

Introduction

The purpose of this supplementary Manual

This KEP*Direct* Setup manual provides enough of the basics to install the EBC* server* software without having to read the complete online help documentation that covers all the details of KEP*Direct*. Exercise caution: This is not intended to replace reading the online help documentation. This is intended only as a supplement. This is only a quick start guide.

Who can and should use KEP*Direct*?

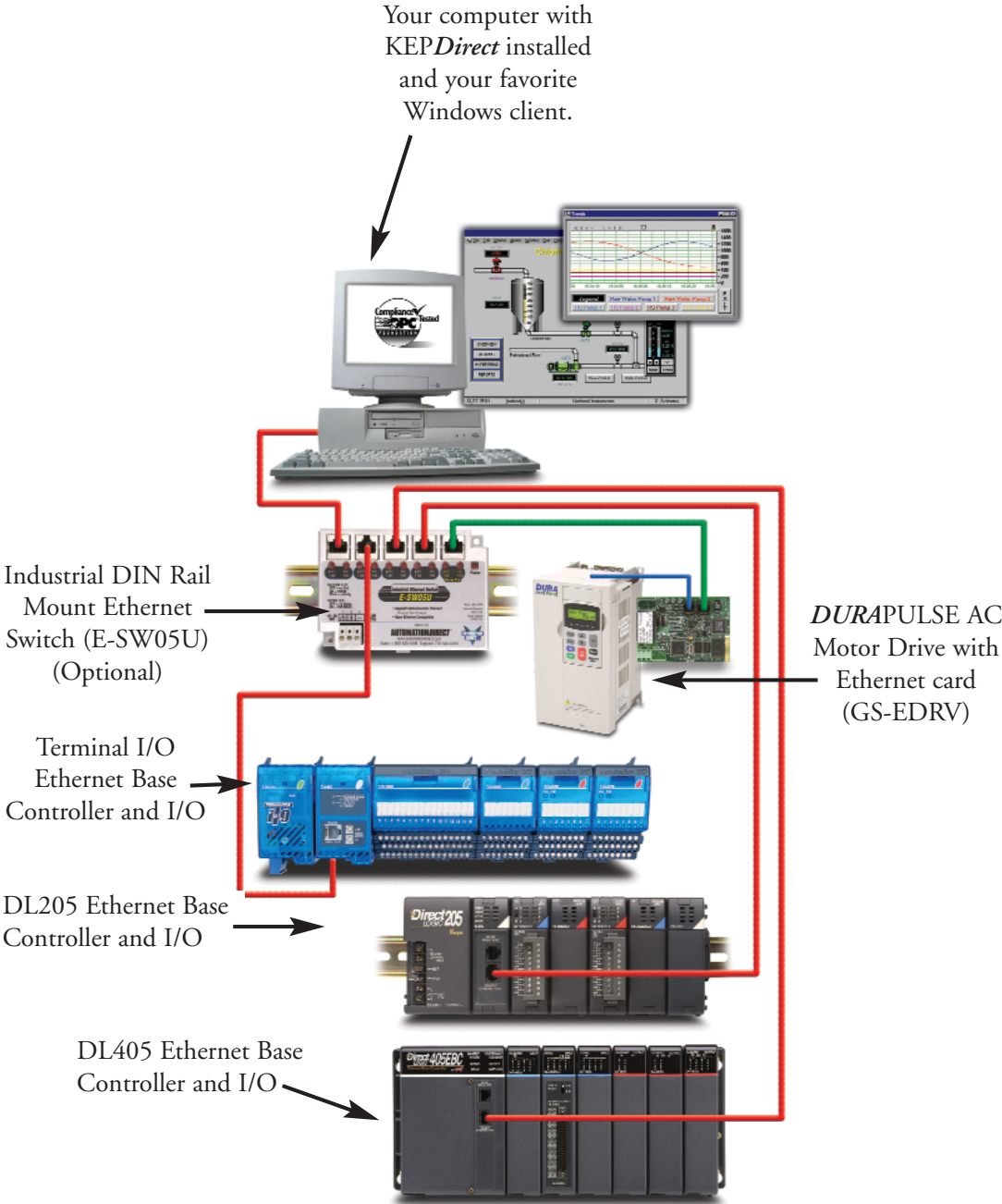
If you have an EBC belonging to the *Direct*LOGIC EBC family, you can use KEP*Direct* to connect your EBCs with your favorite Windows client. The families of EBCs are H2-EBC (-F), H4-EBC (-F), and T1H-EBC. KEP*Direct* can also be used with GS series drives. KEP*Direct* has added the following functionality:

- Connect your favorite HMI/SCADA* software to lowcost **AUTOMATIONDIRECT** EBC I/O systems over a 10Mbit Ethernet network.
- Design a low-cost data acquisition/monitoring system using affordable **AUTOMATIONDIRECT** DL205, DL405, or Terminator I/O. Combine any ASCII* serial devices to the on-board EBC serial port for interfacing intelligent devices like bar code readers, weigh scales, labelers, and more.
- Interface custom VisualBASIC or Visual C++ applications to **AUTOMATIONDIRECT** EBC I/O systems and eliminate time-consuming driver development.

**See Glossary*

KEP*Direct* EBC I/O Server

Diagram Showing the Basic System Compatibility



Preparing for Installation

Getting to Know Windows

KEP*Direct* software runs under 32-bit Windows operating systems (98/2000/NT/XP). If you are more accustomed to using 3.1, please take a moment to study your PC's reference manual on the operation of Windows (95/98/2000/NT/XP).

Check your PC Hardware Requirements

Please check the following requirements when choosing your PC configuration.

Minimum System Requirements

- Windows 98/2000/NT/XP
- Pentium 200MHz
- 32Mb RAM
- 10Mb available disk space
- MS Internet Explorer 5.5 or better
- Available Serial Port or Ethernet Card



Recommended System

- Windows NT (SP6a), Windows 2000 (strongly recommended for industrial settings), XP Pro
- Pentium 400 MHz or better
- 64 Mb RAM
- 10Mb hard disk space
- MS Internet Explorer 5.5 or better



NOTE: The server is designed to run on Windows 98, NT 4.0, and all versions of Windows 2000 and XP. On Win 95 and some older Win 98 PC's you will probably have to load the Microsoft DCOM* patch since DCOM is not native to them.

*See Glossary

KEP*Direct* EBC I/O Server

Power Supply

It is highly recommended that the computer KEP*Direct* operates on has some form of power surge protection. A quality surge protector will protect your computer from most surges and spikes; however, an uninterruptible power supply (UPS) will provide the best protection. A UPS provides isolation between the AC power source and the computer and has battery backup for blackout and brownout conditions.

Now is the time to review the contents of your KEP*Direct* software package. You should have the following items:

- Software Product Showcase CD
- Installation and Setup Manual
- Installation Key Code and Registration Key Code

Supported Devices

One of the benefits with the *Direct*Logic family is the wide variety of I/O Module types. Below is a list of supported devices:

- **H2-EBC:** Ethernet Base Controller for DL205 Base. Supports Discrete Analog I/O H2-CTRIO and H2-SERIO triple port serial module. Supports 10Base-T connections to PC-based controllers supporting UDP/IP protocol.
- **H2-EBC-F:** Ethernet Base Controller for DL205 Base. Supports Discrete Analog I/O H2-CTRIO and H2-SERIO triple port serial module. Supports 10Base-T connections to PC-based controllers supporting UDP/IP protocol.
- **H4-EBC:** Ethernet Base Controller for DL405 Base. Supports Discrete and Analog I/O and H4-CTRIO. Supports 10Base-T connections to PC-based controllers supporting UDP/IP protocol.
- **H4-EBC-F:** Ethernet Base Controller for DL405 Base. Supports Discrete and Analog I/O and H4-CTRIO. Supports Fiber Optic connections to PCbased controllers supporting UDP/IP protocol.
- **T1H-EBC/EBC100:** Ethernet Base Controller for Terminator I/O base and T1H-CTRIO. Supports Fiber Optic connections to PC-based controllers supporting UDP/IP protocol.
- **GS-EDRV:** provides a high-performance Ethernet link for the DURAPULSE and GS series drives.

KEP*Direct* EBC I/O Server

Supported Device Modules

Device	Module	Description	
H2-EBC and H2-EBC-F	D2-08ND3	8 Pt 12-21 VDC Input	
	D2-16ND3-2	16 Pt 24 VDC Input	
	D2-32ND3	32 Pt 24 VDC Input	
	D2-32ND3-2	32 Pt 5-15 VDC Input	
	D2-08NA-1	8 Pt 110 VAC Input	
	D2-08NA-2	8 Pt 170-265 VAC Input, 2 Commons	
	D2-16NA	16 Pt 110 VAC Input	
	F2-08SIM	8 Pt Switch Slide Simulator	
	D2-04TD1	4 Pt 12-24 VDC Sink Output	
	D2-08TD1	8 Pt 12-24 VDC Sink Output	
	D2-16TD1-2	16 Pt 12-24 VDC Sink (2 Terminals)	
	D2-08TD2	8 Pt 12-24 VDC Source Output	
	D2-32TD2	32 Pt 24 VDC Source output	
	D2-16TD2-2	16 Pt 12-24 VDC Out (2 Terminals)	
	D2-32TD1	32 Pt 24 VDC Output	
	D2-08TA	8 Pt 18-220 VAC Output	
	D2-12TA	12 Pt 18-110 VAC Output	
	D2-04TRS		4 Pt Isolated Relay
			5-30 VDC or 5-250 VAC Output
	D2-08TR	8 Pt Relay Output 12-28 VDC / 12-250 VAC	
	F2-08TA	8 Pt 24-140 VAC Output, 1.5A/Pt.	
	F2-08TR		8 Pt Relay Output, 10A/Common,
			12-28 VDC / 12-250 VAC
	F2-08TRS		8 Pt Relay Output
			12-28 VDC or 12-250 VAC
	D2-12TR	12 Pt Relay Output 5-30 VDC or 5-250 VAC	
	D2-08CDR		Combo 4 Pt 24 VDC Input and
			4 Pt Relay Output
	F2-04AD-1	4 CH Analog In 4-20mA 12 Bit Res	
	F2-04AD-1L		4 Pt Analog Input 4-20mA 12 Bit
			12 VDC Power Needed
	F2-04AD-2	4 CH Analog Input Volt 12 Bit Res	
	F2-04AD-2L		4 Pt Analog Input Volt 12 Bit
			12 VDC Power Needed
	F2-08AD-1	8 CH Analog Input 4-20mA 12 Bit	
	F2-08AD-2	8 CH Analog Input Voltage 12 Bit	
	F2-04RTD	4 CH RTD, 0.1 Deg C Resolution	
	F2-04THM	4 CH Thermocouple or 16 bit V Input	
	F2-02DA-1	2 CH Analog Output 4-20mA 12 Bit	
	F2-02DA-2	2 CH Analog Output Voltage 12 Bit	
	F2-02DA-1L		2 CH Analog Output 12 Bit
			Use with 12 VDC Power Supply
	F2-02DA-2L		2 CH Analog Output 12 Bit
			Use with 12 VDC Power Supply
	F2-08DA-1	8 CH Analog Output 4-20mA 12 Bit	
	F2-08DA-2		8 CH 0-5 VDC or 0-10 VDC
			12 Bit Analog Output
F2-02DAS-1	2 CH Isolated 4-20mA 16 Bit Analog Output		
F2-02DAS-2	2 CH Isolated 4-20mA 16 Bit Analog Output		
F2-4AD2DA	4 CH In & 2 CH Out Analog 4-20mA 12 Bit		
H2-SERIO	3-Port RS232 Serial Module		
H2-CTRIO	High Speed Counter/Pulse Output Module		

Supported Device Modules (Continued)

Device	Module	Description
H4-EBC and H4-EBC-F	D4-08ND3S	8 Pt 24-48 VDC Source Input
	D4-16ND2	16 Pt 12-24 VDC Source Input
	D4-16ND2F	16 Pt 12-24 VDC In Fast Response
	D4-16SIM	16 Pt Switch Slide Simulator
	D4-32ND3-1	32 Pt 24 VDC Sink Source Input
	D4-32ND3-2	32 Pt 5-12 VDC Sink Source Input
	D4-64ND2	64 Pt 20-28 VDC Source Input
	D4-08NA	8 Pt 110-220 VAC Input
	D4-16NA	16 Pt 110 VAC Input
	D4-16NA-1	16 Pt 220 VAC Input Module
	D4-16NE3	16 Pt 12-24 VAC/VDC Sink Input
	F4-08NE3S	8 Pt 90-150 VAC/DC Sink Iso. Input
	D4-08TD1	8 Pt 12-24 VDC Sink Output
	F4-08TD1S	8 Pt 24-150 VDC SinkSource Isolated Output
	D4-16TD1	16 Pt 5-24 VDC Sink Output
	D4-16TD2	16 Pt 12-24 VDC Source Output
	D4-32TD1	32 Pt 5-24 VDC Sink Output
	D4-32TD1-1	32 Pt 5-24 VDC Sink Output
	D4-32TD2	32 Pt 12-24 VDC Source Output
	D4-64TD1	64 Pt 5-24 VDC Sink Output
	D4-08TA	8 Pt 18-220 VAC Output
	D4-16TA	16 Pt 18-220 VAC Output
	D4-08TR	8 Pt Relay 5-30 VDC or 5-250 VAC Output
	F4-08TRS-1	8 Pt Relay 12-30 VDC or 12-250 VAC Output
	F4-08TRS-2	8 Pt Relay 12-30 VDC or 12-250 VAC Output
	D4-16TR	16 Pt Relay 5-30 VDC or 5-250 VAC Output
	F4-04AD	4 CH Analog Input Voltage/Current
	F4-04ADS	4 CH Iso. Analog In Voltage/Current
	F4-08AD	8 CH Analog Input Voltage/Current
	D4-02DA	2 CH Analog Output Voltage/Current
	F4-04DA	4 CH Analog Output Voltage/Current
	F4-04DA-1	4 CH Analog Output Current
	F4-04DA-2	4 CH Analog Output Voltage
	F4-04DAS-1	4 CH Isolated. 4-20mA 16 Bit Analog Output
	F4-04DAS-2	4 CH Isolated 16 Bit Analog Voltage Output
	F4-08DA-1	8 CH Analog Output Current
	F4-08DA-2	8 CH 0-5 VDC or 0-10 VDC 12 Bit Analog Output
	F4-16DA-1	16 CH Analog Output Current
	F4-16DA-2	16 CH 0-5 VDC or 0-10 VDC 12 Bit Analog Output
	F4-08THM	8 CH Thermo Module For Type (J,E,K,R,S,T,B,N,C)
	F4-08THM-n	8 CH Thermo Module For Type (J,E,K,R,S,T,B,C,P)
	F4-08RTD	8 CH RTD Module
	H4-CTRIO	High Speed Counter/Pulse Output Module

Go to the Automationdirect.com website for information on any device introduced after publication of this manual.

Supported Device Modules (Continued)

Device	Module	Description
T1H-EBC	T1K-08ND3	8 Pt 12-24 VDC Sink Source Input
	T1K-16ND3	16 Pt 12-24 VDC Sink Source Input
	T1K-08NA-1	8 Pt 110 VAC Input
	T1K-16NA-1	16 Pt 110 VAC Input
	T1K-08TD1	8 Pt 12-24 VDC Sink Output
	T1K-08TD2-1	8 Pt 12-24 VDC Source Output
	T1K-16TD1	16 Pt 12-24 VDC Sink Output
	T1K-16TD2-1	16 Pt 12-24 VDC Source Output
	T1K-08TA	8 Pt 110-240 VAC Output
	T1K-16TA	16 Pt 110-240 VAC Output
	T1K-08TAS	8 Pt 110-240 VAC Output Isolated Commons
	T1K-08TR	8 Pt Relay Output 5-30 VDC or 5-240 VAC
	T1K-16TR	16 Pt Relay Output 5-30 VDC or 5-240 VAC
	T1K-08TRS	8 Pt Isolated Relay Output 5-30 VDC or 5-240 VAC
	T1F-08AD-1	8 CH Analog Input 4-20mA 14 Bit Resolution
	T1F-08AD-2	8 CH Analog Input 14 Bit Resolution
	T1F-16AD-1	16 CH Analog Input 4-20mA 14 Bit Resolution
	T1F-16AD-2	16 CH Analog Input Voltage 14 Bit Resolution
	T1F-08DA-1	8 CH Analog Output 4-20mA 12 Bit Resolution
	T1F-08DA-2	8 CH Analog Output Voltage 12 Bit Resolution
	T1F-16DA-1	16 CH Analog Output 4-20mA 12 Bit Resolution
	T1F-16DA-2	16 CH Analog Output Voltage 12 Bit Resolution
	T1F-16RTD	16 CH. RTD
	T1F-14THM	14 CH Thermocouple 16 Bit Resolution
	T1F-8AD4DA-1	Terminator I/O 8 CH Analog Input 4 CH Analog Output Current
	T1F-8AD4DA-2	Terminator I/O 8 CH Analog Input 4 CH Analog Output Voltage
	T1H-CTRIO	High Speed Counter/Pulse Output Module

Go to the Automationdirect.com website for information on any device introduced after publication of this manual.

Installation of KEP*Direct* EBC I/O Server

Step 1: Load the CD

KEP*Direct* EBC I/O Server Software is available on the **AutomationDirect** Product Showcase CD. To install KEP*Direct*, insert the CD into your PC's CD drive. The CD should start automatically and open the installation window shown below.

NOTE:

If the CD does not start automatically, go to the Windows **START** button, select **RUN** and type:

E:\setup.exe

Change the letter **E**, above, to correspond to the letter for the PC's cd drive. Then click **OK** and the program will start.



Step 2: Select Install Purchased Software Option

The **AutomationDirect** Product Showcase window offers all the options available with this CD. To install KEP*Direct*, select the **Install Purchased Software** option indicated on the figure above. This selection opens the **Product Keycode** window shown below.



Enter
Product Keycode
Here

Then Click on
the **OK** Button

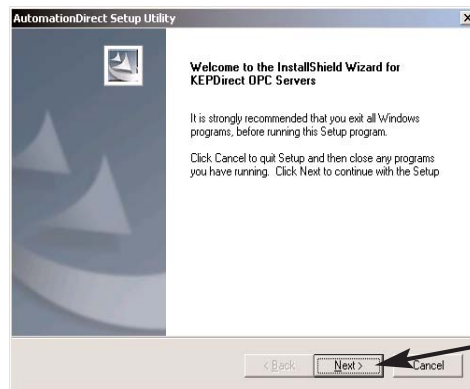
Step 3: Enter the Product Keycode

From this window, enter the **Product Keycode** located on the back cover of your CD case. This software package is protected by this **Product Keycode**. Only licensed users that have a **Product Keycode** may install the software. After entering the keycode click on the **OK** button.

KEP*Direct* EBC I/O Server

Step 4: Welcome Window

The KEP*Direct* installation opens with the **Welcome** window shown below. This window issues a reminder to exit all other Windows applications. If you are unsure of the programs that may be running, open the Task Manager by pressing the Ctrl-ALT-Delete keys at the same time. Close any opened applications by selecting it and clicking on the **Close** button of the Task Manager. Click on the **Next** button on the KEP*Direct* Welcome window to proceed with the installation.

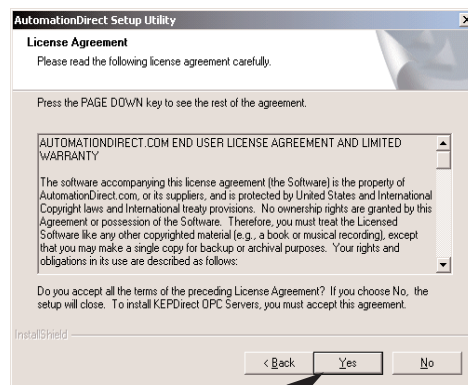


*Click here to proceed with
KEP*Direct* installation*

Step 5: License Agreement

The next screen displays the **Software License Agreement** shown below. Read the agreement and select **Yes** if you agree with the terms and conditions.

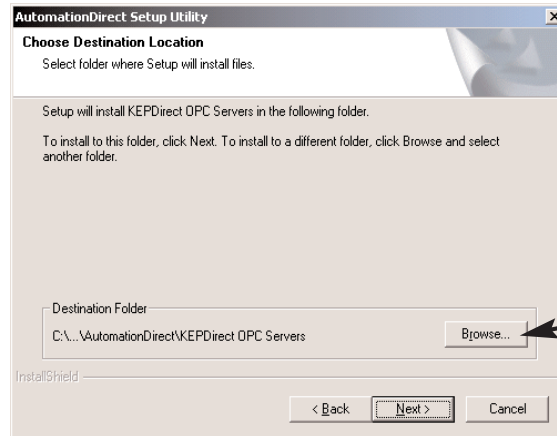
The **Back** button returns you to the Welcome window. Selecting **No** will cancel installation.



*Click here to accept the License
Agreement's terms and conditions*

Step 6: Select the Installation Directory

The **File Destination** window displays the folder (or directory) where the **KEP*Direct*** files will be installed. You may choose a different folder by clicking on the **Browse** button and selecting your preferred directory.



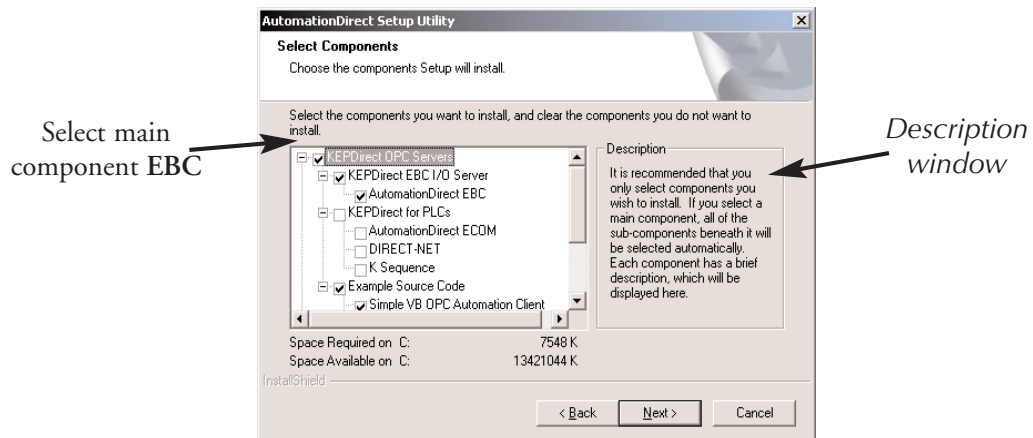
Click here to select a different folder

If you accept the displayed folder or you select your preferred directory, click on the **Next** button to continue with the **KEP*Direct*** installation.

Step 7: Select Components to Install

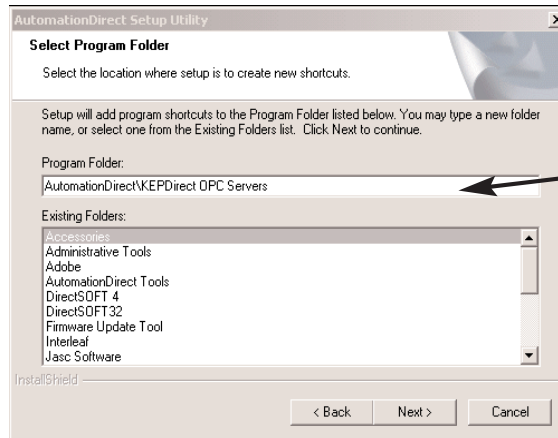
The next window is the **Select Components** window. This window allows you to select what components to install for use with **KEP*Direct***. It is recommended to select only the components you want to install. When a main component is selected, all sub-components are selected automatically. In this case, only the **KEP*Direct*** EBC I/O Server is selected. This window includes a **Description** window, indicated below, which provides a brief description of each component when it is selected.

When all selections are completed, click on the **Next** button to proceed with installation.



Step 8: Select Program Folder

The **Select Program Folder** window will allow you to select or create a folder to install the **KEPDirect** application files. These are the **KEPDirect** launch, help and application files. The installation program automatically creates a new program folder for **KEPDirect**. You may keep this one, type a new name over it or select an existing program folder on the list.



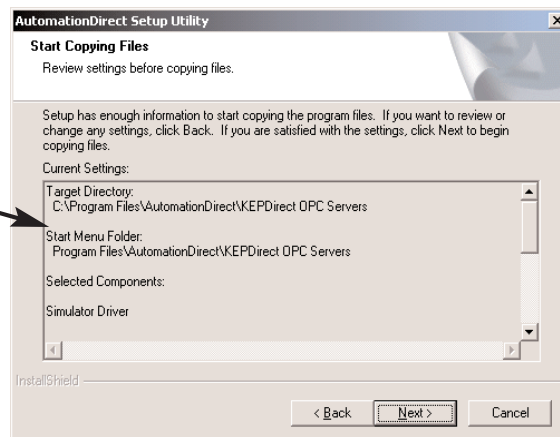
Suggested program folder name. If creating a new name, type over the suggested name.

Once you have chosen or entered a program folder name, click on the **Next** button to continue with the **KEPDirect** installation.

Step 9: Start Copying Files

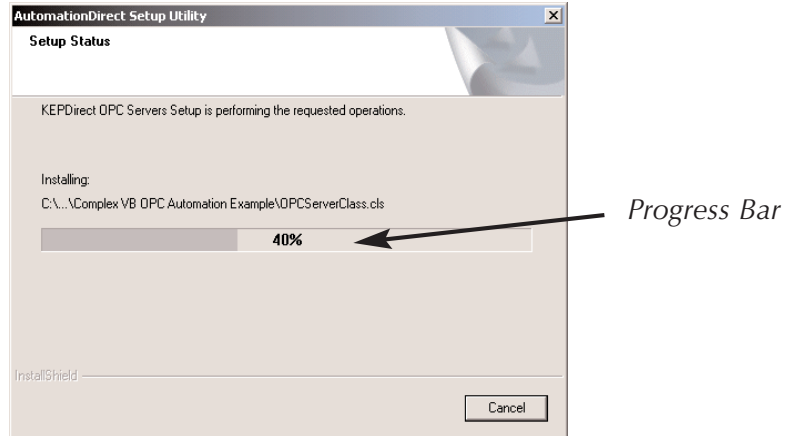
The **Start Copying Files** window, shown below, displays the installation information as configured through all previous steps. Verify that the installation information is correct. If incorrect, click on the **Back** button to return to previous windows. If correct, click on the **Next** button to start copying the **KEPDirect** files to your PC's hard drive.

Displays the installation information configured during the previous installation steps



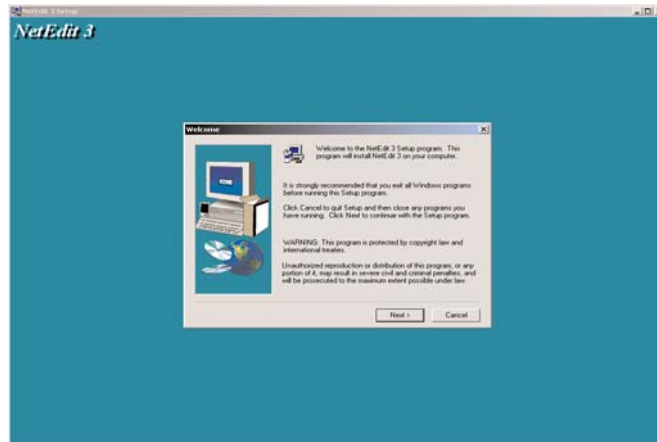
Step 10: KEP*Direct* Files Installation

The installation **Setup Status** window opens and displays the progress of the installation and setup of the KEP*Direct* files. A progress bar is displayed and a number indicating the percentage of the process completed is shown next to the progress bar.



Step 11: Install NetEdit 3

After the setup status bar reaches 100%, the **NetEdit 3** install wizard appears.



If you have not previously installed NetEdit 3 or if you have an earlier version of NetEdit, you can install this new version now. Click on the **Next** button to proceed with the installation.

NetEdit 3 is a tool used to setup **Direct**LOGIC Ethernet devices. NetEdit is the best tool to use to setup a device and to use as a diagnostic and troubleshooting tool. We recommend that you install it at this time.

KEP*Direct* EBC I/O Server

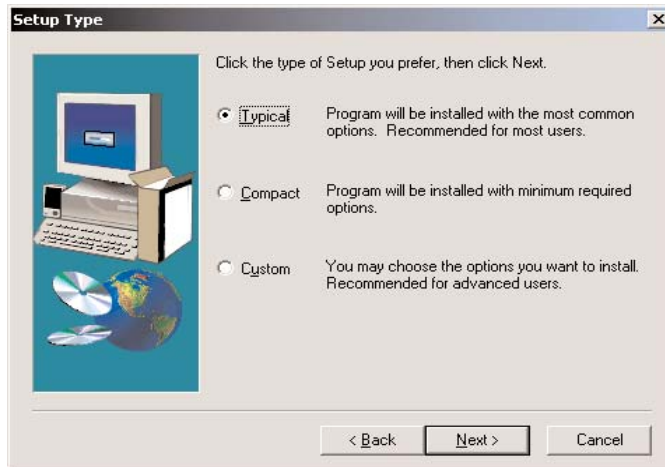
For those who continue with the NetEdit 3 installation you will be asked for an optional name and company name. Type the names and click the **Next** button to continue.



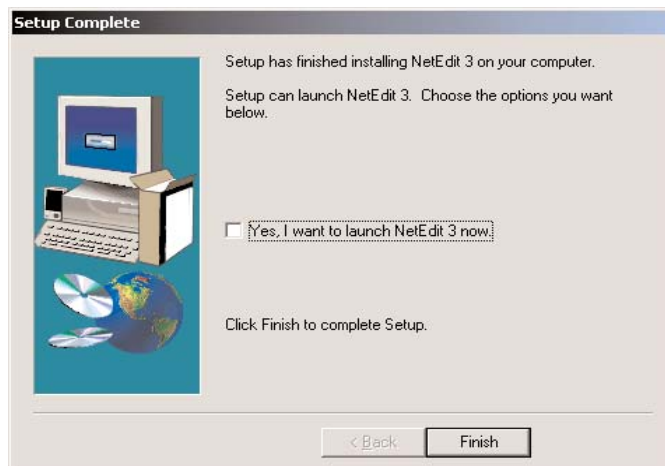
The NetEdit 3 install wizard will ask for the name of the destination folder where NetEdit 3 is to be installed. Either chose a file folder or accept the default folder as shown in the dialog below. Click on the **Next** button to continue.



Select the type of setup in the next dialog window, then click the **Next** button to continue with the installation.



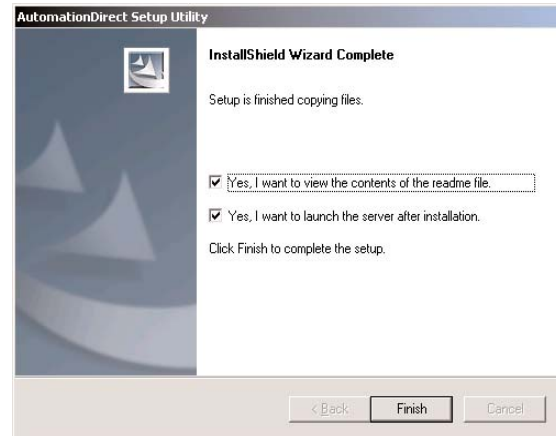
The dialog window below is an indicator that the NetEdit 3 installation is complete



Step 12: Installation Complete

For KEP*Direct*, the install status bar will be showing. When the bar reaches 100% the window closes and the **Installation Complete** window opens. This window is an indication that the KEP*Direct* installation process has finished.

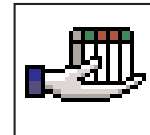
If you wish to cancel either of these options, click on the respective option checkmark to uncheck the option.



This window provides two options that are checked by default. The first option will open the Read Me file which contains the latest information for the KEP*Direct* software. The second option will launch the KEP*Direct* Server Program. Both options will open after clicking on the Finish button. You can either check or uncheck the option selections and, click on the Finish button to end the installation process.

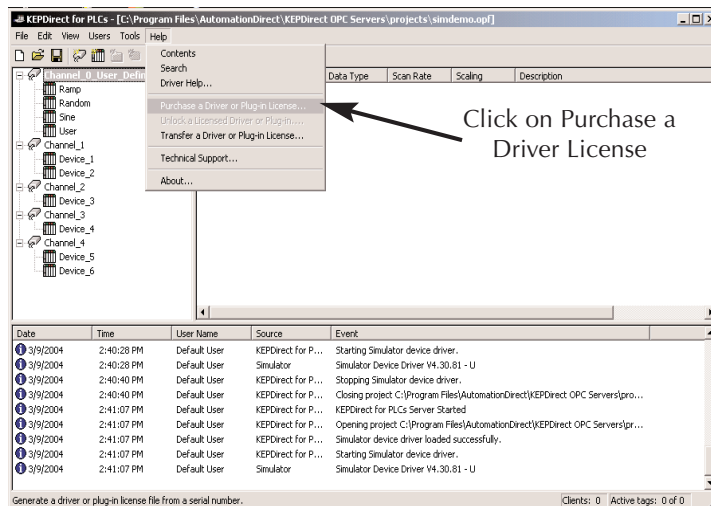
Step 13: Registration

With the installation of KEP*Direct* completed, you may launch the program now by clicking on the KEP*Direct* icon shown on the right.



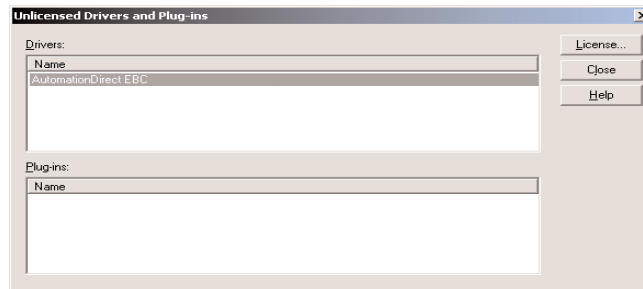
The program opens its main window and offers a set of options on the menu bar located on the top left portion of the window. The menu bar is shown in the figure on the facing page.

KEP*Direct* EBC I/O Server

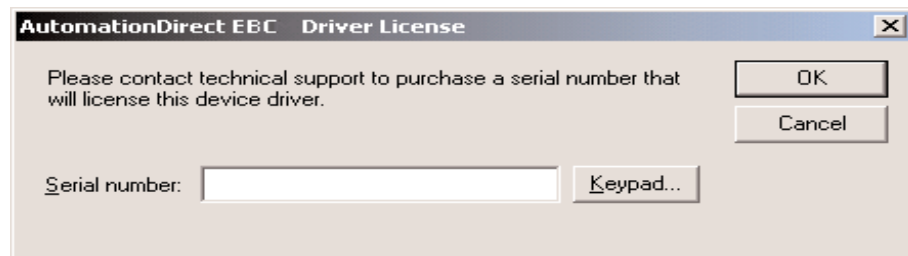


From the menu bar, click on **Help**, then click on **Purchase a Driver License** from the drop down menu shown in the diagram above.

This opens the **Unlicensed Drivers and Plug-ins** window shown below. The only driver shown is the EBC. Click the **License** button to install the driver.



The **EBC Driver License** window, shown below, now opens. Find the label located on the back side of the CD case for the **KEP*Direct*** program. This label shows the license code for the software included with the case. Enter the registration code on the space provided in the window. This code can be entered either from your PC's keyboard or by clicking on **Keypad** and using the pop-up keypad to enter the code. Once the code has been entered, click **OK** to accept the code. The registration is now complete.



KEP*Direct* EBC I/O Server Setup Summary

KEP*Direct* EBC I/O Server is a software driver that provides a means of communication between other software (OPC* clients) and components. Another way of looking at it is that KEP*Direct* is a translator server that provides the communication bridge between the software (OPC* clients) and components.

OPC (OLE* for Process and Control) servers provide a standardized method of allowing multiple industrial applications to share data in a quick and robust manner. The OPC server and LinkMaster* products provided in this package have been designed to meet the demanding requirements found in the industrial environment.

This OPC server has been designed as a two-part program. The primary component provides all of the OPC and DDE* connectivity as well as the user interface functions. The second part is comprised of plug-in communications drivers. This two-part design allows you to add multiple communications options to your SCADA application while utilizing a single OPC server product thus reducing your learning curve as your project grows.

LinkMaster has the capabilities of both a “server” and a “client*” application, allowing it to access, collect, organize, and link data from other OPC servers and offer that data to any OPC/DDE client. It provides the means of linking data between OPC servers; thus, acting as a universal bridge for OPC server/client components.

OPC technology reflects the move from closed proprietary solutions to open architectures that provide more cost-effective solutions based on established standards.

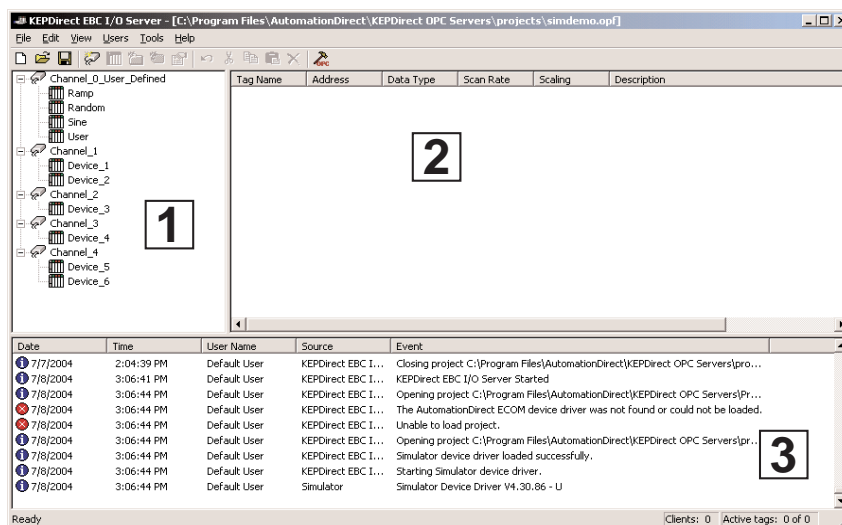
** See Glossary*

Start a New Project

Step 1: Launch **KEP*Direct*** and Get to Know the Work Areas



Click on the **KEP*Direct*** EBC I/O Server icon, shown on the left, to launch the application. The application will start and open the **KEP*Direct*** EBC I/O Server window shown below.



When the **KEP*Direct*** EBC I/O Server application is launched for the first time, it opens with an operational simulation program. This is a sample program which can be used to practice adding devices and channels. The window is divided into three areas as explained below (see window above for reference):

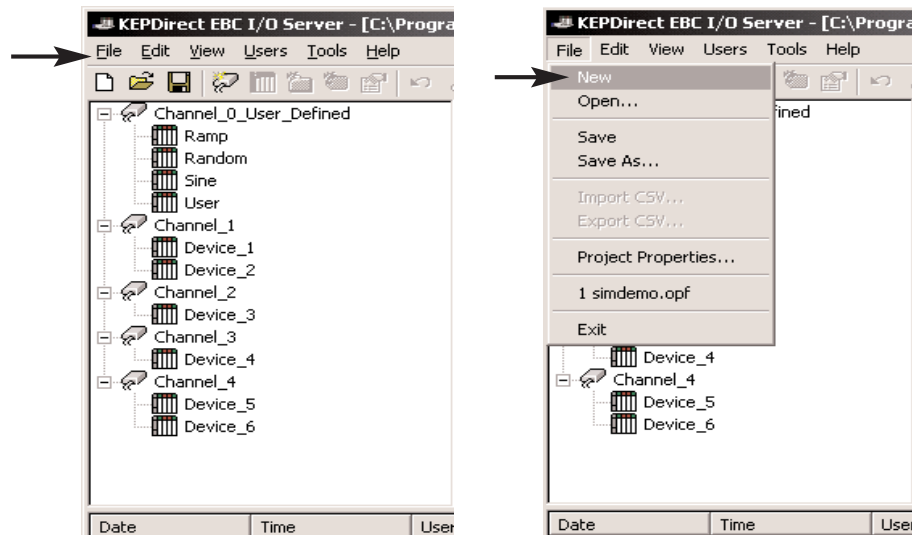
1. This area displays any existing channels, devices, and groups in a project. In addition, from this area new channels, devices, and groups can be added to projects.
2. This is the tag entry window of the server. Tags that you enter for a given device or tag group will be displayed here.
3. This area is the event log window of the server. Any of the messages generated by the server or the underlying driver will be displayed here.

Step 2: Things to confirm before Adding and Configuring a Channel

Before adding a new channel to your new project, first confirm that the following items have been completed or are available:

1. KEP*Direct* EBC I/O Server has been installed and launched properly.
2. Any devices to be added to your project must be installed with modules in place, powered up, and working properly.
3. You must have the correct IP addresses for each of the devices to be used for your project. All the EBC manuals provide instructions on how to setup IP addresses for each device using NetEdit 3. If you have not done so yet, please refer to the EBC manuals and setup the IP addresses for your devices. Make sure to write down the IP address for each device. The IP addresses will be required during the project setup in order to ensure successful communication between the KEP*Direct* EBC I/O Server and your devices.
4. If you will be using more than one device to setup your project, an industrial Ethernet Switch such as the E-SW05U from **AUTOMATIONDIRECT** is strongly recommended.

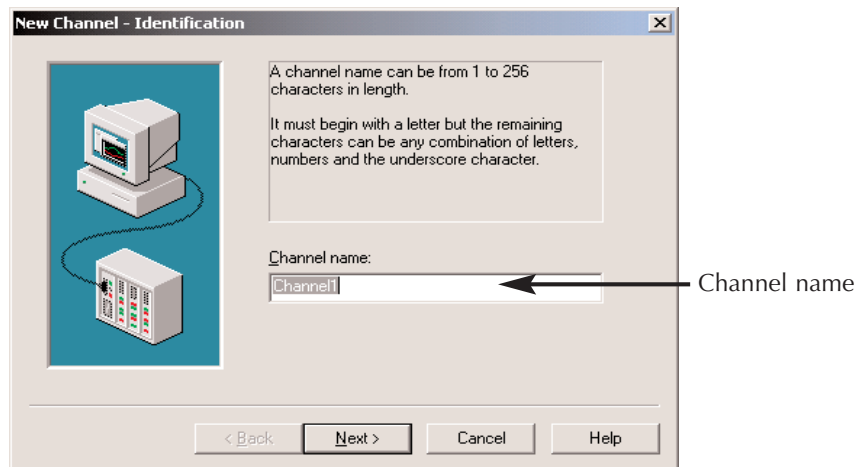
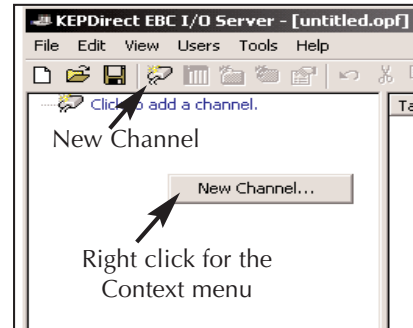
Once all the above items are confirmed and ready, either click on **File** on the menu bar or on the **New Project** icon on the tool bar then on the drop down window click **New** to close the simulator and start your new project.



Step 3: Adding and Configuring a Channel

A channel refers to a specific communications driver. A **KEP*Direct*** project only consists of EBC channels using the same communications driver. You can define a number of channels within a single project for organizational purposes. A channel acts as the basic building block of an OPC link. Each channel name must be unique in a **KEP*Direct*** project. The channel name can be up to 256 alphanumeric characters in length (no spaces or special characters).

To add a new channel to your project, click on either the **Click to add a channel** text in the EBC I/O Server window, **Edit > New Channel**, the toolbar **New channel** or the context menu. As soon as you click on either of the above, the **New Channel** window shown below will open.



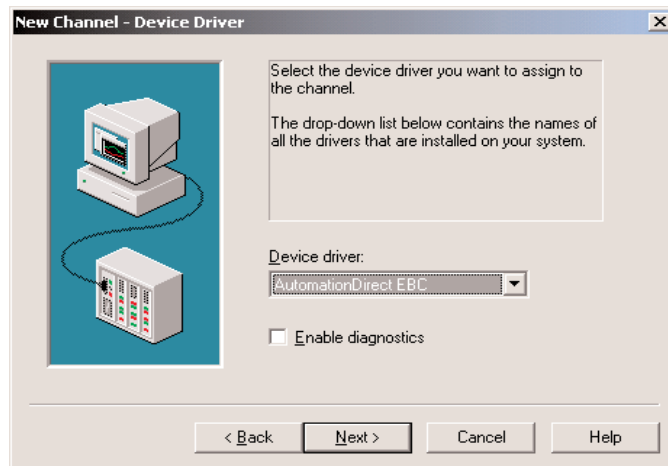
Step 4: Name the Channel

Each channel name must be unique in a **KEP*Direct*** project. Each unique channel name can be up to 256 characters long. 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 channel name entered here will be part of the OPC browser information.

The **New Channel** window prompts you to type a unique name for the channel by typing over the default **Channel1** label. Once the name has been typed, click on the **Next** button.

Step 5: Select the Device Driver

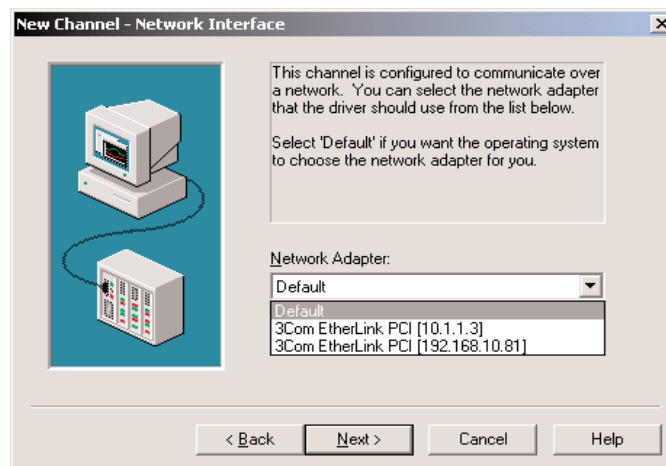
The **Device Driver** window will open. From this window, select **AUTOMATIONDIRECT EBC**. The arrow to the right side of the name provides a drop down list for other device drivers that may be available. After selecting the device driver, click on the **Next** button to go to the next step.



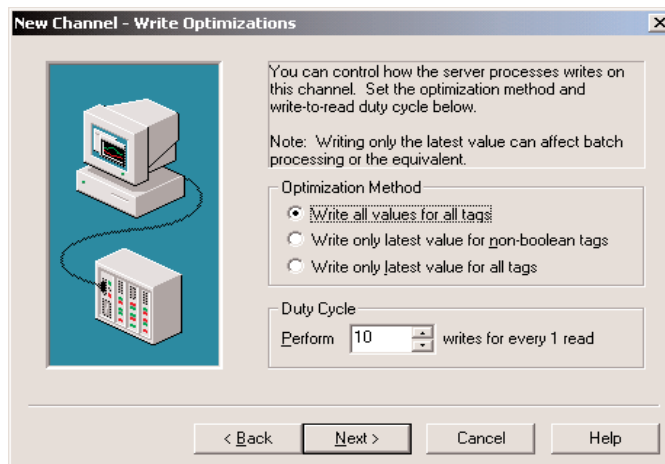
Step 6: Select the Network Interface

The **Network Interface** window will open next. Click on the down arrow to open the drop down list. Select the Ethernet **Network Adapter** that you would like to use to connect to the EBC. This list displays any detected network cards from your computer.

After selecting the network adapter, click on the **Next** button to go to the next step.



Step 7: Write Optimizations Setup



The **Write Optimizations** window will now open. This window provides three choices for a **Optimization Method**. These choices are described in the list below:

- **Write all values for All Tags:** This is the default mode. It forces the server to attempt to write every value to the controller. In this mode the server will continue to gather OPC write requests and add them to the server's internal write queue. The server will then process this write queue and attempt to empty the queue by writing data to the device as quickly as possible.

This mode insures that everything written from your OPC client applications will be sent to the target device. This mode should be selected if the order of your write operations or the content of every write item must uniquely be seen at the target device.

- **Write Only Latest Value for Non-Boolean Tags:** Any value that is not a Boolean value will be updated in the server's internal write queue and will then be sent to the device at the next possible opportunity.

This feature must be used with a clear understanding of how it will affect the operation of your application. This mode does not attempt to optimize writes to Boolean values. This allows you to optimize the operation of HMI data, such as a slide switch, without causing problems with Boolean operations like a momentary push button.

Step 7: Write Optimization Setup (*Continued*)

- **Write Only Latest Value for All Tags:** The final write optimization mode takes the operation described for the second mode and applies it to all tags. If your application needs only to send the latest value to your device, this mode will optimize all writes by updating the tags currently in the write queue before they are sent.

Step 8: Select the Duty Cycle Setting

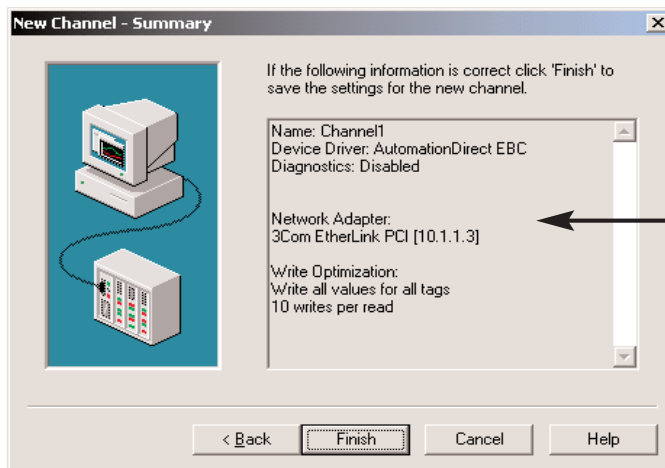
Before ending the write operations, the **Duty Cycle** needs to be set. The **Duty Cycle** selection allows you to control the ratio of write operations to read operations. The ratio is always based on one read for every one to ten writes. By default, the duty cycle is set to ten. This means that ten 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.



NOTE: *It is strongly recommended that you characterize your application for compatibility with these write optimization enhancements before using them in a production environment.*

Step 9: Review the Channel Setup Summary

After setting up the write optimizations, click on the **Next** button. The process ends with a view of the new channel setup. Review the Summary, then click on **Finish** to complete the setup.



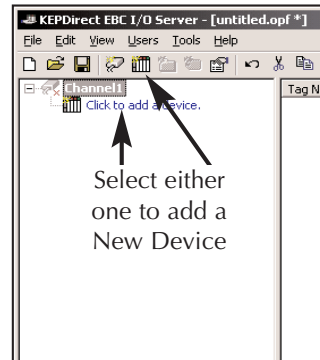
Review the
Channel Setup
Summary

Adding Devices to the New Project

PLCs and HMIs are representative of the types of devices which the server will communicate with. The device driver which the channel is using will restrict the device selection.

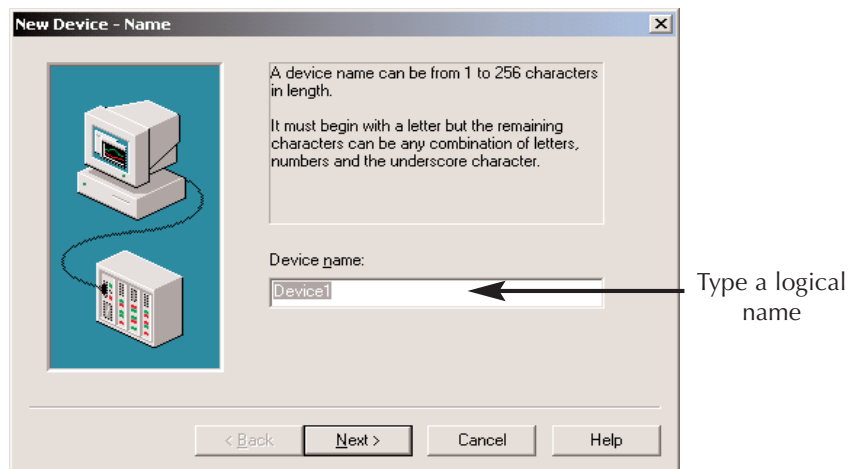
Step 1: Add a Device(s)

Once the channel setup is complete, the application window displays a directory tree on the screen showing the newly created channel name. A new channel can be added in several different ways. One way, the quickest, is to select **Click to add device** next to the device symbol as shown in the figure on the right. Other ways is to select the channel to which you are to add a device and use the **Edit > Add Device**, the Toolbar **Add Device** or the context menu. Any of these methods will initiate the device setup.



Step 2: Name the Device

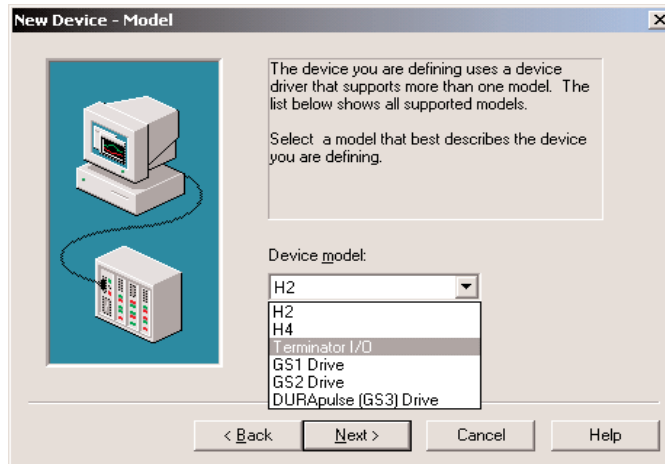
The **New Device** window, shown below, will open. This window allows you to type a logical name from 1 to 256 characters for your device such as a DL205 or accept the default name **Device1**



Once your logical device name has been entered, click on the **Next** button to continue with the device installation.

Step 3: Select the Device Model

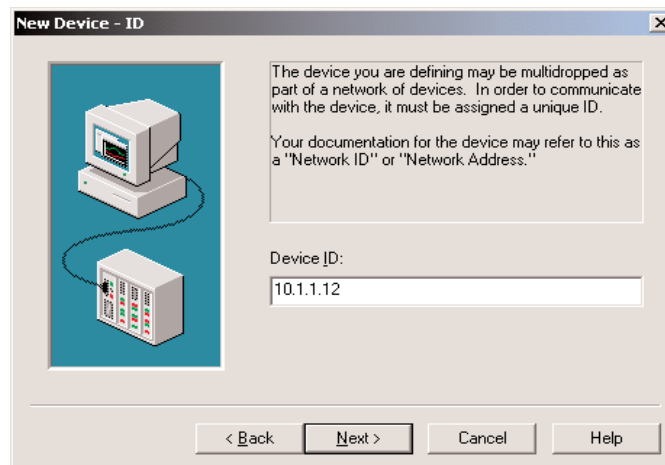
The next window which opens will be the **New Device – Model** window shown below. Click on the arrow to open the drop down list of the devices available for setup and click on the device name to select the model of the device which you are adding, i.e. Terminator I/O.



Once the device model is selected, click on the **Next** button to continue with the device installation.

Step 4: Type the Device IP Address

This will open the **New Device – ID** window shown below. Each device on the channel must be uniquely identified by its own IP address. Up to 1024 devices may be defined on a given channel. The



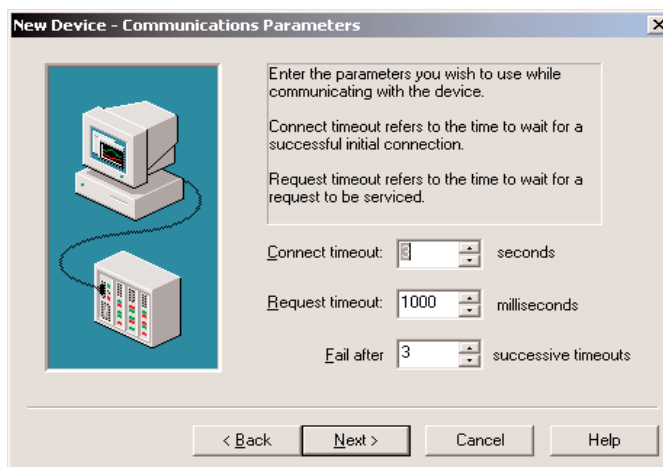
NetEdit 3 configuration software supplied can be used to configure the IP address of an EBC/GS-EDRV module. The IP address is different from

the Unit ID that is configurable on the I/O base unit. In general, the device ID has the following format, YYY.YYY.YYY.YYY where YYY designates the device IP address (each YYY byte should be in the range of 0 - 254).

Type the IP address of the device you are installing. It is important that the IP address is typed correctly and that it corresponds to the device being installed. Once the IP address is typed, click on the **Next** button to continue with the device installation.

Step 5: Set the Device's Communication Parameters

The **Communication Parameters** window shown below opens and displays the available communication settings.



This screen provides three different communication settings for the device being installed. The following list explains the purpose of each available setting.

- **Connection Timeout:** The Connection timeout is used by Ethernet based drivers. The connection timeout allows the time required to establish a socket connection to a remote device to be adjusted. In many cases the connection time to a device can take longer than normal communications request to that same device. The valid range is 1 to 30 seconds. The default is typically 3 seconds but can vary depending on the specific nature of the chosen driver.
- **Request Timeout:** This is used by all drivers to determine how long the driver will wait for a response from the target device. The request timeout has a valid range of 100 to 30000 milliseconds. The default is typically 1000 milliseconds but can vary depending on the specific nature of the chosen driver.

Step 5: Set the Device's Communication Parameters (Continued)

- **Fail After:** This parameter is used to determine how many times the driver will retry a communications request before considering the request to have failed. The valid range is 1 to 10 retries. The default is typically three retries but can vary depending on the specific nature of the driver. The number of retries configured for your application is dependent largely on your communications environment.

If your environment is prone to noise induced communication failures you may want to increase the number of retries the driver performs. Keep in mind, however, that when the driver does encounter a communication issue, it will attempt to reacquire the data for the lost request. Based on the Request timeout and the Fail after count the driver will pause on a specific request until either the device responds or the timeout and retries have been exceeded. With this in mind you wouldn't want to set the timeout to 30000 milliseconds and 10 retries with the hope of covering every possible issue as this would result in a potential communications pause of 5 minutes.

Once the three communication parameter settings have been selected, click on the **Next** button to continue with the device installation.

Step 6: Set the Database Settings

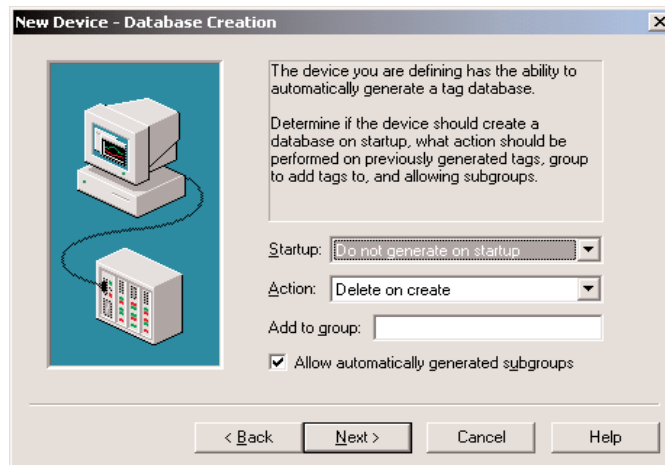
The KEP*Direct* I/O Server has a feature to automatically generate an OPC tag database. This has been designed to make the setup of your OPC application a Plug and Play operation. The **AUTOMATIONDIRECT** EBC I/O driver supports this feature. You can configure the driver to automatically build a list of OPC tags within the server that correspond to device specific data. The automatically generated OPC tags can then be browsed from your OPC client.

If the target device supports its own local tag database, the driver will read the device's tag information and use this data to generate OPC tags within the server. An Ethernet I/O system that supports detection of its' own available I/O module types. The driver in this case will automatically generate OPC tags in the server that are based on the types of I/O modules plugged into the Ethernet I/O rack.



NOTE: The H2-EBC(-F) and Terminator I/O EBCs automatically detect I/O modules in the base on power up. The H4-EBC(-F), however, only detects discrete modules. Use NetEdit 3 to manually configure the H4-EBC(-F) base when using analog modules.

The mode of operation for automatic tag database generation is completely configurable. The following **Database Creation** window, shown below, will open next. It allows you to configure how the server



and the associated communications driver will handle automatic OPC tag database generation. The following list explains the purpose of each available setting and group name.

- **Startup:** This selection allows you to configure when OPC tags will be automatically generated. There are three possible selections:
 - **Do not generate on startup:** This selection prevents the driver from adding any OPC tags to tag space of the OPC Server. This is the default selection.
 - **Always generate on startup:** This selection causes 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.
 - **Generate on first startup:** This selection will cause the driver to evaluate the target device for tag information the first time this OPC Server project is run and to add any OPC tags to the server tag space as needed. When the Startup is selected, any tags that are added to the server's tag space must be saved with the project.

When Startup (automatic tag generation) is enabled, the server needs to know what to do with OPC tags that it may have been added from a previous run or with OPC tags that you may have added or modified after the communications driver has added them.

Step 6: Set the Database Settings (*Continued*)

- **Action:** This selection allows you to control how the server will handle OPC tags that were automatically generated and currently exist in your project. This feature prevents automatically generated tags from piling up in the server. This would occur in an Ethernet I/O system if you continued to change the I/O modules in the rack with the server configured to always generate new OPC tags on startup. Under this condition, every time the communications driver detected a new I/O module, the tags would be added to the server. If the old tags are not removed, a number of unused tags could accumulate in the server's tag space. The Action selection allows you to tailor the server's operation to best fit your application's needs.
 - **Delete on create:** This is the default condition which allows the server to remove any tags that had previously been added to the tag space before the communications driver can add any new tags.
 - **Overwrite as necessary:** This condition allows the server to remove only the tags that the communications driver is replacing with new tags. Any tags that are not being overwritten will remain in the server's tag space.
 - **Do not overwrite:** This condition prevents the server from removing any tags that had been previously generated or that may have already existed in the server. With this selection, the communications driver can only add completely new tags.
 - **Do not overwrite, log error:** This condition has the same effect as the previous condition but it also adds an error message that is posted to the OPC Server's event log when a tag overwrite would have occurred.

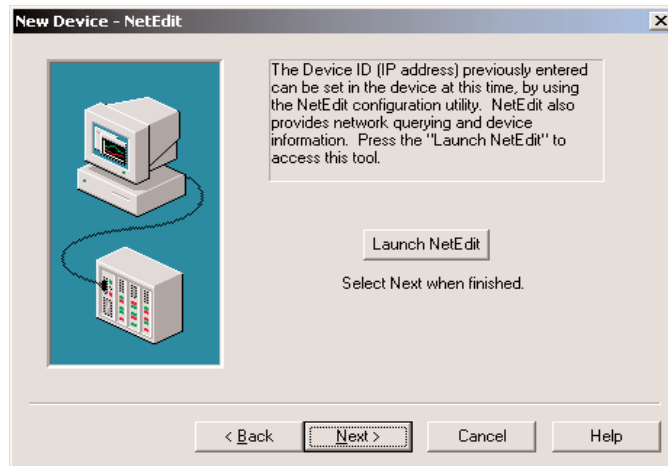


Note: *The removal of OPC tags affects tags that have been automatically generated by the communications driver and any tags you have added using names that match generated tags. It is recommended that you try to avoid adding your own tags to the server using names that match tags that may be automatically generated by the driver.*

- **Add to group:** This parameter can be used to aid in keeping automatically generated tags from mixing with tags that you may enter manually. With this parameter you may specify a sub group that will be used when adding all automatically generated tags for this device. The name of the sub group can be up to 256 characters in length.

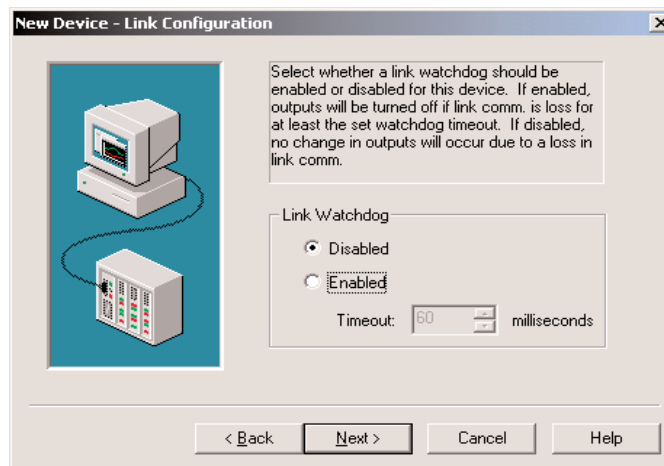
Once the database settings are completed, click on the **Next** button to continue with the device installation.

The following window will appear asking to **Launch NetEdit**. This allows you to check the IP addressing and the Ethernet device connection. If this is not needed at this time, click on the **Next** button to continue with the device setup.



Step 7: Set the Link Configuration

The next window is the **Link Configuration** window shown below. This window offers the options to disable or enable the Link Watchdog.



The Link Watchdog monitors the vital communication link between the PC and the EBC device. It is possible for the communication link to "break" during its use. A "break" in the communications link results in a loss in communication that may be permanent or temporary. An example of a loss in communication would be a physical break in the communication network. An example of a lack of communication

KEP*Direct* EBC I/O Server

would be a significant interval between PC-to-EBC transactions. This interval would have to exceed a set timeout before being considered a loss in communication.

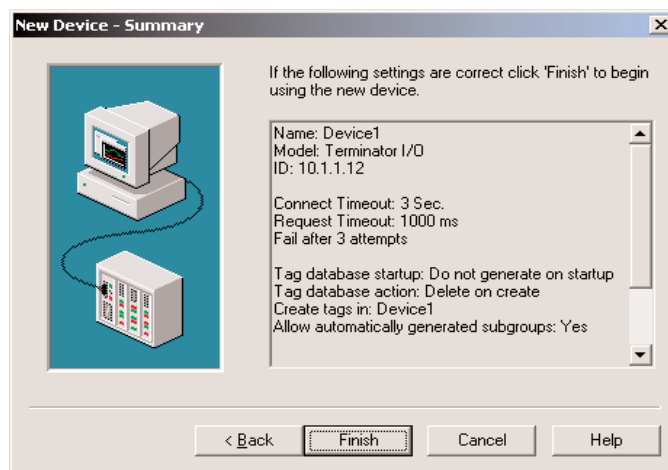
The link watchdog, sometimes referred to as a link monitor, there are two options:

- **Disable:** When the link watchdog is disabled, a "break" in the communication link has no effect on the device's I/O outputs. The outputs will maintain the state they were assigned prior to the "break" until communication is restored and outputs are altered via the client application.
- **Enable:** When link watchdog is enabled, it will continuously monitor the link. If a "break" in the link occurs, a timer is initiated. When the timer reaches the Watchdog Timeout value, all device I/O outputs will be turned off.

Once the link configuration is completed, click on the **Next** button to continue with the device installation.

Step 8: Review the Device Setup Summary

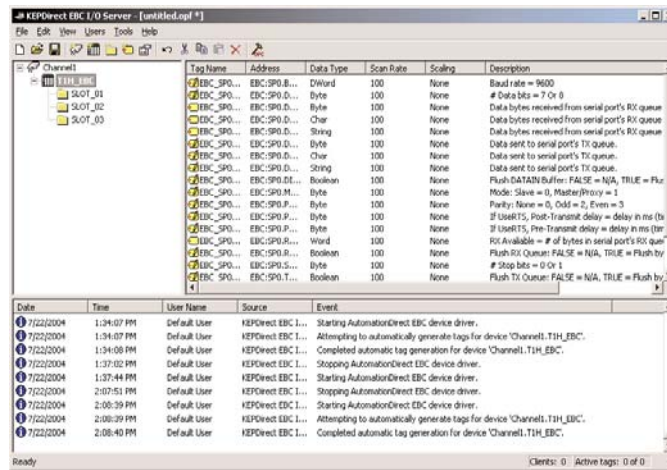
The final window of the device setup is the **New Device Summary** window shown below.



Review the summary and click on **Finish** to complete the device setup.

KEP*Direct* EBC I/O Server

If the I/O had been selected to be generated on start up the **KEP*Direct* Server Program** window will look similar to the following diagram.



Repeat These Steps for Additional Devices

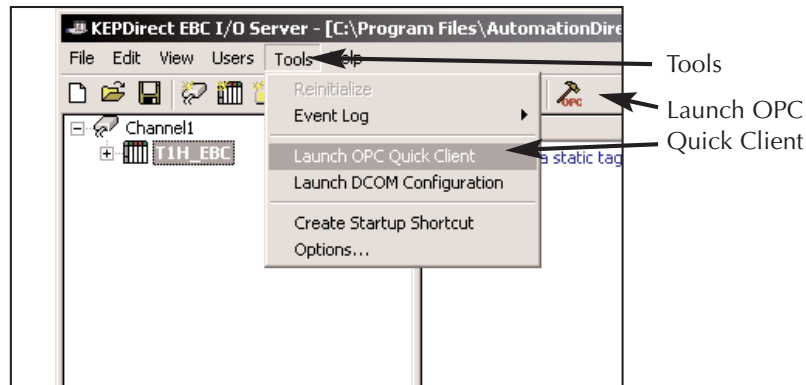
Repeat these steps if more devices are to be added to the selected channel.

Using OPC Quick Client to Test Device Setup

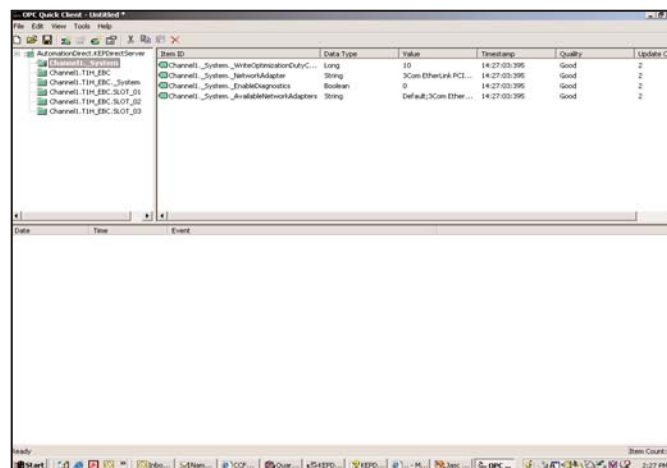
If you follow each of the steps which have been presented, your project is ready to launch the **OPC Quick Client**. Before launching the project you will want to save it to disk. Do this by clicking on **File > Save**. Name the project and save it.

Step 1: Launch the OPC Quick Client

From the KEP*Direct* EBC I/O Server program window, either click on the **Tools** menu and select the **Launch OPC Quick Client** text or click on the **Launch OPC** icon on the tool bar shown in the figure below to launch the **OPC Quick Client** window.

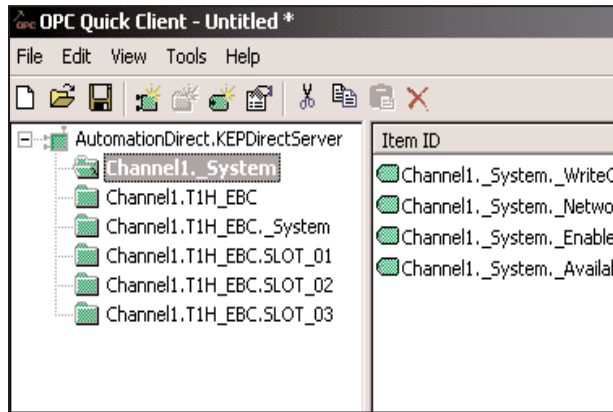


This will open the **OPC Quick Client** window shown here.



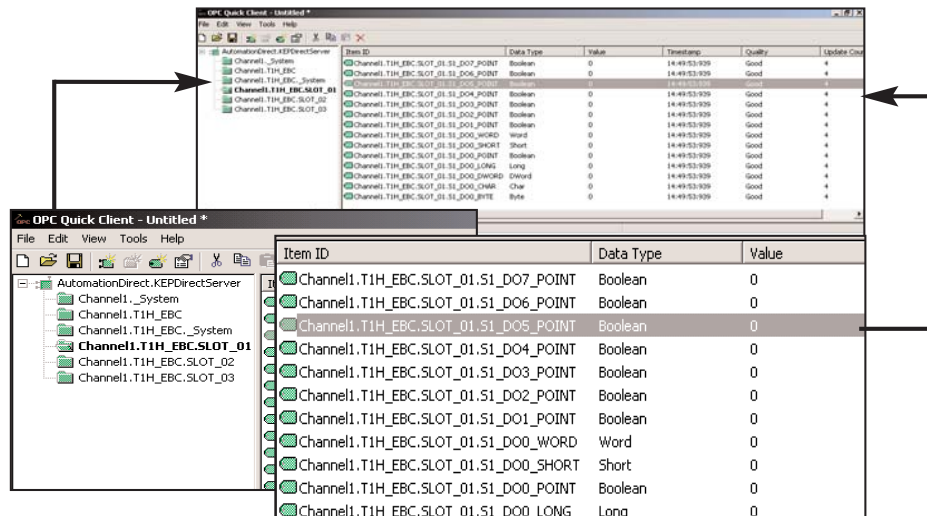
Step 1: Launch the OPC Quick Client (Continued)

In the **OPC Quick Client** window, the first panel on the top left side of the, shown below, displays information about the device connected to the server. Looking at the sample below, the window clearly identifies the server name, **AutomationDirect.KEPDirectServer**. It also shows that the device is connected to **Channel 1** and the device name is **T1H_EBC**. Also that the EBC device has three input/output modules indicated as **SLOT_01**, **SLOT_02** and **SLOT_03**.



Step 2: Select an Output to test

From the available slots, select a slot to test. For our sample we are selecting **Slot 01** as shown in the figure below.



When the slot is selected, the panel on the right displays the available input/output terminals on the device. On the figure above, output point 5 is chosen, shown highlighted.

Step 3: Select an Output to Test (*Continued*)

When selecting an output, it will be identified as shown below.

Item ID	Data Type	Value	Timestamp
Channel1.T1H_EBC.SLOT_01.S1_DO7_POINT	Boolean	0	14:49:53:93
Channel1.T1H_EBC.SLOT_01.S1_DO6_POINT	Boolean	0	14:49:53:93
Channel1.T1H_EBC.SLOT_01.S1_DO5_POINT	Boolean	0	14:49:53:93
Channel1.T1H_EBC.SLOT_01.S1_DO4_POINT	Boolean	0	14:49:53:93
Channel1.T1H_EBC.SLOT_01.S1_DO3_POINT	Boolean	0	14:49:53:93
Channel1.T1H_EBC.SLOT_01.S1_DO2_POINT	Boolean	0	14:49:53:93
Channel1.T1H_EBC.SLOT_01.S1_DO1_POINT	Boolean	0	14:49:53:93
Channel1.T1H_EBC.SLOT_01.S1_D00_WORD	Word	0	14:49:53:93
Channel1.T1H_EBC.SLOT_01.S1_D00_SHORT	Short	0	14:49:53:93
Channel1.T1H_EBC.SLOT_01.S1_D00_POINT	Boolean	0	14:49:53:93
Channel1.T1H_EBC.SLOT_01.S1_D00_LONG	Long	0	14:49:53:93
Channel1.T1H_EBC.SLOT_01.S1_D00_DWORD	DWord	0	14:49:53:93
Channel1.T1H_EBC.SLOT_01.S1_D00_CHAR	Char	0	14:49:53:93
Channel1.T1H_EBC.SLOT_01.S1_D00_BYTE	Byte	0	14:49:53:93

The highlighted item identifies the device as follows:

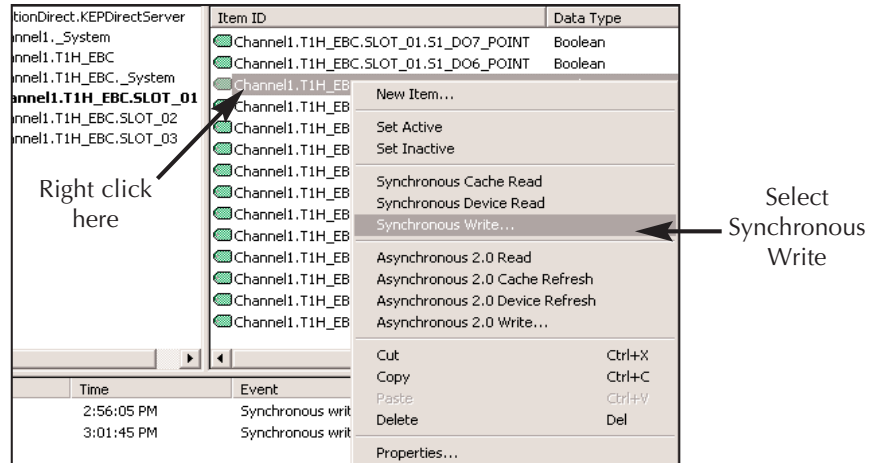
- **T1H_EBC:** Device description connected to Channel 1.
- **Slot_01:** Identifies the location of the I/O module
- **S1_DO05_POINT:** Identifies the connection point or terminal number. On this label, **DO** stands for **Discrete Output** and the **05** indicates point **5** of the I/O module. If the terminal is an input point, the **DO** would be a **DI** representing a **Discrete Input**.
- **Boolean:** Shows the data type used. When selecting outputs to test, ensure that the selected output is a Boolean data type.
- **0:** This is the present value of the output. When the value is **0**, the output is off (deactivated).

The figure below shows a photo of the actual device and indicates the output just described.

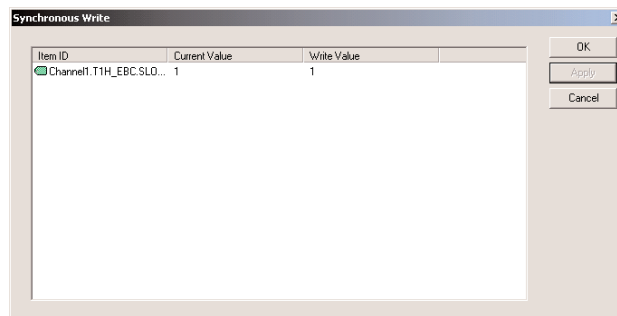


Step 4: Forcing an Output

After selecting the slot and output to test, then right click on the highlighted item from the drop down menu to be displayed.



From the drop down menu, select either **Synchronous Write** or **Asynchronous Write** by clicking on the text in the menu. This selection opens the **Synchronous Write** window shown below.



Click on the 0 (zero) under the **Write Value** in the window, the text bar starts to blink inside the box. Type a 1, then click the **OK** button.



The value on the highlighted item will display a 1 reflecting the **Synchronous Write** just written and the indicator light for point 05 will be illuminated on the Terminator output module.

Step 5: Repeat the Test for all Outputs

To test any remaining discrete outputs, repeat Step 3 and select each Boolean output to be tested.

To select a different slot or module, repeat Step 2 and select the desired slot to test.

To select a different device, select the desired device in the same channel to test if more than one is available.

To learn about other ways to use the OPC Quick Client, refer to the **Help** file selection on the tool bar.

Glossary

Client:

The computers/devices that use the server connection are called the clients. The client devices need to be physically connected to the server via a hub or a serial cable.

DDE:

Dynamic Data Exchange (DDE) is a generic client server technology provided Microsoft. DDE provides a basic architecture that allows many Windows applications from a wide range of vendors to share data.

EBC:

This is the abbreviation for Ethernet Based Controller. An EBC is an Ethernet based slave I/O interface.

LinkMaster:

LinkMaster is a fast and robust Windows application that requires no programming knowledge, simply "drag and drop" to create your links. Built-in scaling, user-access manager, error tracking, and write optimization capabilities, provide total control of your data flow and application access.

OLE:

This is an abbreviation of Object Linking and embedding. OLE is a compound document standard developed by Microsoft Corporation. It enables you to create objects with one application and then link or embed them in a second application. Embedded objects retain their original format and links to the application that created them.

OPC:

This is an abbreviation for Object Process Control. This is the same as OLE (defined above), but for Process Control.

SCADA:

Supervisory Control and Data Acquisition (SCADA). A SCADA system normally consists of a host software program, an HMI unit, a telemetry system, and Remote Terminal Units (RTUs) and/or Programmable Logic Controllers (PLCs) that are typically installed in remote areas and communicate over long distances to the Host system.

Server:

A server is a computer/device, which provides information or services to computers on a network.