Errata Sheet

This Errata Sheet contains corrections or changes made after the publication of this manual.

Product Family:	DirectLOGIC PLCs	**Date**:	January 2019
Manual Number	H24-EBC-M		
Revision and Date	3rd Ed., Rev. A, May 2014		

Changes to Chapter 2. Installing the H2-EBC(100), H2-EBC-F or H4-EBC(-F)

Page 2-2. Setting the Module ID; Setting the Module ID with DIP Switches

Add the following to this section:

This table describes the DIP switch settings for the various EBC modules.

Module Type	Release Version	Dipswitch							
		7	6	5	4	3	2	1	0
H2-EBC **H2-EBC-F**	v2.1.80	Not used		Module ID					
H2-EBC100	v4.0.477	Not used		Module ID					
	v4.0.490	Not used	Recover	Module ID					
H4-EBC **H4-EBC-F**	v2.1.1	Not used		Module ID					

Recover - Dipswitch used to reset IP settings back to factory default on power-up in case device is lost on network.
Module ID - Bit-weighted for manual setting of this parameter.

Changes to Apppendix A. General Specifications

Page A-2. H2 Series and H4 Series EBC Specifications

In the top table, change the "Serial Port (RJ12)" specification in the third column for the H2-EBC100 to read:

"K-Sequence, ASCII IN/OUT, Modbus RTU, Provides 5V 220mA"

Page A-2. Serial Port Specifications

In this table change the description for pin 2 to read:

"5V Power Out, 220 mA"

Ethernet

Base Controller

Manual Number H24-EBC-M

WARNING

Thank you for purchasing automation equipment from ***Automationdirect.com***™, doing business as **AutomationDirect**. We want your new 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.

AVERTISSEMENT

Nous vous remercions d'avoir acheté l'équipement d'automatisation de **Automationdirect.comMC**, en faisant des affaires comme **AutomationDirect**. Nous tenons à ce que votre nouvel équipement d'automatisation 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 commerc

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.

Manual Revisions

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

Title: Ethernet Base Controller
Manual Number: H24-EBC-M

Edition	Date	Description of Changes
Original	10/98	Original issue
2nd Edition	11/01	Added KEPwareEX OPC use
2nd Edition, Rev A	08/02	Minor changes Updated for NetEdit 2.4
3rd Edition	11/04	Added H2-EBC100 Updated for NetEdit3
3rd Edition, Rev A	5/14	Added Appendix F. H2-EBC(100) Analog Module Addressing

Table of Contents

Chapter 1: Introduction

Chapter 2: Installing the H2-EBC(100), H2-EBC-F or H4-EBC(-F)

Chapter 3: Configuring the EBCs Using NetEdit3

Chapter 4: MODBUS TCP/IP for H2-EBC100

Chapter 5: H2-EBC100 DHCP & HTML Configuration

Chapter 6: Troubleshooting Guidelines

Appendix A: General Specifications

Appendix B: Using the H2 Series EBC with Think & Do

Appendix C: Using the H4 Series EBC with Think & Do

Appendix D: Using the H2 Series EBC with KEPDirect OPC Server

Appendix E: Using the KEPDirect OPC Quick Client

Appendix F: H2–EBC(100) Analog Module Addressing

1

Introduction

In This Chapter. . . .

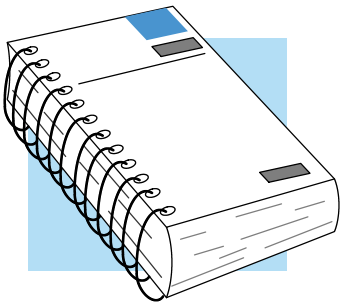
— Manual Overview

— Ethernet Base Controller Overview

Manual Overview

Overview of this Manual

This manual describes the installation and operation of the **Ethernet Base Controller (EBC)**. You will find the necessary information for configuring the H2-EBC, H2-EBC100, H2-EBC-F, H4-EBC and H4-EBC-F, installing the module in a DL205 or DL405 I/O base and connecting the EBC to a 10BaseT, 100BaseT or 10BaseFL Ethernet network. In this manual, the phrase "H2 Series EBCs" will be used when the subject applies to the H2-EBC, H2-EBC100 and H2-EBC-F. Otherwise, the specific H2 Series EBC part number will be listed. Also, the phrase "H4 Series EBCs" is used when the subject applies to both the H4-EBC and H4-EBC-F. Otherwise, the specific H4 Series EBC part number will be listed. The term "EBC" will be used when the subject applies to all of the EBC modules.



Other Reference Materials

You may find other technical manuals useful for your application. For technical information related to your PC-based control software or your PC, please refer to the appropriate manual for that product. For more information about the ***Direct***LOGIC™ products, you may want to read the following:

- DL205 Installation and I/O Manual (D2-INST-M)
- DL405 Installation and I/O Manual (D4-INST-M)

Who Should Read This Manual

You will find this manual helpful for setup and installation if you have chosen to use all of the following:

- Network master – PC-based Control with embedded Ethernet I/O drivers, KEP***Direct*** EBC I/O Server or ***Direct***LOGIC PLCs/WinPLC using the Ethernet Remote Master (ERM) module
- Automationdirect ***Direct***LOGIC DL205 or DL405 I/O

A familiarity with Ethernet communications and with the setup and installation of PLCs is helpful. An understanding of electrical codes and industrial control is essential.

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 technical assistance, please call us at

770-844-4200

Our technical support team is happy to work with you in answering your questions. They are available **weekdays from 9:00 a.m. to 6:00 p.m. Eastern Time**. We also encourage you to visit our website where you can find technical and nontechnical information about our products and our company.

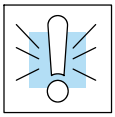
www.automationdirect.com

If you have a comment or question about any of our products, services, or manuals, please fill out and return the 'Suggestions' card that was shipped 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.

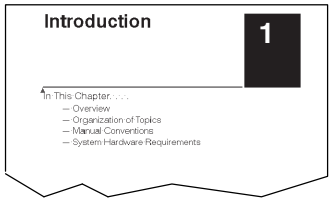
When you see the “notepad” icon in the left-hand margin, the paragraph to its immediate right will be a **special note**.
The word **NOTE:** in boldface will mark the beginning of the text.



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.



Ethernet Base Controller Overview

The Ethernet Base Controllers provide a low-cost, high-performance Ethernet link between a network master controller and an Automationdirect DL205/405 I/O slave system. Network masters include the DL205, DL405 ***Direct***Logic PLCs and WinPLCs using the Ethernet Remote Master module (ERM), and PCs using PC-based control software that includes embedded Ethernet I/O drivers or through a compatible OPC server. The H2-EBC100 also supports the MODBUS TCP/IP client/server protocol.

The Ethernet Base Controller serves as an interface between the master control system and the DL205/405 I/O modules. The control function is performed by the master controller, not the EBC slave. The EBC occupies the CPU slot on the base and communicates across the backplane to input and output modules. The function of the EBC is to:

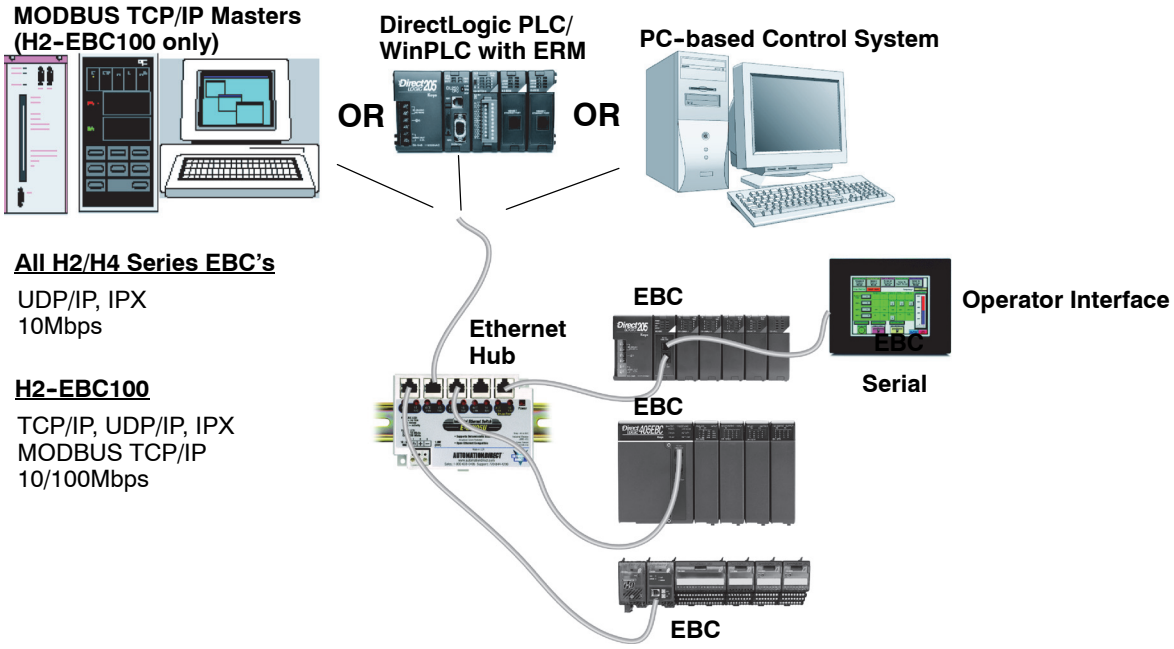
- process analog and digital input signals
- format the I/O signals to conform to the Ethernet standard
- transmit the signals to the network master
- receive and translate output signals from the network master
- distribute the output signals to the appropriate output module in the base

I/O Values Stored in Cache Memory

The EBC module continually scans all I/O and stores the most recent values in cache memory. The cache memory contents are available to the master controller as a block of data or by individual slot location. The EBC reads all channels of digital and analog modules on each scan.

Typically, the network master will request *all* input and output values at the same time from the EBC. The EBC passes the cache memory values for all channels of all input and output modules. By using this method, very fast response times can be achieved by the network master control system. Various master controllers with EBC slaves are shown below.

Example EBC Systems: Various Masters with EBC Slaves



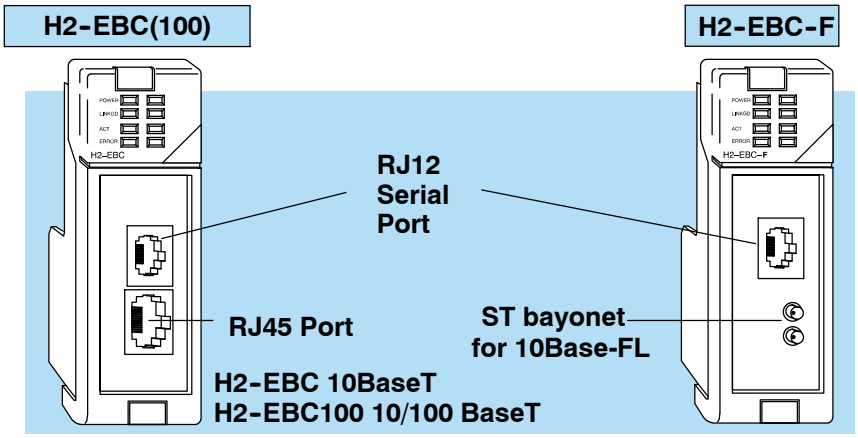
Industry Standard Ethernet

The H2-EBC and H4-EBC modules support industry standard 10BaseT Ethernet communications, the H2-EBC100 module supports industry standard 10/100Base T Ethernet communications and the H2-EBC-F and H4-EBC-F modules support 10BaseFL (fiber optic) Ethernet standards.

WARNING: For deterministic Ethernet communication you must use a dedicated network of EBC modules connected to your master control system. The EBC modules and the master controller must be the only devices on the network.

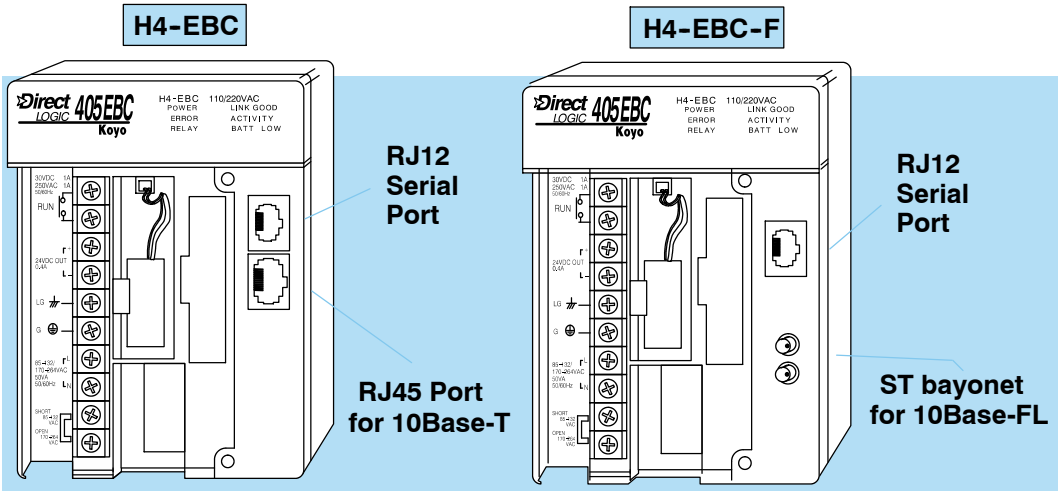
H2-EBC(100) and H2-EBC-F

The H2 Series EBCs install in the CPU slot of a DL205 base and communicates across the backplane to digital and analog input and output modules. The H2 Series EBC modules do not support remote I/O or Specialty Modules, except for the H2-SERIO and H2-CTRIO module. The H2-SERIO is supported when used in a WinPLC/ERM system, but not in a ***Direct***Logic PLC/ERM sytem.



H4-EBC and H4-EBC-F

The H4 Series EBCs install in the CPU slot of a DL405 base and communicates across the backplane to digital and analog input and output modules. The H4 Series EBCs support up to three expansion I/O bases (see page 2-19), and supports the H4-CTRIO and D4-HSC High Speed Counter Module. The H4 Series EBCs also serve as the power supply for the local base. Expansion bases have their own power supplies.



RS232C Serial Port

An RS232C serial port on-board the EBC module allows serial communication to an operator interface device or other serial device. See your PC-based Control software documentation to determine whether this EBC feature is supported.

2

Installing the H2-EBC(100), H2-EBC-F or H4-EBC(-F)

In This Chapter. . . .

EBC Network Identifiers

Each EBC module must be assigned at least one unique identifier to make it possible for PCs or other clients (masters) to recognize it on the network. Two methods of identifying the EBC module give it the flexibility to fit most networking schemes.

The identifiers are:

- Module ID (IPX protocol only)
- IP Address (for TCP/IP and MODBUS TCP/IP protocols); see Chapter 3

Setting the Module ID

If using the IPX protocol for network communications, each Ethernet Base Controller must have a Module ID (Node Address) in order to be recognized on the network, and each Module ID must be unique. Duplicate Module ID on the same network will cause unpredictable results and must be avoided.

Several Methods for Setting Module ID

There are several methods for setting the Module ID:

- The **DIP switch** on the EBC module
- The **NetEdit3** software utility (see Chapter 3)
- HTML Configuration (after IP address is assigned to module using NetEdit3; described in Chapter 5; H2-EBC100 only)
- The software utility in your **PC-based Control software** (if a utility is provided)

Setting Module ID with DIP Switches

We recommend using the DIP switch to set the Module ID because the DIP switch is simple to set, and the Module ID can be determined by looking at the physical module, without reference to a software utility.

See Errata Sheet at the beginning of this file for additional DIP switch information.

You can use the DIP switch to set the Module ID to a number from 1 - 63. 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 (all switches OFF).

The DIP switch settings are read only at powerup. You must cycle power if you change the DIP switches.

Setting Module ID with Software Tool

Software changes to the Module ID do not require cycling power. To set the Module ID using one of the available software tools, do the following:

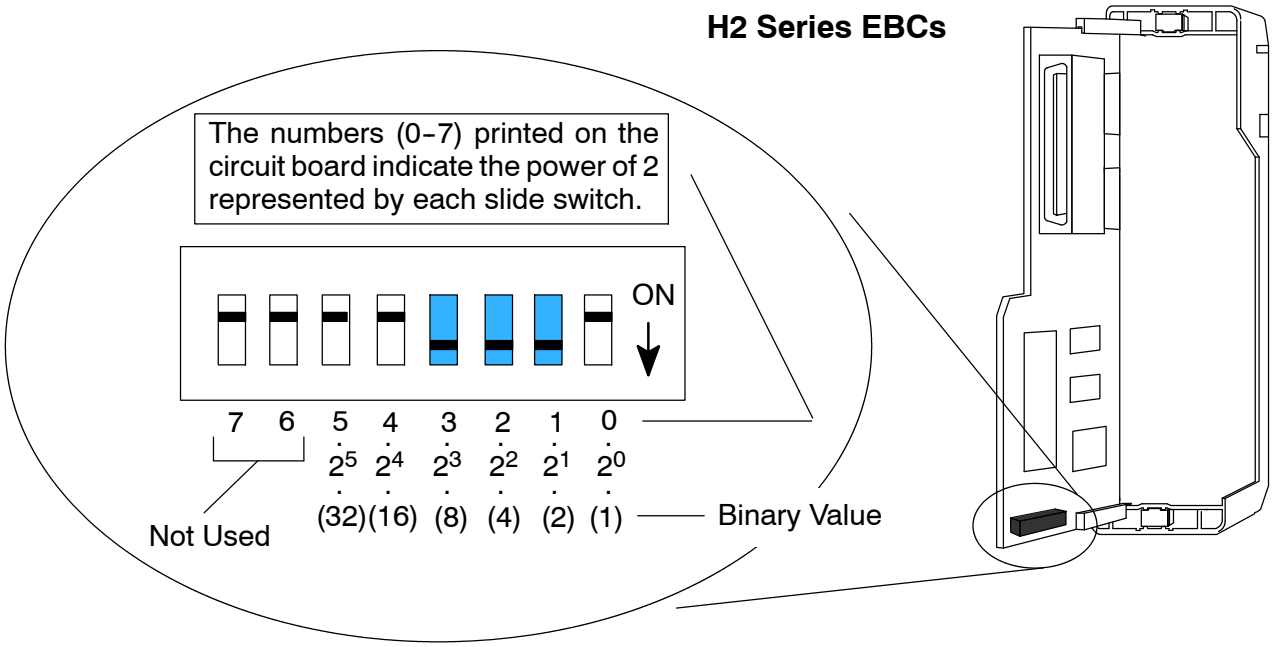
- Check to be sure all DIP switches are set to the off position, Module ID = 0 (see page 2-3 to 2-4)
- Insert the module in the base (see page 2-5)
- Connect the power wiring (see page 2-6 to 2-7)
- Connect module to the Ethernet network (see page 2-8 to 2-10)
- Apply power
- Link to the module and change the Module ID using the software of your choice. Remember to update the module before exiting the software. See note below.

NOTE: Set the Module ID using the method recommended for your PC-based Control software. The use of NetEdit3 to set the Module ID is described in Chapter 3. Some PC-based Control software packages may make automatic updates to the EBC module configuration, overwriting the configuration developed in NetEdit3.

The H2 Series EBC DIP Switch

The H2-EBC(100) & H2-EBC-F DIP Switch

The EBC DIP switch contains eight individual switches, but only six of these are active. You will find that the printed circuit board is labeled 0 – 7. The numbers on the printed circuit board indicate the power of 2 represented by each individual 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 ( ).

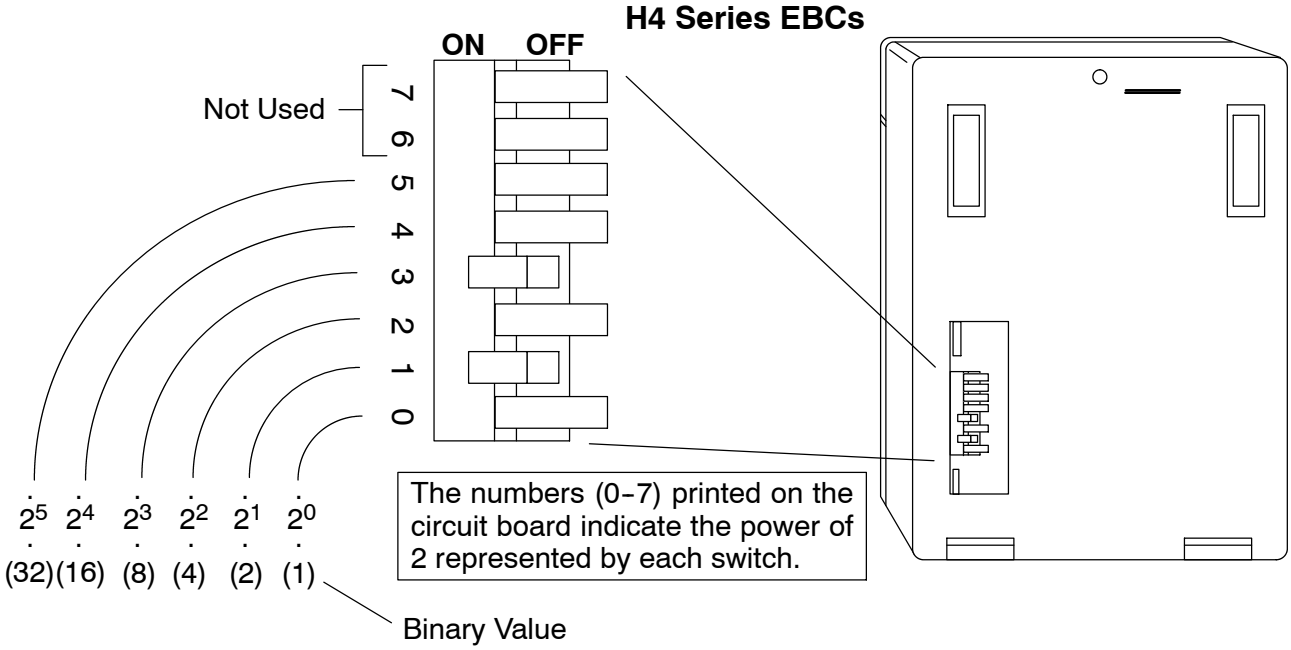


The Module ID equals the *sum* of the binary values of the slide switches set in the ON position. For example, if you set slide switches 1, 2, and 3 to the ON position, the Module ID will be 14. This is found by adding 8+4+2=14. The maximum value you can set on the DIP switch is 32+16+8+4+2+1=63. This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

The H4 Series EBC DIP Switch

The H4-EBC(-F) DIP Switch

The EBC DIP switch contains eight individual switches, but only six of these are active. Two are not used. Notice that the individual switches are labeled 0 – 7 on the printed circuit board. The numbers on the printed circuit board indicate the power of 2 represented by each individual 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 ( ).

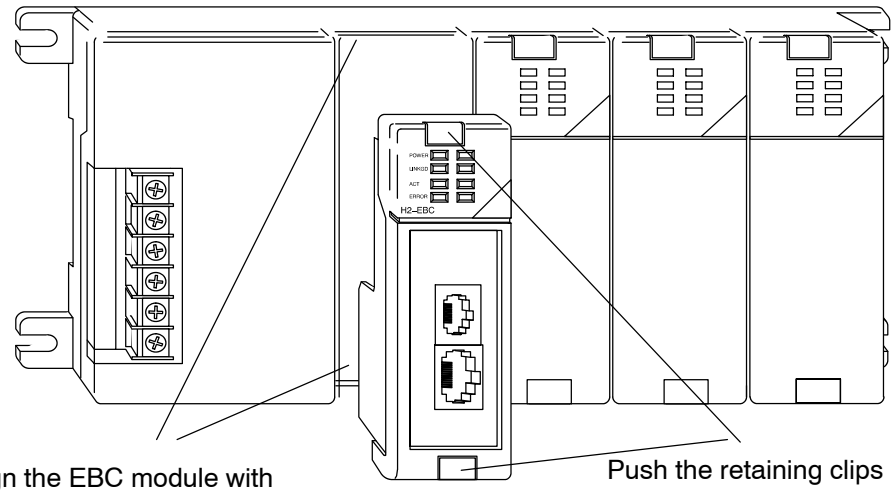


The Module ID equals the *sum* of the binary values of the individual switches set in the ON position. For example, if you set switches 1 and 3 to the ON position, the Module ID will be 10. This is found by adding 8+2=10. The maximum value you can set on the DIP switch is 32+16+8+4+2+1=63. This is achieved by setting switches 0 through 5 to the ON position. The DIP switch must be set to a number greater than zero.

Inserting the H2 Series EBC into the Base

The EBCs plug into the CPU slot of any DL205 base.

- Locate the grooves on the inside top and bottom of the DL205 base.
- Align the module with the grooves and slide the module into the slot until the face of the module is flush with the power supply.
- Push in the retaining clips to secure the module.



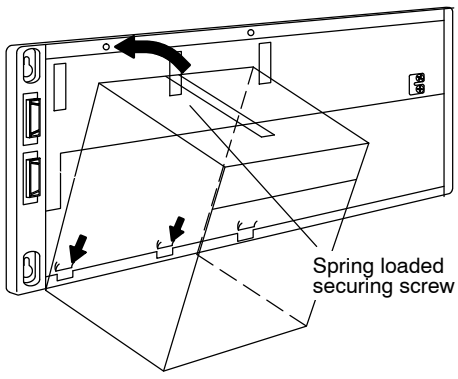
Align the EBC module with grooves in the base and slide it in.

Push the retaining clips in to secure the module in the DL205 base.

Intalling the H4 Series EBCs onto the Base

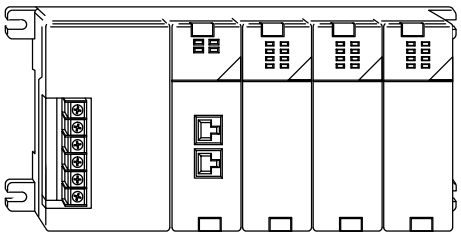
The EBCs installs in the CPU position of any DL405 I/O base.

- The EBC has two plastic tabs at the bottom and a screw at the top.
- With the device tilted as shown, hook the plastic tabs into the notches at the bottom of the base.
- Gently push the top of the module toward the base until the back of the module is flush with the base.
- Tighten the screw at the top of the device to secure it to the base.



DL205 Power Wiring and Grounding

The power wires for the DL205 are ***not*** connected directly to the H2 Series EBCs as they are on the H4 Series EBCs. The DL205 power supply is an integral part of the base and separate from the EBC. The DL205 also has three power options: 12/24VDC, 125VDC, and 120/240VAC.

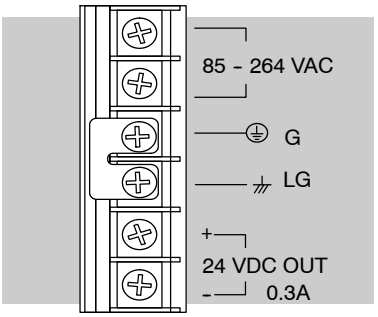


Base Wiring

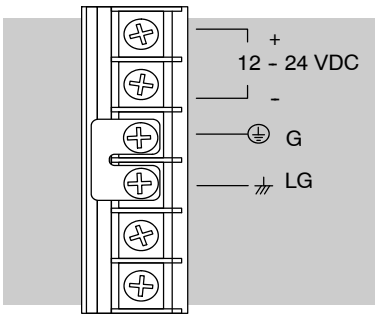
The diagrams show the terminal connections located on the power supply of the DL205 bases. The base terminals can accept up to 16 AWG. You may be able to use larger wiring depending on the type of wire used, but 16 AWG is the recommended size. Do not overtighten the connector screws; recommended torque value is 7.81 pound-inches (0.882 N•m).

NOTE: You can connect either a 115 VAC or 220 VAC supply to the AC terminals. Special wiring or jumpers are not required as with some of the other ***Direct***LOGIC™ products.

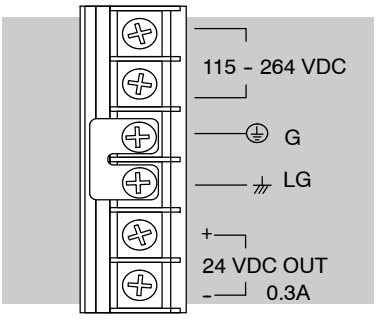
110/220 VAC Base Terminal Strip



12/24 VDC Base Terminal Strip



125 VDC Base Terminal Strip



WARNING: Once the power wiring is connected, install the plastic protective cover. When the cover is removed there is a risk of electrical shock if you accidentally touch the wiring or wiring terminals.

H4 Series EBC Power Wiring and Grounding

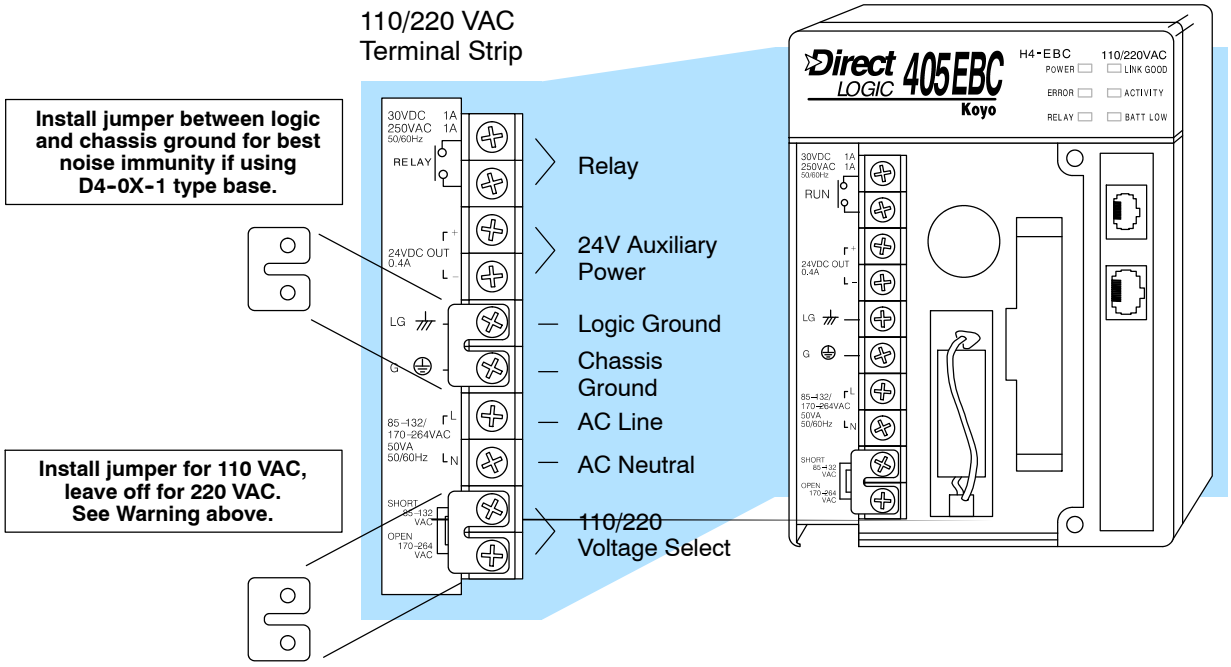
The power connection terminals are under the front cover of the Ethernet Base Controller. The list below describes the function of each of the terminal screws.

- **Relay** – normally-open contact indicates that the EBC's link to hub or PC is good. Link Good indicator light is also on.
- **24VDC Auxiliary Power** – can be used to power field devices or I/O modules requiring external power. It supplies up to 400 mA of current at 20–28VDC, ripple less than 1 V P-P.
- **Logic Ground** – internal ground to the system which can be tied to field devices or communication ports to unite ground signals.
- **Chassis Ground** – where earth ground is connected to the unit.
- **AC Power** –where the line (hot) and the neutral (common) connections are made to the EBC.
- **110/220 Voltage Select** – a jumper across two of the terminals determines the voltage selection. Install the jumper to select 110VAC input power, or remove the jumper to select 220VAC power input.

WARNING: Damage will occur to the power supply if 220 VAC is connected to the terminal connections with the 115 VAC jumper installed. Once the power wiring is connected, install the protective cover to avoid risk of accidental shock.

EBC Wiring

The following diagram shows the appropriate connections for each terminal. Note that you should install a jumper between logic ground and chassis ground for best noise immunity.

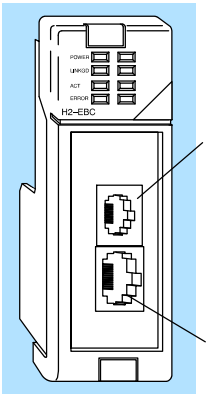


10BaseT/100BaseT Network Cabling

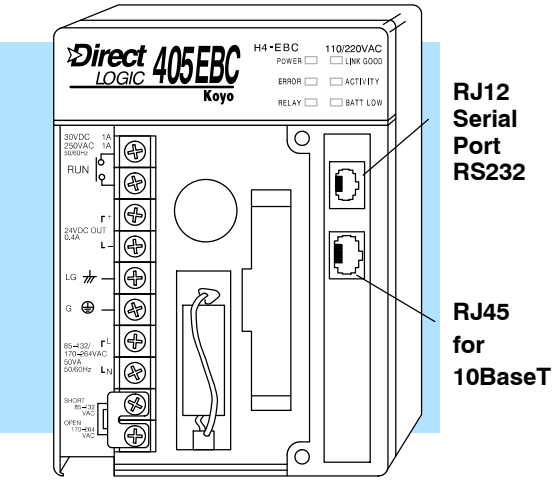
EBC Supports Two Standards

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

H2-EBC(100)



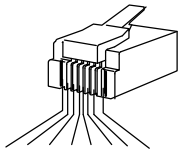
H4-EBC



10/100BaseT Connections

The 10BaseT and 100BaseT EBCs have an eight-pin modular jack that accepts RJ45 connector plugs. 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 (CAT5) cable for all Ethernet 10/100BaseT connections. For convenient and reliable networking, we recommend that you purchase commercially manufactured cables (cables with connectors already attached).

10/100BaseT



8-pin RJ45 Connector (8P8C)

To connect an EBC (or PC) to a hub or repeater, use a **patch cable** (sometimes called a straight-through cable). The cable used to connect a PC *directly* to an EBC or to connect two hubs is referred to as a **crossover cable**.

Patch (Straight-through) Cable

EBC			HUB
TD+	1	OR/WHT — OR/WHT	1 RD+
TD-	2	OR — OR	2 RD-
RD+	3	GRN/WHT — GRN/WHT	3 TD+
	4	BLU — BLU	4
	5	BLU/WHT — BLU/WHT	5
RD-	6	GRN — GRN	6 TD-
	7	BRN/WHT — BRN/WHT	7
	8	BRN — BRN	8
RJ45			RJ45

Crossover Cable

EBC			PC
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

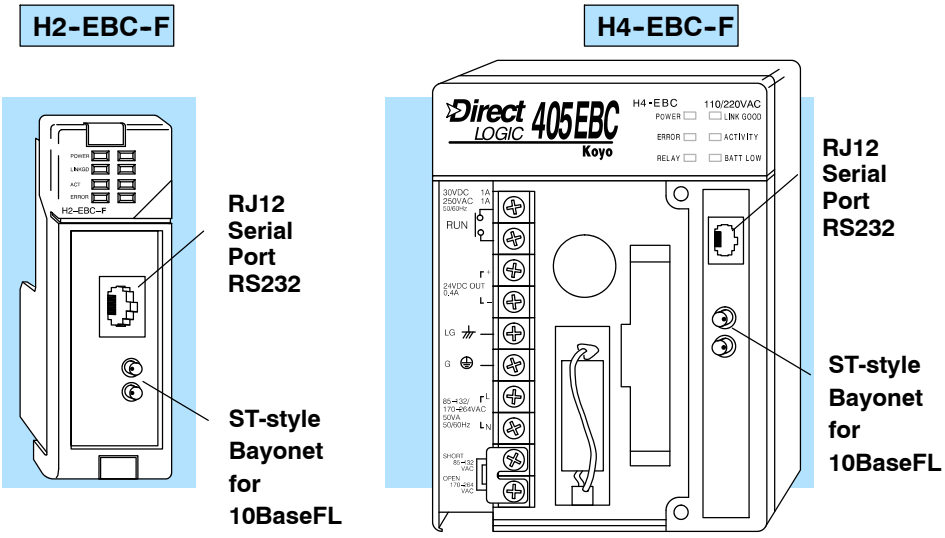
This diagram illustrates the standard wire positions in the RJ45 connector. We recommend all EBC 10/100BaseT cables to be **Category 5**, UTP cable.

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

10BaseFL Network Cabling

EBC Supports Two Standards

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



10BaseFL Connections

Each module has two ST-style bayonet 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.

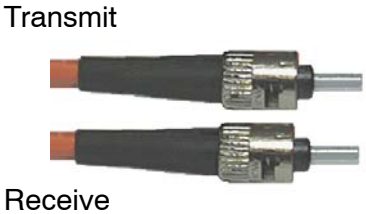
Each cable segment requires two strands of fiber: one to transmit data and one to receive data. The ST-style connectors are used to connect the H4-EBC-F module to a PC or a fiber optic hub or repeater. The modules themselves cannot act repeaters.

Fiber Optic Cable

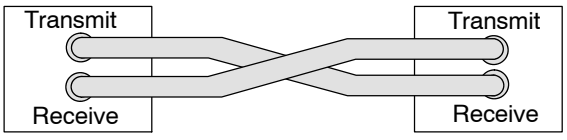
The H4-EBC-F module accepts 62.5/125 multimode fiber optic (MMF) cable. The glass core diameter is 62.5 micrometers, and the glass cladding is 125 micrometers. The fiber optic cable is highly immune to noise and permits communications over much greater distances than 10/100BaseT.

Fiber Optic Module ST Connector

Multimode Fiber Optic (MMF) Cable



62.5/125 MMF cable with bayonet ST-style connectors

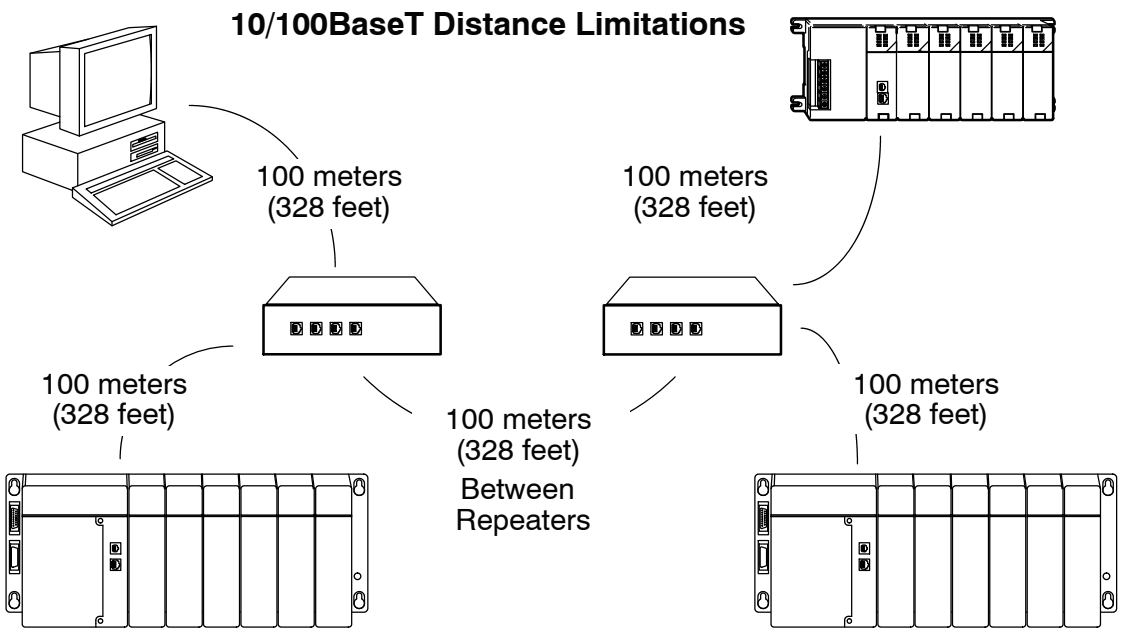


Connecting your fiber optic EBC to a network adapter card or fiber optic hub

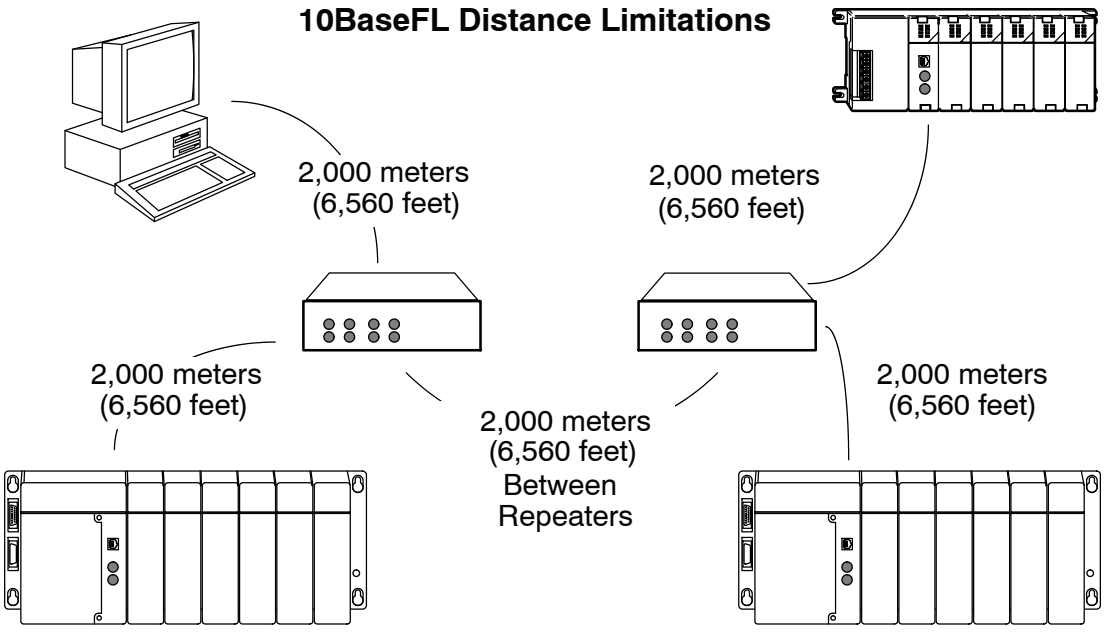
NOTE: See page 2-10 for 10BaseFL distance limitations.

Maximum Ethernet Cable Length

The maximum distance per 10BaseT 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.



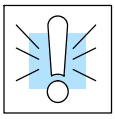
The maximum distance per 10BaseFL cable segment is 2,000 meters or 6,560 feet. Repeaters extend the distance. Each cable segment attached to a repeater can be 2,000 meters. Two repeaters connected together extend the total range to 6,000 meters.



Calculating the Power Budget for the DL205 with H2 Series EBCs

Managing your Power Resource

When determining which I/O modules you will be using in the DL205 EBC system, it is important to remember that there is a limited amount of power available from the power supply. We have provided a table showing the power available from the various DL205 base power supplies and a table showing the maximum power consumed by the EBC and each of the I/O modules supported by the EBC. If any device is connected to the EBC's serial port that uses the 5VDC supply pin, be sure to include the device's power consumption in your 5VDC power budget calculation. Following these two tables is an example of a completed power budgeting worksheet and then a blank worksheet you can use for your own calculations.



WARNING: It is *extremely* important to calculate the power budget. If you exceed the power budget, the system may operate in an unpredictable manner which may result in a risk of personal injury or equipment damage.

EBC Power Specifications

The following table shows the amount of electrical current available at the two voltages supplied from the DL205 base. Use these values when calculating the power budget for you system.

The Auxiliary 24V power source mentioned in the table is available at the base terminal strip. You can connect to external devices or DL205 I/O modules that require 24VDC, but be sure not to exceed the maximum current supplied.

Bases	5V Current Supplied	Auxiliary 24VDC Current Supplied
D2-03B-1	2600 mA	300 mA
D2-04B-1	2600 mA	300 mA
D2-06B-1	2600 mA	300 mA
D2-09B-1	2600 mA	300 mA
D2-03BDC1-1	2600 mA	None
D2-04BDC1-1	2600 mA	None
D2-06BDC1-1	2600 mA	None
D2-09BDC1-1	2600 mA	None
D2-06BDC2-1	2600 mA	300 mA
D2-09BDC2-1	2600 mA	300 mA

Module Power Requirements

The chart on the next page shows the maximum amount of electrical current required to power each of the DL205 EBC or I/O modules. Use these values when calculating the power budget for your system.

Power Consumption Chart (DL205 Modules)

Device	5VDC Base Power Required	External Power Required	Relay Output Modules	5VDC Base Power Required	External Power Required
DC Input Modules			D2-04TRS	250	0
D2-08ND3	50	0	D2-08TR	250	0
D2-16ND3-2	100	0	F2-08TRS	670	0
D2-32ND3(-2)	25	0	F2-08TR	670	0
AC Input Modules			D2-12TR	450	0
D2-08NA-1	50	0	**Combination Modules**		
D2-08NA-2	100	0	D2-08CDR	200	0
D2-16NA	100	0	**EBCs and Specialty Modules**		
DC Output Modules			D2-08SIM	50	0
D2-04TD1	60	20	H2-EBC	320	0
D2-08TD1(-2)	100	0	H2-EBC100	350	0
D2-16TD1-2	200	80	H2-EBC-F	450	0
D2-16TD2-2	200	0	H2-CTRIO	400	0
D2-32TD1(-2)	350	0	H2-SERIO	210	0
AC Output Modules					
D2-08TA	250	0			
F2-08TA	250	0			
D2-12TA	350	0			
Analog Modules					
F2-04AD-1(L)	50	18-30 VDC @ 80 mA max; (-L) 10-15VDC @ 90mA			
F2-04AD-2(L)	60	18-26.4 VDC @ 80 mA max; (-L) 10-15VDC @ 90mA			
F2-08AD-1	50	18-26.4 VDC @ 80 mA max			
F2-08AD-2	60	18-26.4 VDC @ 80 mA max			
F2-02DA-1(L)	40	18-30VDC @ 60mA; (L) 10-15VDC @ 70mA (add 20mA / loop)			
F2-02DA-2(L)	40	18-30 VDC @ 60 mA max; (-L) 10-15VDC @ 70mA			
F2-08DA-1	30	18-30VDC @ 50mA per channel (add 20mA / loop)			
F2-08DA-2	60	18-30 VDC @ 80 mA max			
F2-02DAS-1	100	18-30VDC @ 50mA per channel			
F2-02DAS-2	100	21.6-26.4 VDC @ 60 mA per channel			
F2-4AD2DA	60	18-26.4VDC @ 80mA; add 20mA / loop			
F2-04RTD	90	0			
F2-04THM	100	18-26.4 VDC @ 60 mA max			
F2-8AD4DA-1	35	18-26.4 VDC @ 100 mA max (add 20mA / loop)			
F2-8AD4DA-2	35	18-26.4 VDC @ 80 mA max			

Power Budget Calculation Example

The following example shows how to calculate the power budget for the DL205 system.

Base # 1	Module Type	5 VDC (mA)	Auxiliary Power Source 24 VDC Output (mA)
Available Base Power	**D2-09B-1**	**2600**	**300**
EBC	H2-EBC	+ 320	+ 0
Slot 0	D2-16ND3-2	+ 100	+ 0
Slot 1	D2-16NA	+ 100	+ 0
Slot 2	D2-16NA	+ 100	+ 0
Slot 3	F2-04AD-1	+ 50	+ 80
Slot 4	F2-02DA-1	+ 40	+ 100
Slot 5	D2-08TA	+ 250	+ 0
Slot 6	D2-08TD1	+ 100	+ 0
Slot 7	D2-08TR	+ 250	+ 0
Other (OI, etc.)			
Maximum Power Required		**1310**	**180**
Remaining Power Available		2600-1310= **1290**	300 - 170 = **120**

1. Using the table on the previous page, fill in the information for the base power supply, the EBC, I/O modules, and any other devices that will use system power including devices that use the 24 VDC output. If any device is connected to the EBC's serial port that uses the 5VDC supply pin, be sure to include the device's power consumption in your 5VDC power budget calculation.
2. Add the current columns starting with the row for Slot 0 and working your way down to the "**Other**" category. Put the total in the row labeled "**Maximum power required**".
3. Subtract the row labeled "**Maximum power required**" from the row labeled "**Available Base Power**". Place the difference in the row labeled **"**Remaining Power Available".
4. If **"Maximum Power Required**" is greater than "**Available Base Power**" in either of the two columns, the power budget will be exceeded. It will be unsafe to use this configuration, and you will need to restructure your I/O.

Power Budget Calculation Worksheet

This blank chart is provided for you to copy and use in your power budget calculations.

Base # 0	Module Type	5 VDC (mA)	Auxiliary Power Source 24 VDC Output (mA)
Available Base Power			
CPU Slot			
Slot 0			
Slot 1			
Slot 2			
Slot 3			
Slot 4			
Slot 5			
Slot 6			
Slot 7			
Other (OI, etc.)			
Total Power Required			
Remaining Power Available			

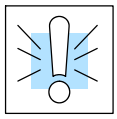
1. Using the table on the previous page, fill in the information for the base power supply, the EBC, I/O modules, and any other devices that will use system power including devices that use the 24 VDC output. If any device is connected to the EBC's serial port that uses the 5VDC supply pin, be sure to include the device's power consumption in your 5VDC power budget calculation.
2. Add the current columns starting with the row for Slot 0 and working your way down to the **"Other"** category. Put the total in the row labeled "**Maximum power required**".
3. Subtract the row labeled "**Maximum power required**" from the row labeled "**Available Base Power**". Place the difference in the row labeled **"Remaining Power Available**".
4. If **"Maximum Power Required"** is greater than "**Available Base Power**" in either of the two columns, the power budget will be exceeded. It will be unsafe to use this configuration, and you will need to restructure your I/O.

Calculating the Power Budget for the H4 Series EBCs

Managing your Power Resource

When determining which I/O modules you will be using in the DL405 EBC system, it is important to remember that there is a limited amount of power available from the power supply. We have provided a table showing the power available from the H4-EBC and H4-EBC-F or Expansion Unit power supplies and a table showing the maximum power consumed by each of the I/O modules supported by the EBC. If any device is connected to the EBC's serial port that uses the 5VDC supply pin, be sure to include the device's power consumption in your 5VDC power budget calculation. Following these two tables is an example of a completed power budgeting worksheet and then a blank worksheet you can use for your own calculations.

If the I/O modules you chose exceed the maximum power available from the power supply you can resolve the problem by shifting some of the modules to an expansion base which contains another power supply.



WARNING: It is *extremely* important to calculate the power budget correctly. If you exceed the power budget, the system may operate in an unpredictable manner which may result in a risk of personal injury or equipment damage.

EBC and Expansion Base Power Specifications

The following chart shows the amount of electrical current available at the two voltages supplied by the EBCs and Expansion units. Use these current values when calculating the power budget for your system.

The Auxiliary 24VDC Power Source mentioned in the table is available at the H4-EBC terminal strip (see page 2-7). You can use this power source to connect to external devices or DL405 I/O modules that require 24VDC.

CPUs	5VDC Current Supplied in mA.	Auxiliary 24VDC Power Source Current Supplied in mA.
H4-EBC	3680	400
H4-EBC-F	3550	400
Expansion Units	**5VDC Current Supplied in mA.**	**Auxiliary 24VDC Power Source Current Supplied in mA.**
D4-EX	4000	400

Module Power Requirements

The chart on the next page shows the maximum amount of electrical current required to power each of the DL405 I/O modules. Use these values when calculating the power budget for your system.

Power Consumption Chart (DL405 Modules)

Device	5V Current Required (mA)	External 24V Current Req. (mA)
I/O Bases		
D4-04B, D4-04BNX, D4-04B-1	80	None
D4-06B, D4-06BNX, D4-06B-1	80	None
D4-08B, D4-08BNX, D4-08B-1	80	None
DC Input Modules		
D4-08ND3S	100	None
D4-16ND2	150	None
D4-16ND2F	150	None
D4-32ND3-1	150	None
D4-32ND3-2	150	None
D4-64ND2	300 (max)	None
AC Input Modules		
D4-08NA	100	None
D4-16NA	150	None
AC/DC Input Modules		
D4-16NE3	150	None
F4-08NES	90	None
DC Output Modules		
D4-08TD1	150	35
F4-08TD1S	295	None
D4-16TD1	200	125
D4-16TD2	400	None
D4-32TD1	250	140
D4-32TD1-1	250	140 (5-15VDC)
D4-32TD2	350	120 / (4A max including loads)
D4-64TD1	800 (max)	None
Specialty Modules		
D4-HSC	300	None
D4-16SIM	150	None
H4-CTRIO	400	None

Device	5V Current Required (mA)	External 24V Current Req. (mA)
AC Output Modules		
D4-08TA	250	None
D4-16TA	450	None
Relay Output Modules		
D4-08TR	550	None
F4-08TRS-1	575	None
F4-08TRS-2	575	None
D4-16TR	1000	None
Analog Modules		
D4-04AD	200	200
F4-04AD	85	100
F4-04ADS	270	120
F4-08AD	75	90
F4-16AD-1	75	100
F4-16AD-2	75	100
D4-02DA	250	300
F4-04DA	120	180
F4-04DA-1	70	75 + 20 per circuit
F4-04DA-2	70	75 + 20 per circuit
F4-08DA-1	70	100 + 20 per circuit
F4-08DA-2	80	150
F4-16DA-1	70	100 + 20 per circuit
F4-16DA-2	80	150
F4-08THM	110	60
F4-08THM-n	120	50 + 20 per circuit
F4-08RTD	80	None
F4-04DAS-1	60	60 per circuit
F4-04DAS-2	60	60 per circuit

Power Budget Calculation Example

The following example shows how to calculate the power budget for the DL405 system.

Base # 1	**Module Type**	**5 VDC (mA)**	**Auxiliary Power Source 24 VDC Output (mA)**
EBC/ Expansion Unit	H4-EBC	**3680**	**400**
Slot 0	D4-16ND2	+ 150	+ 0
Slot 1	D4-16ND2	+ 150	+ 0
Slot 2	D4-02DA	+ 250	+ 300
Slot 3	D4-08ND3S	+ 100	+ 0
Slot 4	D4-08ND3S	+ 100	+ 0
Slot 5	D4-16TD2	+ 400	+ 0
Slot 6	D4-16TD2	+ 400	+ 0
Slot 7	D4-16TR	+ 1000	+ 0
Base	D4-08B	+ 80	+ 0
Other (OI, etc.)			
Maximum power required		**2630**	**300**
Remaining Power Available		3680-2630= **1050**	400 - 300 = **100**

1. Using the table on the previous page, fill in the information for the EBC/Expansion Unit, I/O modules, and any other devices that will use system power including devices that use the 24 VDC output. If any device is connected to the EBC's serial port that uses the 5VDC supply pin, be sure to include the device's power consumption in your 5VDC power budget calculation.Pay special attention to the current supplied by the H4-EBC, the H4-EBC-F or the Expansion Unit. Each one supplies a different amount of current.
2. Add the current columns starting with the row for Slot 0 and working your way down to the **"Other"** category. Put the total in the row labeled "**Maximum power required**".
3. Subtract the row labeled "**Maximum power required**" from the row labeled "**EBC/Expansion Unit**". Place the difference in the row labeled **"Remaining Power Available**".
4. If **"Maximum Power Required"** is greater than "**EBC/Expansion Unit**" in either of the two columns, the power budget will be exceeded. It will be unsafe to use this configuration, and you will need to restructure your I/O. You may need to add expansion bases to accommodate your current requirements.

Power Budget Calculation Worksheet

You may copy and use the following blank chart for your power budget calculations.

Base # _____	Module Type	5 VDC (mA)	Auxiliary Power Source 24 VDC Output (mA)
EBC/ Expansion Unit			
Slot 0			
Slot 1			
Slot 2			
Slot 3			
Slot 4			
Slot 5			
Slot 6			
Slot 7			
Base			
Other (OI, etc.)			
Maximum Power Required			
Remaining Power Available			

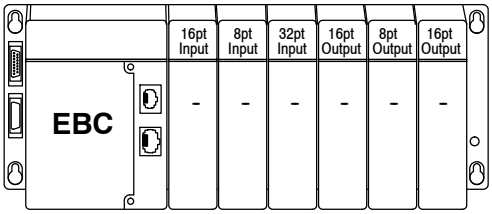
1. Using the tables at the beginning of the Power Budgeting section of this chapter fill in the information for the EBC/Expansion Unit, I/O modules, and any other devices that will use system power including devices that use the 24 VDC output. Pay special attention to the current supplied by the H4-EBC, the H4-EBC-F or the Expansion Unit since they do differ.
2. Add the current columns starting with the row for Slot 0 and working your way down to the **"Other"** category. Put the total in the row labeled "**Maximum power required**".
3. Subtract the row labeled "**Maximum power required**" from the row labeled "**EBC/Expansion Unit**". Place the difference in the row labeled **"Remaining Power Available**".
4. If **"Maximum Power Required"** is greater than "**EBC/Expansion Unit**" in either of the two columns, the power budget will be exceeded. It will be unsafe to use this configuration, and you will need to restructure your I/O. You may need to add expansion bases to accommodate your current requirements.

DL405 Local and Expansion I/O

The H4 Series EBCs support the use of DL405 series I/O local and local expansion bases.

Local Base and I/O

The local base is the base in which the EBC resides. Local I/O modules reside in the same base as the EBC. For example, placing 32-point modules in all eight slots in an 8-slot base will use 256 I/O points .

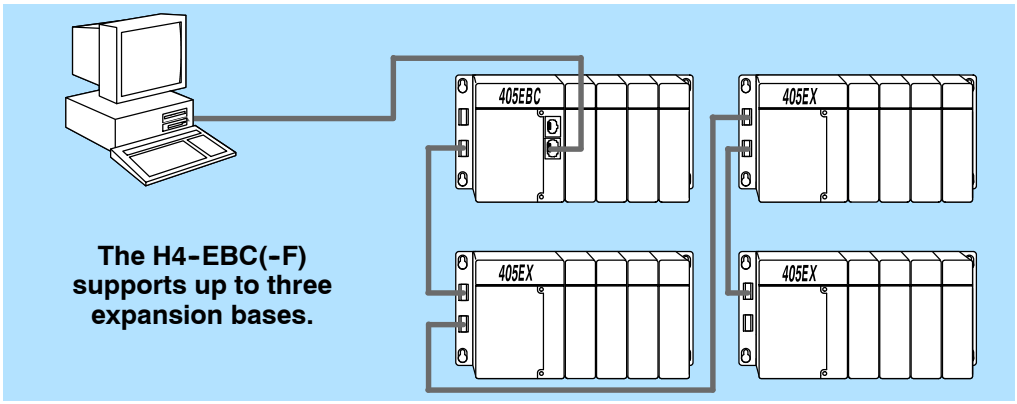


Local Expansion Base and I/O

Use local expansion bases when you need more I/O points or a greater power budget than the local base provides. The expansion bases require a Local Expansion Unit (rather than an EBC) and a cable (either D4-EXCBL-1 or D4-EXCBL-2) to connect to the local EBC base.

The following figure shows one EBC base and three expansion bases. The I/O modules are shown as examples of a usable configuration, but any configuration of I/O modules could be used if it is supported by the power budget. See page 2-15 for information about calculating the power budget.

H4-EBC and H4-EBC-F modules support one local base and a maximum of three expansion bases.



3

Configuring the EBCs Using NetEdit3

In This Chapter. . . .

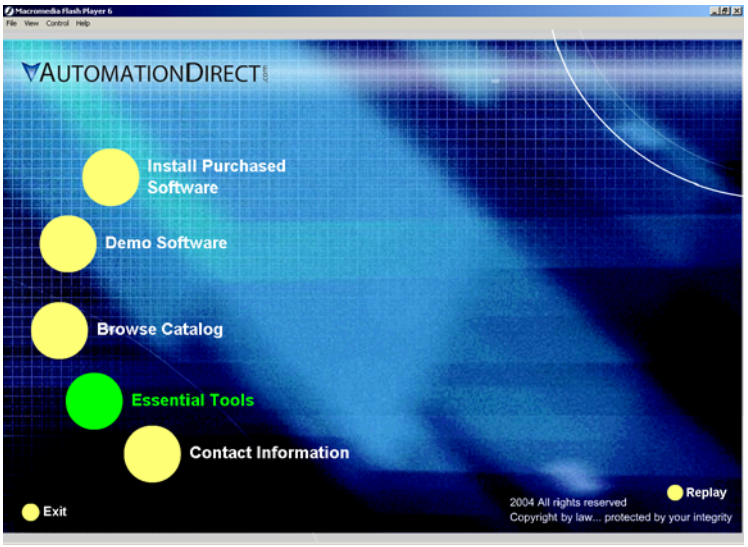
- NetEdit3 Software
- Using NetEdit3
- Locating the MAC Address Label

NetEdit3 Software

NetEdit3 is a software utility which can be used to set network identifiers (Module ID or IP Address), configure the EBC serial port, perform diagnostic and troubleshooting tasks and upgrade the firmware in the EBC module if necessary. The H2-EBC100 requires NetEdit 3.x 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.



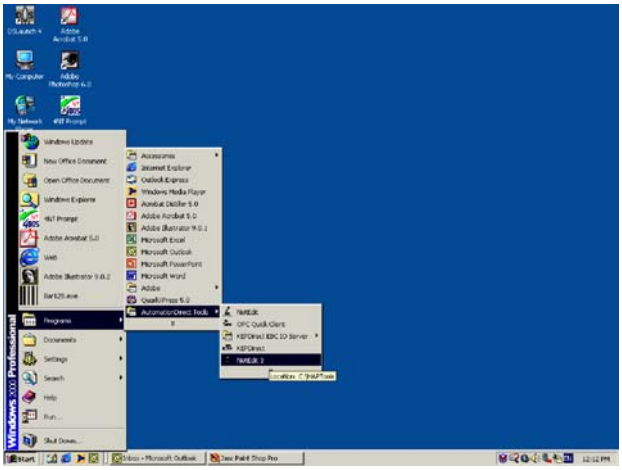
Click on the Essential Tools button. The following window will be displayed.

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

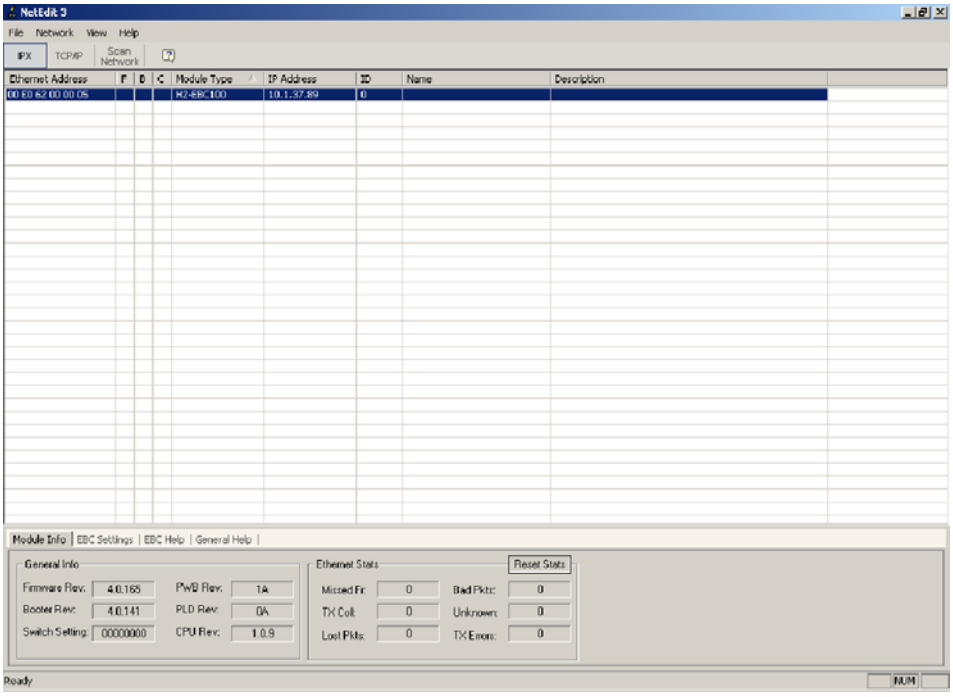
There are three methods to launch NetEdit3.
The three methods are:

- using the Windows Start menu Programs>AutomationDirect Tools> NetEdit3 as shown below
- launching ***Direct***Soft32 (if installed), from the programming window, select PLC>Tools>NetEdit3
- launching ***Direct***Soft32 (if installed), then select Utilities>NetEdit3



The NetEdit3 Screen

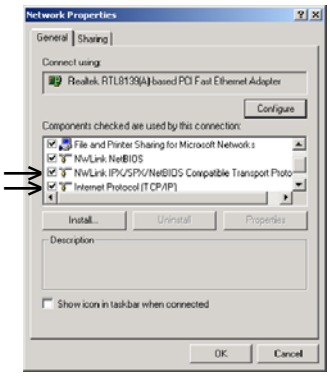
Starting NetEdit brings up the screen below. All NetEdit3 functions are accessed from this screen.



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.

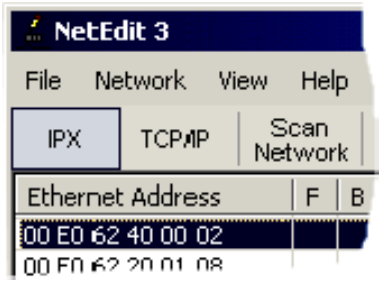
NOTE: Your PC-based Control software may be capable of configuring the EBC module. If so, please refer to the appropriate documentation for that software product to determine the best method to configure the EBC. Depending on which software you are using, it may not be necessary to use NetEdit3.

Ethernet Communication Protocol

In the upper left corner of the NetEdit3 screen, you will find buttons labeled *IPX* and *TCP/IP*. The EBC 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 EBC 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 EBC to link NetEdit3 to the module.



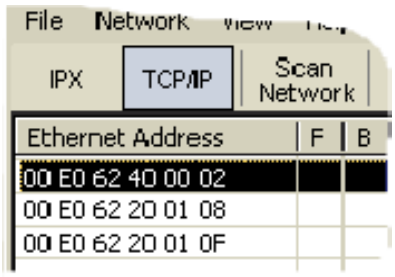
Some PC-based control software products may support only one of these protocols. Read the documentation for your software to be sure the protocol you select is supported.

Ethernet Address

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

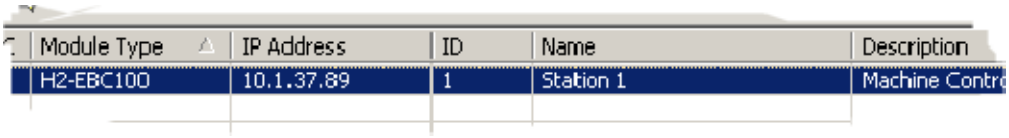
If modules are added or removed from the network, click on the *Scan Network* button to update the list. Notice that the MAC Address is the factory-assigned address that is on the permanent label on the module.

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.

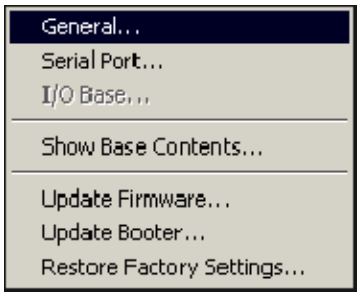
Module Type, IP Address and ID

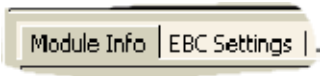


The upper mid section of the NetEdit3 screen displays the *Module Type, IP Address,* module *ID, Name* and *Description* of the modules currently on the network.

A new EBC will have an IP Address of 0.0.0.0, a Module ID of 0 (zero), Module Name (module part number) and a Description (EBC w/ part number). 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 EBC module listed on the NetEdit3 screen will display the window to the right. This is an alternative to using the Module Info or EBC settings tabs (shown below) to access the module's configuration settings. The settings are discussed later in this section.

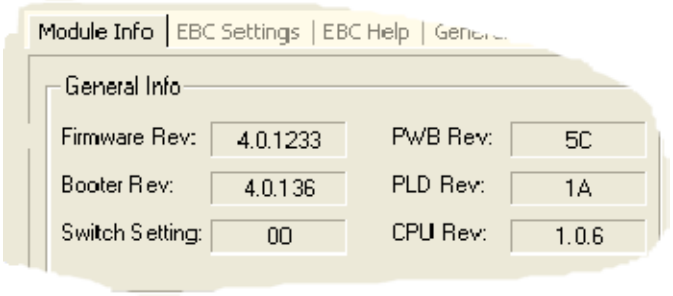




Module Info> General Information

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.

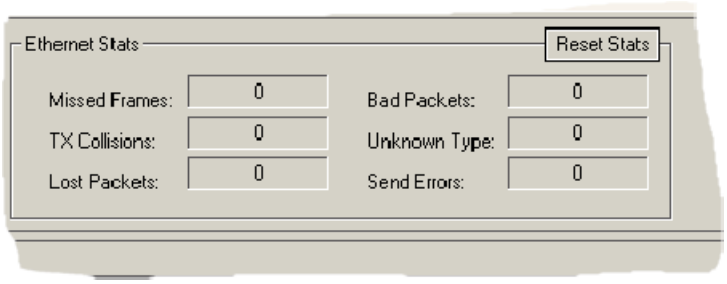
This box is in the lower left section of the NetEdit3 screen.



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).

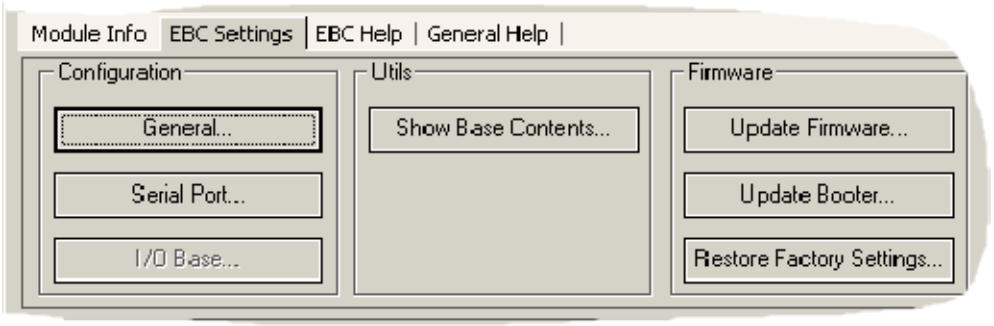
This box is in the lower middle section of the NetEdit3 screen.



EBC Settings

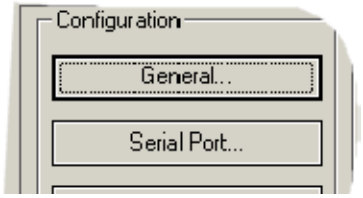
When the EBC Settings tab is selected, the selected module's Configuration, Utilities and Firmware tools can be accessed.

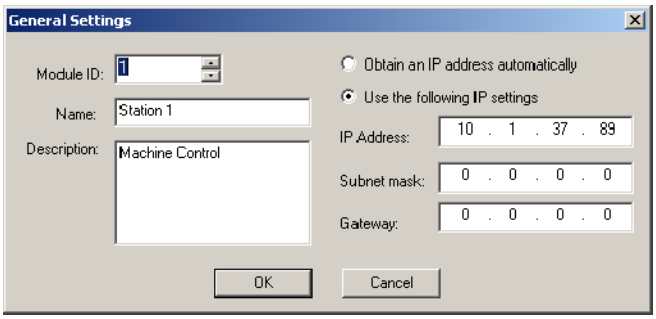
This box is in the lower middle section of the NetEdit3 screen.



EBC Settings> Configuration> General

Clicking the General button in the EBC 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 EBC, 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 and can be used for user identification purposes.

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

IP Address

An **IP Address** is assigned to the EBC module if your network will be using the TCP/IP or MODBUS TCP/IP (H2-EBC100 only) protocols. If you have a separate dedicated network for your EBCs, you may be able to use the Module ID identifier (IPX protocol) for communications instead of an IP address. To set an IP Address, use the twelve-digit number assigned to the EBC module by your network administrator. If you change the IP Address, do not use the number "0" or "255" in any field. Doing so will cause communication problems. The **valid settings** are **1 through 254**. The module ships from the factory with an IP Address of 0.0.0.0. This is not a usable IP Address for communications. Click on *Use the following IP settings* radio button before clicking on the OK button to write the updated settings to the module's flash memory. It is extremely important not to have duplicate IP Addresses on your network.

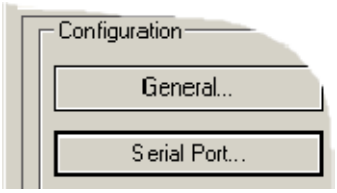
Example

Client Subnet Mask:	255.255.0.0	
Valid Client IP Address:	192.168.50.2	
Valid EBC IP Address:	192.168.**55.5**	**1-254 Valid settings for Bold number fields (Do not duplicate)**
Valid EBC IP Address:	192.168.**70.15**	

WARNING: If your using the H2-EBC100, be sure to read Chapter 5 concerning DHCP issues.

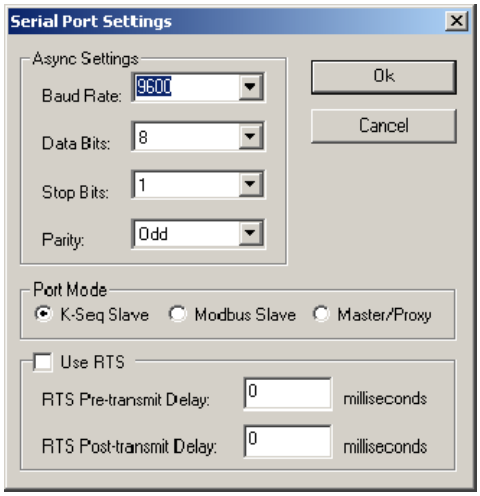
EBC Settings> Configuration> Serial Port

Clicking the Serial Port button in the EBC Settings>Configuration box brings up the Serial Port Settings window below.



NOTE: The serial port has a fixed address of "1". The port is intended to be used as a single RS-232 slave device.

On the *Serial Port Settings* window, make any necessary changes to the serial communication parameters. After making changes, be sure to click on the OK button. Also, Be sure these parameters match the parameters of the serial device with which you are communicating.

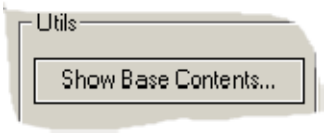


Note: Some PC-based Control software packages may automatically overwrite settings selected here. Refer to the documentation for your PC-based Control software.

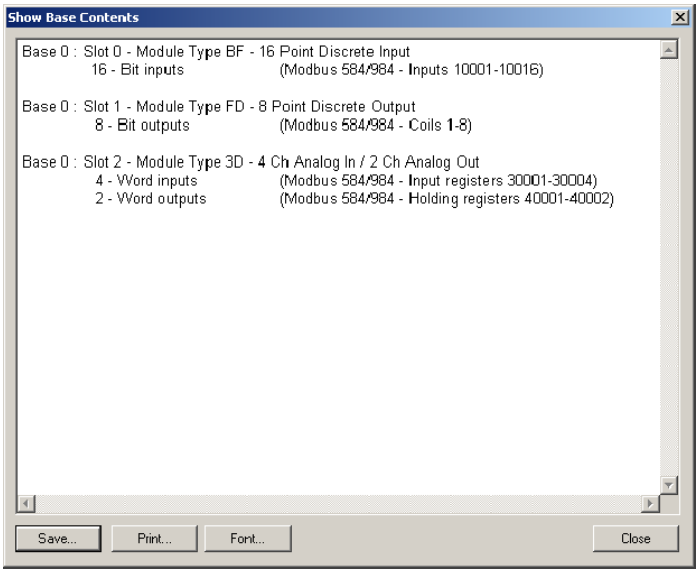
The OK button sends all the entries to the module's flash memory.

EBC Settings> Utils>Show Base Contents

Clicking the Show Base Contents button in the EBC Settings>Utilities box brings up the Show Base Contents Window shown below.

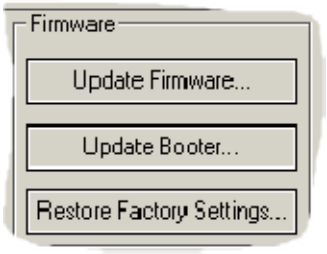


This function queries the EBC for a list of I/O modules it has in its base. The Module Type and the MODBUS 584/984 addressing will be listed as well. This will help confirm that the EBC can recognize all the I/O modules connected to the EBC controller. The Show Base Contents information can be saved as a (*.txt) file or printed for reference or future use.

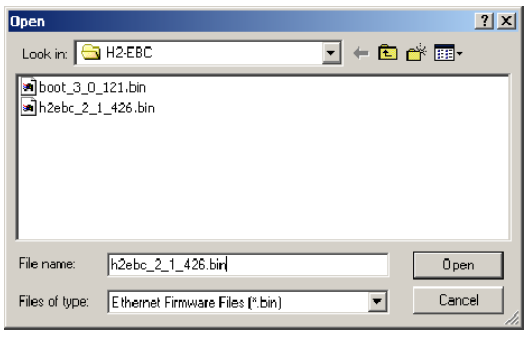


EBC Settings> Firmware

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.

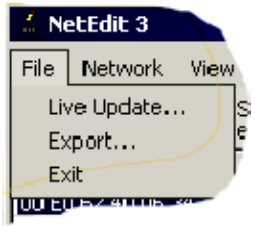


Clicking on the either of the Update buttons opens the appropriate EBC folder within the Images folder, which is created during the install of NetEdit3. The Images folder is located in the same folder as NetEdit3.exe. Each module folder contains the module's firmware and boot loader files. The next section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

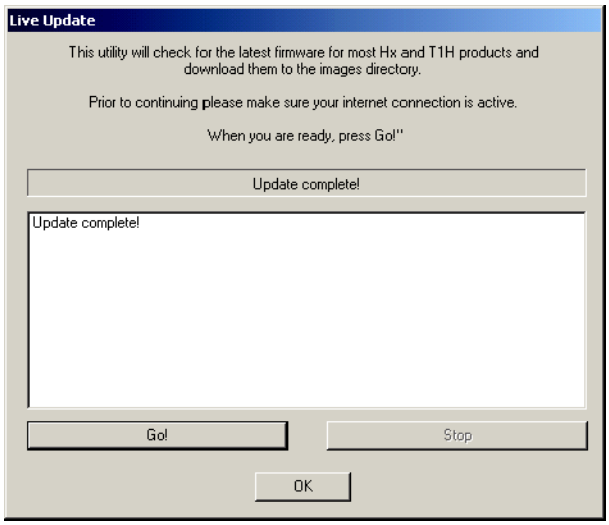


FileMenu>
Live Update

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.

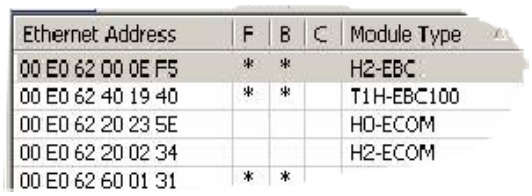


When you click the Go! button on the Live Update window, NetEdit3 will compare the version information of the files on the Host Engineering web site against the files you have locally on your PC, and it will download any newer files. Once this process is complete, NetEdit3 will rescan the devices on 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	*	*		

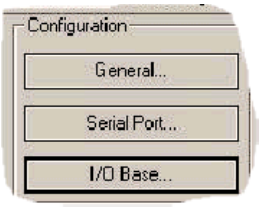
The "F" column will display an asterick beside any device whose firmware is older than its firmware file in your Images folder.
The "B" column will display an asterick beside any device whose boot loader is older than its boot loader file in your Images folder.
The "C" column will display an asterick 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.

Using NetEdit3 to Configure the H4-EBC(-F) Base

NOTE: The following configuration information applies only to the H4-EBC(-F) and the DL405 I/O. The H2-EBC(100) and H2-EBC-F and associated DL205 I/O are self-configuring and do not require this additional step.

EBC Settings> Configuration> I/O Base

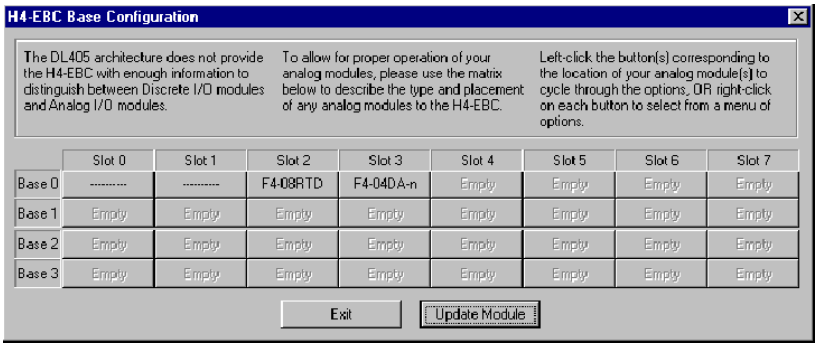
Clicking the I/O Base button in the EBC Settings>Configuration box brings up the Base Configuration window below.



The default symbol "----------" appears on the configuration screen where digital or analog modules are present. For digital modules, you do not need to make any changes. The H4-EBC(-F) recognizes the digital modules and is self-configuring for the digital modules.

If you are using analog modules, you must let the H4-EBC(-F) know that by doing the following. Click on the slot location where the analog module is located. Continue clicking on the same slot location until the part number of your analog module appears.

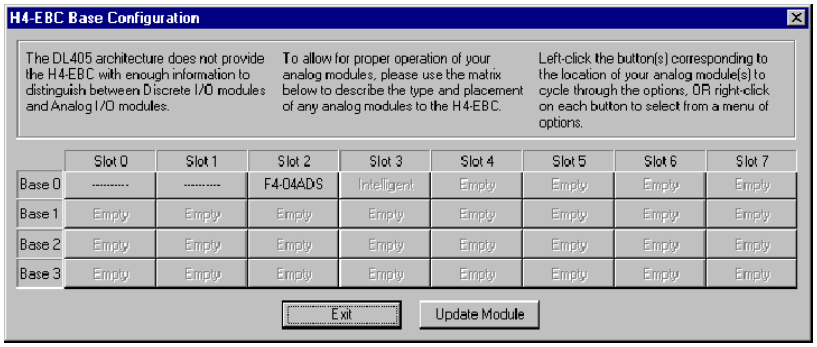
Configuring Analog Modules



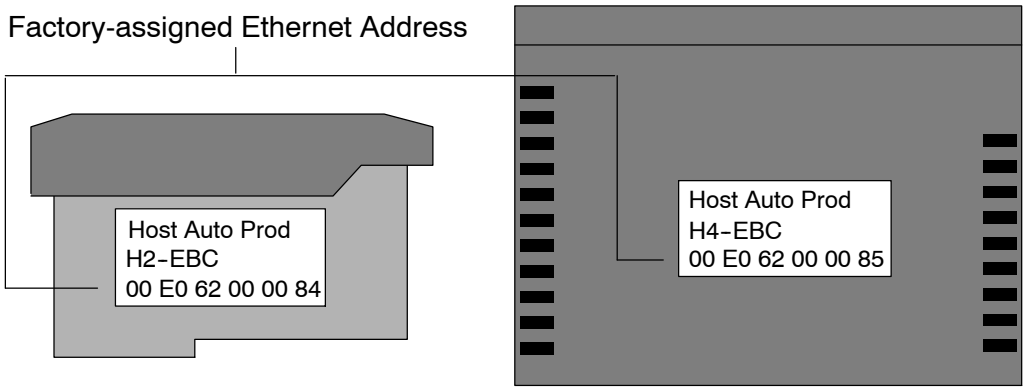
Once the correct part numbers appear for each of your analog modules, click the **Update Module** button to save the configuration into flash memory onboard the module. Leave the symbol "----------" wherever you have a digital module.

If you are using a High Speed Counter module, the word "Intelligent" will appear in gray. The High Speed Counter module is configured automatically (see below). No other action is required other than clicking on the **Update Module** button.

Configuring the High Speed Counter Module



Locating the Ethernet Address Label



Ethernet Address A unique Ethernet Address is assigned to each module at the factory and cannot be changed. It is a twelve digit number, and it is printed on a label permanently attached to the EBC module. NetEdit recognizes the Ethernet Address

4

MODBUS® TCP/IP for H2-EBC100

In This Chapter. . . .

- MODBUS TCP/IP
- Supported MODBUS Function Codes
- MODBUS 584/984 Addressing
- H2-EBC100 System Memory
- Current / Last State Error Codes

MODBUS TCP/IP

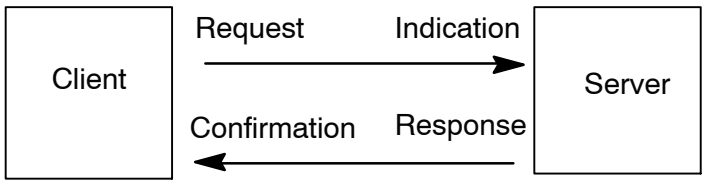
MODBUS TCP/IP is essentially the serial MODBUS RTU protocol encapsulated in a TCP/IP wrapper. MODBUS RTU is used for serial communications between a master and slave(s) devices. MODBUS TCP/IP is used for TCP/IP communications between client and server devices on an Ethernet network. The TCP/IP version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



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 slave/server device (H2-EBC100) 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

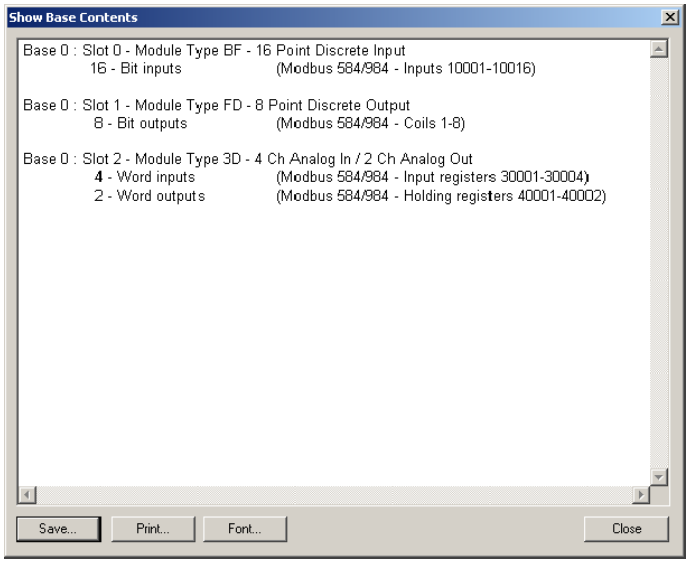
The following MODBUS function codes are supported by the H2-EBC100 base controller.

MODBUS Function Code	Function
01	Read Output Table
02	Read Input Table
03	Read Holding Registers (when addressing mode is 584/984, this function is used to access analog output registers)
04	Read Input Registers (when addressing mode is 584/984, this function is used to access analog input registers)
05	Force Single Output
06	Preset Single Registers
08	Loop back / Maintenance
15	Force Multiple Outputs
16	Preset Multiple Registers

MODBUS 584/984 Addressing

Modbus Data Type (Bit)		H2-EBC100				
		Range (Decimal)	Points		Memory Type	Access
Coil		**1 - 1024**	1024		**Discrete Output**	R/W
		1025 – 10000	–		Reserved	–
Input		**10001 - 11024**	1024		**Discrete Input**	R only
		11025 – 20000	–		Reserved	
Modbus Data Type (Word)						
		Range (Decimal)	**Words (16- bit)**	**Channel (32- bit)**	**Memory Type**	
Input Register	**Analog Input**	**30001 - 30512**	512	256	Analog Input Register	R only
	Input Register	30513 – 32000	–	–	Reserved	–
	Bit Input Register	**32001 - 32064**	64	32	Discrete Input Bit Register	R only
	Input Register	32065 – 37000	–	–	Reserved	–
Hold Register	**Analog output**	**40001 - 40512**	512	256	Analog Output Register	R/W
	Hold Register	40513 – 42000	–	–	Reserved	–
	Bit Output Register	**42001 - 42064**	64	32	Discrete Output Bit Register	R/W
	Hold Register	42065 – 44000	–	–	Reserved	–

Note: NetEdit3 Show Base Contents function will list the MODBUS addressing for each I/O module on the base. For the analog I/O, the module Configuration Data registers are also listed. Refer to Chapter 3 for information on NetEdit3.



MODBUS 584/984 Addressing for Function Code 3 Clients

This memory map offers duplicate registers from the 30001 range and Bit memory data type into the 411000 range for clients/masters that only support function code 3. These ranges are word level data only.

Modbus Word Data Type		**H2-EBC100**				
		Range (Decimal)	**Words**		**Memory Type**	**Access**
Coil		**411000 - 411063**	64		**Discrete Output**	R/W
		411064 - 411124	-		Reserved	-
Input		**411625 - 411688**	64		**Discrete Input**	R only
		411689 - 412062	-		Reserved	
Modbus Worf Data Type						
		Range (Decimal)	**Words (16- bit)**	**Channel (32- bit)**	**Memory Type**	
Input Register	**Analog Input**	**412251 - 412762**	512	256	Analog Input Register	R only
	Input Register	412763 - 414250	-	-	Reserved	-
	Bit Input Register	**414251 - 414314**	64	32	Discrete Input Bit Register	R only
	Input Register	414315 - 419250	-	-	Reserved	-
Hold Register	**Analog output**	**40001 - 40512**	512	256	Analog Output Register	R/W
	Hold Register	40513 - 42000	-	-	Reserved	-
	Bit Output Register	**42001 - 42064**	64	32	Discrete Output Bit Register	R/W
	Hold Register	42065 - 44000	-	-	Reserved	-

H2-EBC100 System Memory

	H2-EBC100			
	Modbus Addressing Range (Decimal)	**Words (16-bit)**	**Word Descriptions**	**Access**
Module Version Information	**37001 - 37006 (419251 - 419256)***	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	R only
	37007 - 37010 (419257 - 419260)	-	Reserved	-
Device Data	**37011 - 37100 (419261 - 419350)***	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 Switch 12 - DIP Settings 13 - Media Type (14-15) - Reserved 16 - Reserved 17 - Reserved 18 - Model Number 19 - Ethernet Speed 20 - Reserved 21 - IO Total Byte Count 22 - Bit Input Byte Count 23 - Bit Output Byte Count 24 - Non-bit Input Byte Count 25 - Non-bit Output Byte Count (26-90) - Reserved	R only
I/O Module ID's	**37101 - 37108 (419351 - 419358)***	8 (1 word per slot)	I/O module ID numbers per slot location	R only
	37133 - 37200 (419359 - 419450)	-	Reserved	-
Module Information	**37201 - 37232 (419451 - 419482)***	32 (4 words per slot)	1 - Bit Input Count 2 - Bit Output Count 3 - Non-bit Input Count 4 - Non-bit Output Count	R only
	37329 - 37400 (419483 - 419650)	-	Reserved	-

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

H2-EBC100 System Memory (continued)

	H2-EBC100			
	Modbus Addressing Range (Decimal)	**Words (16-bit)**	**Word Descriptions**	**Access**
EBC Dynamic Module Data	**410001 - 410020**	20	1 - See Error Codes on p. 4-9. 2 - Error bit-per-slot for first 16 slots If any bit is set, see extended error info of Module Status data for specific problem 3 - Error bit-per-slot for second 16 slots (if present) If any bit is set, see extended error info of Module Status data for specific problem NOTE: Any write to [1], [2], or [3] above will clear the module / slot errors. 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 - 15: Reserved 5 - Reboot Count (LSW) - Read Only 6 - Reboot Count (MSW) - Read Only 7 - Link Monitor Timeout (EBC communication watchdog Timer) - 0 to disable; range 0 - 10000ms. 8-20 - Reserved	R / W
	410021 - 410052	-	Reserved	-
I/O Module Status	**37401 - 37560 (419651 - 419810)***	160 (20 words per slot)	1 - Flags with bits indicating presence of Error, Warning, Info Values Bit 0: If set, indicates that Error Value is non-zero Bit 1: If set, indicates that Warning Value is non-zero Bit 2: If set, indicates that Info Value is non-zero Bit 3: Reserved Bit 4: If set, indicates that Extended error info is present Bit 5: Reserved Bit 6: Reserved Bit 7: Reserved For Words 2-4, refer to Current/Last State Error Codes Table on page 4-9. 2 - Error Code 3 - Warning Code 4 - Info Code 5 - 20: Reserved	R only
	37561 - 40000 (419811 - 422250)	-	Reserved	-

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

Current / Last State Error Codes

The following table lists the error codes for Words 2–4 in the **Module Status** System Memory area.

Error Code (Decimal)	Description
E0	No error.
E121	Channel failure.
E122	Unused analog input channels exist.
E139	Broken transmitter on one of the analog input channels (if supported by analog module)
E142	Multiple channels failed.
E154	I/O configuration has changed. See E153 for reset methods.
E200–E216	Unused analog input channels exist at channel xx (1–16), where xx = Value –200. (example: E212 indicates unused analog channel exists at channel 12.

Extended Error Codes

The following table lists the error codes for Words 5–20 in the **Module Status** System Memory area.

Error Code (Decimal)	Description
E32–E63	Bitwise error where bit 5 is always SET. Look at bit 0 thru bit 4 to get a possible list of errors. Example 34 decimal =22 hexadecimal (Bit 5 SET and Bit 1 SET). BIT — Type of Error 0 — Terminal block off 1 — External P/S voltage low 2 — Fuse blown 3 — Bus error 4 — Module initialization error (intelligent module) 5 — Fault exists in module (this bit is SET if any of the above bits are SET)
E117	Write attempt to an invalid analog channel.
E119	Data not valid. Subnet mask or IP address not allowed // EBC SDK data packet not constructed properly.
E121	Analog input channel error.
E122	Unused analog input channels exist.
E139	Broken transmitter on one of the analog input channels.
E142	Channel failure.
E200–E216	Unused analog input channels exist at channel xx (1–16), where xx = Value –200.

5

H2-EBC100 DHCP & HTML Configuration

In This Chapter. . . .

- H2-EBC100 DHCP
- Disabling DHCP and Assigning a Static IP Address
- Using HTML Configuration

H2-EBC100 DHCP

DHCP Issues

The H2-EBC100 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 (i.e. H2-EBC100).

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 H2-EBC100 device, we **do not** recommend that you use DHCP to assign IP addresses for your runtime operation. Use NetEdit3 or the H2-EBC100's HTML Configuration page to assign a static IP address to the module (shown below).

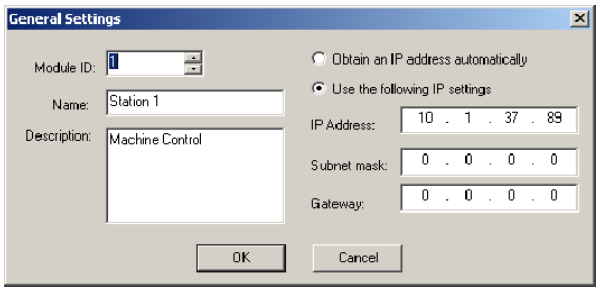
NetEdit3 can be used to connect to a H2-EBC100 using the IPX protocol, regardless of the IP address that was assigned to it by a DHCP server.

Disabling DHCP and Assigning a Static IP Address

You can use NetEdit3 or the H2-EBC100'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 (refer to chapter 3)

.



HTML Configuration

IP Configuration

Mode:	○ Obtain an IP address automatically ◉ Use the following IP address
IP Address:	10.1.37.89
Subnet Mask:	0.0.0.0
Gateway:	0.0.0.0

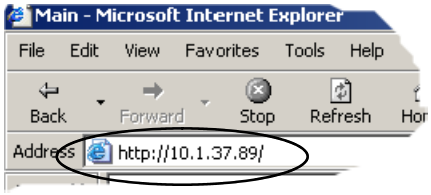
Back Send Reset

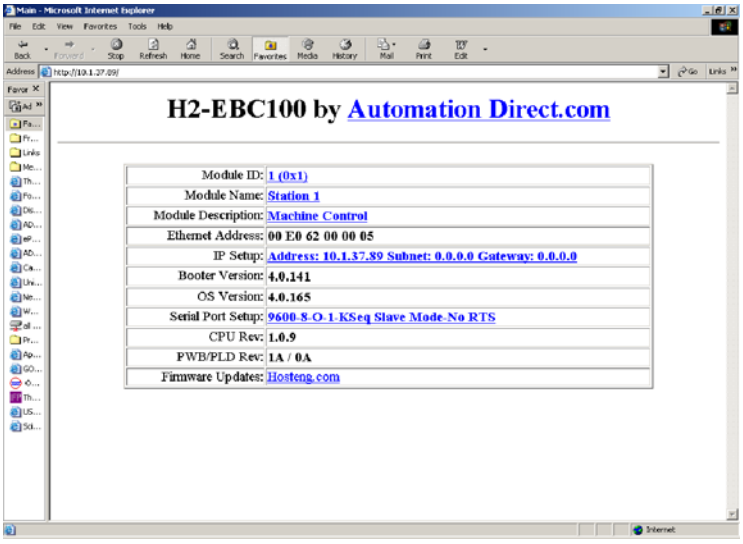
Using HTML Configuration

The H2-EBC100 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 H2-EBC100'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.

Connecting to the H2-EBC100

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

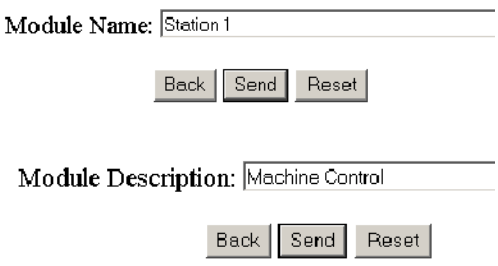




To configure the module, click on the desired parameter field. A new window will open, which are all described below and on the following page. Clicking the Back button will take you back to the main configuration screen shown above. Clicking the Send button writes the entry or change to the module's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

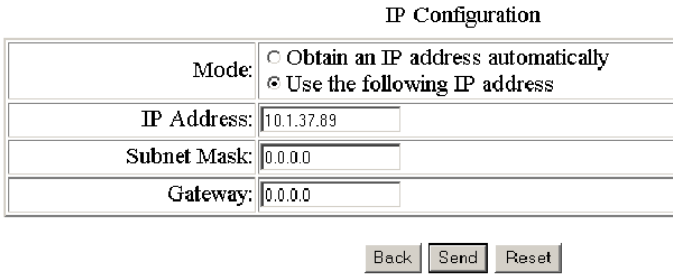
Module ID: Module IDs must be unique for each EBC, but they do not have to be in sequence. The module's Node Address rotary switches must both be set to allow the HTML configuration tool to set a Module ID. Do not use address zero for communications.

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.



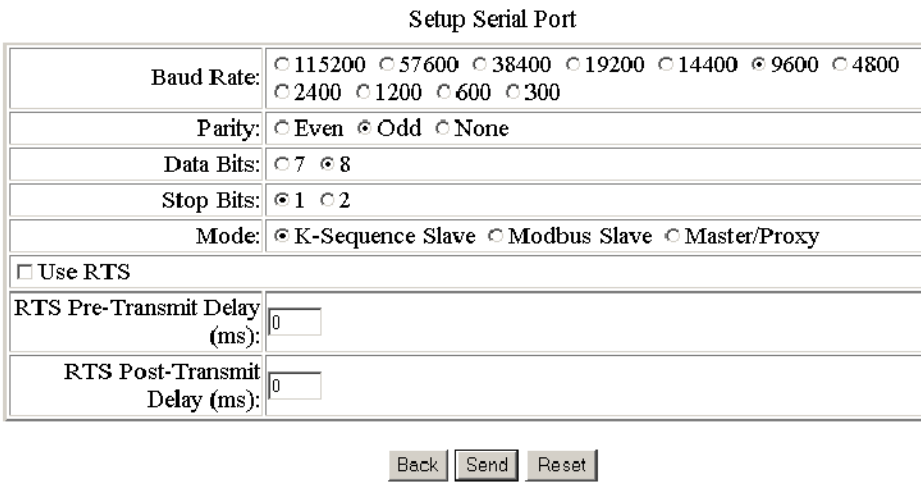
Ethernet Address: this is the MAC Address. It is a factory-assigned address that is on the permanent label on the module.

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.

Serial Port Setup: configure or make necessary changes to the serial port communication parameters. Click the Send button to write to the module's flash memory.



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 to upgrade the module.

6

Troubleshooting Guidelines

In This Chapter. . . .

Isolating a Communication Problem

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

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

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.
- Replacing the module may determine whether the problem is in the module.
- 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.
- Diagnostic tools within your PC-based Control software.

Troubleshooting Chart

The following chart summarizes the different types of communication failures you could experience. In each case the power must be applied to the base, and you must be attempting to communicate with the EBC in question.

The meaning of the **diagnostic LEDs** is explained begining on page 6-4.

Troubleshooting Chart	
Legend: Off On Flash	
EBC Module LEDs	**Corrective Action**
LINKGD ACT ERROR OR LINKGD ACT ERROR	1. Cycle power to the base. This will clear the ERROR if it was due to a transient condition. 2. Replace EBC module
H2-EBC100 STATUS LINKGD ACTIVE ERROR OR **H2-EBC100** STATUS LINKGD ACTIVE ERROR	

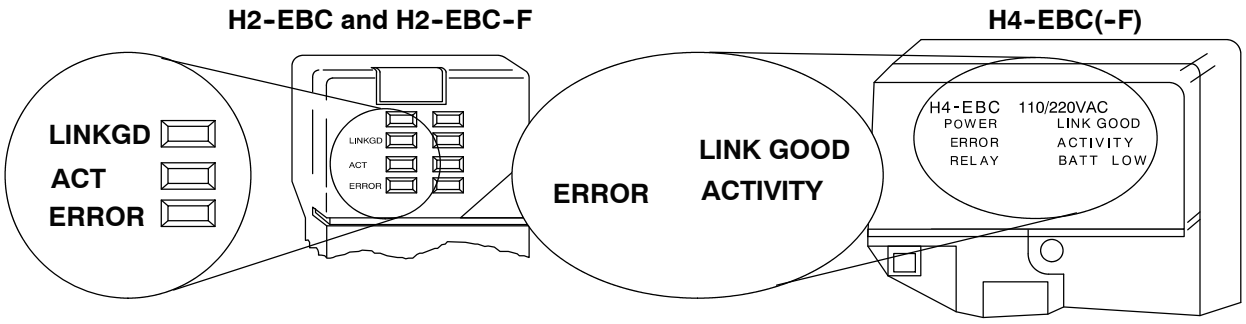
Troubleshooting Chart (Continued)	
Legend: ☐ Off ■ On ▨ Flash	
EBC Module LEDs	**Corrective Action**
H2-EBC100 STATUS ■ LINKGD ☐ LINKGD ☐ ACT ☐ ACTIVE ☐ ERROR ☐ ERROR ☐	1. Try another cable that you know works. Check pinouts (see page 2-8). 2. Try another port on the hub or another hub. 3. Replace EBC module.
H2-EBC100 STATUS ■ LINKGD ■ LINKGD ■ ACT ☐ ACTIVE ☐ ERROR ☐ ERROR ☐	1. Try another cable between PC and hub. 2. Try another port on the hub or another hub. 3. 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. 4. 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. LINKGD ■ ACT ▨ ERROR ☐ OR LINKGD ■ ACT ■ ERROR ☐ **H2-EBC100** STATUS ■ LINKGD ■ ACTIVE ▨ ERROR ☐ OR **H2-EBC100** STATUS ■ LINKGD ■ ACTIVE ■ ERROR ☐	1. Try another cable between PC and hub or EBC and hub. 2. Try another port on the hub or another hub. 3. Look for errors in the setup of the EBC module.

EBC Module Diagnostic LEDs

EBC LEDs

The EBC module has three indicator lights which show the status of the following:

- signal path between the EBC and the hub
- signal between a PC and an EBC
- EBC module hardware



LINK GOOD Indicator

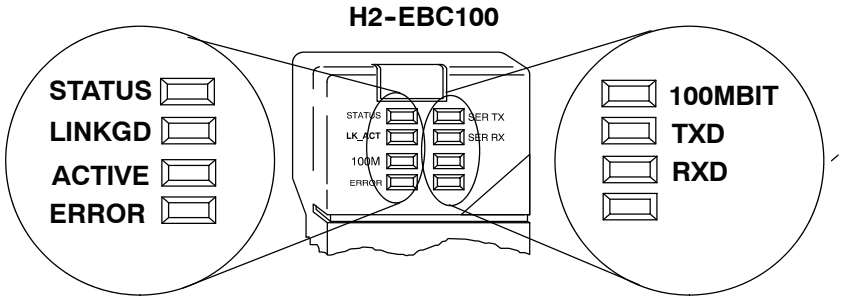
The **green LINKGD** (LINK GOOD) LED is **on steady** when the EBC module is connected properly to an active device on the network and is receiving 5VDC operating voltage from the PLC power supply. The LINKGD LED verifies that the proper cables are connected, and the EBC module is functioning correctly.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

ACTIVITY Indicator

The **red ACT** (ACTIVITY) LED **flashes** to indicate that the module is detecting data 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 **on steady**.

ERROR Indicator

If the EBC module's **red ERROR** indicator is **flashing** or **on steady**, a fatal error has occurred. The error may be in the EBC 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.



STATUS Indicator The **green STATUS** LED is **on steady** when the EBC module is receiving 5VDC operating voltage from the PLC power supply and self diagnostics indicates the module is functioning properly.

LINK GOOD Indicator The **green LINKGD** (LINK GOOD) LED is **on steady** when the EBC module is connected properly 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 EBC module is functioning correctly. If a mismatch with the 10BaseT or 10BaseFL connections occurs this LED will not be illuminated.

ACTIVITY Indicator The **green ACT** (ACTIVITY) LED **flashes** to indicate that the module is detecting data 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 **on steady**.

ERROR Indicator If the EBC module's **red ERROR** indicator is **flashing** or **on steady**, a fatal error has occurred. The error may be in the EBC 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.

Serial TXD Indicator The **green** TXD (Serial TXD) LED flashes when the EBC's serial port is transmitting data.

Serial RXD Indicator The **green** RXD (Serial RXD) LED flashes when the EBC's serial port is receiving data.

Using NetEdit3 for Troubleshooting

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

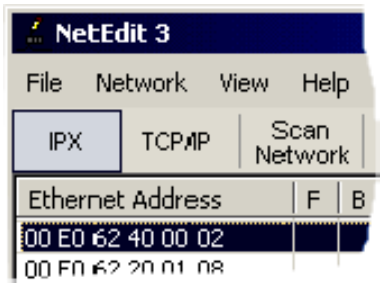
- See active modules on the network.
- Examine and change the modules' configuration settings.
- See the firmware revision number.
- Review statistical information about communication errors by type.

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 inforamtion on the "Ping" command.

Select a Module

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

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

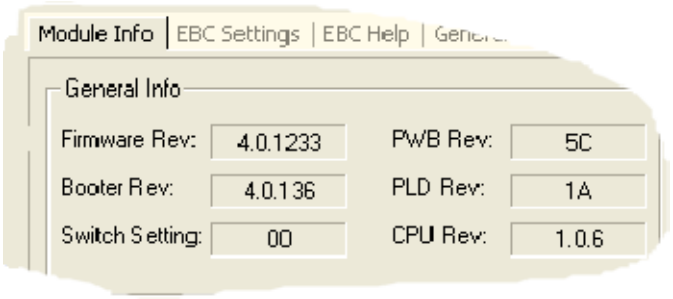


NOTE: The Ethernet Address (MAC) is permanently assigned at the factory, and it is printed on a label on the side of the EBC module. See page 3-13 if you need help locating the label.

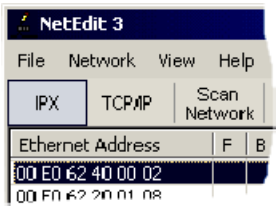
Module Info> General Information

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.



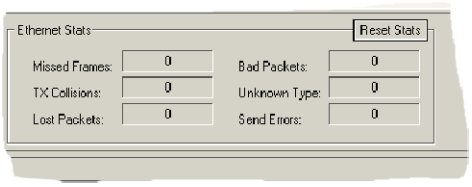
Change Protocol



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 driver is loaded on your PC, refer to page 3-4, as well as your Windows documentation.

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 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 EBC 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.

Replacing the EBC Module

If you are replacing an existing EBC module with a new one, you need to set up the new module with the same **Module ID** 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 page 2-3 or 2-4 to review the procedure for setting the Module ID using the DIP switch.

If you set up your original EBC module using NetEdit3, you will need to duplicate the settings in the new module using the same procedures. See page 3-5 through 3-12 to review the procedures for using NetEdit3.

If you set up your original EBC module using your PC-based Control software, you will need to refer to the appropriate documentation.

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:

- EBC module is working correctly.
- EBC 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 10/100Mbps 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 EBC 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 Good LED on the EBC module is turned off.

Appendix A General Specifications

A

In This Appendix

- H2 Series and H4 Series EBC Specifications
- Serial Port Specifications
- Ethernet Standards

H2 Series and H4 Series EBC Specifications

Specifications	H2-EBC	H2-EBC100	H2-EBC-F
Communications	10Base-T Ethernet	10/100BaseT Ethernet	10Base-FL Ethernet
Data Transfer Rate	10Mbps max.	100Mbps max.	10Mbps max.
Link Distance	100 meters (328 ft)	100 meters (328 ft)	2,000 meters (6,560 ft)
Ethernet Port	RJ45	RJ45	ST-style fiber optic
Ethernet Protocols	TCP/IP, IPX	TCP/IP, IPX, MODBUS TCP/IP	TCP/IP, IPX
Serial Port (RJ12)	K-sequence, ASCII	~~K-sequence, ASCII, MODBUS RTU~~	K-sequence, ASCII
Power Consumption	320mA	350mA	450mA
Manufacturer	Host Automation Prods	Host Automation Prods	Host Automation Prods

Change to:
"K-Sequence, ASCII IN/OUT, Modbus RTU, Provides 5V 220 mA"

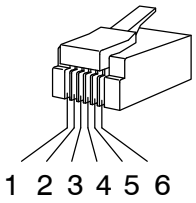
See Errata Sheet at the beginning of this file for updates to this page.

Specifications	H4-EBC	H4-EBC-F
Communications	10BaseT Ethernet	10BaseFL Ethernet
Data Transfer Rate	10Mbps	10Mbps
Link Distance	100 meters (328 ft)	2,000 meters (6,560 ft)
Ethernet Port	RJ45	ST-style fiber optic
Ethernet Protocols	TCP/IP, IPX	TCP/IP, IPX
Serial Port (RJ12)	K-sequence, ASCII	K-sequence, ASCII
Power Supplied	3680mA @ 5VDC	3550mA @ 5VDC
	400mA @ 24VDC	400mA @ 24VDC
Manufacturer	Host Automation Prods	Host Automation Prods

Serial Port Specifications

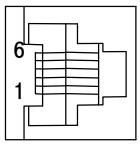
Serial Port Pin Descriptions		
1	0V	Power (-) connection (GND)
2	5V	~~Power (+) connection~~
3	RXD	Receive Data (RS232C)
4	TXD	Transmit Data (RS232C
5	RTS	Request to Send
6	0V	Power (-) connection (GND)

Change to:
"Power Out, 220 mA"



6-pin Male (RJ12)
Modular Plug

6-pin Female (RJ12)
Modular Jack
as oriented on EBC



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 EBC 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.

B

Appendix B
Using the H2 Series EBC with Think & Do

In This Appendix. . . .

Configuring the DL205 I/O Base

The H2 Series EBCs and DL205 I/O are self-configuring. The EBC reads the module and identities it on powerup. Within the Think & Do I/O View tool, the DL205 I/O modules are graphically displayed as soon as a connection is established between your PC and your EBC.

For additional information about establishing a connection between your PC and the H2 Series EBCs, please see the ***Think & Do Software Learning Guide***.

Mapping H2-EBC I/O Points

We recommend that you be familiar with "Getting Started" and "Creating a Project" chapters in the ***Think & Do Studio Learning Guide*** before attempting to map the EBC I/O points/channels to Data Items using ConnectivityCenter.

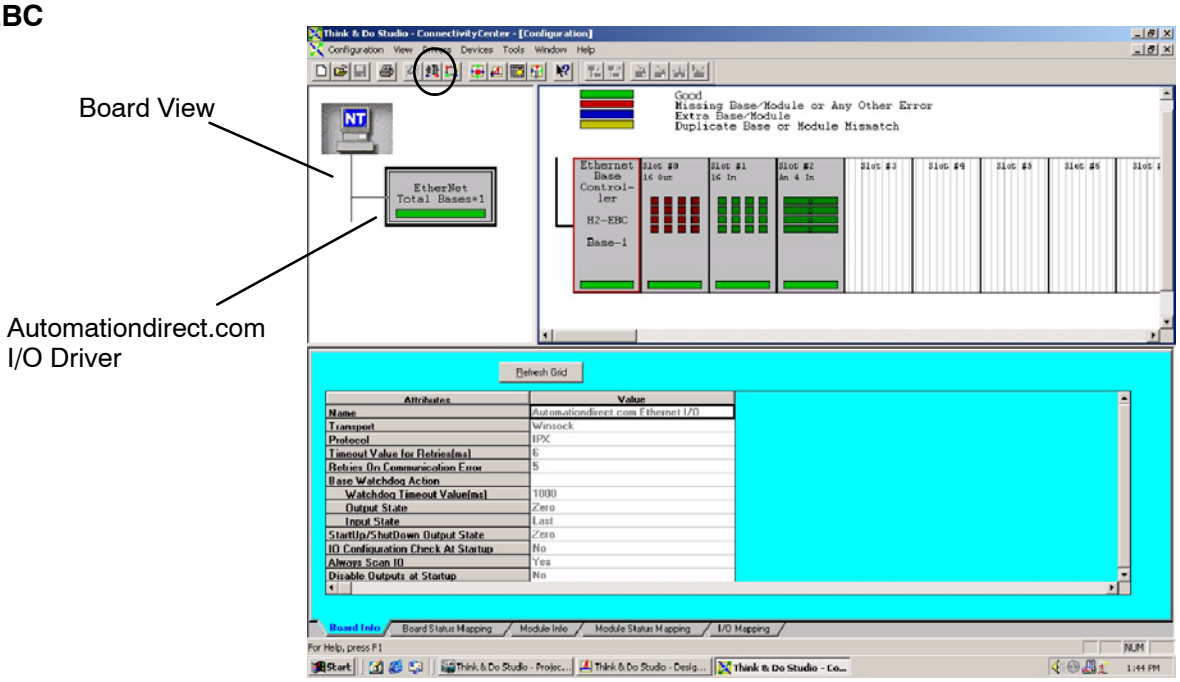
Launching Connectivity Center Tool

To launch ConnectivityCenter:

1) Launch Think & Do Studio ProjectCenter from the Windows desktop by either clicking on **Start**, then **Programs**, next **Think & Do Studio**, finally **ProjectCenter** or click on the **ProjectCenter** icon to start.
2) Click on the File Menu and either open your Think & Do Project or select New.
3) Within ProjectCenter select Windows 2000 or NT Certified PC as the Runtime Target.
4) Then either click **Tools**, then **ConnectivityCenter** to launch the ConnectivityCenter or click on the ConnectivityCenter shortcut in the Project Explorer.
5) Once in ConnectivityCenter click on **Drivers**, then **Add** and select **Automationdirect.com Ethernet I/O Driver**.
5)Then either click on **Configuration**, then **Connect** or click on the Connect toolbar button.

Connecting to the EBC

ConnectivityCenter will draw a picture of your EBC I/O system.

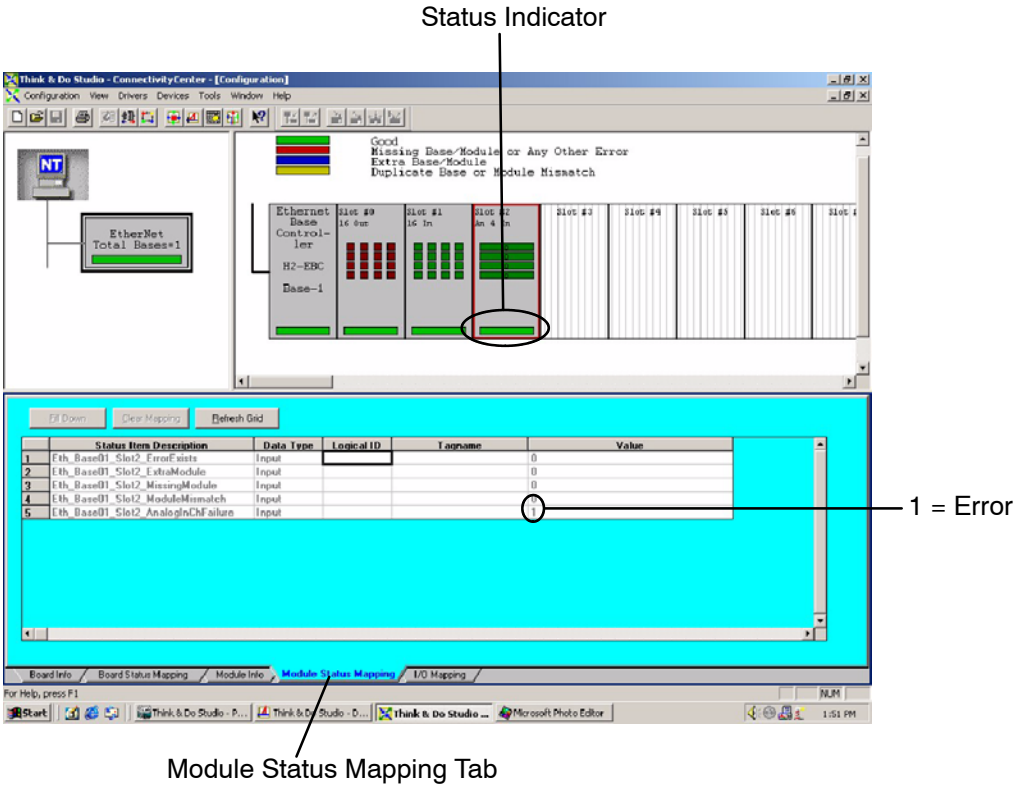


Mapping I/O Points to Data Items

This procedure is discussed in detail in the "Creating a Project" chapter in the ***Think & Do Studio Learning Guide***. This will map your real world I/O to Data Items.

I/O Module Status Word / Bits

I/O Module diagnostic information is listed for each I/O module under the Module Status Mapping tab. Click on a module graphic to display its Status Item Descriptions.

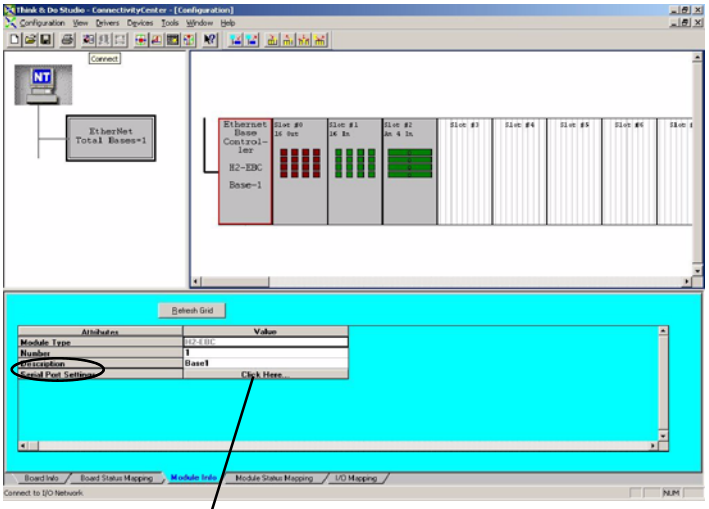


Using EZTouch/EZText Panel with the RJ-12 Serial Port

The H2-EBC has a built-in RS232C serial port that can be used to connect to an operator interface panel. Use ConnectivityCenter to configure the connection from the H2-EBC to the EZTouch or EXText panel. The “HMI Options for Remote Base Controllers” section in the “Operator Screen Techniques” chapter in the ***Think & Do Studio Learning Guide*** discusses configuring and using Optimate Panels with the EBC.

Adding Operator Interface Device

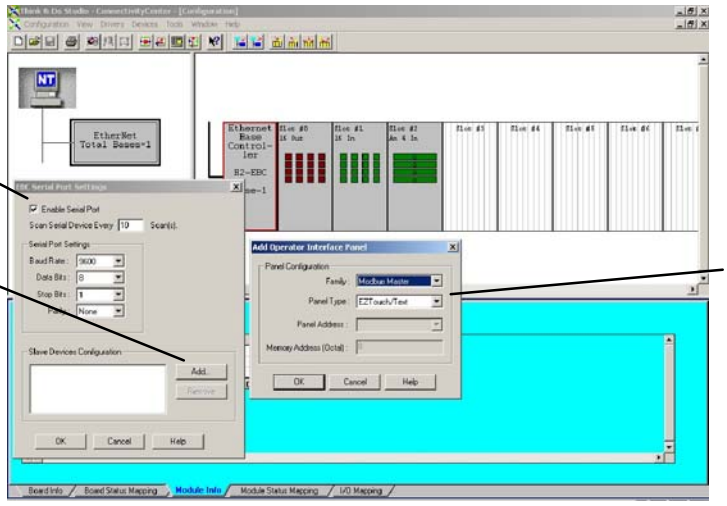
Click on the H2-EBC graphic and Module Info tab in the ConnectivityCenter. The Serial Port Settings attributes are all that will be visible in ConnectivityCenter when the I/O is disconnected. Follow the steps below to configure the EBC’s RJ12 serial port to be used with either the EZTouch or the EZText panels.



1. Click Here to access port settings.

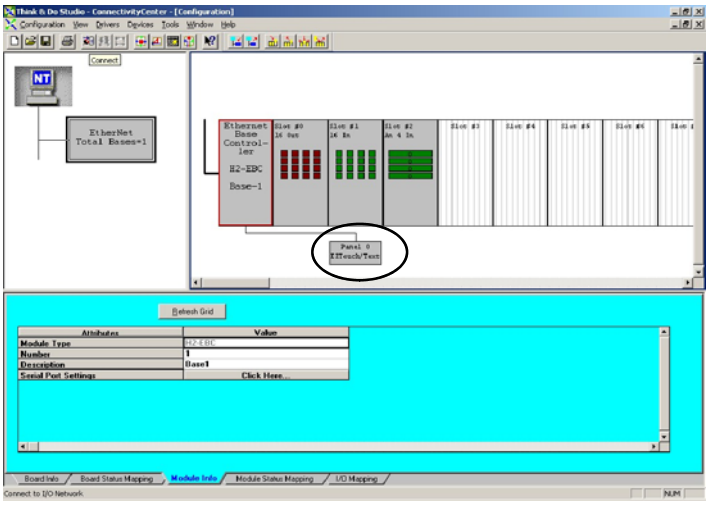
2. Check **Enable Serial Port** to enable the serial port. These settings must match the port configuration of the EZ panel.

3. Click **Add...**



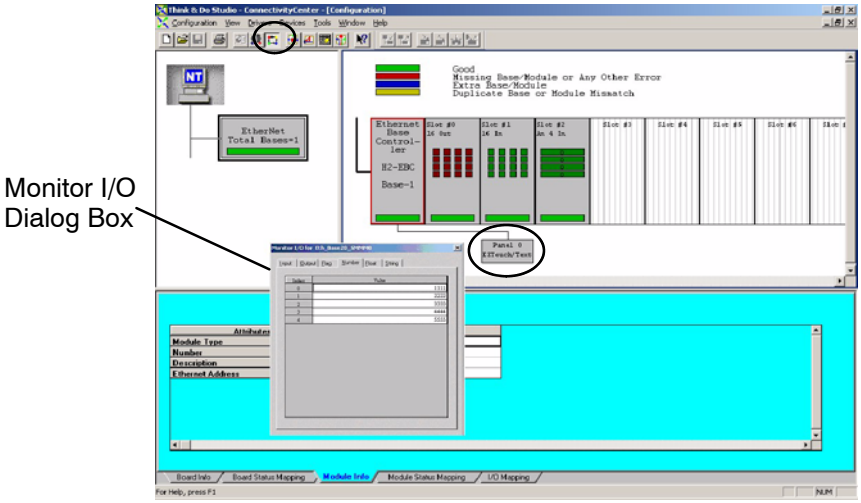
4. Set the Family to Modbus Master and the Panel Type to EZTouch/Text.

Once the EZTouch or EZText panel has been added, it will show up in the list of the configured devices, and an EZTouch/Text panel graphic symbol will be located under the I/O base next to the EBC.



Using Monitor I/O to Verify Panel Operation

Re-connect to the I/O in ConnectivityCenter by either clicking on **Configuration**, then **Connect** or by clicking on the **Connect** toolbar button. Then scan the I/O by either clicking on **Configuration**, then **Scan** or by clicking on the **Scan** toolbar button. Doubleclick on the EZ panel box graphic to launch the Monitor I/O Dialog Box. The Monitor I/O tool allows the user to update the fields at any moment, altough the panel continuously updates the fields with changes as well. All of the "Value" fields in the Monitor I/O Dialog Box are read/write and are updated from the the Monitor I/O Dialog box which takes precedence over updates from the panel.
The user can update bit values (Input, Output and Flag) immediately by one mouse click or by pressing the space bar.
When typing in numbers, the grid will enter the edit mode which will block any conflicting updates from the panel. The edit mode entry is completed after pressing Enter, any arrow key or by selecting a new line.



C

Appendix C
Using the H4 Series EBC with Think & Do

In This Appendix

Configuring the DL405 I/O Base with H4 Series EBCs

Configuring the base is a necessary step in the setup of the H4 Series EBC module. The EBC must know the type and location of each input and output module installed in the base. Once identified, the configuration resides in non-volatile memory on-board the EBC until a change is made.

The H4 Series EBC is partially self-configuring. On powerup, the EBC reads the base to determine the specifications of installed modules. The information available allows the EBC to determine:

- whether the I/O modules are inputs or outputs
- whether the installed modules are standard I/O modules or a High Speed Counter module (other intelligent modules are not supported at this time)

If you **are not** using analog inputs or outputs the H4 Series EBC configures itself. For the following module types, the DL405 I/O system provides the necessary configuration information to the EBC, and the EBC automatically configures the base:

- digital inputs
- digital outputs
- High Speed Counter module

If you **are** using analog inputs or outputs you **must** configure the base manually using a software utility imbedded in Think & Do:

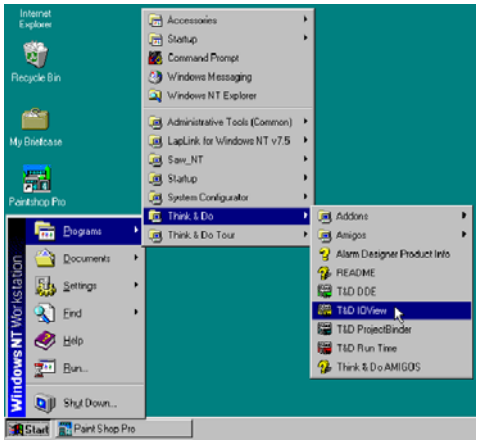
You are ready to configure your base if you have done **all** of the following:

- installed your H4 Series EBC module
- connected power wiring to the EBC terminal strip
- installed I/O modules and expansion bases as necessary for your application
- connected your PC and EBCs to a dedicated Ethernet network
- installed Think & Do (Version 4.4, or later) on your PC

NOTE: The pages that follow explain how to use the Think & Do software utility for configuring the base. For additional information about using the Think & Do software product, please refer to the **Think & Do Software Learning Guide**.

Starting I/O View

I/O View is one of the tools provided by Think & Do software. It is directly accessible from the Start menu after installation. Select **Start**, then **Programs**, then **Think & Do**, then **T&D I/O View**, as shown to the right.

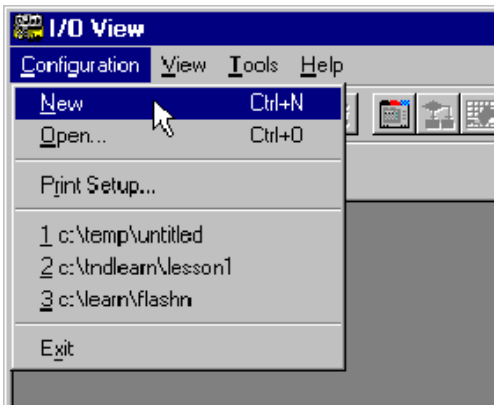


The I/O View window will appear as shown, with a blank I/O configuration screen. If you save this screen without renaming it, the default name will be “IOView1”.

We use a new I/O View window to configure the I/O for an example system. The H4 Series EBC is connected to the PC which is running Think & Do software.

Starting a New Screen in I/O View

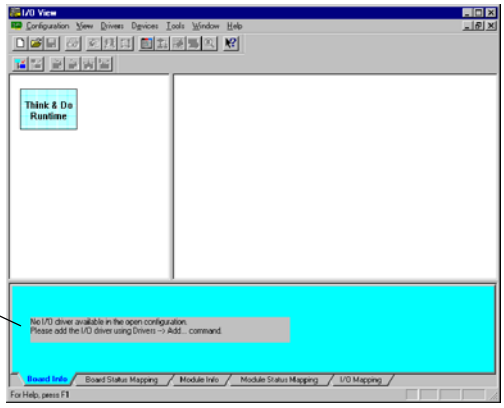
To start a new configuration, select the **Configuration** menu, then **New** as shown.



The next screen is divided into three regions separated by window splitter bars. You can re-size the regions by doing a click-and-drag on a splitter bar.

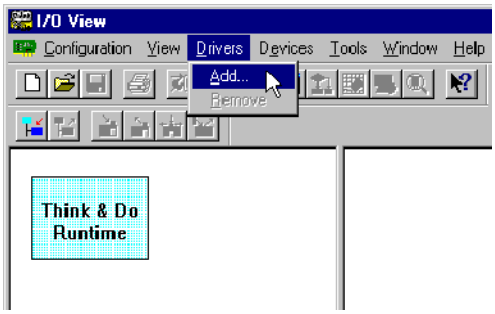
Each project maintains its own record of its I/O configuration.

The first time you open the I/O View window for a new project, it will prompt you to choose an I/O driver, as shown below.

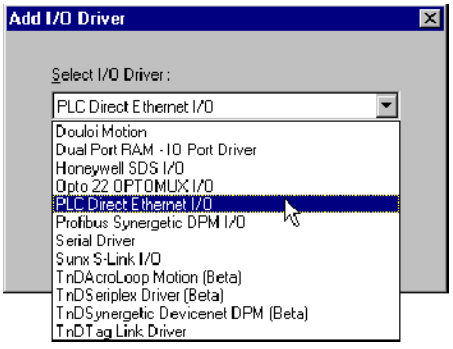


Selecting a Driver

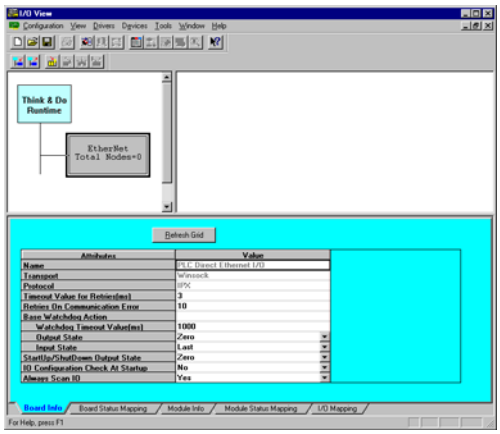
Use the **Drivers** menu and select **Add**, as shown, to access a list of I/O drivers.



Select the **PLC*Direct* Ethernet I/O** driver. Click OK.

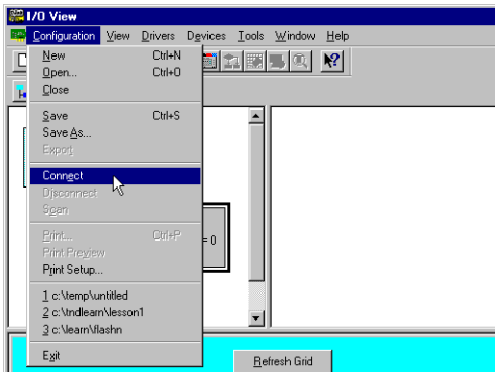


I/O View adds the driver, and attempts to activate the network adapter card. I/O view displays an image of the card as shown to the right.



H4-EBC Base Configuration Screen

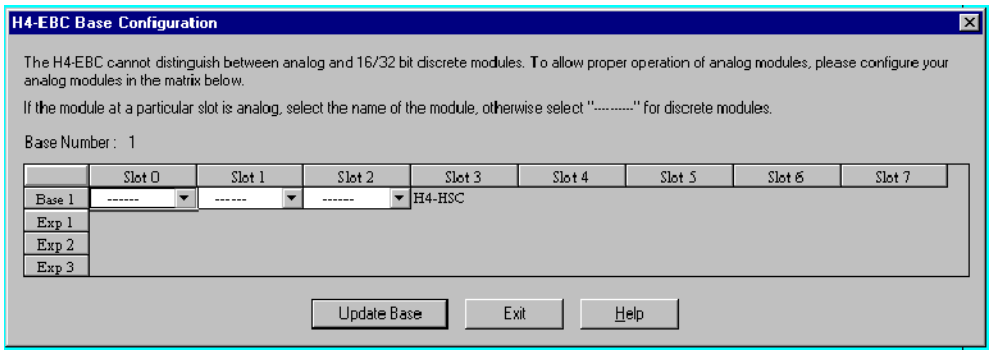
Select the **Configuration** pull-down menu from the I/O View menu bar as shown to the right. Then select **Connect** from the menu. This instructs Think & Do to make the connection to the I/O bases currently on the network.



The module must have a non-zero Module ID set on the DIP switch or an error message will be returned at this point. See page 2-4 “Setting the DIP Switch” for more information.

Think & Do makes a connection to the H4-EBC and automatically displays the **H4-EBC(-F) Base Configuration** screen. The screen will overlay the I/O View screen. Discrete and analog modules are both displayed *initially* as a dashed horizontal line “------” in the block representing the module’s slot location.

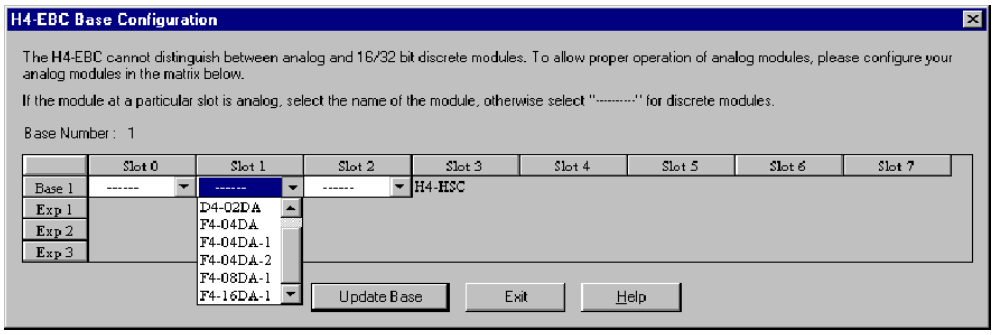
If you have digital inputs and outputs but no analog inputs or outputs, you do not need to do anything additional to configure the base. You may click on Exit or Update Base. Either will return you to the I/O View screen.



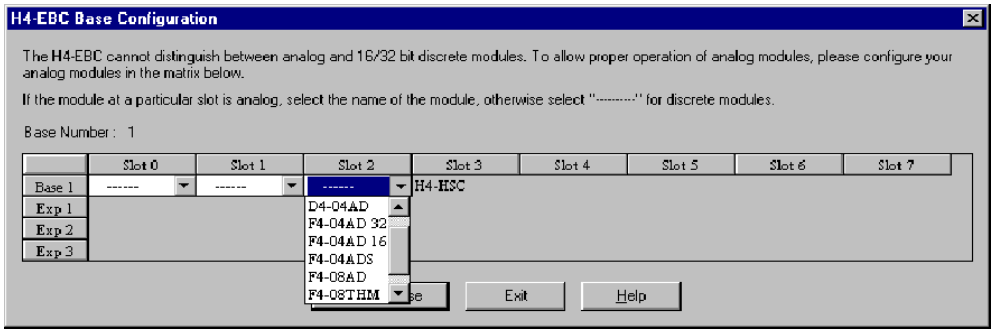
Note: High Speed Counter module may be indicated as D4-HSC or H4-HSC. They are the same.

Identifying Analog Modules

You must identify each analog input or output module by selecting the applicable part number on the Base Configuration screen. The part numbers of all available analog modules appear on the pull–down menu for the appropriate slot (the part number is printed on the face of each module). Click on the arrow beside the slot location to see the menu. The H4 Series EBC automatically distinguishes between input modules and output modules. In the Think & Do implementation of the Base Configuration utility, the pull-down menu for analog input modules lists only *analog input* modules. It does not list analog output modules.

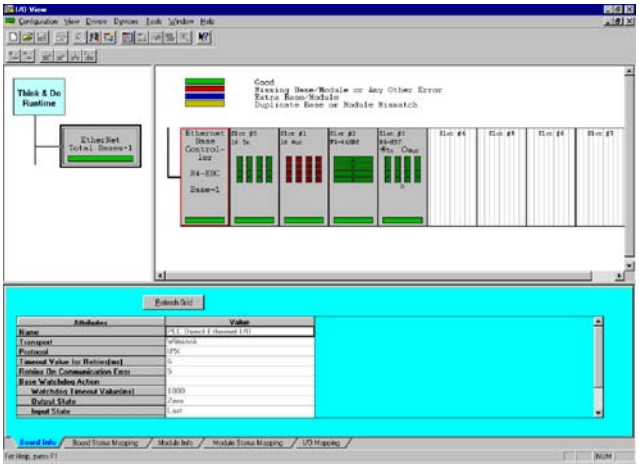


The pull-down menu for analog output modules lists only *analog output* modules.



After selecting the appropriate part number for your analog input or output modules, click on Update. This will save the entries to the H4 Series EBC's non-volatile memory.

After clicking on Update, a graphical representation of the EBC, the base, and I/O modules appears. You have successfully configured the I/O base. The I/O View screen shows an eight-slot base even if you are using a smaller base. Expansion bases are also shown if connected.



D

Appendix D
Using the H2 Series EBC with KEP*Direct* OPC Server

In This Appendix. . . .

Introduction to KEP*Direct* OPC Server

Introduction to OPC

OPC, OLE (**O**bject **L**inking and **E**mbedding) for Process Control, is an industry standard created by a number of worldwide leading hardware and software suppliers in cooperation with Microsoft. The OPC Data Access specification, as maintained by the OPC Foundation, is a non-proprietary technical specification that defines a set of standard interfaces based upon Microsoft's OLE/COM technology. An OPC server (driver) allows items such as distributed control systems, programmable logic controllers, I/O systems and smart field devices to communicate with a wide range of HMI/SCADA (client) software packages residing on a PC. Traditionally, each software or application developer was required to write a custom interface, or server/driver, to exchange information with hardware field devices. OPC eliminates this requirement allowing manufacturing customers true plug and play connectivity and the freedom to choose products based on their automation requirements.

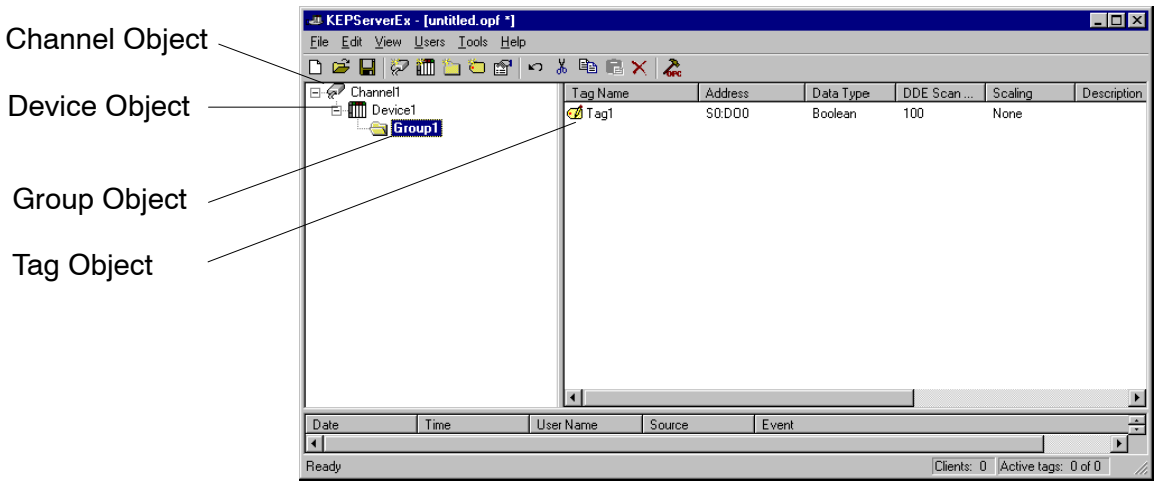
DDE Support

While KEP***Direct*** is first and foremost an OPC server, KEP***Direct*** recognized that a number of legacy applications still depend upon DDE for their underlying client server technology. Early in the development of Windows, Microsoft provided a generic client server technology called DDE (Dynamic Data Exchange). DDE did provide a basic architecture that would allow many windows applications from a wide range of vendors to share data. But there was one problem, DDE was not designed for the industrial market lacking much of the speed and robustness desired in an industrial setting. However, this did not stop DDE from becoming a dominant client/server architecture, largely due to its availability in most windows applications.

KEP*Direct*

KEP***Direct*** Enhanced OPC/DDE Server is a 32 bit windows application that provides a means of bringing data and information from a wide range of industrial devices and systems into client applications on your Windows PC. KEP***Direct*** falls under the category of a "Server" application. It is very common to hear the term "client/server application" in use across many software disciplines and business segments. In the industrial market, it has usually come to mean the sharing of manufacturing or production data between a variety of applications ranging from human machine interface software and data historians, to large MES and ERP applications.

At a high level, the KEP***Direct*** OPC Server is comprised of several objects that are described on the next page.



Channel Object: Each protocol or driver used in a KEP*Direct* project is referred to as a channel. A channel refers to a specific communications driver. A KEP*Direct* project can consist of many channels each with unique communications drivers or each with the same communications driver.
Each channel name must be unique in a KEP*Direct* application. The channel name entered here will be part of the OPC browser information.

Device Object: Unlike the channel name, ”Device names” can be the same from one channel to the next. The device name is a user defined logical name for the device. The device name and channel name will be part of the OPC browser information as well as a DDE item name. Within an OPC client the combination of channel name and device name would appear ”ChannelName.DeviceName”.

Group Object: KEP*Direct* allows tag groups to be added to your project. Tag groups allow you to tailor the layout of OPC data in logical groupings that fit the needs of your application. Using tag groups allows multiple sets of identical tags to be added under the same device. This can be very convenient when a single device handles a number of similar machine segments. From an OPC client standpoint, the use of tag grouping allows you to segregate your OPC data into smaller tag lists, which can make finding a specific tag easier when browsing the server.

Tag Object: KEP*Direct* allows both dynamic tags, (tag entered directly at the OPC client that specify device data) and user defined tags. User defined tags have the benefit of allowing the tag to be browsed from an OPC client that supports tag browsing. User defined tags also support tag scaling. Unlike many of the dialogs you will find in KEP*Direct*, the tag properties dialog has a number of features that are driven by icons. The tag name is part of the OPC browse data. Tag names must be unique within a given device branch or tag group branch. If your application is best suited by using blocks of tags with the same names, use tag groups to segregate the tags.

KEP*Direct* Project: Adding and Configuring a Channel

Running the Server

KEP***Direct***, like any OPC server, can be started a number of ways. One of the benefits of OPC technology is that your OPC client can automatically invoke the server when it attempts to connect and collect data from it. In order for this automatic mode of operation to occur you must first create and configure a project. Once you have created a project, KEP***Direct*** will automatically select the most recently used project when it is invoked by an OPC client.

Initially however, you need to manually invoke KEP***Direct*** using either the desktop icon, if you chose to install it, or by selecting KEP***Direct*** from the windows start menu. Depending on any changes you may have made to the appearance of KEP***Direct***, once invoked you should be presented with the following interface. To learn more about the various elements of the user interface see (Basic KEP***Direct*** Components).

While discussing how to start KEP***Direct*** its important to understand what the system requirements are for running the server. KEP***Direct*** has been designed to place as little strain on your system as possible.

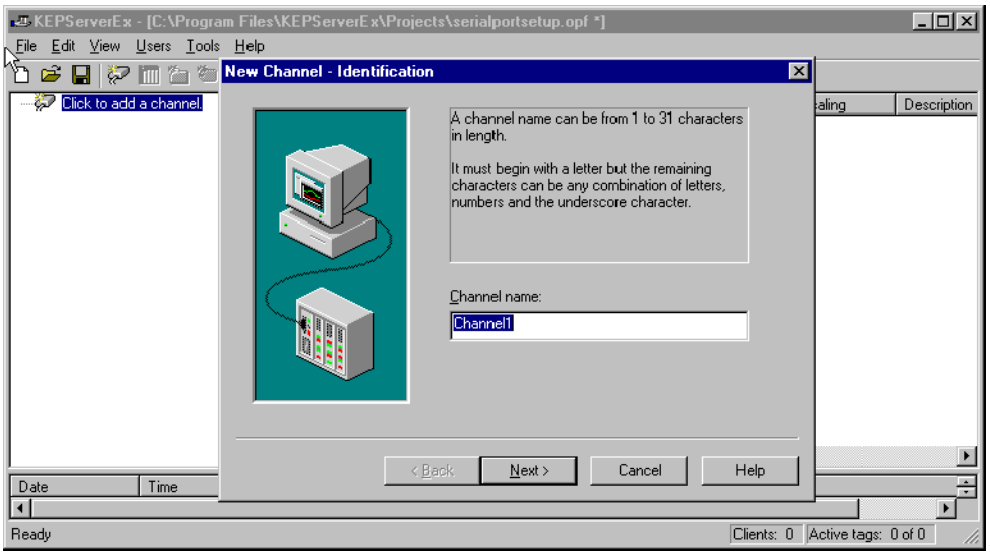
Recommended System Requirements:

400Mhz Pentium
64 Megs of Ram
10 Megs of Hard Disk Space
Windows NT(SP6a)/2000 (Strongly recommended for industrial settings)
Available Ethernet Card

Adding a Channel

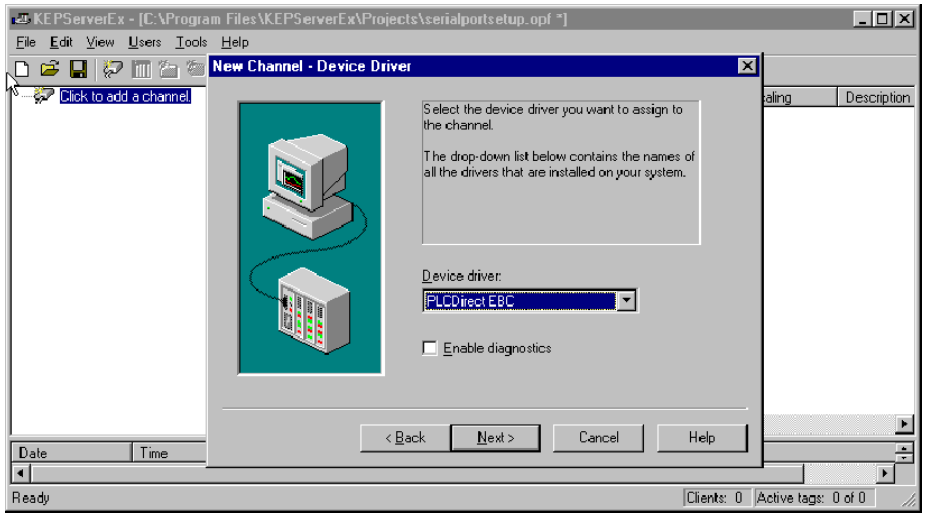
A channel refers to a specific communications driver. A KEP***Direct*** project can consist of many channels each with unique communications drivers or each with the same communications driver. Depending on the driver or drivers you have installed you can define a number of channels within a single project. A channel acts as the basic building block of an OPC link. Properties like communications port, baud rate, and parity are contained at the channel level. Each channel name must be unique in a KEP***Direct*** project. The channel name can be up to 31 characters long.

To add a new channel to your project you can use the Edit menu > New Channel, the Toolbar Add Channel, or the “Click to add a channel” dialog.



Selecting the Device Driver

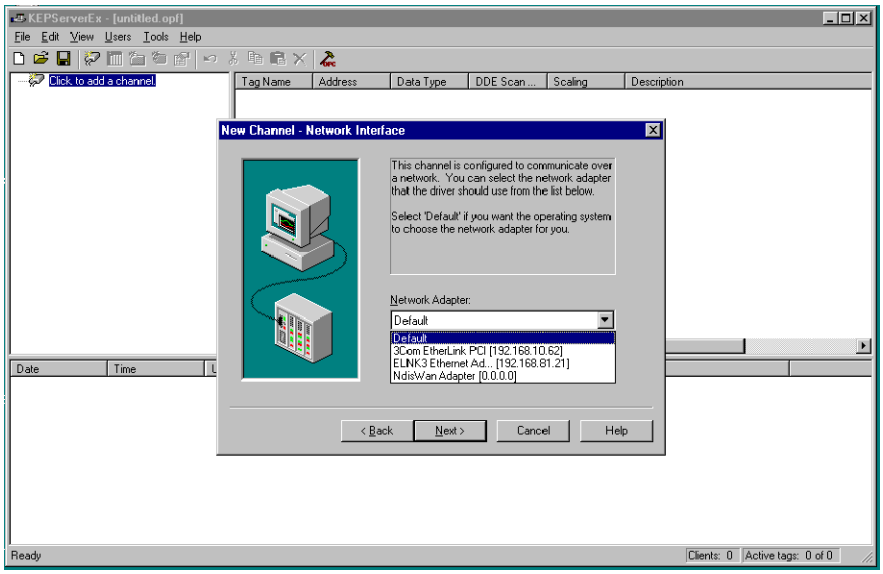
Select the device driver you want to assign to the channel. A driver list will be presented displaying all of the device drivers that are installed in your system.



Selecting the ”Enable diagnostics” check box will enable diagnostic information to be available to your OPC application for this channel. With diagnostic functions enabled, diagnostic tags are available for use within client applications. In addition to diagnostic tags, a diagnostic window is also available when this feature is enabled. The diagnostic features of KEP***Direct*** do require a minimal amount of overhead processing. For this reason it is recommended that you only use the diagnostic features when needed and disable them when not in use which is the default case.

Selecting the Network Adapter

The Network Interface selection allows you to select a specific NIC card for the AutomationDirect EBC Ethernet driver to use based on the NIC name or its assigned IP address. By selecting a specific NIC interface you will be able to force the driver to send all Ethernet communication through the specified NIC. If you do not know which NIC you should use, select the ”Default” condition.



Setting the Server Writes Optimizations

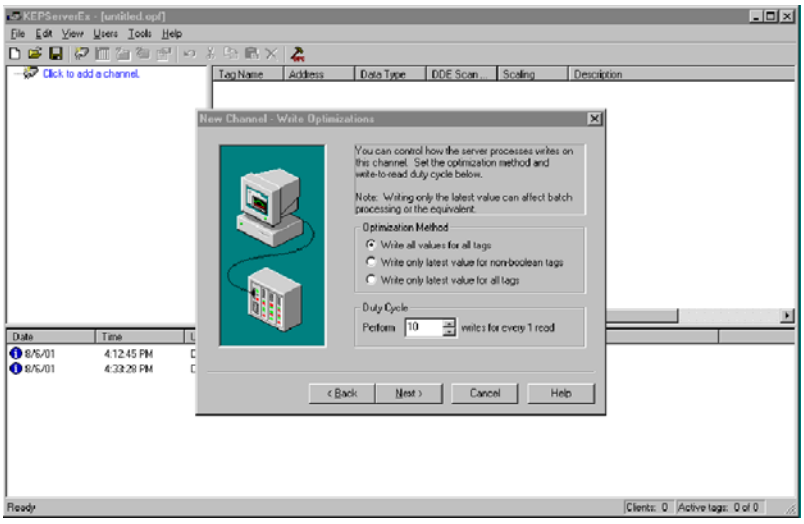
As with any OPC server, writing data to your device may be the most important aspect of your application. Insuring that the data written from your OPC client application gets to the device in a timely manners is the goal of the server. KEP*Direct* provides a number of optimization settings that can be used to tailor the server to meet the needs, and improve the responsiveness of your application.

There are currently three write optimization modes. The following is a brief description of the modes. For a detailed explanation, refer to the “Channel Properties – Write Optimizations” section in the KEP*Direct* on–line help file.

NOTE: We strongly suggest that you characterize your application for compatibility with these write optimization enhancements before using them in a production environment.

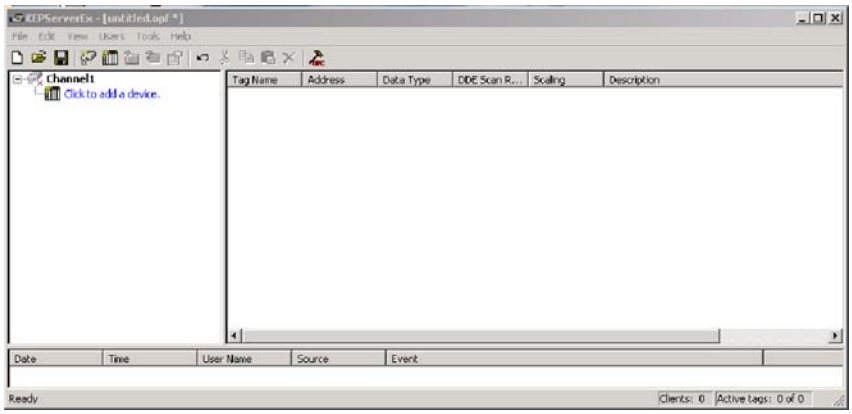
The default mode, ”**Write all values for all tags**” will force the server to attempt to write every value to the controller. This mode insures that everything written from your OPC client applications will be sent to the target device. While writing every value to the device may seem like the best course of action, there are a number of applications where writing every value, many of which may be the same value, over and over may be simply a waste of communications bandwidth.
The ”**Write only latest value for non–boolean tags**” allows any value that is not a boolean value to be updated in the server’s internal write queue and will then be sent to the device at the next possible opportunity. This can dramatically improve the overall performance of your application. This feature must be used with a clear understanding of how it will affect the operation of your application.
The final write optimization mode, ”**Write only the latest value for all tags**”, takes the operation described for the second mode and applies it to all tags.

The **Duty Cycle** selection allows you to control the ratio of write operations to read operations. By default the duty cycle is set to ten. This means that ten writes will occur for each read operation. If your application is doing a large number of continuous writes but you need to insure that read data is still given time to process, you may want to reduce the Duty Cycle. A setting of one will result in one read operation for every write operation. In all cases if there are no write operations to perform, reads will be processed continuously.



Saving the New Channel Settings

With **Channel1** added to the server, the KEP*Direct* window will appear as follows:



Note that the channel is shown using the channel name given, but it also has a small red ”x” below the channel icon. The red ”x” indicates that the channel does not contain a valid configuration. **Channel1** is not valid because a device has not yet been added to the channel.

Using Multiple Channels in a Project

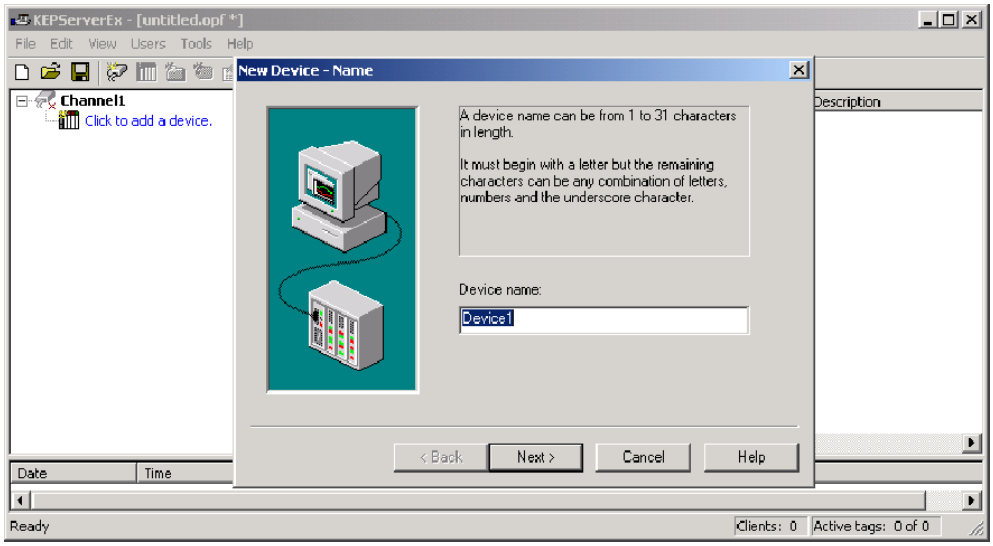
KEP*Direct* supports the use of multiple channels. As you add channels to your project you can specify either the same communications driver or different communications drivers. Most communication drivers offered by KEP*Direct* support operation on up to 16 communications ports or ethernet network connections simultaneously. By defining multiple channels you can improve the overall performance of you application. In the case of either a serial driver or Ethernet driver using multiple channels allows you to spread large communications loads across the multiple channels. A good example of this would be a serial driver that is being used to communicate with eight devices on the serial line. Normally the communications driver used in this application would be responsible for gathering data from all eight devices in a round robin fashion. If this same application is reconfigured to use multiple channels assigned to multiple communications ports, the device load can be divided across the channels. The end result is reduce work load on each channel and dramatic improvements in the responsiveness of your application. The need to use multiple channels is dependent solely on the needs of your application. In either case there is no additional cost involved to use a licensed driver on multiple communications or Ethernet ports.

KEP*Direct* Project: Adding and Configuring a Device

Adding a Device

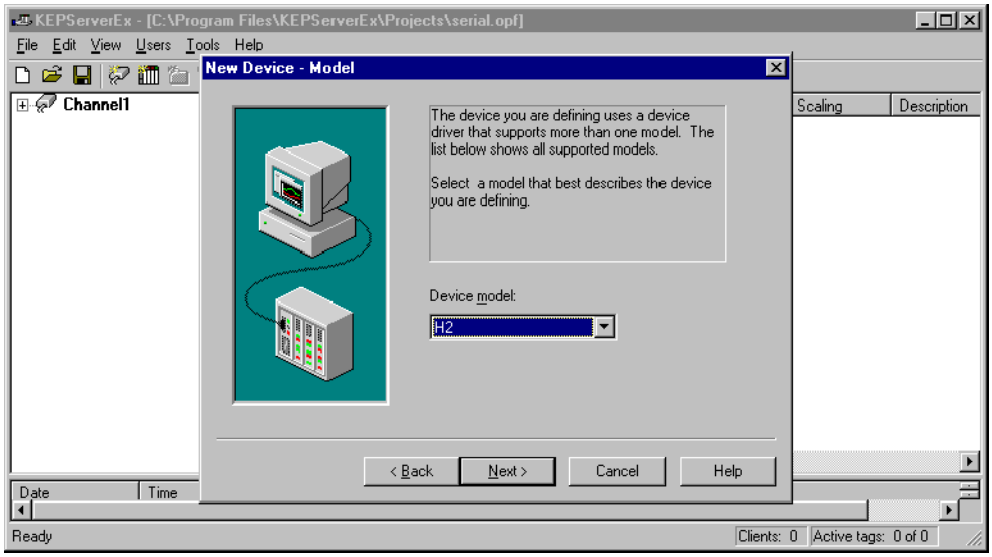
Once a channel has been configured in a KEP***Direct*** project, a device must be added to the channel. Devices represent PLCs, I/O devices or other hardware that the server will communicate with. Device selection is restricted by the device driver the channel is using.

To add a device to a channel, select the desired channel and use the Edit menu > New Device, the Toolbar Add Device, or the “Click to add a device” dialog.



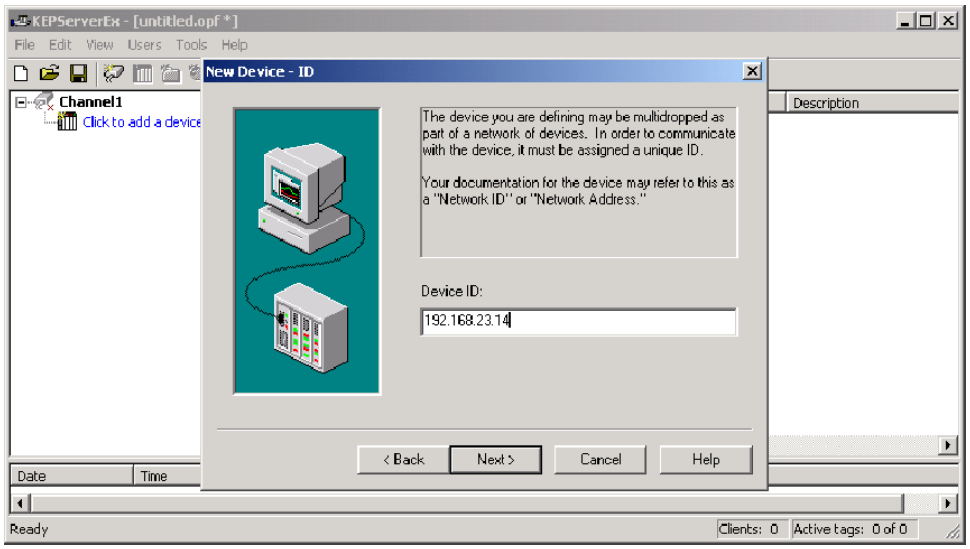
Selecting the Device Model

The ”Model” parameter allows you to select the specific type of the device associated with a device ID. The contents of the model selection drop down will vary depending on the chosen communication driver.



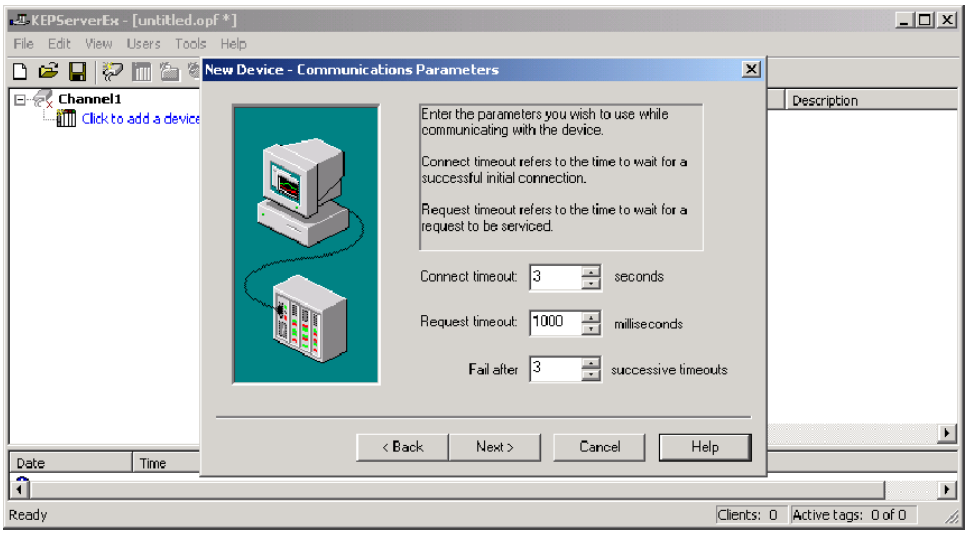
Selecting the Device Model

The ”Device ID” parameter allows you to specify the driver specific station or node address for a given device. Since the Automationdirect EBC driver is an Ethernet based driver, a unique and valid TCP/IP address must be entered. IPX protocol is not supported.



Setting the Device Timeout Properties

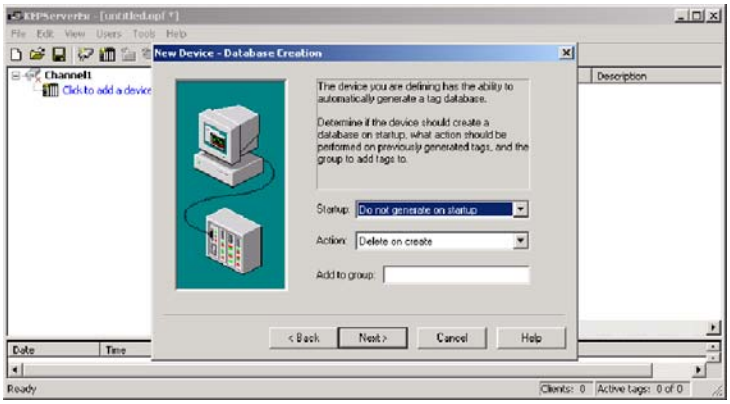
Device timeout parameters allow a driver’s response to error conditions to be tailored to the needs of your application. The timeout parameters are specific to each device you configure. Each of the field parameters is defined in detail in the “Device Properties – Timeout” section in the KEP***Direct*** on-line help file.



The **Connection timeout**: allows the time required to establish a socket connection to a remote device to be adjusted. The **Request timeout**: is used by all drivers to determine how long the driver will wait for a response from the target device. The **Fail after** parameter is used to determine how many times the driver will retry a communications request before considering the request to have failed. If your environment is prone to noise induced communications failures you may want to increase the number of **retries** the driver performs.

Automatic OPC Tag Database Generation

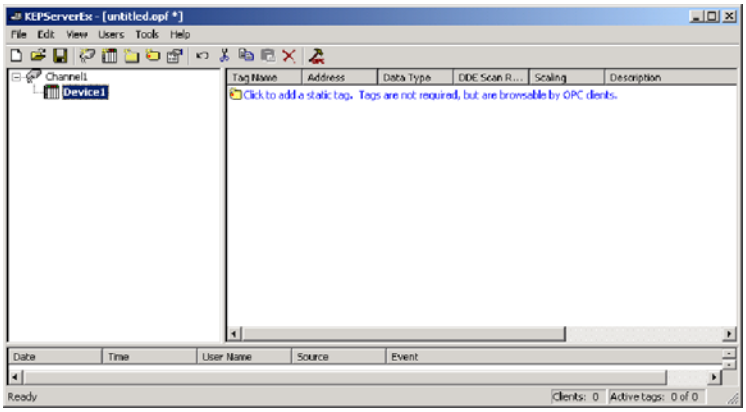
The automatic OPC tag database generation features of KEP*Direct* have been designed to make the setup of your OPC application a Plug and Play operation. Since the Automationdirect EBC communication driver supports this feature, you can configure it automatically build a list of OPC tags within KEP*Direct* that correspond to device specific data. The automatically generated OPC tags are then browsable from your OPC client. The OPC tags that are generated are dependent upon the nature of the supporting driver. Each field selection is defined in detail in the "Automated OPC Tag Base Generation" section in the KEP*Direct* on-line help file.



The "**Automatic tag database generation on device startup**" selection allows you to configure when OPC tags will be automatically generated. There are three possible selections. The default condition, "**Do not generate on startup**", will prevent the driver from adding any OPC tags to tag space of KEP*Direct*. The selection "**Always generate on startup**", will cause the driver to always evaluate the device for tag information and to add OPC tags to the tag space of the server each time the server is launched. The final selection "**Generate on first startup**" will cause the driver to evaluate the target device for tag information the first time this KEP*Direct* project is run and to add any OPC tags to the server tag space as needed. When the automatic generation of OPC tags is selected, any tags that are added to the server's tag space must be saved with the project. You can configure your KEP*Direct* project to auto save from the Tools > Options menu.

Saving the New Device Settings

With **Device1** added to **Channel1**, the KEP*Direct* window will appear as follows:



KEP*Direct* Project: Adding Tags to the Project

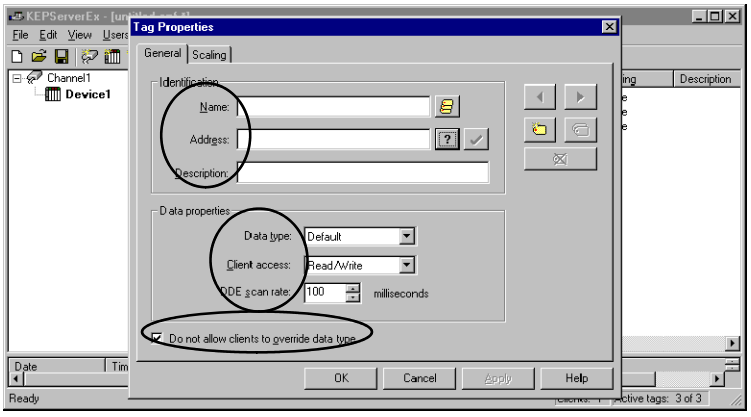
There are two ways to get data from a device to your client application using KEP***Direct***. The first method, and most common method, of defining tags is called **User Defined Tags**. This requires that you define a set of tags in the server project and then use the name you assigned to each tag as the item of each OPC/DDE link between the client and the server. The primary benefit to this method is that all user defined tags are available for browsing within OPC clients. Additionally, user defined tags also support scaling.

The second method of defining tags is called **Dynamic Tags**. Dynamic tags allow you to define tags in the client application. Instead of providing the server with a tag name as the OPC/DDE item, you would provide the device address (and optionally a data type). The server will create a tag for that location and start scanning for data automatically. KEP***Direct*** allows tag groups to be added to your project.

Tag groups allow you to tailor the layout of OPC data in logical groupings that will fit the needs of your application. Using tag groups allows multiple sets of identical tags to be added under the same device. This can be very convenient when a single device handles a number of similar machine segments. From an OPC client standpoint, the use of tag grouping allows you to segregate your OPC data into smaller tag lists, which can make finding a specific tag easier when browsing the server.

User Defined Tags

Each field selection is defined in detail in the **Tag Properties** section in the KEP***Direct*** on-line help file. A brief description of each is listed below.



The tag **Name:** parameter allows you to enter the string that will represent the data available from this tag. The tag name can be up to 31 characters in length. While using long descriptive names is generally a good idea, keep in mind that some OPC client applications may have a limited display window when browsing the tag space of an OPC server. The tag name is part of the OPC browse data. Tag names must be unique within a given device branch or a tag group branch. If your application is best suited by using blocks of tags with the same names, use tag groups to segregate the tags.

The **Address:** parameter allows you to enter the desired driver address for this tag. To determine how an address should be entered, you can use the **Hints button** next to the address parameter. Hints provide a quick reference guide to the address format of the driver. Once you have entered an address you can test it by using the check address button. When pressed, the check address button attempts to validate the address with the driver. If the driver accepts the address as entered no message will be displayed. If an error is detected a pop-up will inform you of the error. Keep in mind that some errors will be related to the data type selection and not the address string.

The **Description:** parameter allows you to attach a comment to this tag. A string of up to 64 characters can be entered for the description. If you are using an OPC client that supports Data Access 2.0 Tag Properties, the description parameter will be accessible from the Item Description property of the tag.

The **Data Type:** selection allows you to specify the format of the tag's data as it is found in the physical device. The data type setting is an important part of how a communication driver reads and writes data to a device. For many drivers the data type of a particular piece of data is rigidly fixed.
The available data type selections are:

- **Default** - This type allows the driver to choose its default data type see the specific driver help for details
- **Boolean** - Single bit data On or Off
- **Char** - Signed 8 bit data
- **Byte** - Unsigned 8 bit data
- **Short** - Signed 16 bit data
- **Word** - Unsigned 16 bit data
- **Long** - Signed 32 bit data
- **Dword** - Unsigned 32 bit data
- **Float** - 32 bit Real value IEEE format
- **String** - Null terminated ASCII string
- **Double** - 64 bit Real value IEEE format
- **BCD** - Two byte packed BCD value range is 0 - 9999
- **LBCD** - Four byte packed BCD value range is 0 - 99999999

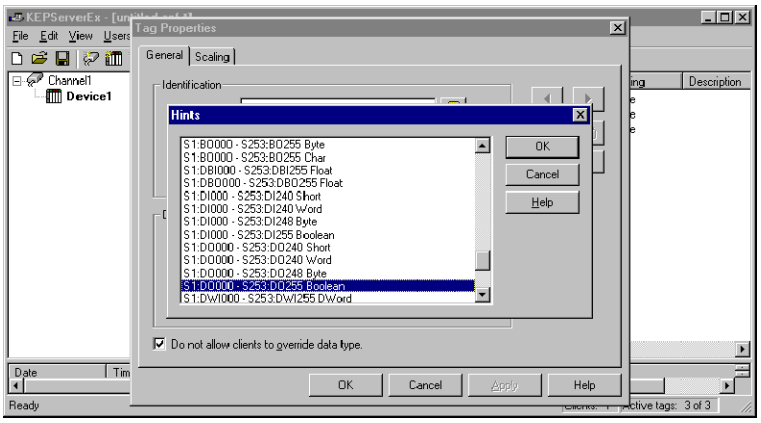
The **Client access:** selection allows you to specify whether this tag is Read Only or Read/Write. By selecting Read Only you can prevent client applications from changing the data contained in this tag. By selecting Read/Write you are allowing client applications to change this tag's value as needed.

The **DDE scan rate:** parameter allows you to specify the the update interval for this tag when used in a DDE client. OPC clients can control the rate at which data is scanned by using the update rate that is part of all OPC groups.

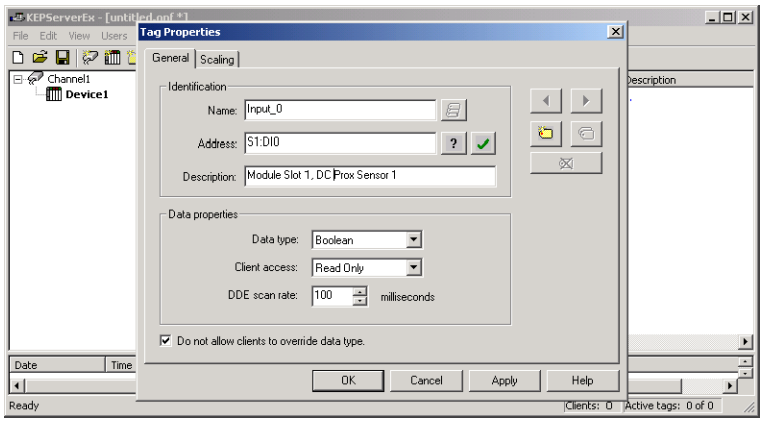
The **Do not allow client to override data type** selection allows you force OPC clients to use the data type you have specified for this tag. OPC clients can specify how they desire to view the data from a particular tag.

Creating a User Define Tag

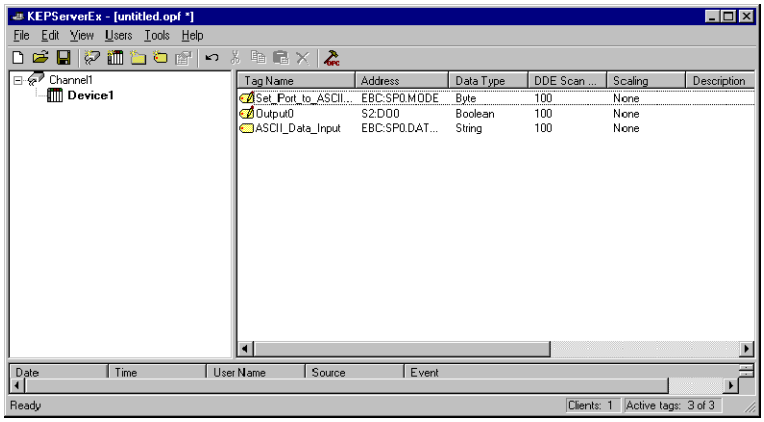
To determine how an address should be entered, use the Hints button "**?**" to the right of the address field. Hints provide a quick reference guide to the address format of the driver.



Once you have entered an address you can test it using the check address "✓" button. When pressed, the check address button attempts to validate the address with the driver. If the driver accepts the address as entered, no message will be displayed. If an error is detected, a pop-up window will inform you of the error. Keep in mind that some errors will be related to the data type selection and not the address string. Below is an example of a valid tag properties.



The window below shows a valid configured channel, device and several user defined tags.



H2 Series EBC I/O Addressing

I/O slots must be individually addressed in the following form: S<ss>:<t><nn> where ss is the slot number (0 to 8), t is the address type (X, Y, K, V, DI, D0, WI, W0, etc.), and nn is the address. The address ranges from 0 to an upper limit determined by the module occupying the slot.

I/O Type	Syntax	Data Type
Discrete Inputs	X or DI<nn> nn = Bit Number (decimal)	**Boolean**, Byte, Char, Word, Short, DWord, Long
Discrete Outputs	Y or DO<nn> nn = Bit Number (decimal)	**Boolean**, Byte, Char, Word, Short, DWord, Long
Byte Inputs	BI<nn> nn = Bit Number (decimal)	**Byte**, Char
Byte Outputs	BO<nn> nn = Bit Number (decimal)	**Byte**, Char
Word Inputs	K or WI<nn> nn = Bit Number (decimal)	**Word**, Short
Word Outputs	V or WO<nn> nn = Bit Number (decimal)	**Word**, Short
DWord Inputs	DWI<nn> nn = Bit Number (decimal)	**DWord**, Long
DWord Outputs	DWO<nn> nn = Bit Number (decimal)	**DWord**, Long
Float Inputs	FI<nn> nn = Bit Number (decimal)	**Float**
Float Outputs	FO<nn> nn = Bit Number (decimal)	**Float**
Double Inputs	DBI<nn> nn = Bit Number (decimal)	**Float**
Double Outputs	DBO<nn> nn = Bit Number (decimal)	**Float**

H2-EBC I/O Addressing Example

Each field selection is defined in detail in the "Tag Properties" section in the KEP*Direct* on-line help file.

H2 Series EBC Module	Slot 0 8 Inputs Addresses S0:X0 to S0:X7	Slot 1 32 Inputs Addresses S1:X0 to S1:X31	Slot 2 4 Analog Inputs Addresses S2:K0 to S2:K3	Slot 3 8 Outputs Addresses S3:Y0 to S3:Y7	Slot 4 16 Outputs Addresses S4:Y0 to S4:YI5	Slot 5 8 Analog Outputs Addresses S5:V0 to S5:V7

Appendix E
Using the KEP*Direct* OPC Quick Client

In This Appendix. . . .

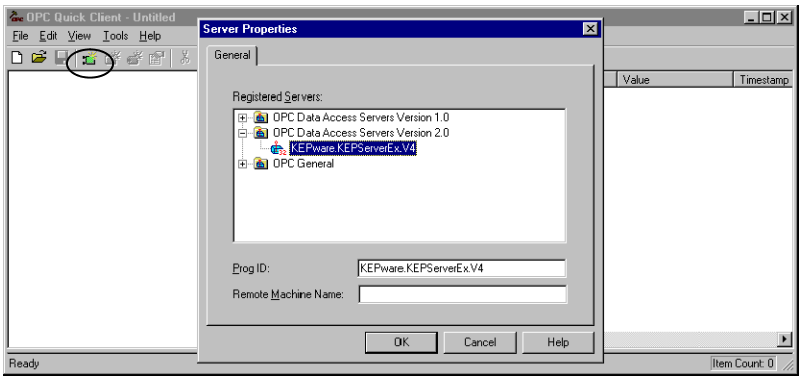
— Creating a KEP*Direct* Quick Client Project
— Using the RJ12 Serial Port in ASCII Mode

Creating a KEP*Direct* Quick Client Project

KEP***Direct*** Quick Client can be used to assist in the test and development of KEPware's OPC Data Access 1.0 and 2.0 Servers.

Connecting the Client to the OPC Server

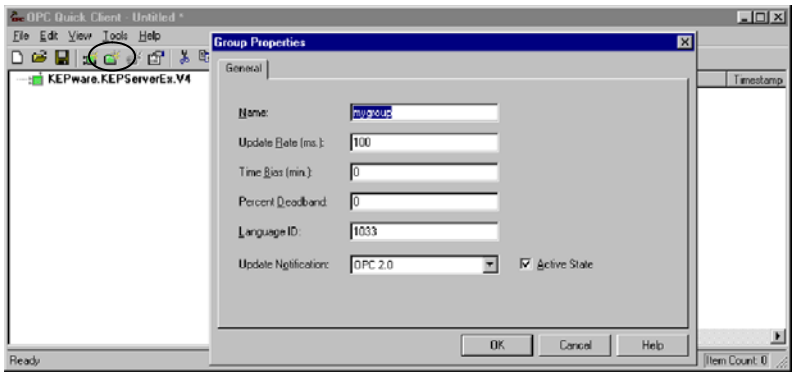
A server connection provides a link between the Quick Client and the KEP***Direct*** OPC server. To add a server connection to the Quick Client, you can use either the Edit menu>New Server Connection or click on the New Server icon in the toolbar menu.



Specify the Prog ID of the OPC Server the client should connect to. You can browse for registered servers by expanding any of the branches. Double-clicking on any registered server will automatically update the Prog ID field. For more information on the registered servers, click on the **Help** button to display the "Server Connection" section of the on-line help file. Once a connection to the OPC server has been established, additional "Server Operations" can be accessed by right clicking on the highlighted server in the right window column or by using the **Tools** menu>Server selection.

Creating a Client Group

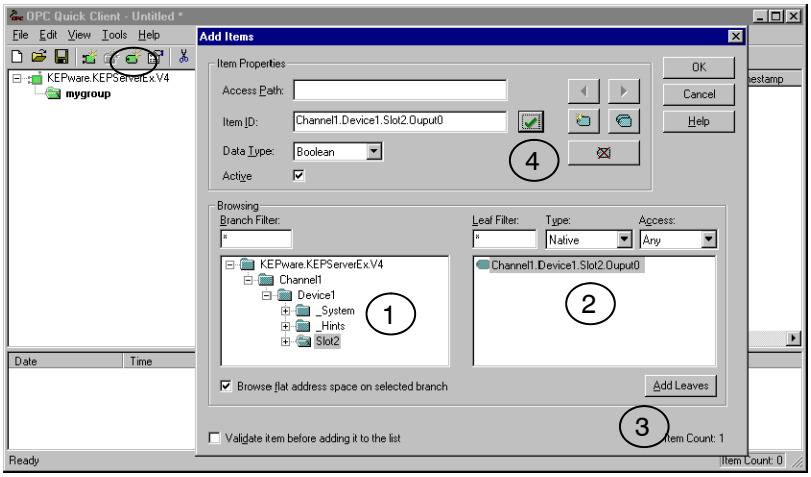
A group is used to organize a collection of items with a common set of properties. To add a Group to the Quick Client, you can use the **Edit** menu>New Group or click on the **New Group** button in the toolbar menu.



A The group specifies the following properties: group **Name**, **Update Rate**, **Time Bias**, **Percent Deadband**, **Language ID**, **Active State** and the typeof data connection that should be made to the server. For detailed information on the group properties, click on the **Help** button to display the "**Group**" section of the on-line help file. Once a Group has been created, additional "Group Operations" can be accessed by right clicking on the highlighted branch Group or by using the Tools menu>Group selection.

Selecting a Group Item

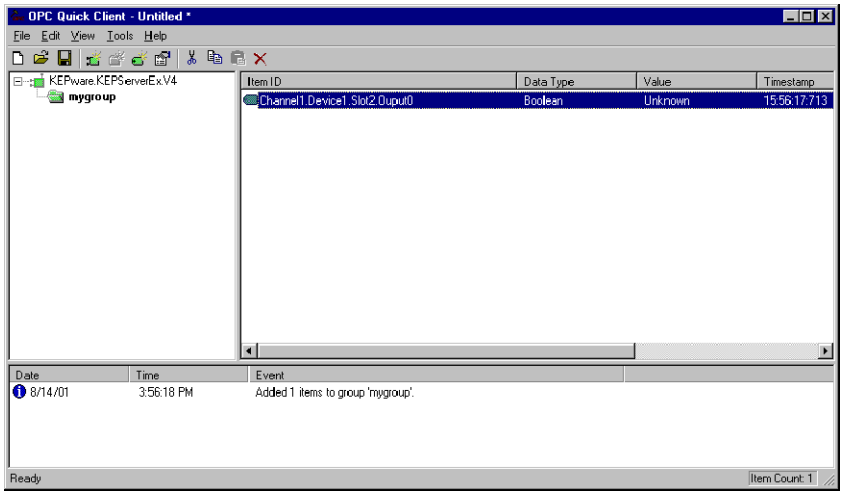
Items represent data that may be accessed via the OPC server. An item specifies the following properties: **Access Path**, **Item ID**, **Data Type** and **Active** state. For detailed information these properties, click on the **Help** button to display the **Item** section of the on-line help file. To add an **Item** to the Quick Client Group, you can either use the Edit menu>New Item or click on the New Item icon on the toolbar.



If the OPC Server was configured to automatically generate OPC tags, the generated tags would be browsable from the OPC client. If automatic tag generation was not selected, create an item by:

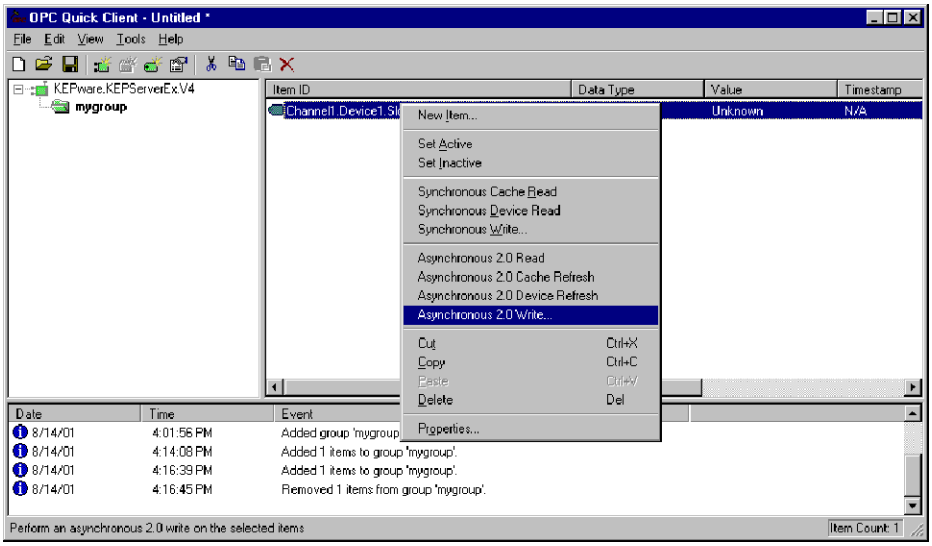
1) browsing the OPC Server branch tags
2) highlighting the desired tag in the right column
3) clicking on the “Add Leaves” button
4) clicking on the “Green Check Mark” button to validate the item
5) and clicking on the “OK” button.

After clicking on the **OK** button, the following window will display the created items.

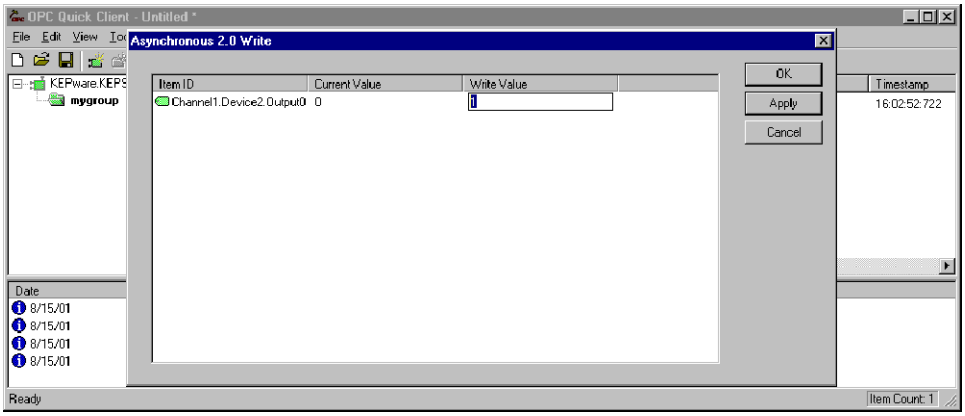


Item Operations

Item operations can be accessed by right clicking on the desired item or by using the **Tools** menu>Group selection.

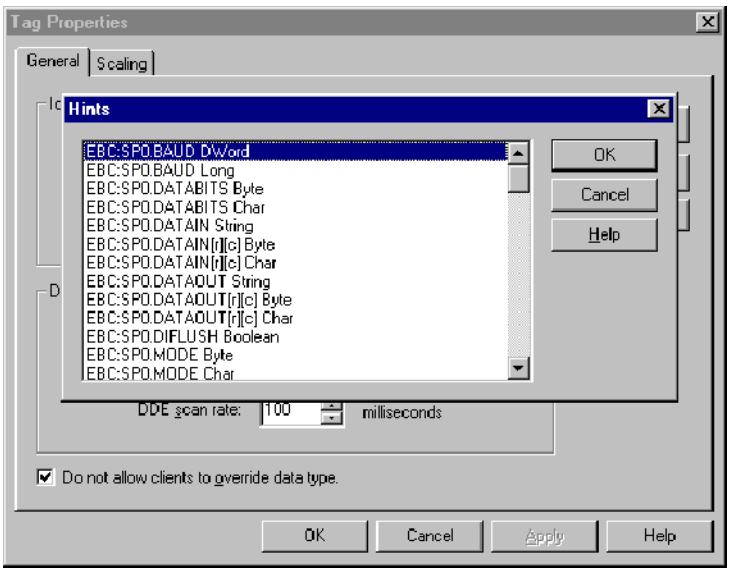


After clicking on the desired item operation, a window similar to the following will be displayed. In this example, a logic 1 value (Boolean data type) is being written to a discrete output to turn it on. The item operations can be used to read discrete/analog inputs and write to discrete/analog outputs, etc.



Using the RJ12 Serial Port in ASCII Mode

The EBC RJ12 serial port can be configured for generic ASCII communications (refer to the "Advanced Settings" section in Chapter 3 to confirm or change the RJ12 serial port settings). Both the transmit buffer and receive buffer of the driver are 127 bytes in size. Thus, the corresponding tags can be a maximum of 127 bytes. Incoming bytes are appended to the receive buffer.
Port specifiers precede the serial port address. It defines which port the serial port address corresponds to. To define an EBC address the mnemonic EBC is used and the mnemonic SP0 specifies serial port 0. For addressing the EBC serial port, no base or slot information is needed.
As shown below in the **Hints** dialog, there are several port address parameters. In many cases the default values can be used. A detailed list explaining the parameters are found by clicking on the **Help** button in the **Hints** window. Then click on the **Index** button in the Terminator I/O, I/O Addressing window. Then locate the "H2, H4, Terminator I/O Serial Port Addressing" help section.

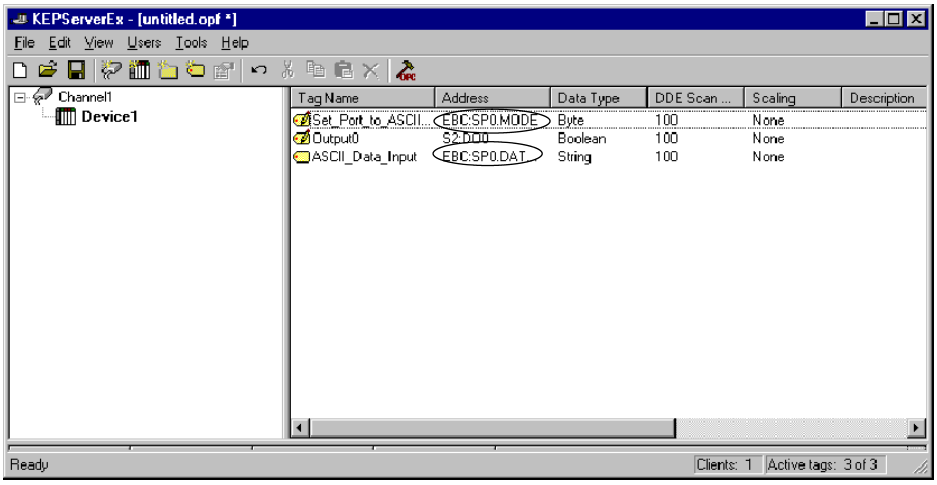


The communication parameter defaults are:

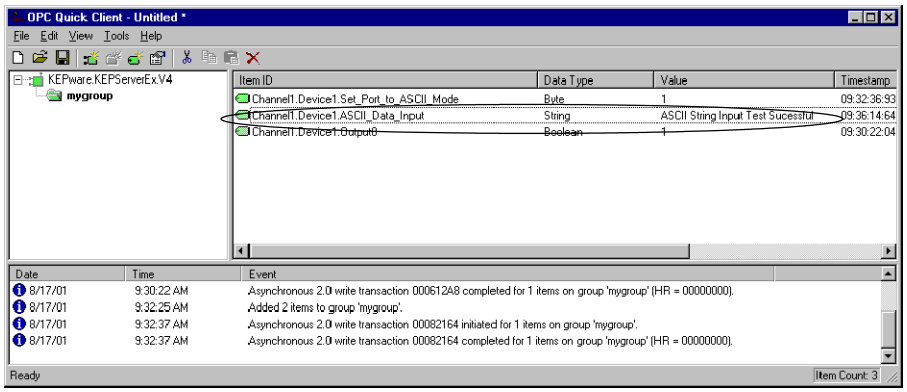
- 9600 baud
- 8 data bits (7 may be selected)
- no parity (odd or even may be selected)
- 1 stop bits (2 may be selected)

The following tags were created in the KEP***Direct*** OPC server for this example.

- EBC:SP0:MODE
- EBC:SP0:DATAIN



The tags created above were browsed and selected as items within the Quick Client as shown below. The **EBC.SP0.MODE** address must be set to a value of 1 to select the ASCII communications mode. The ASCII string **ASCII String Input Test Successful** was entered via the RJ12 serial port. The ASCII Sting displays in the **ASCII_Data_Input** Item ID's Value column.

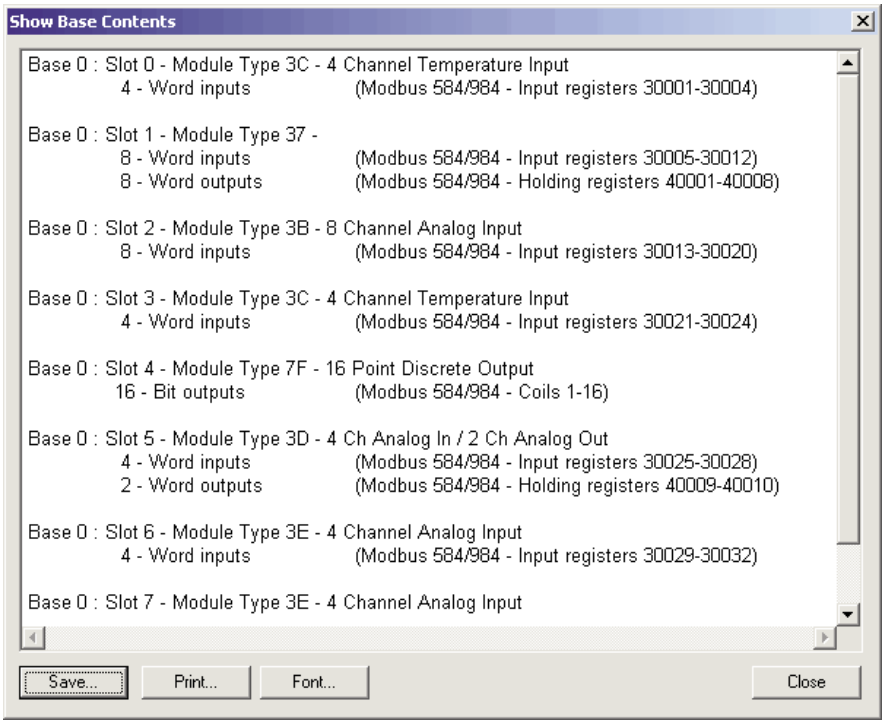


H2-EBC(100) Analog Module Addressing

In This Appendix...

H2-EBC(100) Analog Module Addressing - Modbus TCP

Using the NetEdit3 utility, find and select the IP address of the desired H2-EBC(100). Then select the 'EBC Settings' tab and the 'Show Base Contents' button to see the I/O modules in the H2-EBC(100) base and the Modbus addressing for those modules. You should see something similar to the following:



Use the addresses shown in the 'Show Base Contents' section of NetEdit3 along with the following table to access the analog I/O with your Modbus TCP master.

For example, to configure the range of the F2-8AD4DA-2 module located in Slot 1 of the system, the data provided above along with the table below would show that Modbus address 40006 is required. Also, to read the current temperature detected by Channel 3 of the RTD module in Slot 3, Modbus address 30023 is required.

H2-EBC(100) Analog Module Addressing - Modbus TCP

Part Number	Channel Data	Module Configuration Data	Diagnostics Data
F2-04AD-1 (L) F2-04AD-2 (L)	Input Registers Word 1 = Ch1 Word 2 = Ch2 Word 3 = Ch3 Word 4 = Ch4	None	No Broken Transmitter Detection If No 24VDC or No Terminal Block: • All channels = 0 counts • 'Error Code' =121d* • 'Info Code' = 0*

*** See H2-EBC(100) System Memory Section in Chapter 4 for further details.**

H2-EBC(100) Analog Module Addressing - Modbus TCP			
Part Number	**Channel Data**	**Module Configuration Data**	**Diagnostics Data**
F2-08AD-1	Input Registers Word 1 = Ch1 Word 2 = Ch2 Word 3 = Ch3 Word 4 = Ch4 Word 5 = Ch5 Word 6 = Ch6 Word 7 = Ch7 Word 8 = Ch8	None	Channels with broken transmitter: • Channel=0 counts • 'Error Code' =121d* • 'Info Code' = High Byte - Bit On for Each Failed Channel* If No 24VDC or No Terminal Block: • All channels = 0 counts • 'Error Code' =121d* • 'Info Code' = Cycles 0100h thru 0700h*
F2-08AD-2	Input Registers Word 1 = Ch1 Word 2 = Ch2 Word 3 = Ch3 Word 4 = Ch4 Word 5 = Ch5 Word 6 = Ch6 Word 7 = Ch7 Word 8 = Ch8	None	No Broken Transmitter Detection If No 24VDC or No Terminal Block: • All channels = 0 counts • 'Error Code' =121d* • 'Info Code' = 0Cycles 0100h thru 0700h*
F2-4AD2DA	Input Registers Word 1 = Ch1 Word 2 = Ch2 Word 3 = Ch3 Word 4 = Ch4 Holding Registers Word 1 = Ch1 Word 2 = Ch2	None	No Broken Transmitter Detection If No 24VDC or No Terminal Block: • All channels = 0 counts • 'Error Code' =121d* • 'Info Code' = Cycles 0100h thru 0400h*
F2-8AD4DA-1	Input Registers Word 1 = Ch1 Word 2 = Ch2 Word 3 = Ch3 Word 4 = Ch4 Word 5 = Ch5 Word 6 = Ch6 Word 7 = Ch7 Word 8 = Ch8 Holding Registers Word 1 = Ch1 Word 2 = Ch2 Word 3 = Ch3 Word 4 = Ch4	Holding Registers Word 5 = Input Resolution Word 6 = N/A Word 7 = Track and Hold Word 8 = Not Used	Channels with broken transmitter: • Channel=0 counts • 'Error Code' =121d or 142d* • 'Info Code' = High Byte - Bit On for Each Failed Channel* If No 24VDC or No Terminal Block: • All channels = 0 counts • 'Error Code' =121d* • 'Info Code' = 0FF00h*
F2-8AD4DA-2	Input Registers Word 1 = Ch1 Word 2 = Ch2 Word 3 = Ch3 Word 4 = Ch4 Word 5 = Ch5 Word 6 = Ch6 Word 7 = Ch7 Word 8 = Ch8 Holding Registers Word 1 = Ch1 Word 2 = Ch2 Word 3 = Ch3 Word 4 = Ch4	Holding Registers Word 5 = Input Resolution Word 6 = Range Selection Word 7 = Track and Hold Word 8 = Not Used	No Broken Transmitter Detection If No 24VDC or No Terminal Block: • All channels = 0 counts • 'Error Code' =121d* • 'Info Code' = 0FF00h*

*** See H2-EBC(100) System Memory Section in Chapter 4 for further details.**

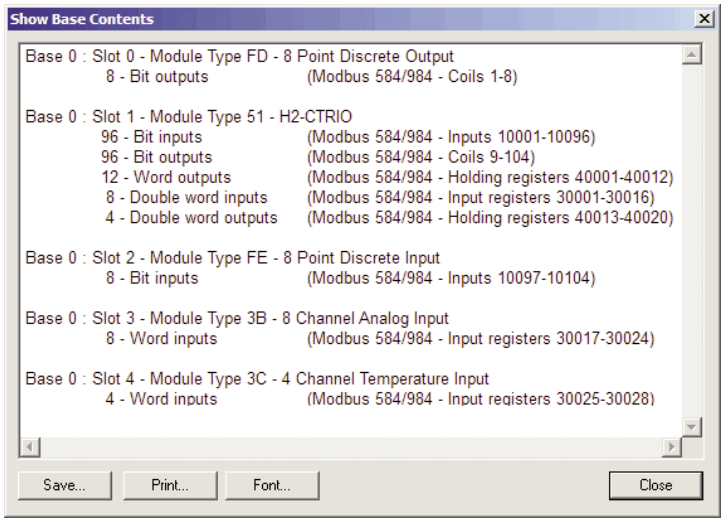
H2-EBC(100) Analog Module Addressing - Modbus TCP			
Part Number	**Channel Data**	**Module Configuration Data**	**Diagnostics Data**
F2-04THM F2-04RTD	Input Registers Word 1 = Ch1 Word 2 = Ch2 Word 3 = Ch3 Word 4 = Ch4	None	Channels with broken transmitter: • Channel=0 counts • 'Error Code' = 142d* • 'Info Code' = High Byte - Bit On for Each Failed Channel* If No 24VDC or No Terminal Block: • All channels = 0 counts • 'Error Code' =121d* • 'Info Code' = 0F00h*
F2-02DA-1(L) F2-02DA-2(L)	Holding Registers Word 1 = Ch1 Word 2 = Ch2	None	None
F2-02DAS-1 F2-02DAS-2	Holding Registers Word 1 = Ch1 Word 2 = Ch2	None	None
F2-08DA-1 F2-08DA-2	Holding Registers Word 1 = Ch1 Word 2 = Ch2 Word 3 = Ch3 Word 4 = Ch4 Word 5 = Ch5 Word 6 = Ch6 Word 7 = Ch7 Word 8 = Ch8	None	None

*** See H2-EBC(100) System Memory Section in Chapter 4 for further details.**

F

F2-04RTD Example (Module in Slot 4)

Using the 'Show Base Contents' dialog below and the 'H2-EBC(100) Analog Module Addressing - Modbus TCP' chart above, we can find the addresses that contain channel data for the F2-04RTD module in Slot 4. Using the I/O Module Status entry in the System Memory table in Chapter 4 of this manual, we can also find the location of the error words. Since there are 20 status words per slot, the first status word for slot 4 will be stored in register 37481.

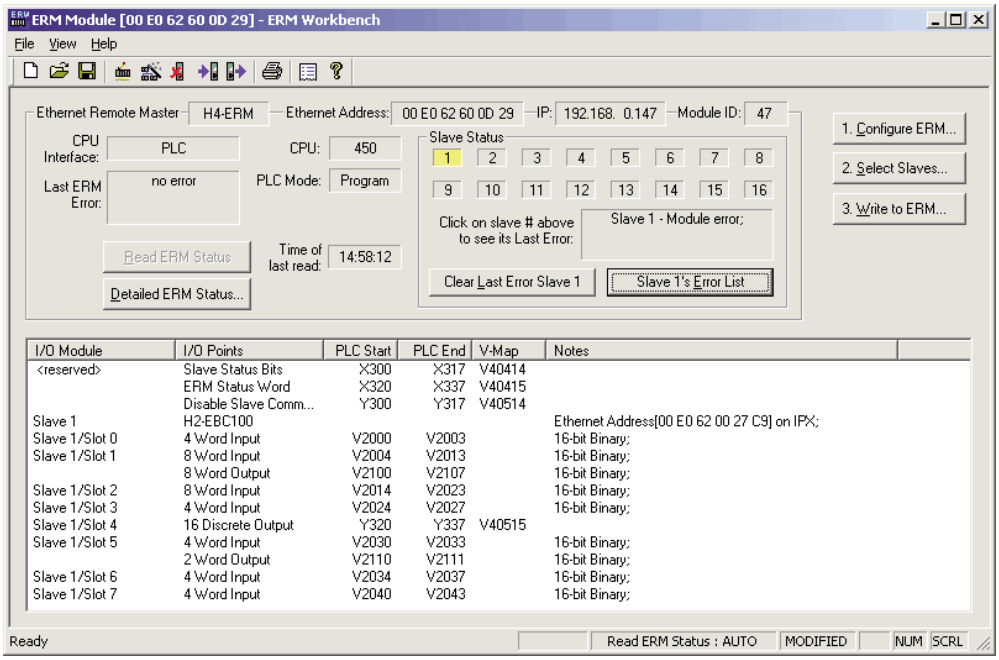


Input Channel	Address	Error Words
Channel 1	30025	• 37481 is the Flags Word, bitmapped error indicators • 37482 is the Error Code Word • 37483 is the Warning Code Word • 37484 is the Info Code Word, with the low byte indicating error codes and the high byte having a bit ON for each failed channel • 37485 -374820 are reserved
Channel 2	30026	
Channel 3	30027	
Channel 4	30028	

H2-EBC(100) Analog Module Addressing - H2/4-ERM(100)

When using an ERM to EBC configuration, most analog module data in the H2-EBC(100) base is mapped to V-memory or Discrete I/O. Certain Diagnostics Data is not automatically mapped. If needed, the Diagnostics Data can be accessed as described in H24-ERM-M Appendix B.

The ERM Workbench software will tell you what the mapping is for each I/O module in the H2-EBC(100) base. Once you have configured the H2-ERM(100) or H4-ERM(100) using ERM Workbench you will get a screen similar to the following:



For the example above, the I/O configuration for Slave 1 is:

Slot 0 = F2-04THM
Slot 1 = F2-8AD4DA-1
Slot 2 = F2-08AD-1
Slot 3 = F2-04RTD
Slot 4 = D2-12TR
Slot 5 = F2-4AD2DA
Slot 6 = F2-04AD-2
Slot 7 = F2-04AD-1

Use the addresses shown in the ERM Workbench along with the following table to access the analog I/O with your ERM master.

For example, to configure the input resolution of the F2-8AD4DA-1 module located in Slot 1 of the system, the data provided above along with the table below would show that V memory location V2104 is required. Also, to read the current temperature detected by Channel 3 of the RTD module in Slot 3, V memory location V2026 is required.

H2-EBC(100) Analog Module Addressing - H2-ERM(100)			
Part Number	**Channel Data**	**Module Configuration Data**	**Diagnostics Data***
F2-04AD-1 (L) F2-04AD-2 (L)	Input Words: Word 1 = Ch1 Word 2 = Ch2 Word 3 = Ch3 Word 4 = Ch4	None	No Broken Transmitter Detection If No 24VDC or No Terminal Block: • All channels = 0 counts • 'Error' = 121** • 'Other' = Cycles 1 thru 3***
F2-08AD-1	Input Words: Word 1 = Ch1 Word 2 = Ch2 Word 3 = Ch3 Word 4 = Ch4 Word 5 = Ch5 Word 6 = Ch6 Word 7 = Ch7 Word 8 = Ch8	None	Channels with broken transmitter: • Channel=0 counts • 'Error' =121** • 'I/O Module Status Word 1' = Channel Number** If No 24VDC or No Terminal Block: • All channels = 0 counts • 'Error' =121** • 'Other' = Cycles 1 thru 7***
F2-08AD-2	Input Words: Word 1 = Ch1 Word 2 = Ch2 Word 3 = Ch3 Word 4 = Ch4 Word 5 = Ch5 Word 6 = Ch6 Word 7 = Ch7 Word 8 = Ch8	None	No Broken Transmitter Detection If No 24VDC or No Terminal Block: • All channels = 0 counts • 'Error' =121** • 'Other' = Cycles 1 thru 7***
F2-4AD2DA	Input Words: Word 1 = Ch1 Word 2 = Ch2 Word 3 = Ch3 Word 4 = Ch4 Output Words: Word 1 = Ch1 Word 2 = Ch2	None	No Broken Transmitter Detection If No 24VDC or No Terminal Block: • All channels = 0 counts • 'Error' =121** • 'Other' = Cycles 1 thru 3***
F2-8AD4DA-1	Input Words: Word 1 = Ch1 Word 2 = Ch2 Word 3 = Ch3 Word 4 = Ch4 Word 5 = Ch5 Word 6 = Ch6 Word 7 = Ch7 Word 8 = Ch8 Output Words: Word 1 = Ch1 Word 2 = Ch2 Word 3 = Ch3 Word 4 = Ch4	Output Words: Word 5 = Input Resolution Word 6 = N/A Word 7 = Track and Hold Word 8 = Not Used	Channels with broken transmitter: • Channel=0 counts • 'Error' = 142** • 'I/O Module Status Word 1' = Bit On for Each Channel with Broken Transmitter** If No 24VDC or No Terminal Block: • All channels = 0 counts • 'Error' =142** • 'Other' = 0xFF***

*** Diagnostics Data is not automatically mapped. If needed, the Diagnostics Data can be accessed via ERM Workbench or ladder as described in H24-ERM-M Appendix B.**

**** See H2-EBC(100) System Memory Section in Chapter 4 for further details.**

***** 'Other' is a field accessible only in ERM Workbench by clicking the button: Slave N's Error List. This field cannot be read programmatically.**

H2-EBC(100) Analog Module Addressing - H2-ERM(100)			
Part Number	**Channel Data**	**Module Configuration Data**	**Diagnostics Data***
F2-8AD4DA-2	Input Words: Word 1 = Ch1 Word 2 = Ch2 Word 3 = Ch3 Word 4 = Ch4 Word 5 = Ch5 Word 6 = Ch6 Word 7 = Ch7 Word 8 = Ch8 Output Words: Word 1 = Ch1 Word 2 = Ch2 Word 3 = Ch3 Word 4 = Ch4	Output Words: Word 5 = Input Resolution Word 6 = Range Selection Word 7 = Track and Hold Word 8 = Not Used	No Broken Transmitter Detection If No 24VDC or No Terminal Block: • All channels = 0 counts • 'Error' =121** • 'Other' = 0xFF***
F2-04THM F2-04RTD	Input Words: Word 1 = Ch1 Word 2 = Ch2 Word 3 = Ch3 Word 4 = Ch4	None	Channels with broken transmitter: • Channel=0 counts • 'Error' = 142d** • 'I/O Module Status Word 1' = Bit On for Each Channel with Broken Transmitter** If No 24VDC or No Terminal Block: • All channels = 0 counts • 'Error' =121** • 'Other' = 0x0F***
F2-02DA-1(L) F2-02DA-2(L)	Output Words: Word 1 = Ch1 Word 2 = Ch2	None	None
F2-02DAS-1 F2-02DAS-2	Output Words: Word 1 = Ch1 Word 2 = Ch2	None	None
F2-08DA-1 F2-08DA-2	Output Words: Word 1 = Ch1 Word 2 = Ch2 Word 3 = Ch3 Word 4 = Ch4 Word 5 = Ch5 Word 6 = Ch6 Word 7 = Ch7 Word 8 = Ch8	None	None

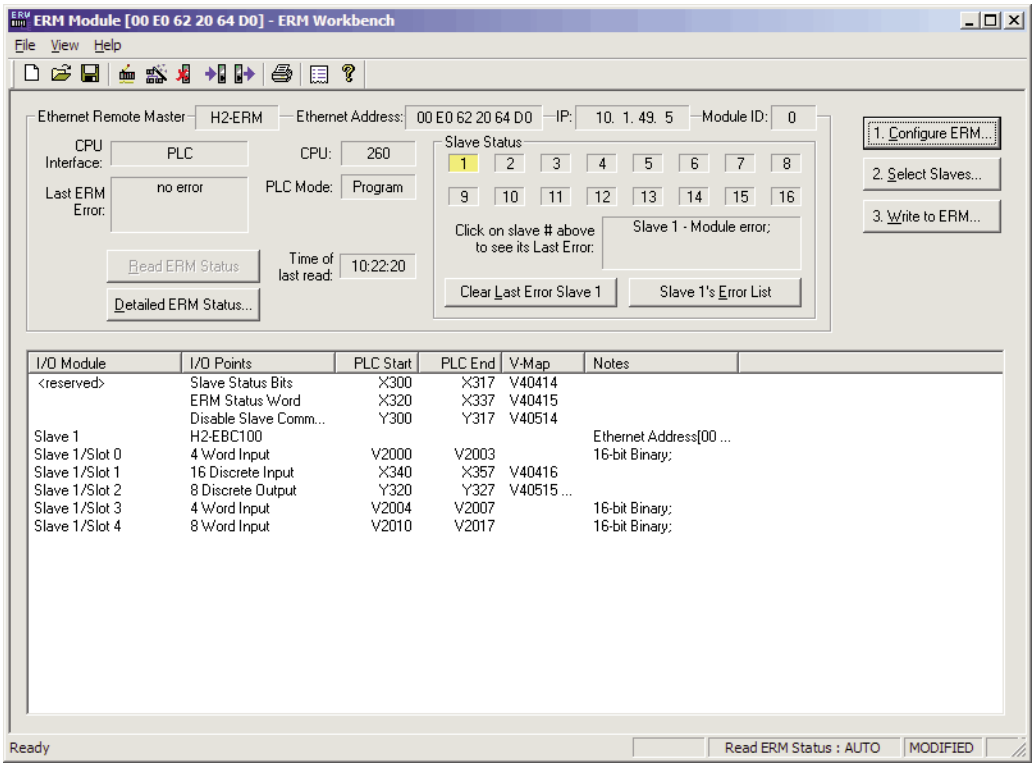
*** Diagnostics Data is not automatically mapped. If needed, the Diagnostics Data can be accessed via ERM Workbench or ladder as described in H24-ERM-M Appendix B.**

**** See H2-EBC(100) System Memory Section in Chapter 4 for further details.**

***** 'Other' is a field accessible only in ERM Workbench by clicking the button: Slave N's Error List. This field cannot be read programmatically.**

F2-04RTD Example (Module in Slot 3)

Using the 'ERM Workbench' dialog below and the 'H2-EBC(100) Analog Module Addressing - H2-ERM(100)' chart above, we can find the addresses that contain channel data for the F2-04RTD module in Slot 3. Diagnostics Data is not automatically mapped. If needed, the Diagnostics Data can be accessed via ERM Workbench or ladder as described in H24-ERM-M Appendix B.



Input Channel	Address	Error Words
Channel 1	V2004	The Slave Diagnostic Word for slot 3 would be mapped to V + 7 (the eighth address in your chosen V-memory range). The H24-ERM-M manual Appendix B contains an example.
Channel 2	V2005	
Channel 3	V2006	
Channel 4	V2007	