stages	S0 – S777	V41000 – V41037	VS0 – VS760

D2–250–1 CPU

D2–250–1 CPU			
Data Types	**Bit Memory**	**Word Memory**	**Alias**
Timer Current Values	None	V0 – V377	TA0 – TA377
Counter Current Values	None	V1000 – V1377	CTA0 – CTA377
User Data Words	None	V1400 – V7377 V10000 – V17777	None
Input Points	X0 – X777	V40400 – V40437	VX0 – VX760
Output Points	Y0 – Y777	V40500 – V40537	VY0 – VY760
Control Relays	C0 – C1777	V40600 – V40677	VC0 – VC1760
Special Relays	SP0 – SP777	V41200 – V41237	VSP0 – VSP760
Timer Status Bits	T0 – T377	V41100 – V41117	VT0 – VT360
Counter Status Bits	CT0 – CT177	V41140 – V41147	VCT0 – VCT160
Stages	S0 – S1777	V41000 – V41077	VS0 – VS1760

D2–260 CPU

D2–260 CPU			
Data Registers	**Bit Memory**	**Word Memory**	**Alias**
Timer Current Values	None	V0 – V377	TA0 – TA377
Counter Current Values	None	V1000 – V1377	CTA0 – CTA377
User Data Words	None	V400 – V777 V1400 – V7377 V10000 – V37777	None
Input Points	X0 – X1777	V40400 – V40477	VX0 – VX1760
Output Points	Y0 – Y1777	V40500 – V40577	VY0 – VY1760
Control Relays	C0 – C3777	V40600 – V40777	VC0 – VC3760
Special Relays	SP0 – SP137 SP320 – SP717	V41200 – V41205 V41215 – V41234	VSP0 – VSP120 VSP320 – VSP700
Timer Status Bits	T0 – T377	V41100 – V41117	VT0 – VT360
Counter Status Bits	CT0 – CT377	V41140 – V41157	VCT0 – VCT360
Stages	S0 – S1777	V41000 – V41077	VS0 – VS1760
Remote I/O	GX0 – GX3777 GY0 – GY3777	V40000 – V40177 V40200 – V40377	VGX0 – VGX3760 VGY0 – VGY3760

D4–430 CPU

D4–430 CPU			
Data Registers	**Bit Memory**	**Word Memory**	**Alias**
Timer Current Values	None	V0 – V177	TA0 – TA177
Counter Current Values	None	V1000 – V1177	CTA0 – CTA177
User Data Words	None	V1400 – V7377	None
Input Points	X0 – X477	V40400 – V40423	VX0 – VX460
Output Points	Y0 – Y477	V40500 – V40523	VY0 – VY460
Control Relays	C0 – C737	V40600 – V40635	VC0 – VC720
Special Relays	SP0 – 137 SP320 – SP617	V41200 – V41205 V41215 – V41230	VSP0 – VSP120 VSP320 – VSP600
Timer Status Bits	T0 – T177	V41100 – V41107	VT0 – VT160
Counter Status Bits	CT0 – CT177	V41140 – V41147	VCT0 – VCT160
Stages	S0 – S577	V41000 – V41027	VS0 – VS560
Remote I/O	GX0 – GX777	V40000 – V40037	VGX0 – VGX760

D4–440 CPU

D4–440 CPU			
Data Registers	**Bit Memory**	**Word Memory**	**Alias**
Timer Current Values	None	V0 – V377	TA0 – TA377
Counter Current Values	None	V1000 – V1177	CTA0 – CTA177
User Data Words	None	V1400 – V7377 V10000 – V17777	None
Input Points	X0 – X477	V40400 – V40423	VX0 – VX460
Output Points	Y0 – Y477	V40500 – V40523	VY0 – VY460
Control Relays	C0 – C1777	V40600 – V40677	VC0 – VC1760
Special Relays	SP0 – 137 SP320 – SP717	V41200 – V41205 V41215 – V41234	VSP0 – VSP120 VSP320 – VSP700
Timer Status Bits	T0 – T377	V41100 – V41117	VT0 – VT360
Counter Status Bits	CT0 – CT177	V41140 – V41147	VCT0 – VCT160
Stages	S0 – S1777	V41000 – V41077	VS0 – VS1760
Remote I/O	GX0 – GX1777	V40000 – V40077	VGX0 – VGX1760

D4–450 CPU

D4–450 CPU			
Data Registers	**Bit Memory**	**Word Memory**	**Alias**
Timer Current Values	None	V0 – V377	TA0 – TA377
Counter Current Values	None	V1000 – V1377	CTA0 – CTA377
User Data Words	None	V1400 – V7377 V10000 – V37777	None
Input Points	X0 – X1777	V40400 – V40477	VX0 – VX1760
Output Points	Y0 – Y1777	V40500 – V40577	VY0 – VY1760
Control Relays	C0 – C3777	V40600 – V40777	VC0 – VC3760
Special Relays	SP0 – SP137 SP320 – SP717	V41200 – V41205 V41215 – V41234	VSP0 – VSP120 VSP320 – VSP700
Timer Status Bits	T0 – T377	V41100 – V41117	VT0 – VT360
Counter Status Bits	CT0 – CT377	V41140 – V41157	VCT0 – VCT360
Stages	S0 – S1777	V41000 – V41077	VS0 – VS1760
Remote I/O	GX0 – GX3777 GY0 – GY3777	V40000 – V40177 V40200 – V40377	VGX0 – VGX3760 VGY0 – VGY3760

Special Relays for Communications

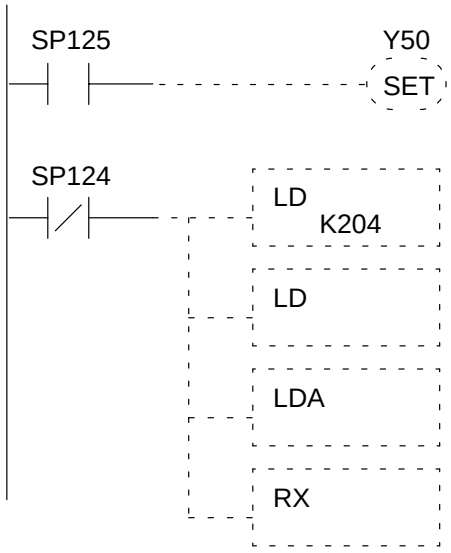
The ***Direct***LOGIC PLCs provide internal contacts (bits) for monitoring the status of communications. The internal contacts are called Special Relays (there are other Special Relays used for other purposes). There are two Special Relays for each slot in the base that will accept the ECOM module. The two relays perform the following functions:

- **Communication Busy** – This bit is on when the communication module is busy transmitting or receiving. You **must** use this bit, or relay contact, to prevent overwriting your Read or Write (RX/WX) instructions.
- **Communication Error** – This bit is on when an error occurred in the last RX or WX communication. This error automatically clears (the bit resets to zero) when another RX or WX instruction executes.

For example, Special Relays SP124 and SP125 correspond to an ECOM module in **slot 3** of the PLC base.

The Special Relay SP125 is used in the example to energize the output Y50, indicating a **communication error** has occurred. This Special Relay must appear earlier in the program than your RX or WX instruction because it is turned off (reset to zero) when a subsequent Read or Write instruction is executed.

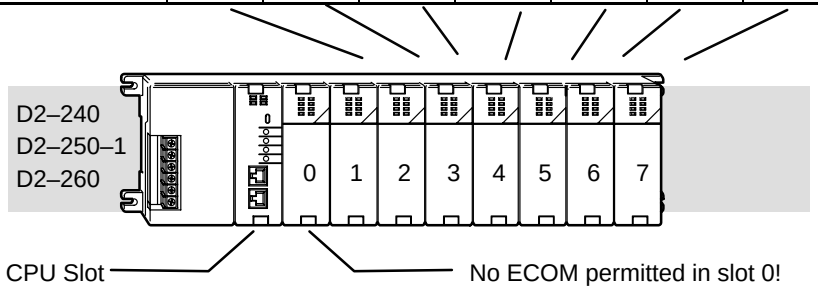
The Special Relay SP124 indicates the ECOM is **busy**. When SP124 is on, the normally closed contact opens to prevent executing another RX or WX instruction until the last one is completed. The appropriate busy bit **must** be used as a NC contact on every RX/WX instruction rung in the program.



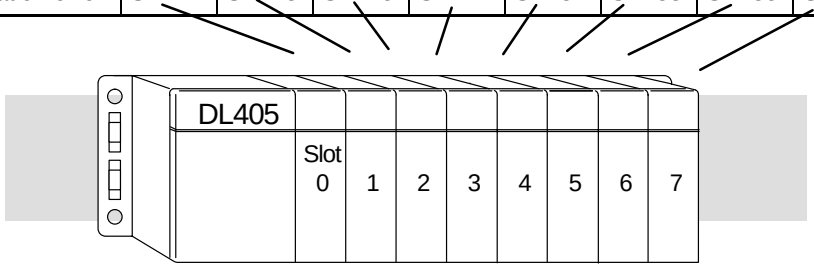
DL05 Special Purpose Communication Relays	
CPU-Base	Option Slot
Communication busy	SP120
Communication error	SP121

DL06 Special Purpose Communication Relays				
CPU-Base	Slot 1	Slot 2	Slot 3	Slot 4
Communication busy	SP120	SP122	SP124	SP126
Communication error	SP121	SP123	SP125	SP127

DL240, DL250–1 and DL260 Special Purpose Communication Relays							
CPU-Base	Slot 1	Slot 2	Slot 3	Slot 4	Slot 5	Slot 6	Slot 7
Communication busy	SP122	SP124	SP126	SP130	SP132	SP134	SP136
Communication error	SP123	SP125	SP127	SP131	SP133	SP135	SP137



D4–430 and D4–440 Special Purpose Communication Relays								
CPU-Base	Slot 0	Slot 1	Slot 2	Slot 3	Slot 4	Slot 5	Slot 6	Slot 7
Communication busy	SP120	SP122	SP124	SP126	SP130	SP132	SP134	SP136
Communication error	SP121	SP123	SP125	SP127	SP131	SP133	SP135	SP137

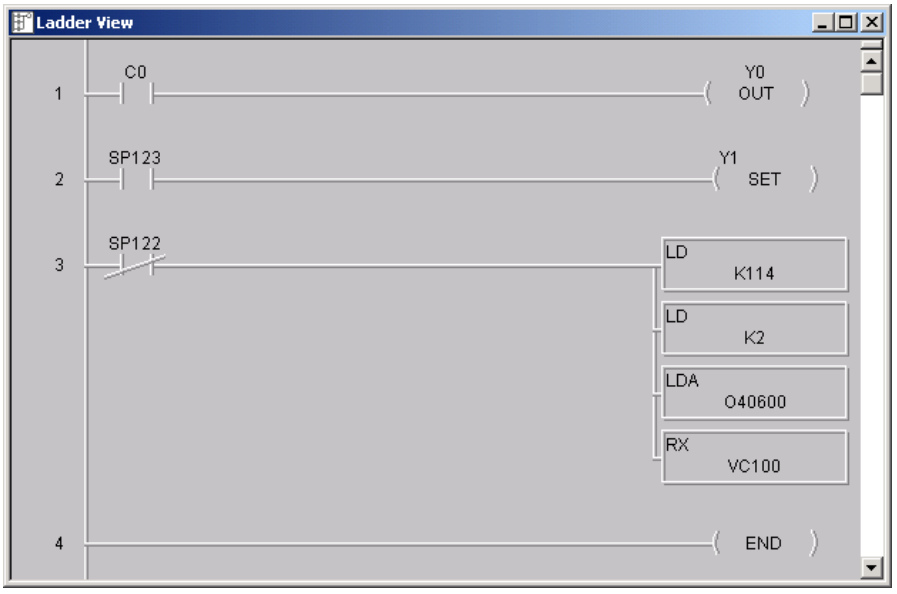


D4–450 Special Purpose Communication Relays								
CPU-Base	Slot 0	Slot 1	Slot 2	Slot 3	Slot 4	Slot 5	Slot 6	Slot 7
Communication busy	SP120	SP122	SP124	SP126	SP130	SP132	SP134	SP136
Communication error	SP121	SP123	SP125	SP127	SP131	SP133	SP135	SP137
Expansion Base 1	Slot 0	Slot 1	Slot 2	Slot 3	Slot 4	Slot 5	Slot 6	Slot 7
Communication busy	SP140	SP142	SP144	SP146	SP150	SP152	SP154	SP156
Communication error	SP141	SP143	SP145	SP147	SP151	SP153	SP155	SP157
Expansion Base 2	Slot 0	Slot 1	Slot 2	Slot 3	Slot 4	Slot 5	Slot 6	Slot 7
Communication busy	SP160	SP162	SP164	SP166	SP170	SP172	SP174	SP176
Communication error	SP161	SP163	SP165	SP167	SP171	SP173	SP175	SP177
Expansion Base 3	Slot 0	Slot 1	Slot 2	Slot 3	Slot 4	Slot 5	Slot 6	Slot 7
Communication busy	SP200	SP202	SP204	SP206	SP210	SP212	SP214	SP216
Communication error	SP201	SP203	SP205	SP207	SP211	SP213	SP215	SP217

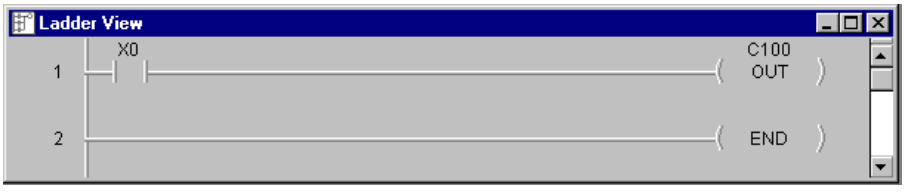
Program with One Read Instruction

The Ladder View screen below is the program development screen in ***Direct***SOFT Programming Software. This four rung program is explained in detail on page 4-14. This is a complete program although its function is very limited. There is also a two rung program that runs in the responding PLC, and it is also explained on page 4-14.

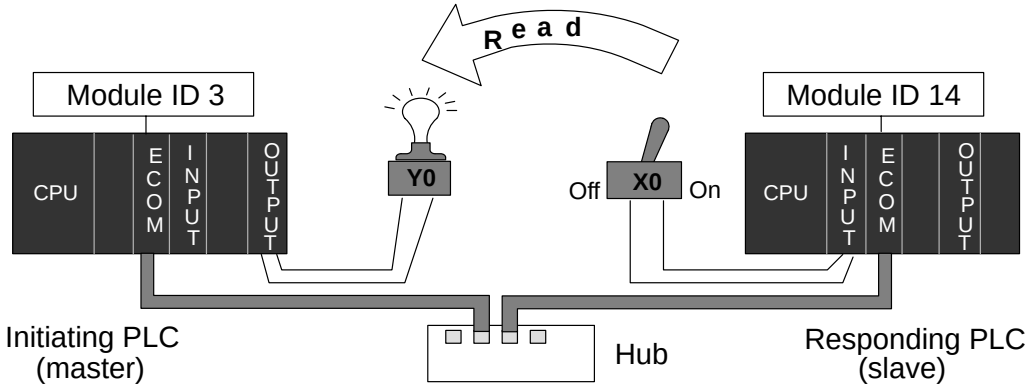
Program for the Initiating PLC



Program for the Responding PLC



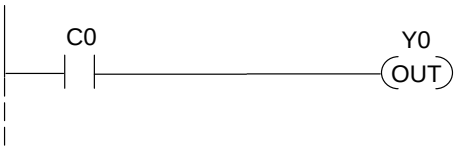
When the toggle switch input to the *responding* PLC is turned on (transitions from 0 to 1), the C0 bit in the *initiating* PLC transitions from 0 to 1. The program in the initiating PLC causes Y0 to turn on in response to the C0 bit.



For the purpose of these example programs, both the initiating PLC and the responding PLC must be in **RUN Mode**.

Rung 1

In our example, the normally open contact labeled **C0** is an internal control relay. When C0 is on, discrete output **Y0** is energized.



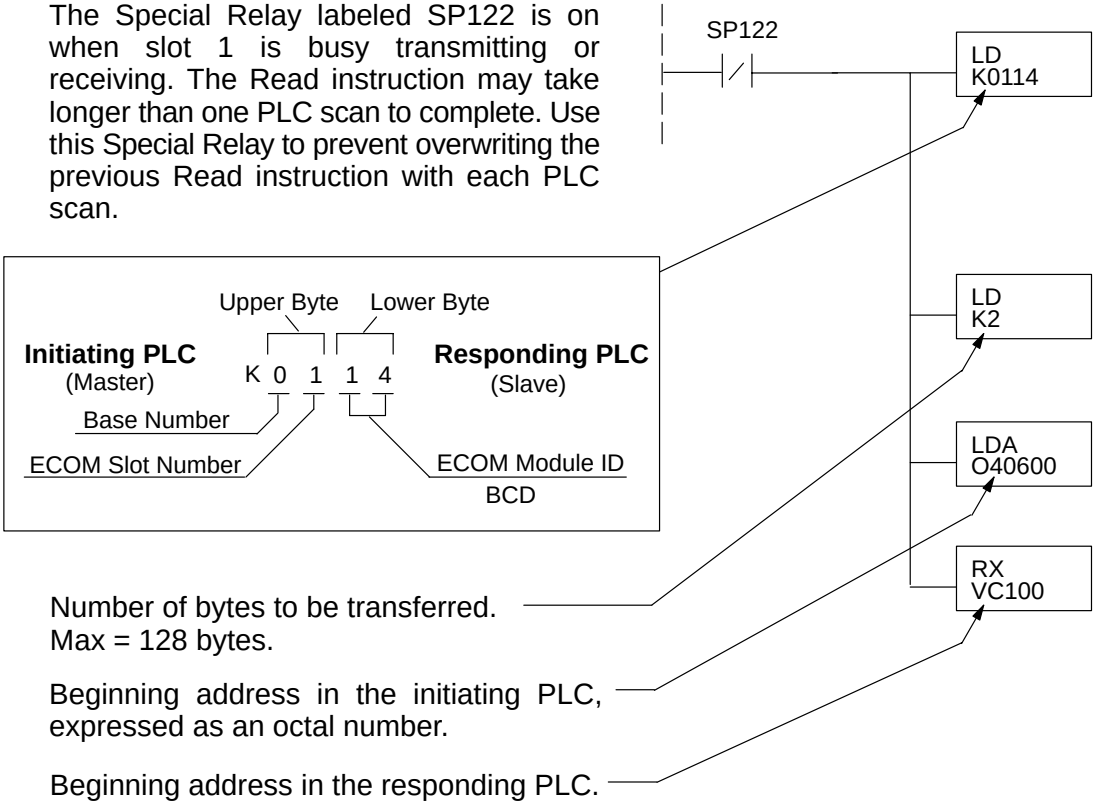
Rung 2

The second rung uses a Special Relay to identify a communication error. In the example, **SP123** is on if a communication error is present for *slot one*. Use different Special Relays if your ECOM module is in a different slot (see page 4-11 and 4-12). We use SP123 to turn on an indicator light connected to a discrete output.



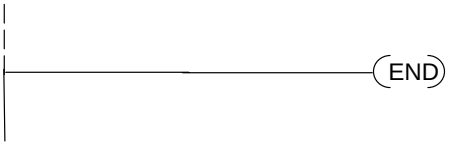
Rung 3

The Special Relay labeled SP122 is on when slot 1 is busy transmitting or receiving. The Read instruction may take longer than one PLC scan to complete. Use this Special Relay to prevent overwriting the previous Read instruction with each PLC scan.



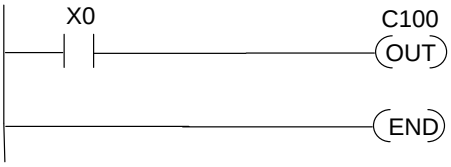
Rung 4

All ***Direct***LOGIC PLCs use an END statement to identify the final rung of the main body of the program.



Program for the Responding PLC

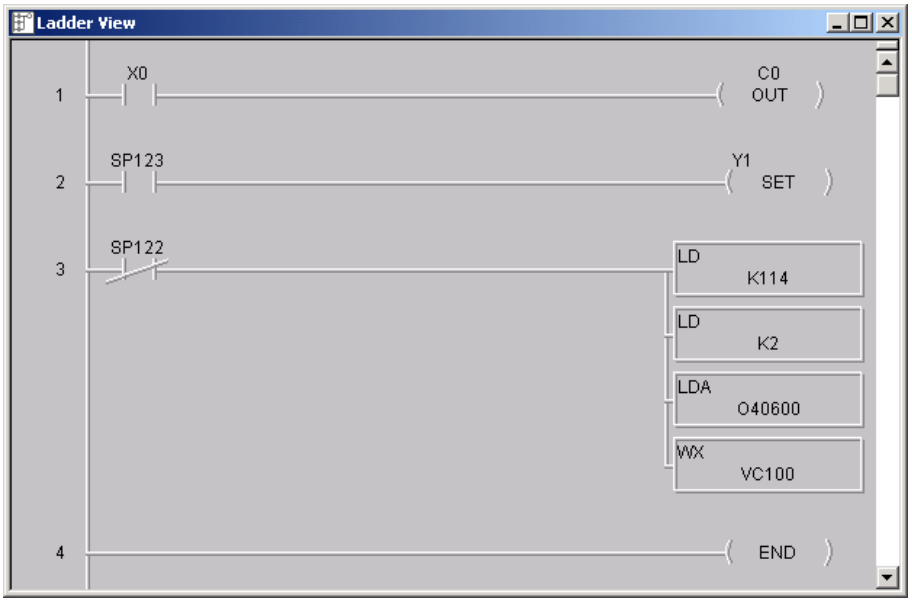
This two-rung program resides in the responding PLC's CPU. Its function is simply to use the X0 contact to turn on the internal control relay, C100.



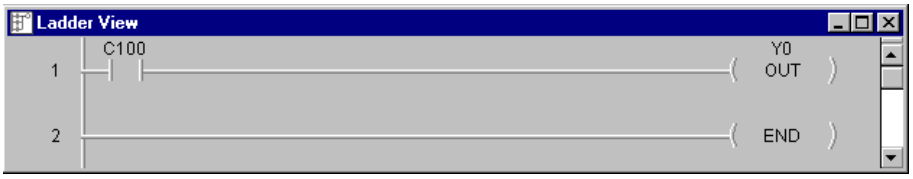
Example Program with One Write Instruction

The Ladder View screen below is the program development screen in ***Direct***SOFT Programming Software. This four-rung program is explained in detail on page 4-16. This is a complete program although its function is very limited. There is also a two-rung program that runs in the responding PLC. It is also explained on page 4-16.

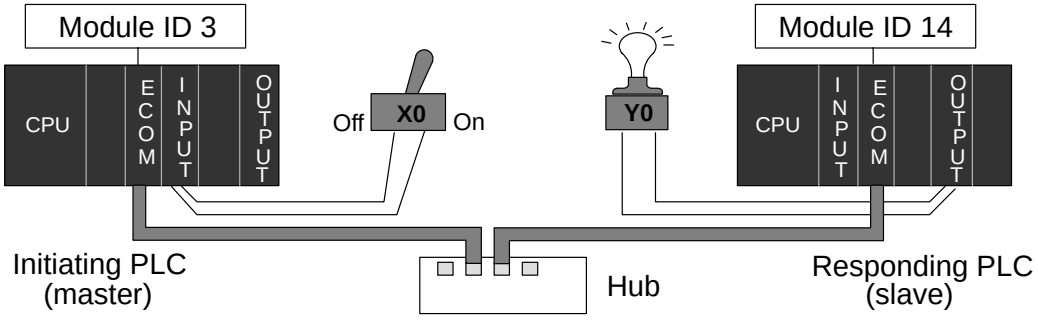
Program for the Initiating PLC



Program for the Responding PLC



When the toggle switch input to the *initiating* PLC is turned on (transitions from 0 to 1), the C100 bit in the *responding* PLC also transitions from 0 to 1. The program in the responding PLC causes Y0 to turn on in response to the C100 bit.



For the purpose of these example programs, both the initiating PLC and the responding PLC must be in **RUN Mode**.

Rung 1

In our example, the normally open contact labeled **X0** is a toggle switch input to a discrete input module. When X0 is on, Control Relay **C0** is energized.



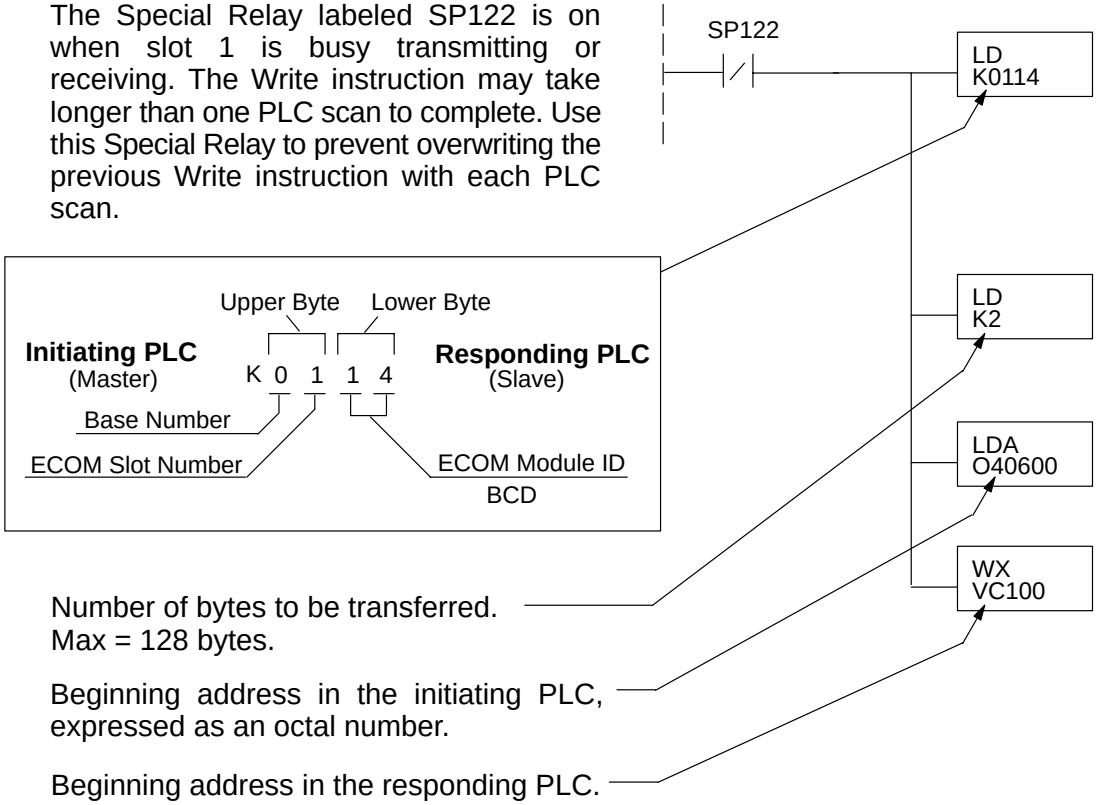
Rung 2

The second rung uses a Special Relay to identify a communication error. In the example, **SP123** is on if there is a communication error present *in slot one*. Use different Special Relays if your ECOM module is in a different slot (see page 4-11 and 4-12). We use SP123 to turn on an indicator light connected to a discrete output.



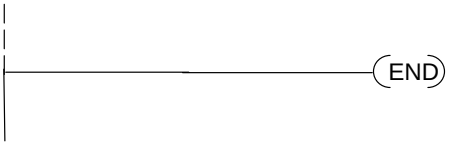
Rung 3

The Special Relay labeled SP122 is on when slot 1 is busy transmitting or receiving. The Write instruction may take longer than one PLC scan to complete. Use this Special Relay to prevent overwriting the previous Write instruction with each PLC scan.



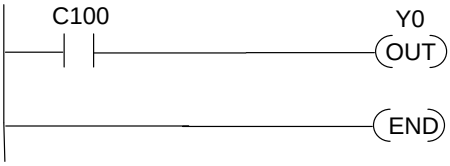
Rung 4

All ***Direct***LOGIC PLCs use an END statement to identify the final rung of the main body of the program.



Program for the Responding PLC

This two-rung program resides in the responding PLC's CPU. Its function is simply to take the C100 contact and convert it to a real output, Y0.



Integrating Multiple Read and Write Instructions

Multiple Read and Write instructions require *interlocks* for sequencing because only one RX/WX instruction can be processed per CPU scan. Using interlocks, one RX/WX instruction is processed in each scan until all RX/WX instructions have been executed. After the last instruction, the sequence then begins again at the first RX/WX instruction.

Without interlocks, the RX/WX instructions would be executed in an unpredictable order, and some might be executed many times before others are executed once. The interlocks serve to *open* (disconnect) the ladder circuits for all Read and Write instructions except the one that should be processed on the current CPU scan.

We show two methods of creating the interlocks necessary for sequencing multiple Read and Write instructions:

- Sequenced Internal Control Relays
- Shift Register

We will step you through the development of the interlocks using both methods. The two examples shown perform the same function. Only the interlocks are different.



NOTE: To fully understand the material in this section, you will first need to understand the Example Programs on pages 4-13 and 4-15, as well as the material in the Network Instructions section, beginning on page 4-3.

The following program segment sequences through three RX/WX instructions (two Write instructions and one Read instruction). You can develop your own program incorporating either of the two interlocking control strategies and expanding the number of interlocks to accommodate the number of RX/WX instructions in your program.

Interlocking Relays

It is easy to see the function of the interlocking relays if we construct a *truth table* first.

Across the top of the truth table we show internal control relays that we are considering using for our sequencing strategy. We have used C50 through C52 for our chart, but any contacts that are not used for other purposes in your program will work just as well.

Truth Table	C52	C51	C50
First RX/WX	0	0	0
Second RX/WX	0	0	1
Third RX/WX	0	1	0
Fourth RX/WX	0	1	1
Fifth RX/WX	1	0	0
Sixth RX/WX	1	0	1
Seventh RX/WX	1	1	0
Eighth RX/WX	1	1	1

Down the left side of the chart, we list the number of RX/WX instructions we may want to use in our RLL program.

The three contacts in this truth table will accommodate as many as eight Read or Write instructions. Our program only has three RX/WX instructions so we only need to use two contacts (see why on page 4-18). We will use C50 and C51. One additional contact (C53) would give us 32 combinations since the number of combinations expands as the power of 2.

Our three RX/WX instructions can be sequenced by the two contacts C50 and C51. Two contacts provide four different binary states:

- both off
- C50 on and C51 off
- C50 off and C51 on
- both on

We only need to use three of the four binary states (circled) since we only have three RX/WX instructions to sequence.

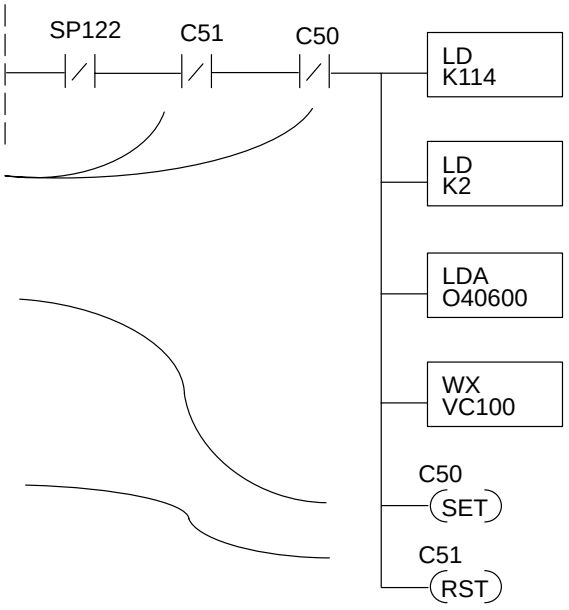
Truth Table	C52	C51	C50
First RX/WX	0	0	0
Second RX/WX	0	0	1
Third RX/WX	0	1	0
Fourth RX/WX	0	1	1
Fifth RX/WX	1	0	0
Sixth RX/WX	1	0	1
Seventh RX/WX	1	1	0
Eighth RX/WX	1	1	1

First RX/WX Instruction

C50 and C51 are interlocking contacts. They are normally closed in this rung to permit power flow to the first WX instruction. Both bits are off, corresponding to the first row of the truth table.

After the WX instruction is executed C50 is SET (turned on) which opens the contact in this rung and closes the C50 contact in the next rung.

C51 is RESET (turned off) which leaves the C51 contact closed for the next rung.

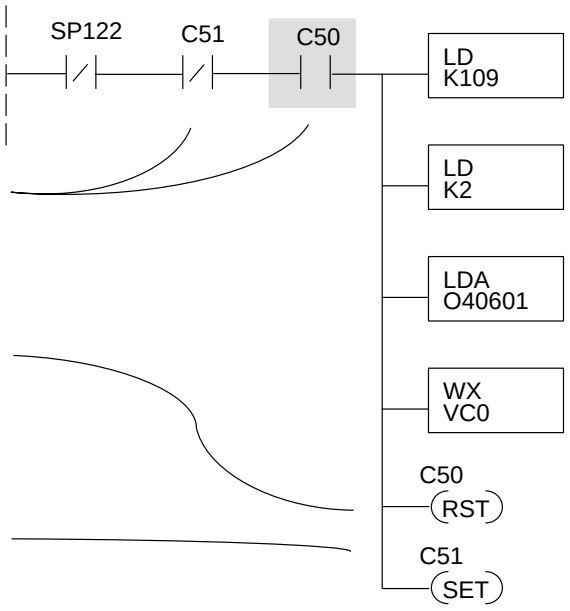


Second RX/WX Instruction

C50 is normally open and C51 is normally closed. For this rung to be executed, the C50 bit must be on and the C51 bit must be off, corresponding to the second row of the truth table. C50 was turned on in the previous rung. C51 was turned off in the previous rung.

After the WX instruction is executed C50 is RESET (turned off) which opens the C50 contact in this rung and closes it in the next rung.

C51 is SET (turned on), which closes the normally open C51 contact in the next rung.

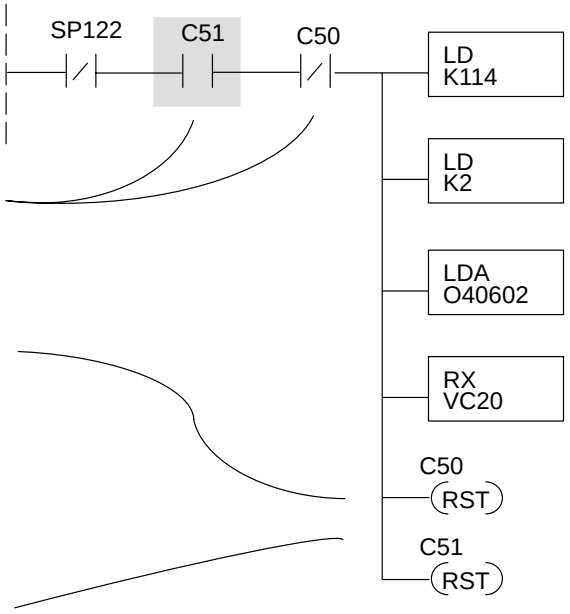


Third RX/WX Instruction

In this last rung, C50 is normally closed and C51 is normally open. For this rung to be executed, the C50 bit must be off and the C51 bit must be on, corresponding to the third row of the truth table. C51 was turned on in the previous rung.

After the RX instruction is executed, C50 is RESET which opens the C50 contact in this rung and allows it to close in preparation for repeating the first communication rung on the next CPU scan (page 4-18).

C51 is also RESET, which allows the C51 contact to close in preparation for repeating the first communication rung on the next CPU scan (page 4-18).



Returning to the First RX/WX Instruction

At the end of the third RX/WX instruction, we cycle back to the top row of the truth table on page 4-18. Both C50 and C51 are off, and the next CPU scan executes the first RX/WX instruction.

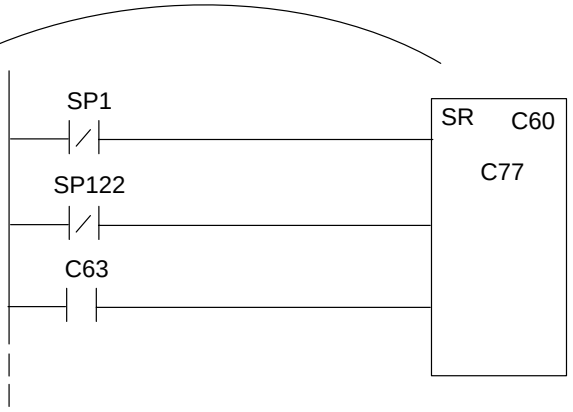
Shift Register

The Shift Register can be used for creating interlocks, as an alternative to using control relays. For a complete explanation of the function of the Shift Register, see the User Manual for your PLC. If you have more than a few RX/WX instructions, using control relays can become cumbersome. The Shift Register allows a single contact to be used in each communication rung as an interlock.

The data input to the Shift Register (SR) is Special Relay SP1. SP1 is the always-on bit. Combined with a normally closed contact it sends zeros to the Shift Register data input.

The clock input to the Shift Register is SP122, the communication busy bit. Each time one of the RX/WX instructions executes, the Shift Register moves the set bit over one place.

C63 is used in this example to reset the Shift Register to all zeros.



C77	C76	C75	C74	C73	C72	C71	C70	C67	C66	C65	C64	C63	C62	C61	C60

Shift Register after first scan.

C77	C76	C75	C74	C73	C72	C71	C70	C67	C66	C65	C64	C63	C62	C61	**C60**

Shift Register after second scan.

C77	C76	C75	C74	C73	C72	C71	C70	C67	C66	C65	C64	C63	C62	**C61**	C60

Shift Register after first RX/WX.

C77	C76	C75	C74	C73	C72	C71	C70	C67	C66	C65	C64	C63	**C62**	C61	C60

Shift Register after second RX/WX.

C77	C76	C75	C74	C73	C72	C71	C70	C67	C66	C65	C64	**C63**	C62	C61	C60

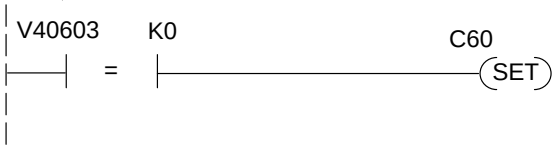
Shift Register after third RX/WX.

C77	C76	C75	C74	C73	C72	C71	C70	C67	C66	C65	C64	C63	C62	C61	C60

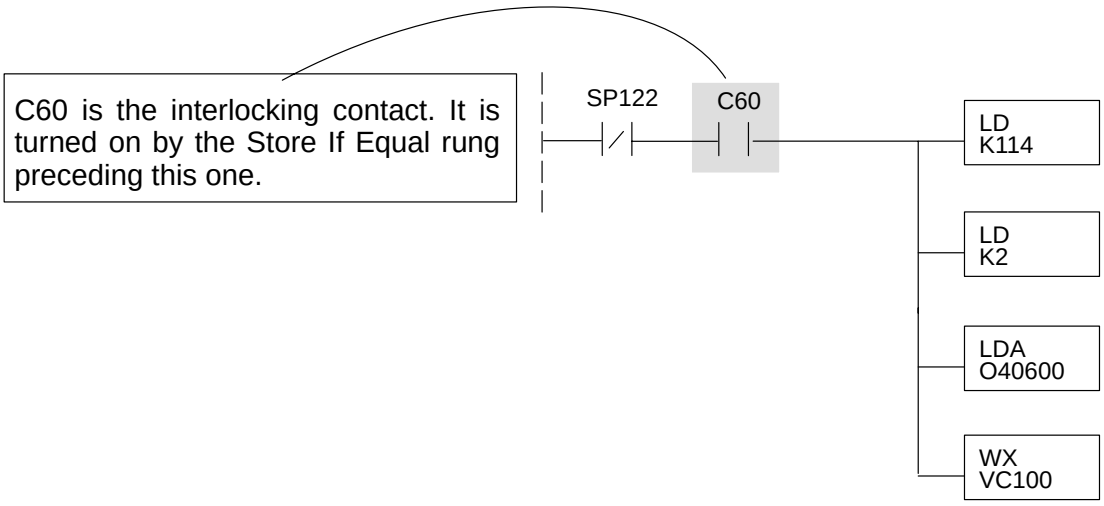
Shift Register after third RX/WX plus one scan.

Store If Equal

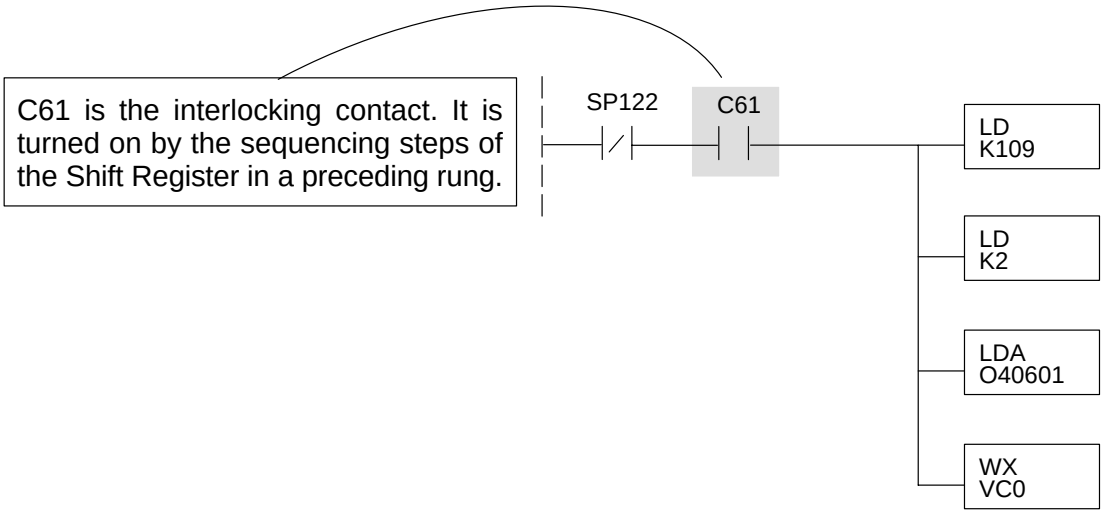
The Store If Equal instruction detects when the Shift Register is reset to zeros. When that condition is true the C60 bit is SET by this rung. The C60 bit becomes the high bit shifted by the Shift Register until each RX/WX instruction is executed in turn.



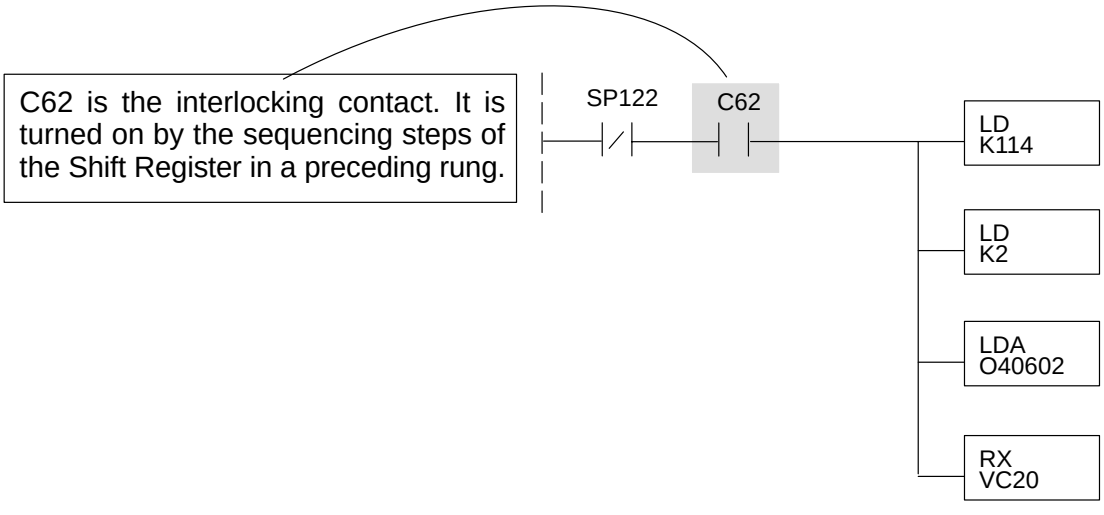
First RX/WX Instruction



Second RX/WX Instruction



Third RX/WX Instruction



After this rung is executed, the Shift Register shifts the high bit from C62 to C63 on the next CPU scan. C63 resets the Shift Register to zeros, the Store If Equal sets the C60 bit, and the CPU executes the first RX/WX instruction.

5

MODBUS® TCP/IP for H0/H2/H4–ECOM100

In This Chapter. . . .

- — MODBUS TCP/IP
- — Supported MODBUS Function Codes
- — Network Server Operation
- — Network Client Operation
- — H0/H2/H4–ECOM100 System Memory

MODBUS TCP/IP

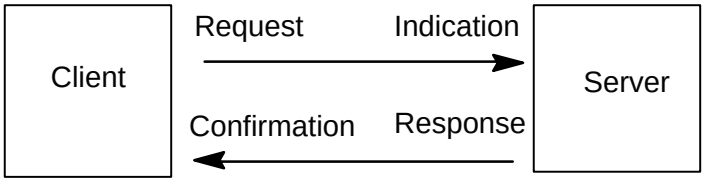
MODBUS TCP/IP is essentially the serial MODBUS RTU protocol encapsulated in a TCP/IP wrapper. MODBUS RTU is used for serial communications between a master and slave(s) devices. MODBUS TCP/IP is used for TCP/IP communications between client and server devices on an Ethernet network. The TCP/IP version of Modbus follows the OSI Network Reference Model.

Client / Server Model

The MODBUS messaging service provides a Client/Server communication between devices connected on an Ethernet TCP/IP network. This client / server model is based on four type of messages:

- MODBUS Request – the message sent on the network by the Client to initiate a transaction
- MODBUS Confirmation – the Response Message received on the Client side
- MODBUS Indication – the Request message received on the Server side
- MODBUS Response – the Response message sent by the Server

Client / Server Model



The H0/H2–ECOM100 can act as a Client or Server

Protocol Description

A typical MODBUS TCP/IP frame consists of the following fields:



The **MBAP header** (MODBUS Application Protocol header) is seven bytes long. It consists of the following fields.

- Transaction Identifier – It is used for transaction pairing, the MODBUS server copies in the response the transaction identifier of the request. (2 bytes)
- Protocol Identifier – It is used for intra–system multiplexing. The MODBUS protocol is identified by the value 0. (2 bytes)
- Length – The length field is a byte count of the following fields, including the Unit Identifier and data fields. (2 bytes)
- Unit Identifier – This field is used for intra–system routing purpose. It is typically used to communicate to a MODBUS or a MODBUS+ serial line slave through a gateway between an Ethernet TCP/IP network and a MODBUS serial line. This field is set by the MODBUS Client in the request and must be returned with the same value in the response by the server. (1 byte)

This header provides some differences compared to the MODBUS RTU application data unit used on serial line:

- The MODBUS “slave address” field usually used on MODBUS Serial Line is replaced by a single byte “Unit Identifier” within the MBAP Header. The “Unit Identifier” is used to communicate via devices such as bridges, routers and gateways that use a single IP address to support multiple independent MODBUS end units.
- All MODBUS requests and responses are designed in such a way that the recipient can verify that a message is finished. For function codes where the MODBUS PDU has a fixed length, the function code alone is sufficient. For function codes carrying a variable amount of data in the request or response, the data field includes a byte count.
- Protocol Identifier – It is used for intra–system multiplexing. The MODBUS protocol is identified by the value 0. (2 bytes)

The **function code field** of a message contains 8 bits. Valid function codes are in the range of 1 – 255 decimal. The function code instructs the slave what kind of action to take. Some examples are to read the status of a group of discrete inputs; to read the data in a group of registers; to write to an output coil or a group of registers; or to read the diagnostic status of a slave.
When a slave responds to the master, it uses the function code field to indicate either a normal response or that some type of error has occurred. For a normal response, the slave echoes the original function code. In an error condition, the slave echoes the original function code with its MSB set to a logic 1.

The **data field** is constructed using sets of two hexadecimal digits in the range of 00 to FF. According to the network’s serial transmission mode, these digits can be made of a pair of ASCII characters or from one RTU character.
The data field also contains additional information that the slave uses to execute the action defined by the function code. This can include internal addresses, quantity of items to be handled, etc.
The data field of a response from a slave to a master contains the data requested if no error occurs. If an error occurs, the field contains an exception code that the master uses to determine the next action to be taken. The data field can be nonexistent in certain types of messages.



Note: ModScan32 is a Windows based application program that can be used as a MODBUS master to access and change data points in a connected device (H0/H2/H4–ECOM100) The utility is ideally suited for quick and easy testing of MODBUS TCP network slave devices. Visit www.win–tech.com to download a free ModScan32 trial demo and for more information on ModScan32.

Supported MODBUS Function Codes

The following MODBUS function codes are supported by the H0/H2/H4–ECOM100. Not all function codes are supported when the ECOM100 serves as a network client. The “Network Client Operation” section later in this chapter lists the function codes that are supported in client mode.

MODBUS Function Code	Function	Server Mode	Client Mode
01	Read Output Table	yes	yes
02	Read Input Table	yes	yes
03	Read Holding Registers (when addressing mode is 584/984, this function is used to access analog output registers)	yes	yes
04	Read Input Registers (when addressing mode is 584/984, this function is used to access analog input registers)	yes	no
05	Force Single Output	yes	no
06	Preset Single Registers	yes	yes
08	Loop back / Maintenance	yes	no
15	Force Multiple Outputs	yes	yes
16	Preset Multiple Registers	yes	yes

Network Server (slave) Operation

This section describes how other MODBUS TCP/IP clients on a network can communicate with an H0/H2/H4–ECOM100 that you have configured for MODBUS TCP/IP protocol. A network client must send a MODBUS function code and MODBUS address to specify a PLC memory location the DL05/06/205/405 CPU. No CPU ladder logic is required to support MODBUS TCP/IP server operation.

MODBUS Function Codes Supported

The H0/H2/H4–ECOM100 supports the following MODBUS function codes when acting as a MODBUS TCP/IP server.

MODBUS Function Code	Function	DL05/06/205 /405 Data Types Available
01	Read Output Table	Y, C, T, CT
02	Read Input Table	X, SP
03	Read Holding Registers (when addressing mode is 584/984, this function is used to access analog output registers)	V
04	Read Input Registers (when addressing mode is 584/984, this function is used to access analog input registers)	V
05	Force Single Output	Y, C, T, CT
06	Preset Single Registers	V
08	Loop back / Maintenance	
15	Force Multiple Outputs	Y, C, T, CT
16	Preset Multiple Registers	V

Determining the MODBUS Address

There are typically two ways that most MODBUS addressing conventions allow you to specify a PLC memory location. These are:

- By specifying the MODBUS data type and address
- By specifying a MODBUS address only.

If Your Host Software or Client Requires the Data Type and Address

Many MODBUS TCP/IP clients allow you to specify the MODBUS data type and the MODBUS address that corresponds to the PLC memory location. This is the easiest method, but not all packages allow you to do it this way.

The actual equation used to calculate the address depends on the type of PLC data you are using. The PLC memory types are split into two categories for this purpose.

- Discrete – X, SP, Y, C, S, T(contacts), CT (contacts)
- Word – V memory, Timer current value, Counter current value

In either case, you basically convert the PLC octal address to decimal and add the appropriate MODBUS starting address (as required). The following tables show the exact range used for each group of data.



Note: For an automated MODBUS/Koyo address conversion utility, download the file **modbus_conversion.xls** from the **www.automationdirect.com** technical support website.

DL05 Memory Type	QTY (Dec.)	PLC Range (Octal)	MODBUS Address Range	MODBUS Data Type
For Discrete Data Types .... Convert PLC Addr. to Dec. +			**Start of Range +**	**Data Type**
Inputs (X)	256	X0 – X377	2048 – 2303	Input
Special Relays (SP)	512	SP0 – SP777	3072 – 3583	Input
Outputs (Y)	256	Y0 – Y377	2048 – 2303	Coil
Control Relays (C)	512	C0 – C777	3072 – 3583	Coil
Timer Contacts (T)	128	T0 – T177	6144 – 6271	Coil
Counter Contacts (CT)	128	CT0 – CT177	6400 – 6527	Coil
Stage Status Bits (S)	256	S0 – S377	5120 – 5375	Coil
For Word Data Types .... Convert PLC Addr. to Dec. +				**Data Type**
Timer Current Values (V)	128	V0 – V177	0 – 127	Input Register
Counter Current Values (V)	128	V1000 – V1177	512 – 639	Input Register
V Memory, user data (V)	3072	V1400 – V7377	768 – 3839	Holding Register

DL06 Memory Type	QTY (Dec.)	PLC Range (Octal)	MODBUS Address Range	MODBUS Data Type
For Discrete Data Types .... Convert PLC Addr. to Dec. +			**Start of Range +**	**Data Type**
Inputs (X)	512	X0 – X777	2048 – 2560	Input
Special Relays (SP)	512	SP0 – SP777	3072 – 3583	Input
Outputs (Y)	512	Y0 – Y777	2048 – 2560	Coil
Control Relays (C)	1024	C0 – C1777	3072 – 4095	Coil
Timer Contacts (T)	256	T0 – T377	6144 – 6399	Coil
Counter Contacts (CT)	128	CT0 – CT177	6400 – 6527	Coil
Stage Status Bits (S)	1024	S0 – S1777	5120 – 6143	Coil
Global Inputs (GX)	2048	GX0 – GX3777	0 – 2047	Input
Global Outputs (GY)	2048	GY0 – GY3777	0 – 2047	Coil
For Word Data Types .... Convert PLC Addr. to Dec. +				**Data Type**
Timer Current Values (V)	256	V0 – V377	0 – 255	Input Register
Counter Current Values (V)	128	V1000 – V1177	512 – 639	Input Register
V Memory, user data (V)	256 3072 4096	V400 – V677 V1400 – V7377 V10000 – V17777	256 – 511 768 – 3839 4096 – 8191	Holding Register

DL240 Memory Type	QTY (Dec.)	PLC Range (Octal)	MODBUS Address Range	MODBUS Data Type
For Discrete Data Types .... Convert PLC Addr. to Dec. +			**Start of Range +**	**Data Type**
Inputs (X)	320	X0 – X477	2048 – 2367	Input
Special Relays (SP)	144	SP0 – SP137 SP540 – SP617	3072 – 3167 3280 – 3471	Input
Outputs (Y)	320	Y0 – Y477	2048 – 2367	Coil
Control Relays (C)	256	C0 – C377	3072 – 3551	Coil
Timer Contacts (T)	128	T0 – T177	6144 – 6271	Coil
Counter Contacts (CT)	128	CT0 – CT177	6400 – 6527	Coil
Stage Status Bits (S)	512	S0 – S777	5120 – 5631	Coil
For Word Data Types .... Convert PLC Addr. to Dec. +				**Data Type**
Timer Current Values (V)	128	V0 – V177	0 – 127	Input Register
Counter Current Values (V)	128	V1000 – V1177	512 – 639	Input Register
V Memory, user data (V)	1024	V2000 – V3777	1024 – 2047	Holding Register
V Memory, user data (V) non–volatile	256	V4000 – V4377	2048 – 2303	Holding Register
V Memory, system (V)	106	V7620 – V7737 V7746 – V7777	3984 – 4063 4070 – 4095	Holding Register

DL250–1 Memory Type	QTY (Dec.)	PLC Range (Octal)	MODBUS Address Range	MODBUS Data Type
For Discrete Data Types .... Convert PLC Addr. to Dec. +			**Start of Range +**	**Data Type**
Inputs (X)	512	X0 – X777	2048 – 2560	Input
Special Relays (SP)	512	SP0 – SP137 SP320 – SP777	3072 – 3167 3280 – 3583	Input
Outputs (Y)	512	Y0 – Y777	2048 – 2560	Coil
Control Relays (C)	1024	C0 – C1777	3072 – 4095	Coil
Timer Contacts (T)	256	T0 – T377	6144 – 6399	Coil
Counter Contacts (CT)	128	CT0 – CT177	6400 – 6527	Coil
Stage Status Bits (S)	1024	S0 – S1777	5120 – 6143	Coil
For Word Data Types .... Convert PLC Addr. to Dec. +				**Data Type**
Timer Current Values (V)	256	V0 – V377	0 – 255	Input Register
Counter Current Values (V)	128	V1000 – V1177	512 – 639	Input Register
V Memory, user data (V)	3072 4096	V1400 – V7377 V10000 – V17777	768 – 3839 4096 – 8191	Holding Register
V Memory, system (V)	256	V7400 – V7777	3840 – 4095	Holding Register

DL260 Memory Type	QTY (Dec.)	PLC Range (Octal)	MODBUS Address Range	MODBUS Data Type
For Discrete Data Types .... Convert PLC Addr. to Dec. +			**Start of Range +**	**Data Type**
Inputs (X)	1024	X0 – X1777	2048 – 3071	Input
Special Relays (SP)	512	SP0 – SP137 SP320 – SP717	3072 – 3167 3280 – 3535	Input
Outputs (Y)	1024	Y0 – Y1777	2048 – 3071	Coil
Control Relays (C)	2048	C0 – C3777	3072 – 5119	Coil
Timer Contacts (T)	256	T0 – T377	6144 – 6399	Coil
Counter Contacts (CT)	256	CT0 – CT377	6400 – 6655	Coil
Stage Status Bits (S)	1024	S0 – S1777	5120 – 6143	Coil
Global Inputs (GX)	2048	GX0 – GX3777	0 – 2047	Input
Global Outputs (GY)	2048	GY0 – GY3777	0 – 2047	Coil
For Word Data Types .... Convert PLC Addr. to Dec. +				**Data Type**
Timer Current Values (V)	256	V0 – V377	0 – 255	Input Register
Counter Current Values (V)	256	V1000 – V1377	512 – 767	Input Register
V Memory, user data (V)	256 3072 11264	V400– V777 V1400 – V7377 V10000 – V35777	256 – 511 768 – 3839 4096 – 15359	Holding Register
V Memory, system (V)	256	V7600 – V7777 V36000 – V37777	3968 – 4095 15360 – 16383	Holding Register

DL430 Memory Type	QTY (Dec.)	PLC Range (Octal)	MODBUS Address Range (Decimal)	MODBUS Data Type
For Discrete Data Types .... Convert PLC Addr. to Dec. +			**Start of Range +**	**Data Type**
Inputs (X)	320	X0 – X477	2048 – 2367	Input
Special Relays (SP)	288	SP0 – SP137 SP320 – SP617	3072 – 3167 3280 – 3471	Input
Outputs (Y)	320	Y0 – Y477	2048 – 2367	Coil
Control Relays (CR)	512	C0 – C737	3072 – 3583	Coil
Timer Contacts (T)	128	T0 – T177	6144 – 6271	Coil
Counter Contacts (CT)	128	CT0 – CT177	6400 – 6527	Coil
Stage Status Bits (S)	384	S0 – S577	5120 – 5503	Coil
Global I/O (GX)	512	GX0 – GX777	0 – 511	Input
For Word Data Types .... Convert PLC Addr. to Dec. +				**Data Type**
Timer Current Values (V)	128	V0 – V177	0 – 127	Input Register
Counter Current Values (V)	128	V1000 – V1177	512 – 639	Input Register
V Memory, user data (V)	3072	V1400 – V7377	768 – 3839	Holding Register
V Memory, system (V)	256	V7400 – V7777	3840 – 4095	Holding Register

DL440 Memory Type	QTY (Dec.)	PLC Range (Octal)	MODBUS Address Range (Decimal)	MODBUS Data Type
For Discrete Data Types .... Convert PLC Addr. to Dec. +			**Start of Range +**	**Data Type**
Inputs (X)	320	X0 – X477	2048 – 2367	Input
Special Relays (SP)	352	SP0 – SP137 SP320 – SP717	3072 – 3167 3280 – 3535	Input
Outputs (Y)	320	Y0 – Y477	2048 – 2367	Coil
Control Relays (CR)	1024	C0 – C1777	3072 – 4095	Coil
Timer Contacts (T)	256	T0 – T377	6144 – 6399	Coil
Counter Contacts (CT)	128	CT0 – CT177	6400 – 6527	Coil
Stage Status Bits (S)	1024	S0 – S1777	5120 – 6143	Coil
Global I/O (GX)	1024	GX0 – GX1777	0 – 1023	Input
For Word Data Types .... Convert PLC Addr. to Dec. +				**Data Type**
Timer Current Values (V)	256	V0 – V377	0 – 255	Input Register
Counter Current Values (V)	128	V1000 – V1177	512 – 639	Input Register
V Memory, user data (V)	3072 4096	V1400 – V7377 V10000 – V17777	768 – 3839 4096 – 8191	Holding Register
V Memory, system (V)	288	V700 – V737 V7400 – V7777	448 – 479 3840 – 4095	Holding Register

DL450 Memory Type	QTY (Dec.)	PLC Range (Octal)	MODBUS Address Range (Decimal)	MODBUS Data Type
For Discrete Data Types .... Convert PLC Addr. to Dec. + Start of Range +				**Data Type**
Inputs (X)	1024	X0 – X1777	2048 – 3071	Input
Special Relays (SP)	512	SP0 – SP137 SP320 – SP717	3072 – 3167 3280 – 3535	Input
Outputs (Y)	1024	Y0 – Y1777	2048 – 3071	Coil
Control Relays (CR)	2048	C0 – C3777	3072 – 5119	Coil
Timer Contacts (T)	256	T0 – T377	6144 – 6399	Coil
Counter Contacts (CT)	256	CT0 – CT377	6400 – 6655	Coil
Stage Status Bits (S)	1024	S0 – S1777	5120 – 6143	Coil
Global Inputs (GX)	1536	GX0 – GX2777	0 – 1535	Input
Global Outputs (GY)	1536	GY0 – GY2777	0 – 1535	Coil
For Word Data Types .... Convert PLC Addr. to Dec. +				**Data Type**
Timer Current Values (V)	256	V0 – V377	0 – 255	Input Register
Counter Current Values (V)	256	V1000 – V1377	512 – 767	Input Register
V Memory, user data (V)	3072 12288	V1400 – V7377 V10000 – V37777	768 – 3839 4096 – 16383	Holding Register
V Memory, system (V)	320	V700 – V777 V7400 – V7777	448 – 768 3840 – 4095	Holding Register

The following examples show how to generate the MODBUS address and data type for hosts which require this format.

Example 1: V2100

Find the MODBUS address for User V location V2100.

1. Find V memory in the table.
2. Convert V2100 into decimal (1089).
3. Use the MODBUS data type from the table.

PLC Address (Dec.) + Data Type

V2100 = 1088 decimal

1088 + Hold. Reg. = **Holding Reg. 1089**

Timer Current Values (V)	128	V0 – V177	0 – 127	Input Register
Counter Current Values (V)	128	V1000 – V1177	512 – 639	Input Register
V Memory, user data (V)	1024	V2000 – V3777	1024 – 2047	Holding Register

Example 2: Y20

Find the MODBUS address for output Y20.

1. Find Y outputs in the table.
2. Convert Y20 into decimal (16).
3. Add the starting address for the range (2049).
4. Use the MODBUS data type from the table.

PLC Addr. (Dec) + Start Addr. + Data Type

Y20 = 16 decimal

16 + 2049 + Coil = **Coil 2065**

Outputs (Y)	320	Y0 – Y477	2049 – 2367	Coil
Control Relays (CR)	256	C0 – C377	3072 – 3551	Coil

Example 3: T10 Current Value

Find the MODBUS address to obtain the current value from Timer T10.

1. Find Timer Current Values in the table.
2. Convert T10 into decimal (8).
3. Use the MODBUS data type from the table.

PLC Address (Dec.) + Data Type

TA10 = 8 decimal

8 + Input Reg. = **Input Reg. 8**

Timer Current Values (V)	128	V0 – V177	0 – 127	Input Register
Counter Current Values (V)	128	V1000 – V1177	512 – 639	Input Register

Example 4: C54

Find the MODBUS address for Control Relay C54.

1. Find Control Relays in the table.
2. Convert C54 into decimal (44).
3. Add the starting address for the range (3072).
4. Use the MODBUS data type from the table.

PLC Addr. (Dec) + Start Addr. +Data Type

C54 = 44 decimal

44 + 3073 + Coil = **Coil 3117**

Outputs (Y)	320	Y0 – Y477	2048 – 2367	Coil
Control Relays (CR)	256	C0 – C377	3073 – 3551	Coil

If the Host Software or Client Requires an Address ONLY

Some MODBUS TCP/IP clients do not allow you to specify the MODBUS data type and address. Instead, you specify an address only. This method requires another step to determine the address, but it is not difficult. Basically, MODBUS also separates the data types by address ranges as well. This means an address alone can actually describe the type of data and location. This is often referred to as “adding the offset”.

The actual equation used to calculate the address depends on the type of PLC data you are using. The PLC memory types are split into two categories for this purpose.

- Discrete – X, GX, SP, Y, CR, S, T, C (contacts)
- Word – V memory , Timer current value, Counter current value

In either case, you basically convert the PLC octal address to decimal and add the appropriate MODBUS starting address (as required). The following tables show the exact range used for each group of data.



Note: For an automated MODBUS/Koyo address conversion utility, download the file **modbus_conversion.xls** from the **www.automationdirect.com** website.

Discrete Data Types*				
PLC Memory Type	**QTY (Dec.)**	**PLC Range (Octal)**	**MODBUS Address Range**	**Access**
Global Inputs (GX)	2048	GX0–GX1746	10001–10999	Read only
		GX1747 – GX3777	11000–12048	
Inputs (X)	1024	X0 – X1777	12049 – 13072	
Special Relays (SP)	512	SP0– SP777	13073 – 13584	
Reserved	–	–	13585 – 20000	
Global Outputs (GY)	2048	GY0– GY3777	1 – 2048	Read/ Write
Outputs (Y)	1024	Y0 – Y1777	2049 – 3072	
Control Relays (CR)	2048	C0 – C3777	3073 – 5120	
Timer Contacts (T)	256	T0 – T377	6145 – 6400	
Counter Contacts (CT)	256	CT0 – CT377	6401 – 6656	
Stage Status Bits (S)	1024	S0 – S1777	5121 – 6144	
Reserved	–	–	6657 – 10000	

***** Refer to your PLC user manual for the correct memory mapping size of your PLC. Some of the addresses shown above might not pertain to your particular CPU.

Word Data Types*					
Registers (Word)	**QTY (Dec.)**	**PLC Range (Octal)**	**MODBUS 40001 Address Range**	**MODBUS 30001 Address Range**	**Access**
V Memory (Timers)	256	V0 – V377	40001 – 40256	30001 – 30256	Read/ Write
V Memory (Counters)	256	V1000 – V1377	40513 – 40768	30513 – 30768	
V Memory (Data Words)	256	V400 – V777	40257 – 40512	30257 – 30512	
	3072	V1400 – 7377	40769 – 43840	30769 – 33840	
	5903	V10000 – V23416	44097 – 49999	34097 – 39999	
	5361	V23417 – V35777	410000 – 415360	310000 – 315360	
V Memory (System Parameters)	128	V7600 – V7777	43969 – 44096	33969 – 34096	
	1024	V36000 – V37777	415361 – 416384	315361 – 316384	
V Memory (Remote Inputs)	128	V40000 – V40177	416385 – 416512	316385 – 316512	Read only
V Memory (Remote Outputs)	128	V40200 – V40377	416513 – 416640	316513 – 316640	Read/ Write
V Memory (Input Points)	64	V40400 – V40477	416641 – 416704	316641 – 316704	Read only
V Memory (Output Points)	64	V40500 – V40577	416705 – 416768	316705 – 316768	Read/ Write
V Memory (Control Relays)	128	V40600 – V40777	416769 – 416896	316769 – 316896	
V Memory (Timers Status Bits)	16	V41100 – V41117	416961 – 416976	316961 – 316976	
V Memory (Counter Status Bits)	16	V41140 – V41157	416993 – 417008	316993 – 317008	
V Memory (Special Relays)	32	V41200 – V41237	417025 – 417056	317025 – 317056	Read only

***** Refer to your PLC user manual for the correct memory mapping size of your PLC.
Some of the addresses shown above might not pertain to your particular CPU.

Example 1: V2100

Find the MODBUS address for User V location V2100.

1. Find V memory in the table.
2. Convert V2100 into decimal (1088).
3. Add the MODBUS starting address for the mode (40001).

PLC Address (Dec.) + Mode Address

V2100 = 1088 decimal

1088 + 40001 = **41089**

For Word Data Types	PLC Address (Dec.)		+	Appropriate Mode Address		
Timer Current Values (V)	128	V0 – V177	0 – 127	3001	30001	Input Reg
Counter Current Values (V)	128	V1000 – V1177	512 – 639	3001	30001	Input Reg
V Memory, user data (V)	1024	V2000 – V3777	1024 – 2047	4001	40001	Hold Reg.

Example 2: Y20

Find the MODBUS address for output Y20.

1. Find Y outputs in the table.
2. Convert Y20 into decimal (16).
3. Add the starting address for the range (2048).
4. Add the MODBUS address for the mode (1).

PLC Addr. (Dec) + Start Address + Mode

Y20 = 16 decimal

16 + 2048 + 1 = **2065**

Outputs (Y)	320	Y0 – Y477	2048 – 2367	1	1	Coil
Control Relays (CR)	256	C0 – C377	3072 – 3551	1	1	Coil
Timer Contacts (T)	128	T0 – T177	6144 – 6271	1	1	Coil

Example 3: C54

Find the MODBUS address for Control Relay C54.

1. Find Control Relays in the table.
2. Convert C54 into decimal (44).
3. Add the starting address for the range (3072).
4. Add the MODBUS address for the mode (1).

PLC Addr. (Dec) + Start Address + Mode

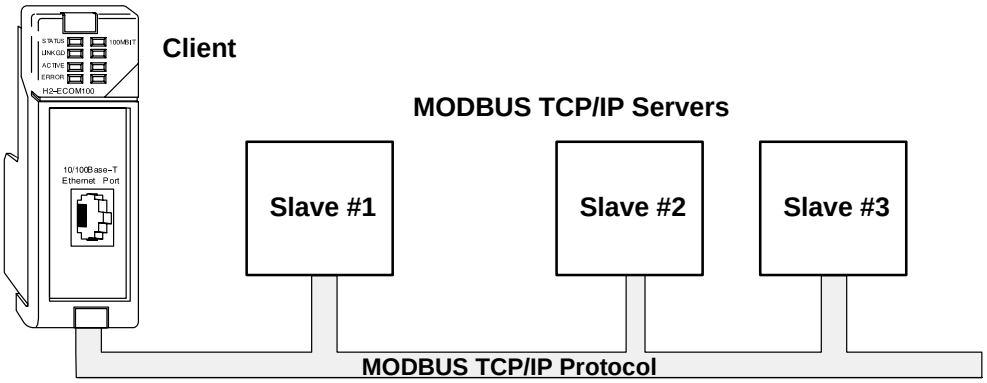
C54 = 44 decimal

44 + 3072 + 1 = **3117**

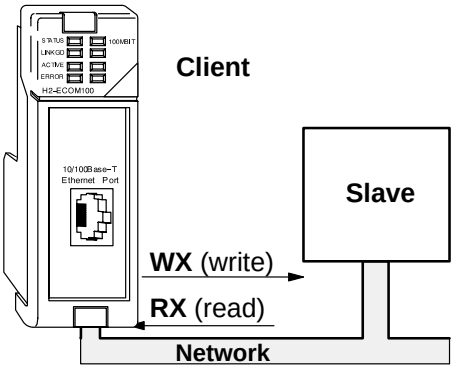
Outputs (Y)	320	Y0 – Y477	2048 – 2367	1	1	Coil
Control Relays (CR)	256	C0 – C377	3072 – 3551	1	1	Coil
Timer Contacts (T)	128	T0 – T177	6144 – 6271	1	1	Coil

Network Client (master) Operation

This section describes how the DL05/06/205/405 CPU can serve as a client on a MODBUS TCP/IP network using the H0/H2/H4–ECOM100. This section discusses how to design the required ladder logic for network client operation.



When using the ECOM100 as a client on the network, you use simple RLL instructions to initiate the requests. The WX instruction initiates network write operations, and the RX instruction initiates network read operations. Before executing either the WX or RX commands, we need to load data related to the read or write operation onto the CPU's accumulator stack. When the WX or RX instruction executes, it uses the information on the stack combined with data in the instruction box to completely define the task.



MODBUS Function Codes Supported

The H0/H2/H4–ECOM100 supports the following MODBUS function codes when acting as a MODBUS TCP/IP client.

MODBUS Function Code	Function	DL05/06/205/405 Data Types Available
01	Read Output Table	Y, C, T, CT
02	Read Input Table	X, SP
03	Read Holding Registers (when addressing mode is 584/984, this function is used to access analog output registers)	V
06	Preset Single Registers	V
15	Force Multiple Outputs	Y, C, T, CT
16	Preset Multiple Registers	V

Note: The H0/H2/H4–ECOM100, as a client/master, does not support function code 4. Thus, 30001 address ranges cannot be read from a server/slave device.

PLC Memory Supported for Client Operation

The actual equation used to calculate the address depends on the type of PLC data you are using. The PLC memory types are split into three categories for this purpose.

- Discrete – X, GX, SP
- Discrete – Y, CR, S, T, C
- Word – Timer current value, Counter current value, Data Words

In either case, you basically take the MODBUS address you are trying to target, subtract the starting MODBUS of that range, convert the result to octal and add the octal number to the begining PLC address in the appropriate PLC range. See the conversion examples on the following page. The following tables show the exact range used for each group of data.



Note: For an automated MODBUS/Koyo address conversion utility, download the file **modbus_conversion.xls** from the **www.automationdirect.com** website.

Discrete Data Types*				
PLC Memory Type	**QTY (Dec.)**	**PLC Range (Octal)**	**MODBUS Address Range**	**Access**
Global Inputs (GX)	2048	GX0–GX1746	10001–10999	Read only
		GX1747 – GX3777	11000–12048	
Inputs (X)	1024	X0 – X1777	12049 – 13072	
Special Relays (SP)	512	SP0– SP777	13073 – 13584	
Reserved	–	–	13585 – 20000	
Global Outputs (GY)	2048	GY0– GY3777	1 – 2048	Read/ Write
Outputs (Y)	1024	Y0 – Y1777	2049 – 3072	
Control Relays (CR)	2048	C0 – C3777	3073 – 5120	
Timer Contacts (T)	256	T0 – T377	6145 – 6400	
Counter Contacts (CT)	256	CT0 – CT377	6401 – 6656	
Stage Status Bits (S)	1024	S0 – S1777	5121 – 6144	
Reserved	–	–	6657 – 10000	

Word Data Types*				
Registers (Word)	**QTY (Dec.)**	**PLC Range (Octal)**	**MODBUS Address Range**	**Access**
V Memory (Timers)	256	V0 – V377	40001 – 40256	Read/ Write
V Memory (Counters)	256	V1000 – V1377	40513 – 40768	
V Memory (Data Words)	256	V400 – V777	40257 – 40512	
	3072	V1400 – 7377	40769 – 43840	
	5903	V10000 – V23416	44097 – 49999	
	5361	V23417 – V35777	410000 – 415360	
V Memory (System Parameters)	128	V7600 – V7777	43969 – 44096	
	1024	V36000 – V37777	415361 – 416384	

***** Refer to your PLC user manual for the correct memory mapping size of your PLC. Some of the addresses shown above might not pertain to your particular CPU.



Note: Your PC's Windows calculator can be used for number conversions (i.e. decimal to octal). The Windows calculator must be in Calculator>View>Scientific mode to enable number conversions capability.

Example 1: Calculating Word PLC Address

Find the PLC address to use to target MODBUS address **41025** in a server device.

1. Subtract the begining of the MODBUS word address range (40001) from the desired MODBUS address to target.
2. Convert decimal result into octal.
3. Add octal result to begining PLC range (Input, Output or Word).

1. 41025 – 40001 = 1024 decimal
2. 1024 decimal = 2000 octal
3. V0 (octal) + 2000 (octal) = **V2000** octal

Example 2: Calculating Discrete Input PLC Address

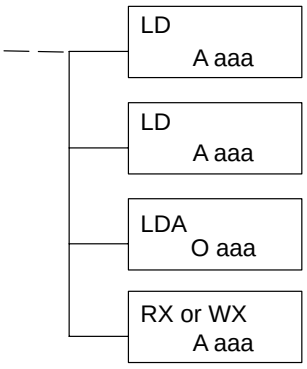
Find the PLC address to use to target MODBUS address **12060** in a server device.

1. Subtract the begining of the MODBUS Input address range (12049) from the desired MODBUS address to target.
2. Convert decimal result into octal.
3. Add octal result to begining PLC range (Input, Output or Word).

1. 12060 – 12049 = 11 decimal
2. 11 decimal = 13 octal
3 X0 (octal) + 13 (octal) = **X13** octal

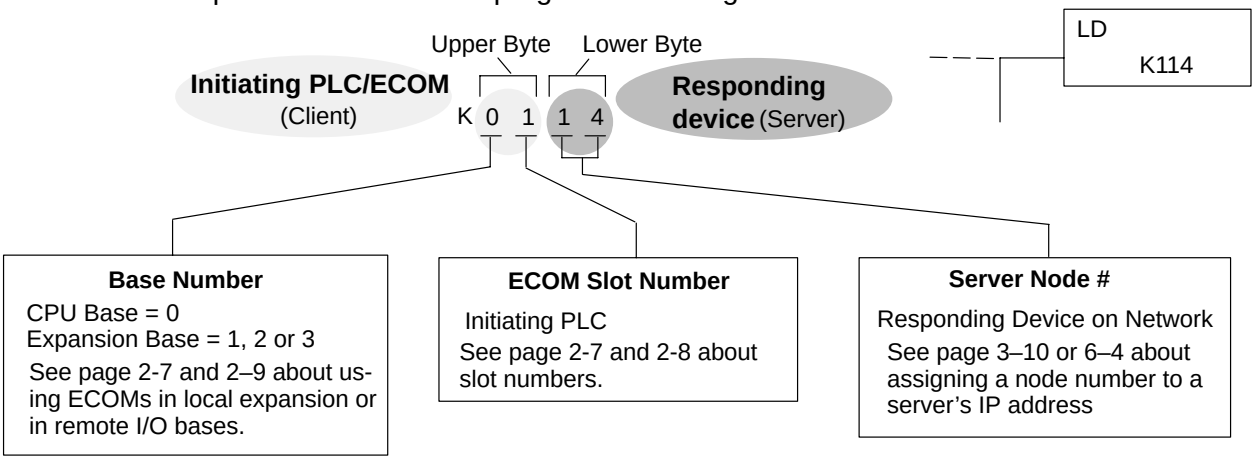
Building the Read (RX) or Write (WX) Routine

For network communications, you build the Read (RX) or Write (WX) instructions into a **routine** which requires the four instructions you see to the right. They must be used in the sequence shown. The following step-by-step procedure will provide you the information necessary to set up your ladder program to receive data from a network server.



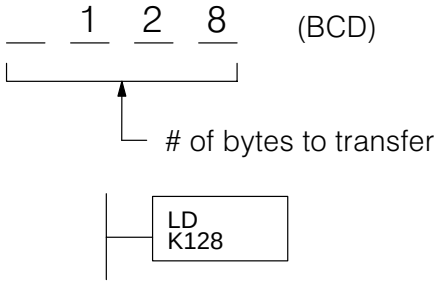
Step 1: Identify ECOM Slot Location and Server Node #

The first Load (LD) instruction accepts either a constant or a variable. Use a “K” to designate the number as a constant. Use a “V” if you are entering the address of a register. The contents of that register perform the same function as the constant shown below. For example, you could use V2000 in place of K0114. If the contents of V2000 is the number “114,” the function would be the same. Using a variable allows changing parameters while the program is running.



Step 2: Load Number of Bytes to Transfer

The second Load (LD) instruction determines the number of bytes which will be transferred between the master and slave in the subsequent WX or RX instruction. The value to be loaded is in BCD format (decimal), from 1 to 128 bytes.

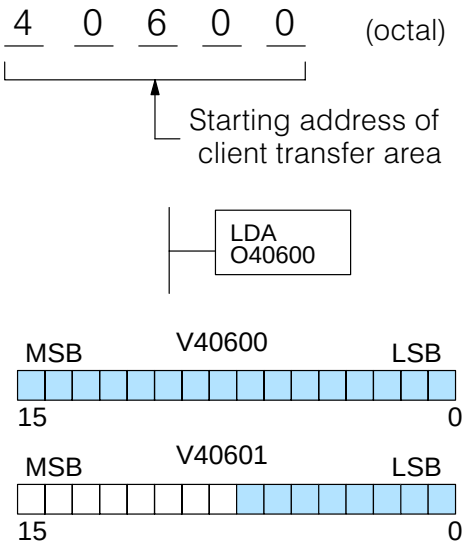


Step 3: Specify Master Memory Area

The third instruction in the RX or WX sequence is a Load Address (LDA) instruction. Its purpose is to load the starting address of the memory area to be transferred. Entered as an octal number, the LDA instruction converts it to hex and places the result in the accumulator.

For a WX instruction, the CPU sends the number of bytes previously specified from its memory area beginning at the LDA address specified.

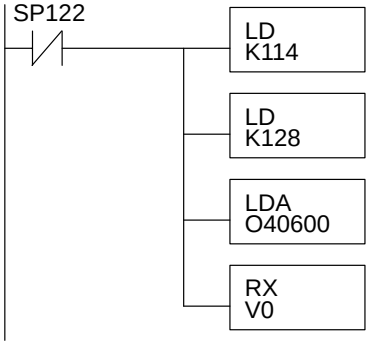
For an RX instruction, the CPU reads the number of bytes previously specified from the server, placing the received data into its memory area beginning at the LDA address specified.



NOTE: Since V memory words are always 16 bits, you may not always use the whole word. For example, if you only specify to read 3 bytes, you will only get 24 bits of data. In this case, only the 8 least significant bits of the last word location will be modified. The remaining 8 bits are not affected.

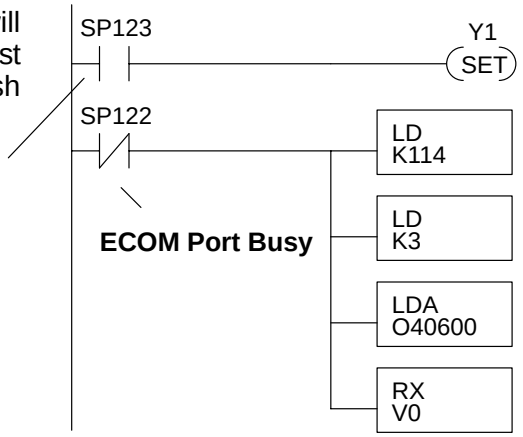
Step 4: Specify Slave Memory Area

The last instruction in our sequence is the WX or RX instruction itself. Use WX to write to the server, and RX to read from the server. All four of our instructions are shown to the right. In the last instruction, you must specify the starting address and a valid data type for the server.



Communications from a Ladder Program

Typically network communications will last longer than 1 scan. The program must wait for the communications to finish before starting the next transaction.



Depending on which slot the ECOM is in, it has two Special Relay contacts associated with it (see page 4–11 to 4–12 for special relays). One indicates “Port busy”, and the other indicates “Port Communication Error”. The example above shows the use of these contacts for an ECOM that is in slot 1. The “Port Busy” bit is on while the PLC communicates with the slave. When the bit is off the program can initiate the next network request.

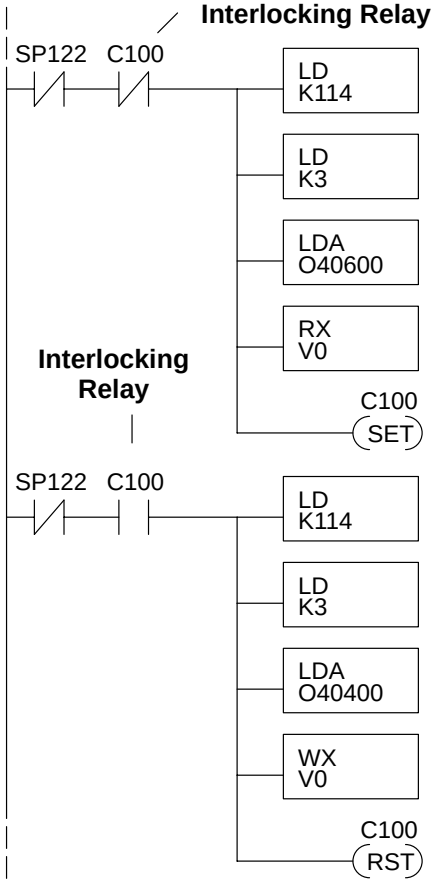
The “Port Communication Error” bit turns on when the PLC has detected an error. Use of this bit is optional. When used, it should be ahead of any network instruction boxes since the error bit is reset when an RX or WX instruction is executed.

Multiple Read and Write Interlocks

If you are using multiple reads and writes in the RLL program, you have to interlock the routines to make sure all the routines are executed. If you don’t use the interlocks, then the CPU will only execute the first routine. This is because each port can only handle one transaction at a time.

In the example to the right, after the RX instruction is executed, C0 is set. When the port has finished the communication task, the second routine is executed and C0 is reset.

If you’re using RLL*PLUS* Stage Programing, you can put each routine in a separate program stage to ensure proper execution and switch from stage to stage allowing only one of them to be active at a time.



H0/H2/H4–ECOM100 System Memory

	H0/H2/H4–ECOM100			
	Modbus Addressing Range (Decimal)	**Words (16–bit)**	**Word Descriptions**	**Access**
Module Version Information	**317501 – 317506;** **(417501 – 417506)***	6	1 – OS Major Version 2 – OS Minor Version 3 – OS Build Version 4 – Booter Major Version 5 – Booter Minor Version 6 – Booter Build Version	Read only
	317507 – 317510 (417507 – 417510)	–	Reserved	–
Device Data	**317511 – 317600;** **(417511 – 417600)***	90	1 – Version of Device 2 – Family 3 – Processor 4 – Module Type 5 – Status Code (6–8) – Ethernet Address 9 – RAM Size 10 – Flash Size 11 – Batt RAM Size 12 – DIP Settings 13 – Media Type (14–15) – EPF Count (if supported) 16 – Run Relay State (if supported) 17 – Batt Low (if supported) 18 – Model Number 19 – Ethernet Speed (20–90) – Reserved	Read only
	317601 – 318500 (417601 – 418500)	–	Reserved	–
Dynamic Module Data	**418001 – 418020**	20	(1–3) – Reserved 4 – Flags: Bit 0: If 1, module has rebooted since this bit was cleared, a write to the Flags word with this bit set will clear this reboot bit. Bit (1–7) – Reserved 5 – Reboot Count (LSW) – Read Only 6 – Reboot Count (MSW) – Read Only (7–20) – Reserved	Read/ Write
	418021 – 419250	–	Reserved	–

***For clients that only support function code 3 to read word data.

6

H0/H2/H4–ECOM100 DHCP & HTML Configuration

— H0/H2/H4–ECOM100 DHCP
— Disabling DHCP and Assigning a Static IP Address
— Using HTML Configuration

H0/H2/H4–ECOM100 DHCP

DHCP Issues

The H0/H2/H4–ECOM100 is configured at the factory to look for a DHCP (Dynamic Host Configuration Protocol) server at power up. DHCP provides a way to allocate IP address dynamically to devices on a local area network (LAN). A system or network administrator configures a DHCP server with a range of IP addresses that can be assigned to DHCP enabled clients/servers (i.e. H0/H2/H4–ECOM100).

In addition to an IP address, a DHCP server can provide other information such as DNS domain or a gateway IP address.

DHCP uses the concept of a “lease” or amount of time that an assigned IP address will be valid for a client. The lease time can vary depending on how long a user is likely to require the network connection at a particular location. Since the TCP/IP configuration is “leased” to the client, that is, it’s not a permanent configuration. This information can change from one power up session to the next. While this is an acceptable solution for the initial testing and setup of your H0/H2/H4–ECOM100 device, we **do not** recommend that you use DHCP to assign IP addresses for your runtime operation. Use NetEdit3.5 or the H0/H2/H4–ECOM100’s HTML Configuration page to assign a static IP address to the module (shown below).

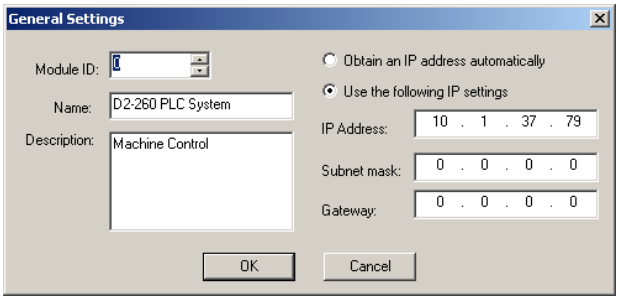
NetEdit3.5 can be used to connect to a H0/H2/H4–ECOM100 using the IPX protocol, regardless of the IP address that was assigned to it by a DHCP server.

Disabling DHCP and Assigning a Static IP Address

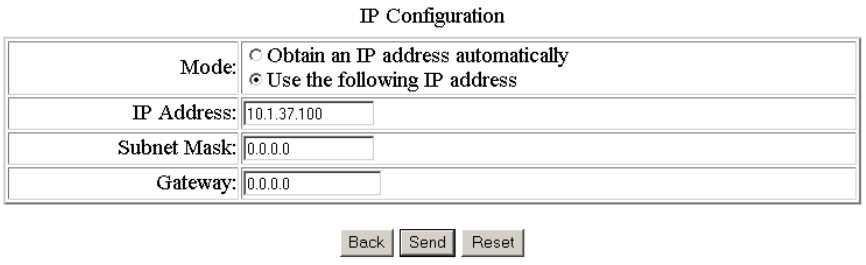
You can use NetEdit3.5 or the H0/H2/H4–ECOM100’s HTML Configuration page to disable DHCP and assign a static IP address to the module. Click on the *Use the following IP Address* button and enter a valid IP address for your network application.

NetEdit3.5 (refer to chapter 3)

.



HTML Configuration

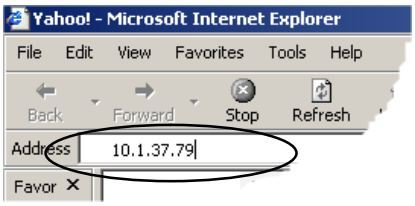


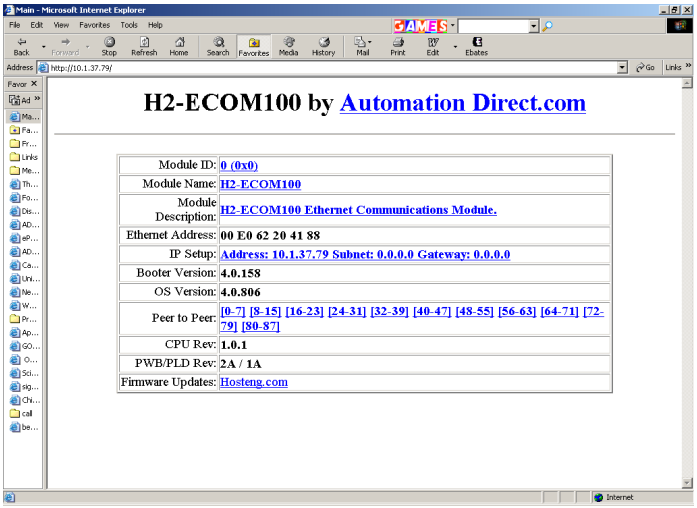
Using HTML Configuration

The H0/H2/H4–ECOM100 can be configured by using your PC's internet browser to access the module's HTML configuration page. This method of configuration uses the TCP/IP protocol, so you must know the H0/H2/H4–ECOM100's IP address to establish communications. The IP address may have been assigned by a DHCP server (default) or may have been set by using NetEdit3.5 or later.

Connecting to the H0/H2/H4–ECOM100

Enter the module's IP address in your browsers Address field. Connecting to the module's HTML Configuration utility brings up the window below

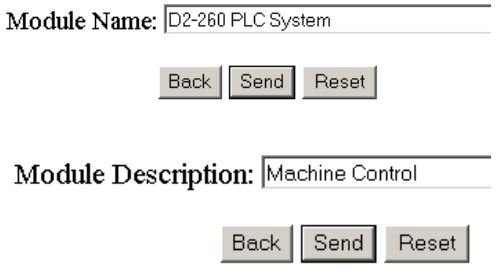




To configure the module, click on the desired parameter field. A new window will open, which are all described below and on the following page. Clicking the Back button will take you back to the main configuration screen shown above. Clicking the Send button writes the entry or change to the module's flash memory and clicking the Reset button reads the module's flash memory.

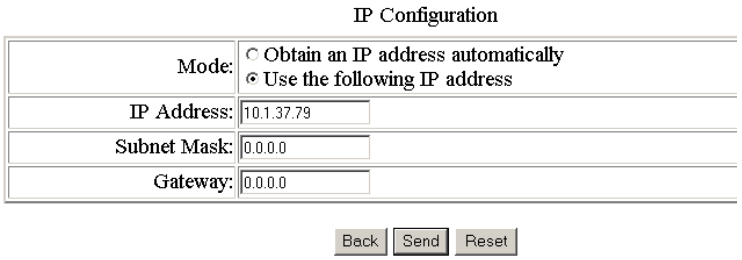
Module ID: Module IDs must be unique for each ECOM, but they do not have to be in sequence. The module's DIP switches must both be set to allow the HTML configuration tool to set a Module ID. Do not use address zero for communications.

Module Name field and **Module Description** fields are optional to identify the module. Click the Send button to write to the module's flash memory.



Ethernet Address: this is the MAC Address. It is a factory-assigned address that is on the permanent label on the module.

IP Configuration: Set IP Address, Subnet Mask and Gateway addresses. Click the Send button to write to the module's flash memory.



The module's current **Booter Version** and **OS Version** are listed. The latest versions can be found by clicking Hosteng.com in the Firmware Updates field.
The module's current **CPU Rev** and **PWB/PLD Rev** are listed. The latest versions can be found by clicking Hosteng.com in the Firmware Updates field.

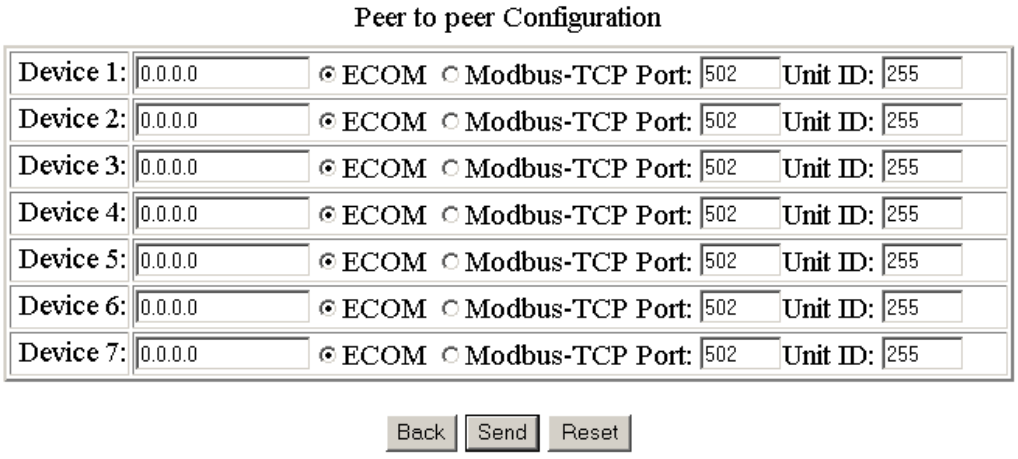
Firmware Updates: If your PC is connected to the internet, clicking on Hosteng.com will take you to Host Engineering's web site where the most current firmware files are available for downloading to your PC. You must use NetEdit3.5 or later to upgrade the module.

H0/H2/H4–ECOM100 Client Peer to Peer Configuration

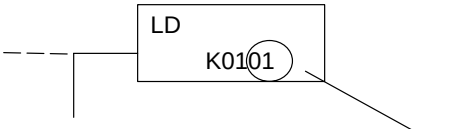
Peer to Peer: The following are reasons or conditions of when to configure the H0/H2/H4–ECOM100 for peer to peer communications with a TCP/IP server device:

- To eliminate network broadcast traffic when the ECOM100 is a client executing RX/WX network instructions.
- When an ECOM100 is an “ECOM protocol client” and needs to access other server ECOMs through a router.
- When an ECOM100 needs to serve as MODBUS TCP/IP client

The default configuration window is shown below.

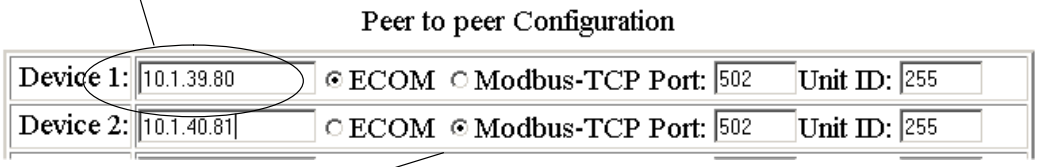


Device Number: This is a “RX/WX node number” that is associated with the server’s IP address that is entered in the field beside the Device number. This node or server (slave) number is used in the RX/WX instruction.



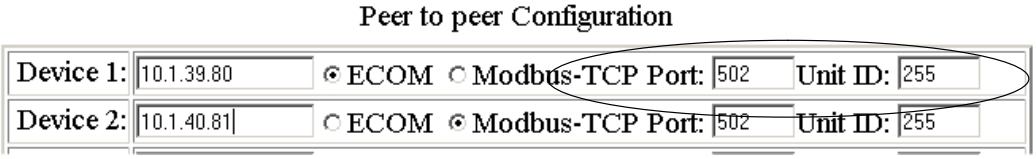
This is the first LD box of the RX/WX instruction. It specifies the ECOM slot number and the server (slave) node number. Use the Device number here.

IP Address: Enter the server’s IP address in the field beside the Device Number. Again, this associates the Device Number to the the entered IP address.



IP Protocol: Select the protocol that the client H0/H2/H4–ECOM100 will be using to communicate with the server device. Select the ECOM radio button if the target device is another ECOM module. Select the MODBUS TCP/IP button if the target device is a MODBUS TCP/IP server.

The **Port Number** and **Unit ID** are only used if the H0/H2/H4–ECOM100 client is trying to communicate with a MODBUS TCP/IP server device through a firewall or an Ethernet–to–serial converter.



Port Number: The MODBUS TCP/IP specification states that all MODBUS TCP devices respond to (server) and send (client) on port 502. This will allow a MODBUS TCP/IP server device to communicate with an H0/H2/H4–ECOM100 MODBUS TCP/IP client through a firewall.

Unit ID: The Unit ID may be required for the following applications:

- MODUS/TCP–to–MODUS/RTU converters: These devices convert Ethernet (TCP) packets to serial (RTU) communications. The Unit ID field is used to direct the TCP packet to the correct, serially connected RTU slave device.
- Some MODUS/TCP slave devices require this; It may be a holdover from an old serial device driver that was converted to Ethernet. The slave device usually requires the Unit ID to be set to “255”, but in some instances the slave requires the Unit ID to be set to “0”. The Modicon specification states to use Unit ID set to “255”. If communications are unsuccessful, it is recommended to try Unit ID of “0”. It is possible you may need to get the actual Unit ID setting from the slave device documentation or manufacturer.

7

Maintenance and Troubleshooting

In This Chapter. . . .

- Isolating a Communication Problem
- Troubleshooting Chart
- ECOM Module Diagnostic LEDs
- Using NetEdit3 for Troubleshooting
- Replacing the ECOM Module

Isolating a Communication Problem

If you are experiencing a problem communicating with an ECOM module, the problem can usually be isolated to one of four components of the communication link:

- the ECOM module itself (hardware or firmware)
- the communication program or the setup of the ECOM module
- the cabling and connections
- other external influences, such as electrical noise, heavy communication traffic on the network or exceeding the PLC power budget

You can also use your PC's DOS "Ping" command to verify communications with a network server. The PC's NIC must have the TCP/IP protocol enabled and the server must have a valid IP address. Visit www.microsoft.com for information on the "Ping" command.

Diagnostic Tools and Techniques

Several available tools and techniques can help you isolate a communication problem:

- The LEDs on the face of the module indicate the status of the link, the module, and the network communications.
- NetEdit3 displays a list of the active modules on the network and their protocol and configuration settings.
- Cable testing devices can pinpoint short or open circuits or diagnose attenuation problems and other cabling problems.

Troubleshooting Chart

The following chart summarizes the different types of communication failures you could experience. In each case the **CPU PWR LED** must be **on**, and you must be attempting to communicate with the ECOM in question. You must be attempting to send or receive RX/WX instructions or you must be attempting to link to the ECOM using our KEP***Direct*** for PLCs or other software driver.

The meaning of the **diagnostic LEDs** is explained begining on page 7-4.

Troubleshooting Chart	
Legend: Off On Flash	
ECOM Module LEDs	**Corrective Action**
ECOM(–F) LINK/LINKGD ACT ERROR/ERR OR **ECOM(–F)** LINK/LINKGD ACT ERROR/ERR	1. Cycle power to the PLC. This will clear the error if it was due to a transient condition. 2. Replace ECOM module
ECOM100 OK / STATUS LINKGD/LINK ACTIVE/ACT ERROR/ERR OR **ECOM100** OK / STATUS LINKGD/LINK ACTIVE/ACT ERROR/ERR	

Troubleshooting Chart (Continued)	
Legend: ▭ Off ■ On ▨ Flash	
ECOM Module LEDs	**Corrective Action**

ECOM100

OK / STATUS ■
LINKGD/LINK ▭
ACTIVE/ACT ▭
ERROR/ERR ▭

ECOM(–F)

LINK/LINKGD ▭
ACT ▭
ERROR/ERR ▭

1. Try another cable that you know works. Check pinouts (see page 2-12).
2. Try another port on the hub or another hub.
3. Replace ECOM module.

ECOM100

OK / STATUS ■
LINKGD/LINK ■
ACTIVE/ACT ▭
ERROR/ERR ▭

ECOM(–F)

LINK/LINKGD ■
ACT ▭
ERROR/ERR ▭

1. Are PLCs in RUN mode? Some communications require initiating and responding PLCs to be in RUN mode.
2. Try another cable between PC and hub or the other communicating ECOM and hub.
3. Try another port on the hub or another hub.
4. Make sure you have not exceeded the recommended cable length for your network cable. The link signal could arrive with sufficient strength even though the data transmission does not.
5. Could be related to Windows configuration. Consult Windows documentation.

Note: This is also the indication of proper operation! Troubleshoot only if you are failing to exchange data.

ECOM(–F)

LINK/LINKGD ■
ACT ▨
ERROR/ERR ▭

OR

ECOM(–F)

LINK/LINKGD ■
ACT ■
ERROR/ERR ▭

ECOM100

OK / STATUS ■
LINKGD/LINK ■
ACTIVE/ACT ▨
ERROR/ERR ▭

OR

ECOM100

OK / STATUS ■
LINKGD/LINK ■
ACTIVE/ACT ■
ERROR/ERR ▭

1. Try another cable between PC and hub or other ECOM and hub.
2. Try another port on the hub or another hub.
3. Confirm that ECOM module is in a usable slot in the PLC base (see pages 2-7 to 2-9) and that the CPU and CPU firmware support the ECOM module.
4. Look for errors in the setup of the ECOM module or in the communication program.

ECOM Module Diagnostic LEDs

H0 Series Indicators

The H0 series ECOM module has four status indicators. The table below explains the status for each indicator.

Indicator	Status	Description
OK (H0–ECOM only)	ON – GREEN	Module is powered up and functional
	OFF	Module powerup failed
LINK	ON – GREEN	Properly connected to network
	OFF	Not connected to network or incorrect configuration
ACT	ON or FLASHING RED	Active network data
	OFF	Network idle
ERR	ON or FLASHING RED	A fatal error has occurred
	OFF	No error present
100 (H0–ECOM100 only)	ON –GREEN	Ethernet data is detected at 100MBit
	OFF	(with ACT LED ON) – Ethernet data is detected at 10MBit

OK Indicator

The **green OK** LED on the H0–ECOM module is **on steady** after a successful powerup. If the LED fails to turn on, the module failed to powerup. It may not be properly installed or it may be defective.

LINK indicator

The **green LINK** (link good) LED is **on steady** when the ECOM module is correctly connected to an active device on the network and is receiving 5VDC operating voltage from the PLC power supply. The LINK LED verifies that the proper cables are connected, and the ECOM module is functioning correctly. If a mismatch with the 10/100BaseT connections occurs this LED will not be illuminated.

ACT Indicator

The **red ACT** (activity) LED **flashes** to indicate that the module sees data travelling on the network. If any network device is sending or receiving data, the ACT LED will be illuminated. In idle mode (no network traffic) this LED is OFF. During heavy communication loads this LED will be **steady on**.

ERR Indicator

If the ECOM module's **red ERROR** indicator is **flashing** or **steady on**, a fatal error has occurred. The error may be in the ECOM module itself, or a network problem may be causing this symptom. The ERROR indication can be caused by a faulty ground, an electrical spike or other types of electrical disturbances. Cycle power to the system to attempt clearing the error.

100MBIT Indicator

The **green 100** (100MB) LED is **on steady** when Ethernet data is detected at 100BaseT frequency. (H0–ECOM100 only)

H24–ECOM–(F) Indicators

The table below describes the three status indicators on the H24–ECOM modules.

Indicator	Status	Description
LINKGD	ON – GREEN	Properly connected to network
	OFF	No connected to network or incorrect configuration
ACT	FLASHES RED	Active network data
	OFF	Network idle
ERROR	ON or FLASHING RED	A fatal error has occurred
	OFF	No error present

LINKGD indicator

The **green LINKGD** (link good) LED is **on steady** when the H24–ECOM–(F) module is correctly connected to an active device on the network and is receiving 5VDC operating voltage from the PLC power supply. The LINKGD LED verifies that the proper cables are connected, and the H24–ECOM–(F) module is functioning correctly. If a mismatch with the 10BaseT or 10BaseFL connections occurs this LED will not be illuminated.

ACT Indicator

The **red ACT** (activity) LED **flashes** to indicate that the module sees data travelling on the network. If any network device is sending or receiving data, the ACT LED will be illuminated. In idle mode (no network traffic) this LED is OFF. During heavy communication loads this LED will be **steady on**.

ERROR Indicator

If the H24–ECOM–(F) module's **red ERROR** indicator is **flashing** or **steady on**, a fatal error has occurred. The error may be in the H24–ECOM–(F) module itself, or a network problem may be causing this symptom. The ERROR indication can be caused by a faulty ground, an electrical spike or other types of electrical disturbances. Cycle power to the system to attempt clearing the error.

H2/H4–ECOM100 Indicators

The H2/H4–ECOM100 module has several status indicators. The table below explains the status for each indicator.

Indicator	Status	Description
STATUS (H2–ECOM100 only)	ON – GREEN	Module is powered up and functional
	OFF	Module powerup failed
LINKGD	ON – GREEN	Properly connected to network
	OFF	Not connected to network or incorrect configuration
ACTIVE	ON or FLASHING RED	Active network data
	OFF	Network idle
ERROR	ON or FLASHING RED	A fatal error has occurred
	OFF	No error present
100MBIT	ON	Ethernet data is detected at 100MBit
	OFF	(with Active LED ON) – Ethernet data is detected at 10MBit

STATUS Indicator

The **green OK** LED on the H2–ECOM100 module is **on steady** after a successful powerup. If the LED fails to turn on, the module failed to powerup. It may not be properly installed or it may be defective.

LINKGD indicator

The **green LINK** (link good) LED is **on steady** when the H2/H4–ECOM100 module is correctly connected to an active device on the network and is receiving 5VDC operating voltage from the PLC power supply. The LINK LED verifies that the proper cables are connected, and the H2/H4–ECOM100 module is functioning correctly. If a mismatch with the 10/100BaseT or 10BaseFL connections occurs this LED will not be illuminated.

ACTIVE Indicator

The **red ACTIVE** (activity) LED **flashes** to indicate that the module sees data travelling on the network. If any network device is sending or receiving data, the ACT LED will be illuminated. In idle mode (no network traffic) this LED is OFF. During heavy communication loads this LED will be **steady on**.

ERROR Indicator

If the H2/H4–ECOM100 module's **red ERROR** indicator is **flashing** or **steady on**, a fatal error has occurred. The error may be in the H2–ECOM100 module itself, or a network problem may be causing this symptom. The ERROR indication can be caused by a faulty ground, an electrical spike or other types of electrical disturbances. Cycle power to the system to attempt clearing the error.

100MBIT Indicator

The **green 100M** (100MB) LED is **on steady** when Ethernet data is detected at 100BaseT frequency.

Using NetEdit3 for Troubleshooting

NetEdit3 is a software utility which came with this manual. To review the procedures for running and using NetEdit3, see pages 3–5 through 3–12. NetEdit3 allows you to:

- See active modules on the network.
- Examine and change the modules' configuration settings.
- See the firmware revision number.
- Review statistical information about communication errors by type.

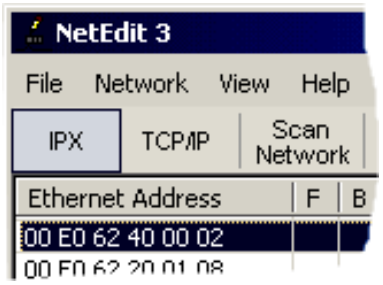
If you can see the ECOM module on the list in the Module box (described below), you are *linking* to the module from your PC. If you are linking to the module but failing to communicate with the module, you can conclude that:

- The module is working.
- The cabling is satisfactory from the PC to the hub and from the hub to the ECOM module.
- The hub is working.
- The problem is in one of the other components of the communication link.

Select a Module

The Module box shows the Ethernet (MAC) Addresses of all modules which are currently linked to the NetEdit3 utility. If your ECOM module is not on this list, try the following:

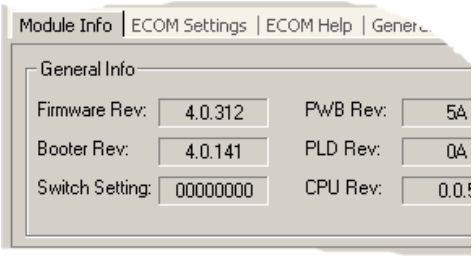
- Change Protocol selection and click on Scan Network. See **Change Protocol** on the next page.
- Confirm that your PC has IPX or TCP/IP protocol loaded.
- Confirm that the ECOM module's LINKGD LED is on.



NOTE: The Ethernet (MAC) Address is permanently assigned at the factory, and it is recorded on a label on the side of the ECOM module. See page 2-4 if you need help locating the label.

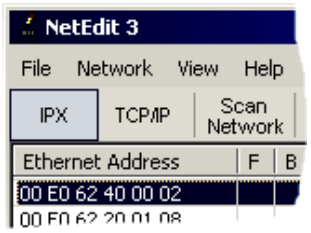
Module Information

The Module Information box gives the ECOM module Type, firmware Version, Booter firmware version and the address DIP switch setting. Verify that all modules *of the same type* have the same firmware version.



Change Protocol

If you are experiencing a problem communicating from your PC to a module that *does not* appear on the list of active modules, try changing the protocol and clicking on Scan Network. You may be able to link to your module with the other protocol.



If you are not sure which protocol you have loaded on your PC, refer to pages 3–4 and 3–5, as well as your Windows documentation. This protocol selection is only for PC-to-PLC communications and has no effect on PLC-to-PLC communications.

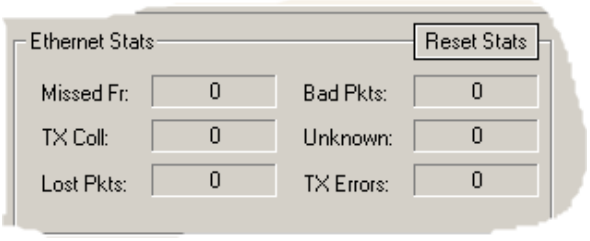
Ethernet Stats

If you are able to see the *problem* module on the list of modules currently active on the network, you can **select** the module to see the Ethernet Stats for that module. Select the module by clicking on the Ethernet (MAC) Address in the Module box.

To begin a new statistical record, click the Clear Stats button.

The diagnostic information available in the Ethernet Stats box is:

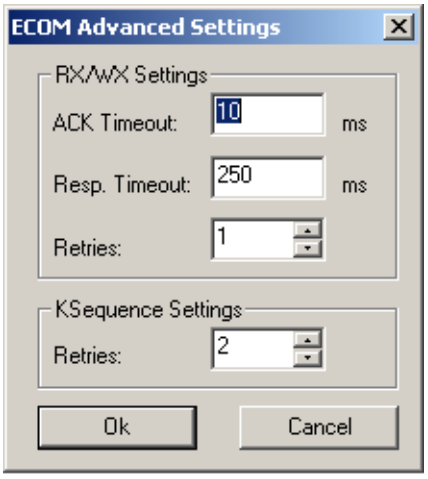
- Missed Frames – frames lost due to unavailability of buffer space.
- TX Collisions – detected when RXD+ and RXD– become active during a data transmission. Two devices are trying to communicate at the same time.
- Lost Packets – packets that overflow the queue.
- Bad Packets – packets that fit the Ethernet standard but are not in the right format for the ECOM module.
- Unknown Type – a foreign command was received and could not be interpreted. This will probably happen only during software driver development.
- Send Errors – the Ethernet standard number of retries were attempted for a transmission.



RX/WX Settings

The RX/WX Settings box in this window provides a place to make changes that affect PLC-to-PLC communications *only*.
Change these settings only if:

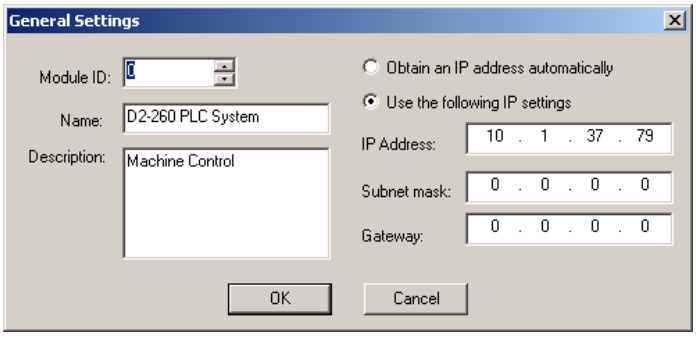
- The LINK/LINKGD LED is on.
- The ACT LED is flashing.
- You have explored the module configuration and RLL programming for possible errors.



Record the Module Settings

If you are replacing an existing module, record the existing network identifier settings in the Configuration box. Set up the replacement module by linking to it with NetEdit3 and inserting the same network identifiers in the Configuration box. Click on Update Module to store the settings in the ECOM module's flash memory.

Each module on the network must have a unique network identifier.



Replacing the ECOM Module

If you are experiencing a communication problem, and you think the ECOM may be defective, try replacing the ECOM module with a new one or one that you know is working properly. If that corrects the problem, then the original module is probably defective (assuming the replacement module was set up the same as the original and no changes have been made to the communication program or other network components). The diagnostic LEDs will help you determine which applies in this case. If replacing the module does not correct the problem, then the problem is probably not in the module.

If you are replacing an existing ECOM module with a new one, you need to set up the new module with the same network identifiers as the module you are replacing. If you used the **DIP switch** to set the **Module ID**, you will need to set the DIP switch on the replacement module to the same Module ID. See pages 2-5 or 2-6 to review the procedure for setting the Module ID using the DIP switch.

If you set up your original ECOM module using one of the **software utilities**, you will need to refer to your record of the module's network identifiers (see page 2-2) and duplicate the settings in the new module using the same procedures. See page 3–5 through 3-12 to review the procedures for using the software utilities.

To replace the ECOM module, disconnect power from the PLC and remove the original module. Insert the replacement ECOM module, connect the module to your network and apply power to the PLC. Links to other network devices will automatically be reestablished and communications between devices will resume.



WARNING: Your system can be damaged if you install or remove system components before disconnecting the system power. To minimize the risk of equipment damage, electrical shock, or personal injury, always disconnect the system power before installing or removing any system component.

Diagnosing Network Cable Problems

If you are experiencing communication problems, swapping cables is one of the simplest diagnostic procedures you can perform. If the network operates correctly with a different cable, you have isolated and cured the problem. If possible, use a short run of cable to test the network because problems with longer cable runs can be more difficult to diagnose and are more often intermittent.

If you are unable to swap cables, verify the proper operation of all other network components. You probably have a cable problem if you have verified that your:

- ECOM module is working correctly.
- ECOM module configuration is correct.
- RLL program or PC program is correct.
- hubs are working correctly.
- Windows configuration is correct.
- network adapter card is the correct type, and it is working correctly.

It is a good maintenance practice to test network cables periodically and maintain a permanent record of cable characteristics. A number of cable test instruments are available to test 10/100BaseT and 10BaseFL networks. These instruments will check the electrical or optical characteristics of your cabling, including:

- Continuity – This is a check to make sure the communication pairs are wired correctly, and that the wires are continuous from end to end. In the case of fiber optic network this is a test to be sure light is transmitted from one end of the cable to the other.
- Attenuation – This refers to the amount of signal loss over the cable segment at the signal frequency of interest. The 10/100BaseT specification allows for a maximum signal loss of 11.5 decibels (dB) for the entire link at the signal frequency used by 10Mbps Ethernet. The 10BaseFL specification calls for the optical loss in link segment to be no greater than 12.5 dB.
- Crosstalk – Crosstalk occurs when a signal in one pair of wires is electromagnetically coupled to an adjacent pair. This is critical for10BaseT networks which are susceptible to noise interference. 10BaseFL networks are virtually immune to noise interference.



NOTE: Any significant difference between the cable characteristics of the transmitter and receiver can cause communication errors.

Ethernet devices continually monitor the receive data path for activity as a means of verifying their link is working correctly. When the network is idle, each network device (including the ECOM module) sends a periodic *link test* signal to verify that the network is working. If the link test signal or other network activity is not received periodically, the LINK/LINKGD LED on the ECOM module is turned off.

Appendix A General Specifications

A

In This Appendix

— H0–ECOM Specifications
— H2–ECOM and H4–ECOM Specifications
— H0–ECOM100 Specifications
— H2–ECOM100 and H4–ECOM100 Specifications
— H2–ECOM–F and H4–ECOM–F Specifications
— Ethernet Standards

General Specifications

ECOM Specifications

H0–ECOM	
Module Type	Intelligent Data Communications Module
Quantity of Modules Per Base	Defined by CPU and base configuration
Diagnostics	LEDs/Network Monitoring Software (NetEdit3)
Communications	10BaseT Ethernet
Data Transfer	10 Million bits per second
Extension Port	RJ45
OK Indicator (OK)	Green LED
Link Good Indicator (LINK)	Green LED
Activity Indicator (ACT)	Red LED
Error Indicator (ERR)	Red LED
Power Consumption	250mA (Supplied by DL05/DL06 base)
Operating Temperature	32° to 140° F (0° to 60° C)
Storage Temperature	–4° to 158° F (–20° to 70° C)
Relative Humidity	30% – 95% RH (non-condensing)
Environmental Air	No corrosive gases permitted
Networking Protocols Supported	TCP/IP, IPX
Manufacturer	Host Automation Products
Link Distance	100 meters (328 feet)

H2–ECOM / H4–ECOM	
Module Type	Intelligent Data Communications Module
Quantity of Modules Per Base	Defined by CPU and base configuration
Diagnostics	LEDs, Network Monitoring Software (NetEdit3)
Communications	10BaseT Ethernet
Data Transfer	10 Million bits per second
Extension Port	RJ45
Link Good Indicator (LINKGD)	Green LED
Activity Indicator (ACT)	Red LED
Error Indicator (ERROR)	Red LED
Power Consumption	H2: 450mA; H4: 530mA (Supplied by base)
Operating Temperature	32° to 140° F (0° to 60° C)
Storage Temperature	–4° to 158° F (–20° to 70° C)
Relative Humidity	30% – 95% RH (non-condensing)
Environmental Air	No corrosive gases permitted
Networking Protocols Supported	TCP/IP, IPX
Manufacturer	Host Automation Products
Link Distance	100 meters (328 feet)

H0–ECOM100	
Module Type	Intelligent Data Communications Module
Quantity of Modules Per Base	Defined by CPU and base configuration
Diagnostics	LEDs, Network Monitoring Software (NetEdit3)
Communications	10/100BaseT Ethernet
Data Transfer	100 Million bits per second
Extension Port	RJ45
Link Good Indicator (LINK)	Green LED
Activity Indicator (ACT)	Green LED
Error Indicator (ERR)	Red LED
100MB Transfer Rate (100M)	Green LED
Power Consumption	300mA (Supplied by DL05/06 base)
Operating Temperature	32° to 140° F (0° to 60° C)
Storage Temperature	–4° to 158° F (–20° to 70° C)
Relative Humidity	30% – 95% RH (non-condensing)
Environmental Air	No corrosive gases permitted
Networking Protocols Supported	TCP/IP, IPX, MODBUS TCP/IP, DHCP, HTML Configuration
Manufacturer	Host Automation Products
Link Distance	100 meters (328 feet)

H2–ECOM100 / H4–ECOM100	
Module Type	Intelligent Data Communications Module
Quantity of Modules Per Base	Defined by CPU and base configuration
Diagnostics	LEDs, Network Monitoring Software (NetEdit3)
Communications	10/100BaseT Ethernet
Data Transfer	100 Million bits per second
Extension Port	RJ45
Status Indicator (STATUS) (H2–ECOM100 only)	Green LED
Link Good Indicator (LINKGD)	Green LED
Activity Indicator (ACTIVE)	Green LED
Error Indicator (ERROR)	Red LED
100MB Transfer Rate (100MBIT)	Green LED
Power Consumption	300mA (Supplied by DL base)
Operating Temperature	32° to 140° F (0° to 60° C)
Storage Temperature	–4° to 158° F (–20° to 70° C)
Relative Humidity	30% – 95% RH (non-condensing)
Environmental Air	No corrosive gases permitted
Networking Protocols Supported	TCP/IP, IPX, MODBUS TCP/IP, DHCP, HTML Configuration
Manufacturer	Host Automation Products
Link Distance	100 meters (328 feet)

H2–ECOM–F / H4–ECOM–F	
Module Type	Intelligent Data Communications Module
Quantity of Modules Per Base	Defined by CPU and base configuration
Diagnostics	LEDs, Network Monitoring Software (NetEdit3)
Communications	10BaseFL Ethernet (fiber optic)
Data Transfer	10 Million bits per second
Extension Port	ST-style fiber optic connector
Link Good Indicator (LINKGD)	Green LED
Activity Indicator (ACT)	Red LED
Error Indicator (ERROR)	Red LED
Power Consumption	H2: 640mA; H4: 670mA (Supplied by base)
Operating Temperature	32° to 140° F (0° to 60° C)
Storage Temperature	–4° to 158° F (–20° to 70° C)
Relative Humidity	30% – 95% RH (non-condensing)
Environmental Air	No corrosive gases permitted
Networking Protocols Supported	TCP/IP, IPX
Manufacturer	Host Automation Products
Link Distance	Up to 2,000 meters (2Km), 6,560ft (1.2 miles)

Ethernet Standards

Various institutes and committees have been involved in establishing Ethernet data communication standards. These specification standards assure Ethernet network compatibility for products from a broad variety of manufacturers.

The ECOM module complies with American National Standards Institute (ANSI) and Institute of Electrical and Electronic Engineers standard ANSI/IEEE 802.3, Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Methods and Physical Layer Specifications. This standard has been adopted by the International Organization for Standardization (ISO) as document ISO/IEC 8802–3.

The Electronic Industries Association (EIA) and Telecommunications Industries Commercial Building Telecommunications Wiring Standard designated EIA/TIA–568A defines implementation of 10BaseT (twisted pair) and 10BaseF (fiber optics) for Ethernet communications.

The same two organizations produced EIA/TIA TSB40–Additional Transmission Specifications for Unshielded Twisted-Pair Connecting Hardware. The purpose of this document is to specify transmission performance requirements and connecting hardware requirements.

Artisan Technology Group is an independent supplier of quality pre-owned equipment

Gold-standard solutions

Extend the life of your critical industrial, commercial, and military systems with our superior service and support.

We buy equipment

Planning to upgrade your current equipment? Have surplus equipment taking up shelf space? We'll give it a new home.

Learn more!

Visit us at **artisantg.com** for more info on price quotes, drivers, technical specifications, manuals, and documentation.

Artisan Scientific Corporation dba Artisan Technology Group is not an affiliate, representative, or authorized distributor for any manufacturer listed herein.

We're here to make your life easier. How can we help you today?

(217) 352-9330 | sales@artisantg.com | artisantg.com

ARTISAN® TECHNOLOGY GROUP