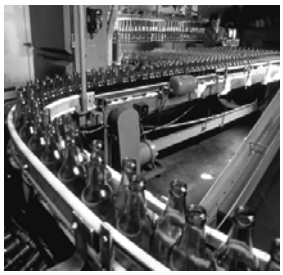
LISTEN.
THINK.
SOLVE.®

PowerFlex® 755 Drive
Embedded EtherNet/IP Adapter



USER MANUAL

Firmware Version 1.xxx

ALLEN-BRADLEY • ROCKWELL SOFTWARE

Rockwell Automation

Important User Information

Solid state equipment has operational characteristics differing from those of electromechanical equipment. *Safety Guidelines for the Application, Installation and Maintenance of Solid State Controls* (Publication SGI-1.1 available from your local Rockwell Automation sales office or online at http://www.rockwellautomation.com/literature) describes some important differences between solid state equipment and hard-wired electromechanical devices. Because of this difference, and also because of the wide variety of uses for solid state equipment, all persons responsible for applying this equipment must satisfy themselves that each intended application of this equipment is acceptable.

In no event will Rockwell Automation, Inc. be responsible or liable for indirect or consequential damages resulting from the use or application of this equipment.

The examples and diagrams in this manual are included solely for illustrative purposes. Because of the many variables and requirements associated with any particular installation, Rockwell Automation, Inc. cannot assume responsibility or liability for actual use based on the examples and diagrams.

No patent liability is assumed by Rockwell Automation, Inc. with respect to use of information, circuits, equipment, or software described in this manual.

Reproduction of the contents of this manual, in whole or in part, without written permission of Rockwell Automation, Inc. is prohibited.

Throughout this manual, when necessary we use notes to make you aware of safety considerations.

WARNING: Identifies information about practices or circumstances that can cause an explosion in a hazardous environment, which may lead to personal injury or death, property damage, or economic loss.

Important: Identifies information that is critical for successful application and understanding of the product.

ATTENTION: Identifies information about practices or circumstances that can lead to personal injury or death, property damage, or economic loss. Attentions help you identify a hazard, avoid a hazard, and recognize the consequences.

Shock Hazard labels may be located on or inside the equipment (e.g., drive or motor) to alert people that dangerous voltage may be present.

Burn Hazard labels may be located on or inside the equipment (e.g., drive or motor) to alert people that surfaces may be at dangerous temperatures.

This is the first release of the PowerFlex 755 Drive Embedded EtherNet/IP Adapter User Manual.

Table of Contents

About This Manual

Related Documentation

For:	Refer to:	Publication
EtherNet/IP	EtherNet/IP Planning and Installation Manual EtherNet/IP Performance and Application Guide	ENET-IN001 ENET-AP001
DriveExplorer™	http://www.ab.com/drives/driveexplorer, and DriveExplorer online help (installed with the software)	—
DriveTools™ SP (includes DriveExecutive™)	http://www.ab.com/drives/drivetools, and DriveExecutive online help (installed with the software)	—
Enhanced PowerFlex 7-Class HIM	Enhanced PowerFlex 7-Class HIM User Manual	20HIM-UM001
PowerFlex® 750-Series AC Drives	PowerFlex 750-Series AC Drives User Manual PowerFlex 750-Series Reference Manual	750-UM001 750-RM001
RSLinx® Classic	Getting Results with RSLinx Guide, and online help (installed with the software)	LINX-GR001
RSLogix™ 5 RSLogix™ 500 RSLogix™ 5000	RSLogix 5 Getting Results Guide* RSLogix 500 Getting Results Guide* RSLogix 5000 Getting Results Guide* * And online help (installed with the software)	LG5-GR001 LG500-GR001 9399-RLD300GR
ControlLogix® and 1756-ENBT or 1756-EN2T	EtherNet/IP Modules in Logix5000 Control Systems User Manual	ENET-UM001
PLC-5®	Enhanced and Ethernet PLC-5 Programmable Controllers User Manual	1785-UM012
SLC™ 500 and 1747-L5-xxx	SLC 500 Modular Hardware Style User Manual	1747-UM011
MicroLogix™ 1100	MicroLogix 1100 Programmable Controllers User Manual	1763-UM001

You can view or download publications at www.rockwellautomation.com/literature. To order paper copies of technical documentation, contact your local Rockwell Automation distributor or sales representative.

To find your local Rockwell Automation distributor or sales representative, visit www.rockwellautomation.com/locations.

For information such as firmware updates or answers to drive-related questions, go to the Drives Service & Support web site at www.ab.com/support/abdrives and click on the “Downloads” or “Knowledgebase” link.

Rockwell Automation Support

Rockwell Automation, Inc. offers support services worldwide, with over 75 sales/support offices, over 500 authorized distributors, and over 250 authorized systems integrators located through the United States alone. In addition, Rockwell Automation, Inc. representatives are in every major country in the world.

Local Product Support

Contact your local Rockwell Automation, Inc. representative for:

- Sales and order support
- Product technical training
- Warranty support
- Support service agreements

Technical Product Assistance

For technical assistance, please review the information in Chapter 7, Troubleshooting, first. If you still have problems, then access the Allen-Bradley Technical Support web site at www.ab.com/support/abdrives or contact Rockwell Automation, Inc.

Conventions Used in This Manual

The following conventions are used throughout this manual:

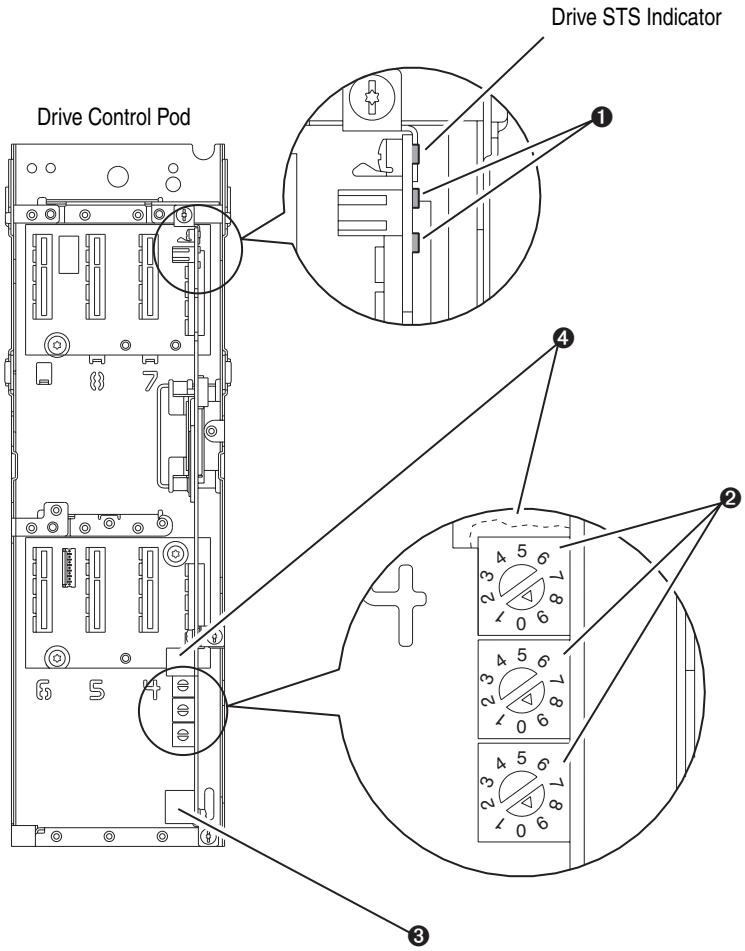
- Parameter names are shown in the format **Parameter xx - [*]**. The xx represents the parameter number. The * represents the parameter name — for example **Parameter 01 - [DL From Net Cfg 01]**.
- Menu commands are shown in bold type face and follow the format **Menu > Command**. For example, if you read “Select **File > Open**,” you should click the **File** menu and then click the **Open** command.
- The firmware release is displayed as FRN X.xxx. The “FRN” signifies Firmware Release Number. The “X” is the major release number. The “xxx” is the minor update number.
- RSLinx Classic (version 2.52), RSLogix 5 (version 7.20), RSLogix 500 (version 7.20), and RSLogix 5000 (version 16) were used for the screen shots in this manual. Different versions of the software may differ in appearance and procedures.
- This manual provides information about the EtherNet/IP adapter embedded on the Main Control Board in PowerFlex 755 drives, and using it for network communication.

Getting Started

The EtherNet/IP adapter, embedded on the Main Control Board in PowerFlex 755 drives, is used for network communication.

Components

Figure 1.1 Components of the Embedded EtherNet/IP Adapter (shown with HIM bezel open and drive cover removed)



Item	Part	Description
❶	Status Indicators	Two LEDs that indicate the status of the adapter and network communications. Refer to Chapter 7, Troubleshooting.
❷	IP Address Switches	Sets the IP address of the embedded adapter when not using BOOTP or adapter parameters. Refer to Setting the IP Address Switches on page 2-2.
❸	Ethernet Connector	An RJ-45 connector for the Ethernet cable. The connector is CAT-5 compliant to ensure reliable data transfer on 100Base-TX Ethernet connections.
❹	DPI Port 2	Cable connection for handheld and remote options.

Features

The features of the embedded EtherNet/IP adapter include:

- Switches to set an IP address before applying power to the drive—or you can disable the switches and use a BOOTP server or adapter parameters to configure the IP address.
- Compatibility with various configuration tools to configure the embedded EtherNet/IP adapter and host drive. The tools include the enhanced PowerFlex 7-Class HIM (Human Interface Module) on the drive, and drive-configuration software such as DriveExplorer (version 6.01 or higher) or DriveExecutive (version 5.01 or higher).
- Status indicators that report the status of the embedded EtherNet/IP adapter and network communications. They are visible when the drive cover is open or closed.
- Parameter-configurable 32-bit Datalinks in the I/O to meet application requirements (16 for writing data from the network, and 16 for reading data to the network).
- Explicit Messaging support.
- Master-Slave or Peer-to-Peer hierarchy that can be configured to transmit data to and from either a controller or another PowerFlex 750-Series drive on the network.
- User-defined fault actions to determine how the embedded EtherNet/IP adapter and its host PowerFlex 755 drive respond to:
  - I/O messaging communication disruptions (Comm Flt Action)
  - Controllers in idle mode (Idle Flt Action)
  - Peer device communication disruptions (Peer Flt Action)
  - Explicit messaging disruptions for drive control via PCCC and the CIP Register Object (Msg Flt Action)
- Web pages, viewed using a web browser, that show information about the embedded EtherNet/IP adapter, its host drive, and DPI devices connected to the drive.
- Configurable e-mail messaging to desired addresses when selected drive faults occur and/or are cleared, and/or when the embedded EtherNet/IP adapter takes a communication or idle fault action.
- Support for DPI routing, enabling access to any networked PowerFlex 7-Class drive using DriveExplorer (version 6.01 or higher) to monitor and configure that drive and its connected peripherals.

Compatible Products

At the time of publication, the embedded EtherNet/IP adapter is compatible with Allen-Bradley PowerFlex 750-Series drives.

Required Equipment

Equipment Shipped with the Drive

Since the EtherNet/IP adapter is embedded on the Main Control Board in the PowerFlex 755 drive, it is always an integral part of the drive and, therefore, is not shipped with installation instructions.

User-Supplied Equipment

To configure the embedded EtherNet/IP adapter, you must supply:

- ❑ A small flathead screwdriver
- ❑ Ethernet cable (for details, refer to the *EtherNet/IP Media Planning and Installation Manual, publication ENET-IN001*)
- ❑ Ethernet switch (for details, refer to the *EtherNet/IP Performance Application Solution, publication ENET-AP001*)
- ❑ Configuration tool, such as:
  - Enhanced PowerFlex 7-Class HIM (20-HIM-A6/C6S)
  - DriveExplorer (version 6.01 or higher)
  - DriveExecutive stand-alone software (version 5.01 or higher) or bundled with the DriveTools SP suite (version 5.01 or higher)
  - BOOTP Server (version 2.1 or higher) for network setup only
- ❑ Controller configuration software (such as RSLogix 5/500/5000)
- ❑ A PC connection to the EtherNet/IP network

Safety Precautions

Please read the following safety precautions carefully.

ATTENTION: Risk of injury or equipment damage exists. Only personnel familiar with drive and power products and the associated machinery should plan or implement the installation, start up, configuration, and subsequent maintenance of the drive using this embedded adapter. Failure to comply may result in injury and/or equipment damage.

ATTENTION: Risk of equipment damage exists. The embedded adapter contains ESD (Electrostatic Discharge) sensitive parts that can be damaged if you do not follow ESD control procedures. Static control precautions are required when handling the adapter. If you are unfamiliar with static control procedures, refer to *Guarding Against Electrostatic Damage (publication 8000-4.5.2)*.

ATTENTION: Risk of injury or equipment damage exists. If the adapter is transmitting control I/O to the drive, the drive may fault when you reset the adapter. Determine how your drive will respond before resetting the adapter.

ATTENTION: Risk of injury or equipment damage exists. Embedded adapter **Parameters 54 - [Comm Flt Action]**, **55 - [Idle Flt Action]**, **56 - [Peer Flt Action]**, and **57 - [Msg Flt Action]** let you determine the action of the adapter and drive if I/O communication is disrupted, the controller is idle, Peer I/O is disrupted, or explicit messaging for drive control is disrupted. By default, these parameters fault the drive. You can set these parameters so that the drive continues to run. Precautions should be taken to ensure that the settings of these parameters do not create a risk of injury or equipment damage. When commissioning the drive, verify that your system responds correctly to various situations (for example, a disconnected cable or a controller in idle state).

ATTENTION: Risk of injury or equipment damage exists. When a system is configured for the first time, there may be unintended or incorrect machine motion. Disconnect the motor from the machine or process during initial system testing.

ATTENTION: Risk of injury or equipment damage exists. The examples in this publication are intended solely for purposes of example. There are many variables and requirements with any application. Rockwell Automation, Inc. does not assume responsibility or liability (to include intellectual property liability) for actual use of the examples shown in this publication.

Quick Start

This section is provided to help experienced users quickly start using the embedded EtherNet/IP adapter. If you are unsure how to complete a step, refer to the referenced chapter.

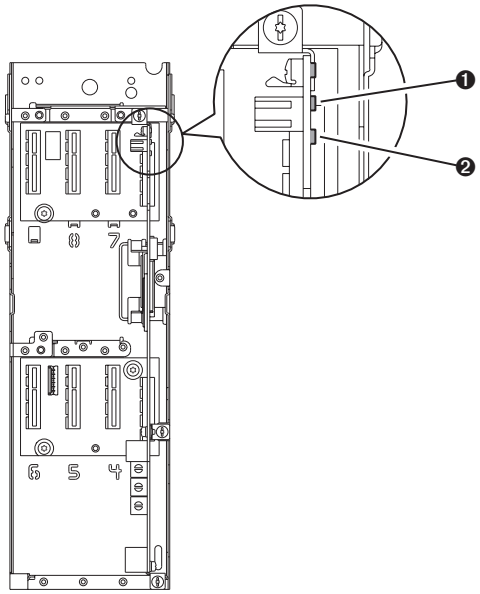
Step	Action	Refer to...
1	**Review the safety precautions for the adapter.**	Throughout This Manual
2	**Verify that the PowerFlex drive is properly installed.**	PowerFlex 750-Series AC Drive Installation Instructions
3	**Set the adapter IP address.** A. When using the adapter switches, set the IP address. When using a BOOTP server or adapter parameters to set the IP address, first perform Step 3B and all of Step 4. Then proceed with Step 5. B. Verify that the PowerFlex drive is not powered. Then, connect the embedded EtherNet/IP adapter to the network using an Ethernet cable.	Chapter 2, Installing the Adapter
4	**Apply power to the drive.** A. Replace the drive cover or close the drive door. B. The embedded EtherNet/IP adapter receives power from the drive. Apply power to the drive. The status indicators should be green. If they flash red, there is a problem. Refer to Chapter 7, Troubleshooting. C. Configure/verify key drive parameters.	Chapter 2, Installing the Adapter
5	**Configure the adapter for your application.** Set embedded EtherNet/IP adapter parameters for the following functions as required by your application: • IP address, subnet mask, and gateway address (only when not using adapter switches) • Data rate • I/O configuration • Master-Slave or Peer-to-Peer hierarchy • Fault actions • Web enable/features	Chapter 3, Configuring the Adapter
6	**Configure the controller to communicate with the adapter.** Use a controller configuration tool such as RSLogix to configure the master on the EtherNet/IP network to recognize the embedded EtherNet/IP adapter and drive.	Chapter 4, Configuring the I/O
7	**Create a ladder logic program.** Use a controller configuration tool such as RSLogix to create a ladder logic program that enables you to: • Control the embedded EtherNet/IP adapter and drive using I/O. • Monitor or configure the drive using Explicit messages.	Chapter 5, Using the I/O Chapter 6, Using Explicit Messaging

Status Indicators

The embedded EtherNet/IP adapter uses two status indicators to report its operating status. They can be viewed with the HIM cradle closed or open (Figure 1.2).

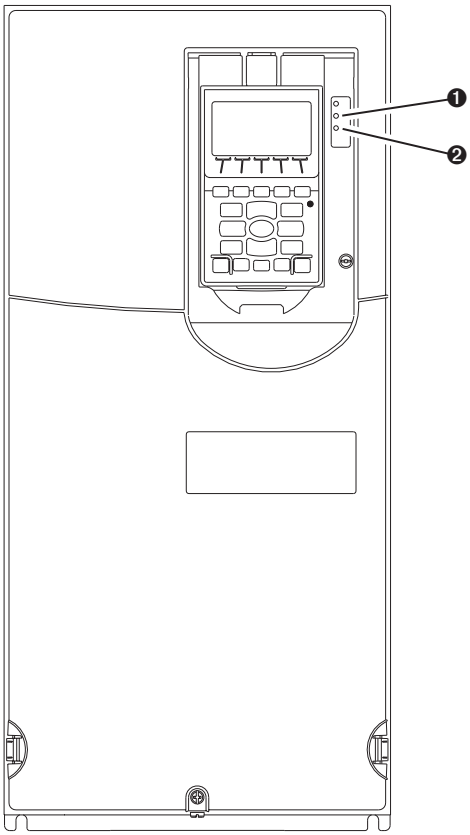
Figure 1.2 Status Indicators

Embedded EtherNet/IP adapter status indicators on main control board in drive control pod.



Item	Name
❶	ENET
❷	LINK

Indicators shown with HIM bezel closed and drive cover installed.



After connecting the embedded EtherNet/IP adapter to the network and applying power to the drive, refer to Start-Up Status Indications on page 2-5 for possible start-up status indications and their descriptions.

Installing the Adapter

Since the EtherNet/IP adapter is embedded on the Main Control Board in the PowerFlex 755 drive, the only required adapter installation is setting its IP address and connecting it to the network.

Preparing for an Installation

Before installing the embedded EtherNet/IP adapter:

- Read the *EtherNet/IP Performance and Application Guide (publication ENET-AP001)* and *EtherNet/IP Media Planning and Installation Manual (publication ENET-IN001)*.
- Understand IGMP Snooping/Ethernet Switches

  Much of EtherNet/IP implicit (I/O) messaging uses IP multicast (including this embedded EtherNet/IP adapter) to distribute I/O control data, which is consistent with the CIP producer/consumer model. Historically, most switches have treated multicast packets the same as broadcast packets. That is, all multicast packets are re-transmitted to all ports.

  IGMP snooping constrains the flooding of multicast traffic by dynamically configuring switch ports so that multicast traffic is forwarded only to ports associated with a particular IP multicast group.

  Switches that support IGMP snooping “learn” which ports have devices that are part of a particular multicast group and only forward the multicast packets to the ports that are part of the multicast group.

  Be careful as to what level of support a switch has of IGMP snooping. Some layer 2 switches that support IGMP snooping require a router (which could be a layer 3 switch) to send out IGMP polls to learn what devices are part of the multicast group. Some layer 2 switches can use IGMP snooping without a router sending polls. If your control system is a stand-alone network or is required to continue performing if the router is out of service, make sure the switch you are using supports IGMP snooping without a router being present.
- Verify that you have all required equipment. Refer to Required Equipment on page 1-3.

Setting the IP Address Switches

There are three methods for configuring the embedded EtherNet/IP adapter's IP address:

- **Adapter Rotary Switches** — Use the switches when working on a simple, isolated network (for example, 192.168.1.xxx) that has other products with switches to set their IP addresses, does not need to be accessed from outside the network, and you prefer a simplified node addressing method. The three adapter switches are read when the drive powers up, and represent three decimal digits from top to bottom (see Figure 2.1). If set to a valid address (001-254), the adapter will use that value as the lower octet of its IP address (192.168.1.xxx, where xxx = rotary switch settings), along with a subnet mask of 255.255.255.0 and there will be no gateway configured. Also, the setting for adapter **Parameter 36 - [BOOTP]** is automatically ignored.

  See Figure 2.1 and its accompanying table for all possible switch settings and their related descriptions.

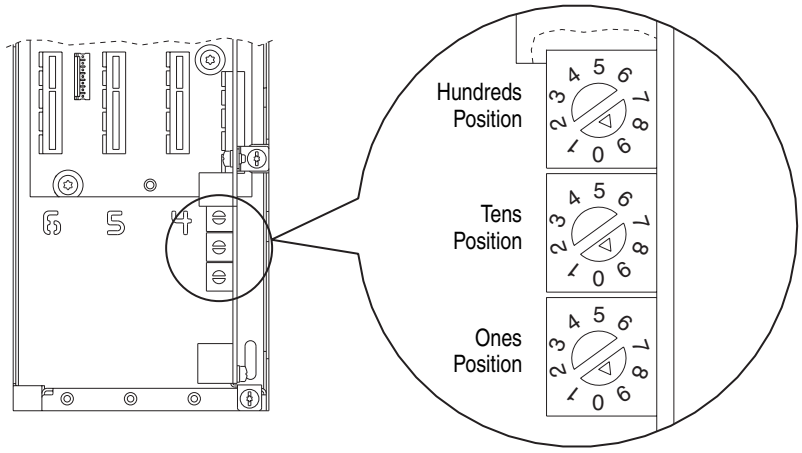
  Important: When using the adapter rotary switches, set the IP address before power is applied because the adapter uses the IP address it detects when it first receives power.

- **BOOTP Server** — Use BOOTP if you prefer to control the IP addresses of devices using a server. The IP address, subnet mask, and gateway addresses will then be provided by the BOOTP server.

- **Adapter Parameters** — Use adapter parameters when you want more flexibility in setting up the IP address, or need to communicate outside the control network using a gateway. The IP address, subnet mask, and gateway addresses will then come from the adapter parameters you set.

Important: Regardless of the method used to set the adapter's IP address, each node on the EtherNet/IP network must have a unique IP address. To change an IP address, you must set the new value and then remove and reapply power to (or reset) the adapter.

Figure 2.1 Setting the IP Address Switches



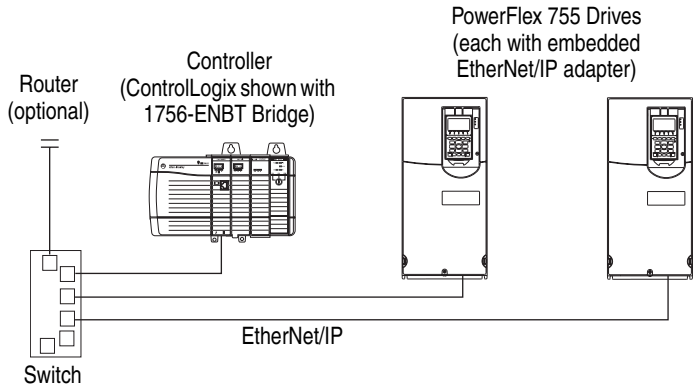
Possible Settings	Description
000	Adapter will use, depending on **Parameter 36 - [BOOTP]**, the BOOTP setting or the adapter parameter settings for the IP address.
001 - 254	Adapter will use the rotary switch settings for the IP address (192.168.1.xxx, where xxx = rotary switch settings).
255 - 887	Adapter will use, depending on **Parameter 36 - [BOOTP]**, the BOOTP setting or the adapter parameter settings for the IP address.
888	Resets the adapter IP address function to factory defaults. Thereafter, the drive must be powered down, the switches set to a setting other than 888, and then the drive must be powered up again to accept the new address.
889 - 998	Adapter will use, depending on **Parameter 36 - [BOOTP]**, the BOOTP setting or the adapter parameter settings for the IP address.
999 (default settings)	Disables the rotary switches. Adapter will use, depending on **Parameter 36 - [BOOTP]**, the BOOTP setting or the adapter parameter settings for the IP address.

Connecting the Adapter to the Network

ATTENTION: Risk of injury or death exists. The PowerFlex drive may contain high voltages that can cause injury or death. Remove power from the drive, and then verify power has been discharged before connecting the embedded EtherNet/IP adapter to the network.

1. Remove power from the drive.
2. Remove the drive cover.
3. Use static control precautions.
4. Connect an Ethernet cable to the EtherNet/IP network. See Figure 2.2 for an example of wiring to an EtherNet/IP network.

Figure 2.2 Connecting the Ethernet Cable to the Network



5. Route the Ethernet cable through the bottom of the PowerFlex 755 drive, and insert the cable's plug into the embedded EtherNet/IP adapter's mating socket (item 3 in Figure 1.1).

Applying Power

ATTENTION: Risk of equipment damage, injury, or death exists. Unpredictable operation may occur if you fail to verify that parameter settings are compatible with your application. Verify that settings are compatible with your application before applying power to the drive.

Install the drive cover, and apply power to the drive. The embedded EtherNet/IP adapter receives its power from the drive. When you apply power to the embedded EtherNet/IP adapter for the first time, its "ENET" status indicator should be solid or flashing green after an initialization. If it is red, there is a problem. Refer to Chapter 7, Troubleshooting.

Start-Up Status Indications

After power has been applied, status indicators for the PowerFlex 755 drive and embedded EtherNet/IP adapter can be viewed on the front of the drive (Figure 2.3). Possible start-up status indications are shown in Table 2.A.

Figure 2.3 Drive and Adapter Status Indicators

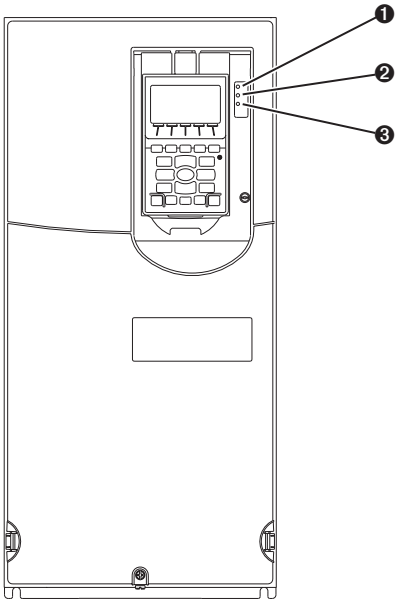


Table 2.A Drive and Adapter Start-Up Status Indications

Item	Name	Color	State	Description
				Drive STS Indicator
1	STS (Status)	Green	Flashing	Drive ready but not running, and no faults are present.
			Steady	Drive running, no faults are present.
		Yellow	Flashing	When running, a type 2 (non-configurable) alarm condition exists – drive continues to run. When stopped, a start inhibit condition exists and the drive cannot be started (see drive parameter 933 - [Start Inhibit]).
			Steady	A type 1 (user configurable) alarm condition exists, but the drive continues to run.
		Red	Flashing	A major fault has occurred. Drive will stop. Drive cannot be started until fault condition is cleared.
			Steady	A non-resettable fault has occurred.
		Red/Yellow	Flashing Alternately	A minor fault has occurred. Use drive parameter 950 - [Minor Flt Config] to enable. If not enabled, acts like a major fault. When running, the drive continues to run. System is brought to a stop under system control. The fault must be cleared to continue.
		Yellow/Green	Flashing Alternately	When running, a type 1 alarm exists.
		Green/Red	Flashing Alternately	Drive is flash updating.
				Embedded EtherNet/IP Adapter Status Indicators
2	ENET	Unlit	Off	Adapter and/or network is not powered, adapter is not properly connected to the network, or adapter needs an IP address.
		Red	Flashing	An EtherNet/IP connection has timed out.
			Steady	Adapter failed the duplicate IP address detection test.
		Red/Green	Flashing Alternately	Adapter is performing a self-test.
		Green	Flashing	Adapter is properly connected, but is not communicating with any devices on the network.
			Steady	Adapter is properly connected and communicating on the network.
3	LINK	Unlit	Off	Adapter is not powered or is not transmitting on the network.
		Green	Flashing	Adapter is properly connected and transmitting data packets on the network.
			Steady	Adapter is properly connected, but is not transmitting on the network.

Configuring/Verifying Key Drive Parameters

The PowerFlex 755 drive can be separately configured for the control and Reference functions in various combinations. For example, you could set the drive to have its control come from a peripheral or terminal block with the Reference coming from the network. Or you could set the drive to have its control come from the network with the Reference coming from another peripheral or terminal block. Or you could set the drive to have both its control and Reference come from the network.

The following steps in this section assume that the drive will receive the Logic Command and Reference from the network.

1. Use drive Parameter 545 - [Speed Ref A Sel] to set the drive speed Reference to "Port 13 Reference" (the drive port dedicated to the embedded EtherNet/IP adapter).
2. Verify that drive Parameter 930 - [Speed Ref Source] is reporting that the source of the Reference to the drive is "Port 13 Reference." This ensures that any Reference commanded from the network can be monitored by using drive Parameter 002 - [Commanded SpdRef]. If a problem occurs, this verification step provides the diagnostic capability to determine whether the drive/adapter or the network is the cause.
3. If hard-wired discrete digital inputs are not used to control the drive, verify that all unused digital input parameters are set to "Not Used."

Commissioning the Adapter

To commission the embedded EtherNet/IP adapter, you must set a unique IP address. See the Glossary for details about IP addresses. When using the adapter switches, refer to Setting the IP Address Switches on page 2-2. When not using the adapter switches, use either a BOOTP server or adapter parameters to set the IP address after connecting the adapter to the network and applying power to the drive.

By default, the adapter is configured so that you must set the IP address using a BOOTP server. For details, see Using BOOTP on page 3-2. To set the IP address using adapter parameters, refer to Setting the IP Address, Subnet Mask, and Gateway Address on page 3-5.

Important: New settings for some adapter parameters (for example, **Parameters 38 - [IP Addr Cfg 1]** through **41 - [IP Addr Cfg 4]**) are recognized only when power is applied to the adapter or it is reset. After you change parameter settings, cycle power or reset the adapter.

Configuring the Adapter

This chapter provides instructions and information for setting the parameters to configure the embedded EtherNet/IP adapter.

For a list of parameters, refer to Appendix B, Adapter Parameters. For definitions of terms in this chapter, refer to the Glossary.

Configuration Tools

The embedded EtherNet/IP adapter stores parameters and other information in its own non-volatile memory. You must, therefore, access the adapter to view and edit its parameters. The following tools can be used to access the adapter parameters:

Tool	Refer to…
Enhanced PowerFlex 7-Class HIM	Enhanced PowerFlex 7-Class HIM User Manual
BOOTP Server	page 3-2
DriveExplorer Software (version 6.01 or higher)	http://www.ab.com/drives/driveexplorer, or DriveExplorer online help (installed with the software)
DriveExecutive Software (version 5.01 or higher)	http://www.ab.com/drives/drivetools, or DriveExecutive online help (installed with the software)

Using the Enhanced PowerFlex 7-Class HIM

If your drive has an enhanced PowerFlex 7-Class HIM, it can be used to access parameters in the adapter. For details on viewing and editing parameters, refer to the *Enhanced PowerFlex 7-Class HIM User Manual (publication 20HIM-UM001)*.

Using BOOTP

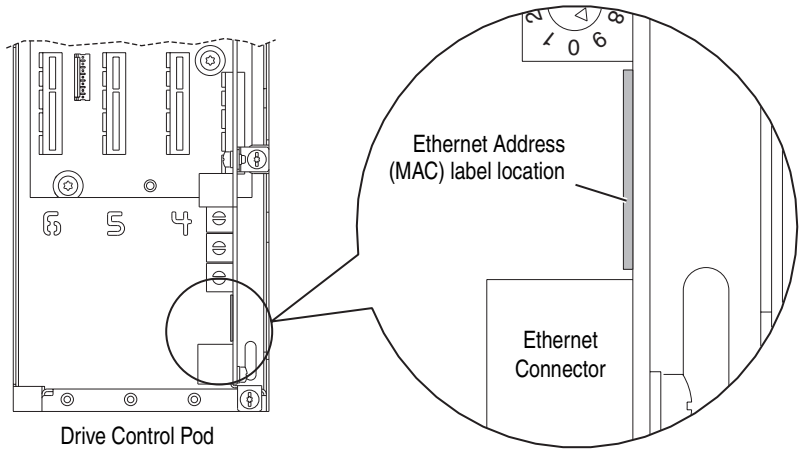
By default, the embedded EtherNet/IP adapter is configured to set its IP address, subnet mask, and gateway address by using a BOOTP utility. You can select from a variety of BOOTP utilities. These instructions use Rockwell's BOOTP Server (version 2.3 or higher), a stand-alone program that incorporates the functionality of standard BOOTP utilities with a graphical interface. It is available from www.ab.com/networks/bootp.html. Refer to the Readme file and online Help for detailed directions and information.

TIP: If desired, you can disable BOOTP and configure the IP address, subnet mask, and gateway address by setting adapter parameters. For details, see Setting the IP Address, Subnet Mask, and Gateway Address on page 3-5.

Configuring the Adapter Using BOOTP Server

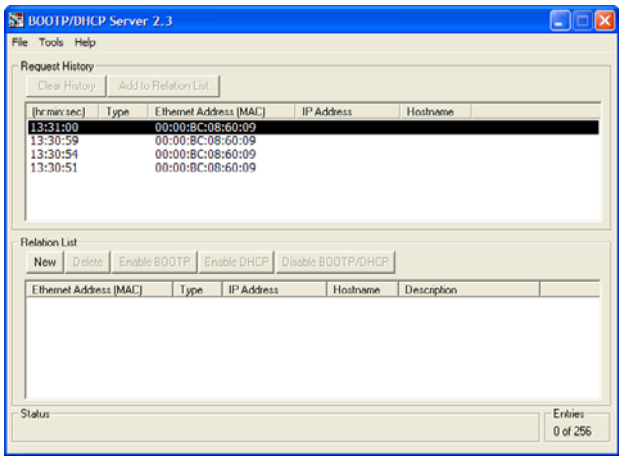
1. Verify and note the adapter's hardware Ethernet Address (MAC), which will be used in Step 6. There are two ways to do this:
   - Use the HIM to scroll to Port 13 and access the embedded EtherNet/IP adapter's DIAGNOSTIC folder screen. Then scroll to Diagnostic Items 43 (HW Addr 1) through 48 (HW Addr 6) to view the adapter's hardware Ethernet Address (MAC).
   - Remove the drive cover and locate the adapter's hardware Ethernet Address (MAC) label on the drive's Main Control Board (Figure 3.1).

Figure 3.1 Adapter Hardware Address Label Location



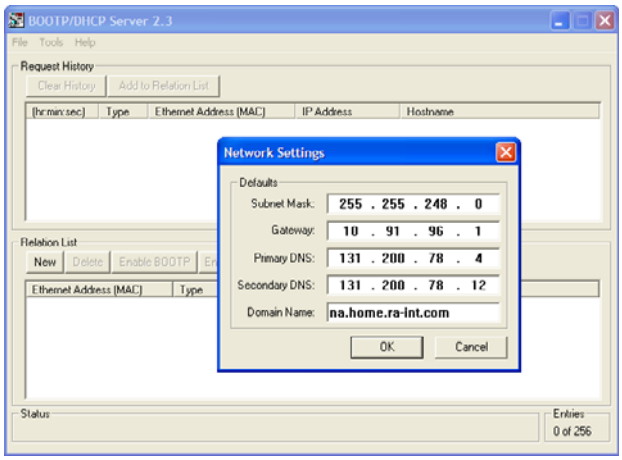
2. On a computer connected to the EtherNet/IP network, start the BOOTP software. The BOOTP Server window (Figure 3.2) appears.

Figure 3.2 BOOTP Server Window



3. To properly configure devices on your EtherNet/IP network, you must configure settings in the BOOTP software to match the network. Select **Tools > Network Settings** to display the Network Settings window (Figure 3.3).

Figure 3.3 Network Settings Window



4. Edit the following:

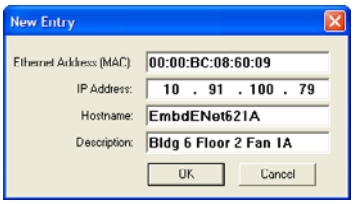
Box	Type
Subnet Mask (1)	The subnet mask for the embedded EtherNet/IP adapter's network.
Gateway (1)	The IP address of the gateway device on the adapter's network.
Primary DNS	The address of the primary DNS server to be used on the local end of the link for negotiating with remote devices.
Secondary DNS	Optional—the address of the secondary DNS server to be used on the local end of the link for negotiating with remote devices when the primary DNS server is unavailable.
Domain Name	The text name corresponding to the numeric IP address that was assigned to the server that controls the network.

(1) For definitions of these terms, refer to the Glossary.

5. Click **OK** to apply the settings. Devices on the network issuing BOOTP requests appear in the BOOTP Request History list.

6. In the BOOTP Request History list, either double-click the adapter's Ethernet Address (MAC) noted in Step 1A or Step 1B, or click **New** in the Relation List. The New Entry dialog box (Figure 3.4) appears. In the first case, the Ethernet Address (MAC) is automatically entered. In the latter case, you must manually enter it.

Figure 3.4 New Entry Dialog Box



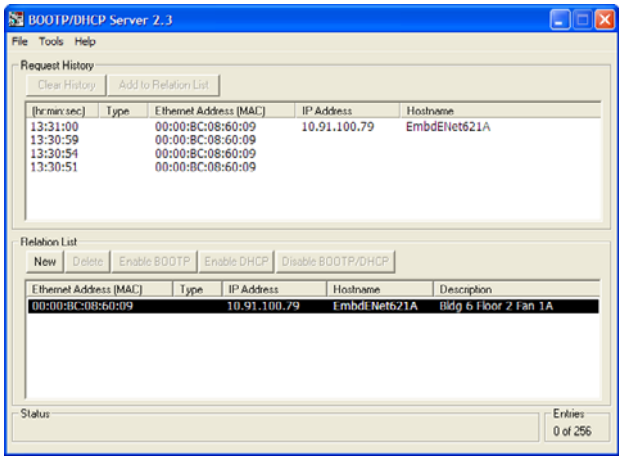
7. Edit the following:

Box	Type
IP Address (1)	A unique IP address for the adapter
Host Name	Optional
Description	Optional

(1) For definitions of these terms, refer to the Glossary.

8. Click **OK** to apply the settings. The adapter appears in the Relation List (Figure 3.5) with the new settings.

Figure 3.5 BOOTP Server Window with Adapter in the Relation List



9. To assign this configuration to the adapter permanently, select the device in the Relation List and click **Disable BOOTP/DHCP**. When power is cycled on the adapter, it will use the configuration you assigned it and not issue new BOOTP requests.

TIP: To enable BOOTP for an embedded adapter that has had BOOTP disabled, first select the adapter in the Relation List, then click **Enable BOOTP**, and finally reset the adapter or power cycle the drive.

10. To save the Relation List, select **File > Save**.

Setting the IP Address, Subnet Mask, and Gateway Address

By default, the adapter is configured to set its IP address, subnet mask, and gateway address using a BOOTP server. If you want to set these attributes using the adapter parameters instead, you must first disable BOOTP and then set these network address parameters in the adapter.

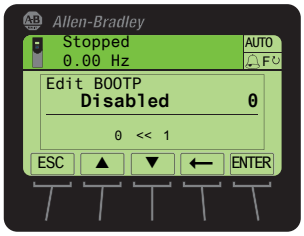
Accessing Parameters in the Adapter

1. Display the Status screen, which is shown on HIM power up.
2. Use the 4 or 6 key to scroll to Port 13, which is the port always dedicated to the embedded EtherNet/IP adapter.
3. Press the PAR# *soft key* to display the Jump to Param # entry pop-up box.
4. Use the numeric keys to enter the desired parameter number, or use the ▲ or ▼ *soft key* to scroll to the desired parameter number.

Disabling the BOOTP Feature

1. Set the value of **Parameter 36 - [BOOTP]** to “0” (Disabled).

Figure 3.6 Edit BOOTP HIM Screen



Value	Setting
0	Disabled
1	Enabled (Default)

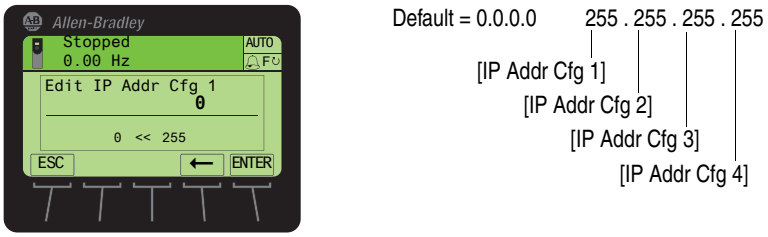
2. Reset the adapter by power cycling the drive or by using the HIM’s Reset Device function located in the drive’s DIAGNOSTIC folder.

   After disabling the BOOTP feature, you can then configure the IP address, subnet mask, and gateway address using adapter parameters.

Setting an IP Address Using Parameters

1. Verify that **Parameter 36 - [BOOTP]** is set to “0” (Disabled). This parameter must be set to Disabled to configure the IP address using the adapter parameters.
2. Set the value of **Parameters 38 - [IP Addr Cfg 1]** through **41 - [IP Addr Cfg 4]** to a unique IP address.

Figure 3.7 Edit IP Addr Cfg 1 HIM Screen



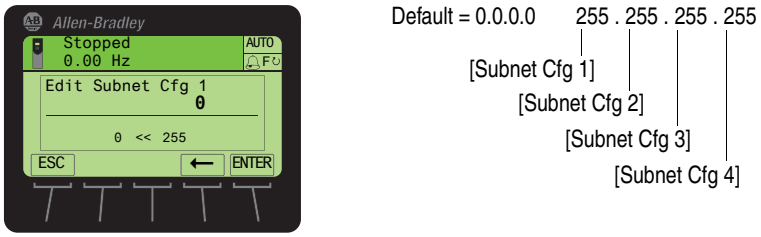
3. Reset the adapter by power cycling the drive or by using the HIM's Reset Device function located in the drive's DIAGNOSTIC folder.

   The ENET status indicator will be solid green or flashing green if the IP address is correctly configured.

Setting a Subnet Mask Using Parameters

1. Verify that **Parameter 36 - [BOOTP]** is set to "0" (Disabled). This parameter must be set to Disabled to configure the subnet mask using the adapter parameters.
2. Set the value of **Parameters 42 - [Subnet Cfg 1]** through **45 - [Subnet Cfg 4]** to the desired value for the subnet mask.

Figure 3.8 Edit Subnet Cfg 1 HIM Screen

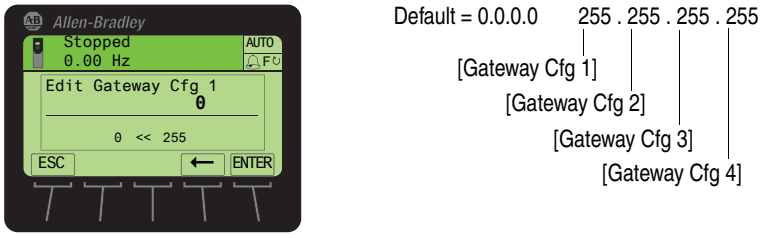


3. Reset the adapter by power cycling the drive or by using the HIM's Reset Device function located in the drive's DIAGNOSTIC folder.

Setting a Gateway Address for the Adapter Using Parameters

1. Verify that **Parameter 36 - [BOOTP]** is set to "0" (Disabled). This parameter must be set to Disabled to configure the gateway address using the adapter parameters.
2. Set the value of **Parameters 46 - [Gateway Cfg 1]** through **49 - [Gateway Cfg 4]** to the IP address of the gateway device.

Figure 3.9 Edit Gateway Cfg 1 HIM Screen



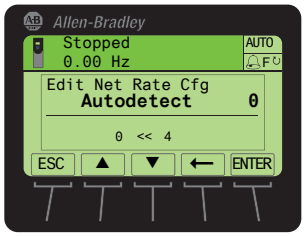
3. Reset the adapter by power cycling the drive or by using the HIM's Reset Device function located in the drive's DIAGNOSTIC folder.

Setting the Data Rate

By default, the adapter is set to autodetect, so it automatically detects the data rate and duplex setting used on the network. If you need to set a specific data rate and duplex setting, the value of **Parameter 50 - [Net Rate Cfg]** determines the Ethernet data rate and duplex setting that the adapter will use to communicate. For definitions of data rate and duplex, refer to the Glossary.

1. Set the value of **Parameter 50 - [Net Rate Cfg]** to the data rate at which your network is operating.

Figure 3.10 Edit Net Rate Cfg HIM Screen



Value	Data Rate
0	Autodetect (default)
1	10Mbps Full
2	10Mbps Half
3	100Mbps Full
4	100Mbps Half

TIP: Auto detection of baud rate and duplex works properly only if the device (usually a switch) on the other end of the cable is also set to automatically detect the baud rate/duplex. If one device has the baud rate/duplex hard-coded, the other device must be hard-coded to the same settings.

2. Reset the adapter by power cycling the drive or by using the HIM's Reset Device function located in the drive's DIAGNOSTIC folder.

Selecting Master-Slave or Peer-to-Peer

A hierarchy determines the type of device with which the adapter exchanges data. In a Master-Slave hierarchy, the adapter exchanges data with a master, such as a bridge or controller. In a Peer-to-Peer hierarchy, the adapter exchanges data with one or more EtherNet/IP adapters in other PowerFlex 750-Series drives. (The drives must have compatible Logic Command/Status words).

For both Master-Slave and Peer-to-Peer hierarchies, the devices exchanging data must be on the same IP subnet. See “IP Addresses” in the Glossary for information about IP subnets.

Setting a Master-Slave Hierarchy

Enabling Datalinks To Write Data

The controller output image (controller outputs-to-drive) can have anywhere from 0 to 16 additional 32-bit parameters (Datalinks). They are configured using **Parameters 01 - [DL From Net 01]** through **16 - [DL From Net 16]**. The number of Datalinks actively used is controlled by the connection size in the controller. See the respective controller example sections in Chapter 4 for more information on setting the connection sizes.

Important: Always use the Datalink parameters in consecutive numerical order, starting with the first parameter. For example, use Parameters 01, 02, and 03 to configure 3 Datalinks to write data. Otherwise, the network I/O connection will be larger than necessary, which needlessly increases controller response time and memory usage.

TIP: When using a ControlLogix controller and an RSLogix 5000 Add-On Drive Profile (v16 or higher), there is no need to configure Datalink parameters at this time. They will be assigned when configuring the RSLogix 5000 Add-On Drive Profile (see Adding the Drive/Adapter to the I/O Configuration on page 4-5).

When using a ControlLogix controller and the Generic Profile, or a PLC-5, SLC 500 or MicroLogix 1100 controller, configure the Datalink parameters now as described in this section.

Parameters 01 - [DL From Net 01] through **16 - [DL From Net 16]** control which parameters in the drive, adapter, or any other connected peripheral receive the values from the network. The enhanced PowerFlex 7-Class HIM, DriveExplorer, and DriveExecutive provide user-friendly screens to help select the drive or peripheral by port number and the parameter by name. As an alternate method, the parameter value can be set manually by number using this formula:

From Net Parameter Value = (10000 * Port Number) + (Destination Parameter Number)

For example, suppose you want to use **Parameter 01 - [DL From Net 01]** to write to Parameter 03 of an optional encoder card plugged into

drive Port 5. Using the formula, the value for **Parameter 01 - [DL From Net 01]** would be (10000 * 5) + (3) = 50003.

1. Set the values of only the required number of contiguous Datalink parameters needed to write data to the drive and that are to be included in the network I/O connection.
2. Reset the adapter by power cycling the drive or by using the HIM's Reset Device function located in the drive's DIAGNOSTIC folder.
3. Since the Logic Command and Reference are always used in the adapter, configure the parameters in the drive to accept the Logic Command and Reference from the adapter. If the controller is going to be used for speed reference via the adapter, set Parameter 545 - [Speed Ref A Sel] in a PowerFlex 755 drive to "Port 13 Reference" (the drive port dedicated to the embedded EtherNet/IP adapter). Also, verify that the mask parameters in the drive (for example, Parameter 324 - [Logic Mask]) are configured to receive the desired logic from the adapter. Refer to the documentation for your drive for details.

The adapter is ready to receive input data from the master (controller). You must now configure the controller to recognize and transmit I/O to the adapter. Refer to Chapter 4, Configuring the I/O.

Enabling Datalinks To Read Data

The controller input image (drive-to-controller inputs) can have anywhere from 0 to 16 additional 32-bit parameters (Datalinks). They are configured using **Parameters 17 - [DL To Net 01]** through **32 - [DL To Net 16]**. The number of Datalinks actively used is controlled by the connection size in the controller. See the respective controller example sections in Chapter 4 for more information on setting the connection sizes.

Important: Always use the Datalink parameters in consecutive numerical order, starting with the first parameter. For example, use Parameters 17, 18, 19, 20, and 21 to configure 5 Datalinks to read data. Otherwise, the network I/O connection will be larger than necessary, which needlessly increases controller response time and memory usage.

TIP: When using a ControlLogix controller and an RSLogix 5000 Add-On Drive Profile (v16 or higher), there is no need to configure Datalink parameters at this time. They will be assigned when configuring the RSLogix 5000 Add-On Drive Profile (see Adding the Drive/Adapter to the I/O Configuration on page 4-5).

When using a ControlLogix controller and the Generic Profile, or a PLC-5, SLC 500 or MicroLogix 1100 controller, configure the Datalink parameters now as described in this section.

Parameters 17 - [DL To Net 01] through **32 - [DL To Net 16]** control which parameters in the drive, adapter, or any other connected peripheral send the values to the network. The enhanced PowerFlex 7-Class HIM, DriveExplorer, and DriveExecutive provide user-friendly screens to help select the drive or peripheral by port number and the parameter by name. As an alternate method, the parameter value can be set manually by number using this formula:

To Net Parameter Value = (10000 * Port Number) + (Origination Parameter Number)

For example, suppose you want to use **Parameter 17 - [DL To Net 01]** to read Parameter 2 of an optional I/O card plugged into drive Port 4. Using the formula, the value for **Parameter 17 - [DL To Net 01]** would be (10000 * 4) + (2) = 40002.

1. Set the values of only the required number of contiguous Datalink parameters needed to read data from the drive and that are to be included in the network I/O connection.

2. Reset the adapter by power cycling the drive or by using the HIM's Reset Device function located in the drive's DIAGNOSTIC folder.

The adapter is ready to send output data to the master (controller). You must now configure the controller to recognize and transmit I/O to the adapter. Refer to Chapter 4, Configuring the I/O.

Setting the Adapter to Transmit Peer-to-Peer Data

Simple Peer I/O Configuration

The most common use of Peer I/O is to take the Logic Command and Reference from one drive and repeat it over Ethernet to one or more other drives. If scaling of the Reference is needed to enable drives to run at different but related speeds, use drive Parameter 609 - [TrmPct RefA Stpt]. The embedded EtherNet/IP adapter provides a simplified configuration method for simple Peer I/O.

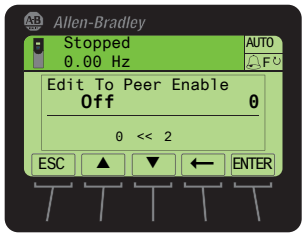
Important: Because of the 32-bit REAL (floating point) Reference, the following method works only if the drives transmitting and receiving are PowerFlex 750-Series drives.

To set up the master (broadcast) side of simple Peer I/O:

1. Set **Parameters 89 - [To Peer Period]** and **90 - [To Peer Skip]** as desired for your application. Parameter 89 controls how frequently the adapter will transmit present data. Parameter 90 controls how many transmit opportunities can be skipped if the data to be transmitted has not changed.

2. Set **Parameter 91 - [To Peer Enable]** to a value of "1" (Cmd/Ref).

Figure 3.11 Edit To Peer Enable HIM Screen



Value	Setting
0	Off (Default)
1	Cmd/Ref
2	Custom

To set up the slave (receiver) side of simple Peer I/O:

1. Set **Parameter 80 - [Fr Peer Timeout]** to a suitable timeout value for your application. This value should be greater than the product of **Parameter 89 - [To Peer Period]** and **Parameter 90 - [To Peer Skip]** in the transmitting drive.
2. Set **Parameters 81 - [Fr Peer Addr 1]** through **84 - [Fr Peer Addr 4]** to the IP address of the drive transmitting Peer I/O.
3. In each PowerFlex 750-Series slave drive, set drive parameter 308 - [Direction Mode] to "1" (Bipolar) to ensure that it properly follows the master drive's speed reference and commanded direction.
4. Set **Parameter 85 - [Fr Peer Enable]** to a value of "1" (Cmd/Ref).

Custom Peer I/O Configuration

Peer I/O also allows more flexibility in sending custom data over the network, but requires more configuration.

Important: Because of the 32-bit REAL (floating point) Reference, the following method works only if the drives transmitting and receiving are PowerFlex 750-Series drives.

To set up the master (broadcast) side of custom Peer I/O:

1. Decide how many Datalink parameters you want to transmit. Set **Parameter 87 - [DLs To Peer Cfg]** to that value.
2. Determine how the Datalinks are allocated. The highest numbered of the 16 Datalinks are allocated to Peer I/O. For example, if **Parameter 87 - [DLs To Peer Cfg]** is set to "3," then Datalinks 14, 15, and 16 are allocated to Peer I/O. To avoid an overlap between Master-Slave and Peer I/O, make sure that **Parameter 35 - [DLs To Net Act]** plus **Parameter 87 - [DLs To Peer Cfg]** does not total more than 16.
3. Set **Parameters 17** through **32 - [DL To Net 01-16]** to the parameters you want to transmit, based on the allocation in Step 2.
4. Reset the adapter by power cycling the drive or by using the HIM's Reset Device function located in the drive's DIAGNOSTIC folder so that changes to **Parameter 87 - [DLs To Peer Cfg]** take effect.

5. Set **Parameters 89 - [To Peer Period]** and **90 - [To Peer Skip]** as required for your application. Parameter 89 controls how frequently the adapter will transmit present data. Parameter 90 controls how many transmit opportunities can be skipped if the data to be transmitted has not changed.

6. Set **Parameter 91 - [To Peer Enable]** to a value of "2" (Custom).

To set up the slave (receiver) side of custom Peer I/O:

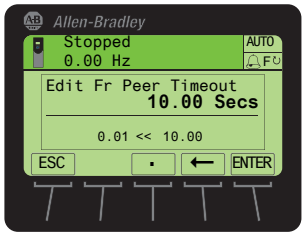
1. Decide how many pieces of data (Logic Command, Reference, and Datalink parameters) you want to receive. This must match the number of parameters transmitted by the master. Set **Parameter 76 - [DLs Fr Peer Cfg]** to that value.

2. Determine how the Datalinks are allocated. The highest numbered of the 16 Datalinks are allocated to Peer I/O. For example, if **Parameter 76 - [DLs Fr Peer Cfg]** is set to "3," Datalinks 14, 15, and 16 are allocated to Peer I/O. To avoid an overlap between Master-Slave and Peer I/O, make sure that **Parameter 34 - [DLs From Net Act]** plus **Parameter 76 - [DLs Fr Peer Cfg]** does not total more than 16.

3. Set **Parameters 1** through **16 - [DL From Net 01-16]** to the parameters you want to receive, based on the allocation in Step 2.

4. Set **Parameter 80 - [Fr Peer Timeout]** to a suitable timeout value for your application. This value should be greater than the product of **Parameter 89 - [To Peer Period]** and **Parameter 90 - [To Peer Skip]** in the transmitting drive.

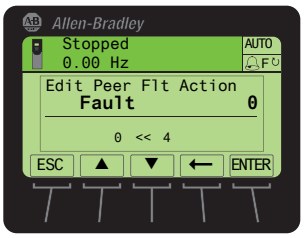
Figure 3.12 Edit Fr Peer Timeout HIM Screen



5. Set **Parameter 56 - [Peer Flt Action]** to the desired action if Peer I/O data is not received before the timeout is reached.

ATTENTION: Risk of injury or equipment damage exists. **Parameter 56 - [Peer Flt Action]** lets you determine the action of the adapter and connected drive if the adapter is unable to communicate with the designated peer. By default, this parameter faults the drive. You can set this parameter so that the drive continues to run. Precautions should be taken to ensure that the setting of this parameter does not create a hazard of injury or equipment damage. When commissioning the drive, verify that your system responds correctly to various situations (for example, a disconnected cable).

Figure 3.13 Edit Peer Flt Action HIM Screen

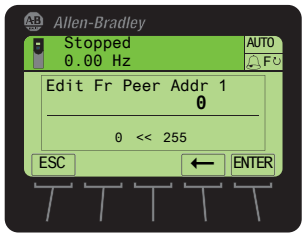


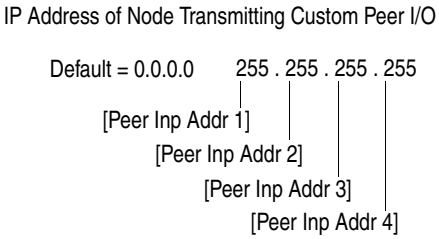
Value	Description
0	Fault (Default)
1	Stop
2	Zero Data
3	Hold Last
4	Send Flt Cfg

For more details about fault action, see Setting a Fault Action on page 3-14.

6. Set **Parameters 81 - [Fr Peer Addr 1]** through **84 - [Fr Peer Addr 4]** to the IP address of the drive transmitting the custom Peer I/O.

Figure 3.14 Edit Fr Peer Addr 1 Screen on the HIM





7. If a Logic Command is being sent, use **Parameter 78 - [Logic Src Cfg]** to set the Datalink number containing the Logic Command. Otherwise, set Parameter 78 to a value of "0." For bit definitions, refer to Appendix D or the drive documentation.

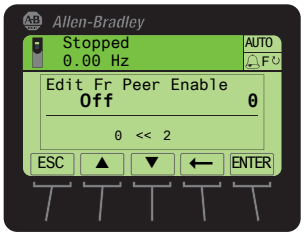
8. If a Reference is being sent, use **Parameter 79 - [Ref Src Cfg]** to set the Datalink number containing the Reference. Otherwise, set Parameter 79 to a value of "0."

9. In each PowerFlex 750-Series slave drive, set drive parameter 308 - [Direction Mode] to "1" (Bipolar) to ensure that it properly follows the master drive's speed reference and commanded direction.

10. Reset the adapter by power cycling the drive or by using the HIM's Reset Device function located in the drive's DIAGNOSTIC folder so that changes to **Parameter 76 - [DLs Fr Peer Cfg]** take effect.

11. Set **Parameter 85 - [Fr Peer Enable]** to a value of "2" (Custom).

Figure 3.15 Edit Fr Peer Enable HIM Screen



Value	Setting
0	Off (Default)
1	Cmd/Ref
2	Custom

Setting a Fault Action

By default, when I/O communications are disrupted (for example, the network cable is disconnected), the controller is idle (in program mode or faulted), and/or Peer I/O or explicit messaging for drive control is disrupted, the drive responds by faulting. You can configure a different response to:

- Disrupted I/O communication by using **Parameter 54 - [Comm Flt Action]**.
- An idle controller by using **Parameter 55 - [Idle Flt Action]**.
- Disrupted Peer I/O by using **Parameter 56 - [Peer Flt Action]**.
- Disrupted explicit messaging for drive control via PCCC and the CIP Register Object by using **Parameter 57 - [Msg Flt Action]**.

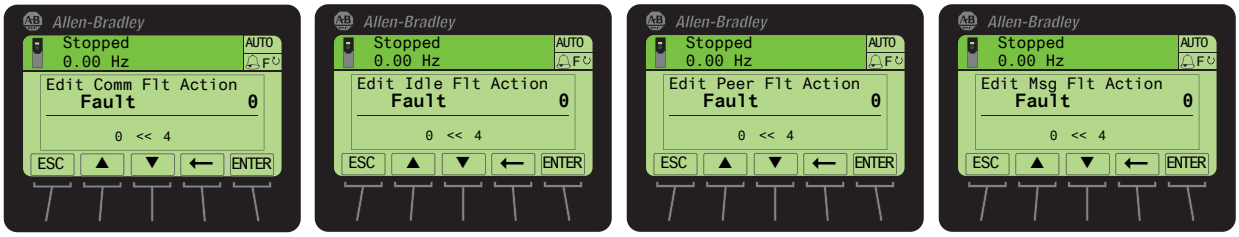
ATTENTION: Risk of injury or equipment damage exists. Embedded adapter **Parameters 54 - [Comm Flt Action]**, **55 - [Idle Flt Action]**, **56 - [Peer Flt Action]**, and **57 - [Msg Flt Action]** respectively let you determine the action of the adapter and drive if I/O communication is disrupted, the controller is idle, Peer I/O is disrupted, or explicit messaging for drive control is disrupted. By default, these parameters fault the drive. You can set these parameters so that the drive continues to run. Precautions should be taken to ensure that the settings of these parameters do not create a risk of injury or equipment damage. When commissioning the drive, verify that your system responds correctly to various situations (for example, a disconnected network cable, controller in idle state or explicit message control disruption).

Changing the Fault Action

Set the values of **Parameters 54 - [Comm Flt Action]**, **55 - [Idle Flt Action]**, **56 - [Peer Flt Action]**, and **57 - [Msg Flt Action]** to the desired responses:

Value	Action	Description
0	Fault	The drive is faulted and stopped. (Default)
1	Stop	The drive is stopped, but not faulted.
2	Zero Data	The drive is sent 0 for output data. This does not command a stop.
3	Hold Last	The drive continues in its present state.
4	Send Flt Cfg	The drive is sent the data that you set in the fault configuration parameters (**Parameters 58 - [Flt Cfg Logic]**, **59 - [Flt Cfg Ref]**, and **60 - [Flt Cfg DL 01]** through **75 - [Flt Cfg DL 16]**).

Figure 3.16 Edit Fault Action HIM Screens



Changes to these parameters take effect immediately. A reset is not required.

Setting the Fault Configuration Parameters

If you set **Parameter 54 - [Comm Flt Action]**, **55 - [Idle Flt Action]**, **56 - [Peer Flt Action]** or **Parameter 57 - [Msg Flt Action]** to "Send Flt Cfg," the values in the following parameters are sent to the drive after an I/O communications fault, idle fault, Peer I/O fault, and/or explicit messaging for drive control fault occurs. You must set these parameters to values required by your application.

Parameter	Description
Parameter 58 - [Flt Cfg Logic]	A 32-bit value sent to the drive for Logic Command.
Parameter 59 - [Flt Cfg Ref]	A 32-bit REAL (floating point) value sent to the drive for Reference.
Parameter 60 - [Flt Cfg DL 01] through **Parameter 75 - [Flt Cfg DL 16]**	A 32-bit integer value sent to the drive for a Datalink. If the destination of the Datalink is a REAL (floating point) parameter, you must convert the desired value to the binary representation of the REAL value. (An internet search of "hex to float" provides a link to a tool to do this conversion.)

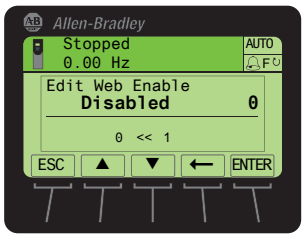
Changes to these parameters take effect immediately. A reset is not required.

Setting Web Access Control

By using a web browser to access the IP address set for the adapter, you can view the adapter's web pages for information about the adapter, the drive, and other DPI devices connected to the drive, such as HIMs or converters. Additionally, the adapter can be configured to automatically send e-mail messages to desired addresses when selected drive faults occur and/or are cleared, and/or when the adapter takes a communication or idle fault action. For more details on the adapter's web pages, refer to Chapter 8, Viewing the Adapter Web Pages.

By default, the adapter web pages are disabled. To enable the adapter web pages, set **Parameter 52 - [Web Enable]** to "1" (Enabled) and then reset the adapter for the change to take effect.

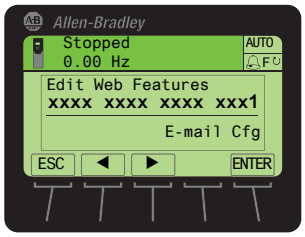
Figure 3.17 Edit Web Enable HIM Screen



Value	Description
0	Disabled (Default)
1	Enabled

Bit 0 of **Parameter 53 - [Web Features]** is used to protect the configured settings for e-mail messaging. By default, settings are not protected and the user can make changes. To protect the configured settings, set the value of E-mail Cfg Bit 0 to "0" (Disabled). You can unprotect the configuration by changing Bit 0 back to "1" (Enabled). E-mail messaging will always remain active regardless of whether or not its settings are protected—unless e-mail messaging was *never* configured. For more information about configuring adapter e-mail messaging or to stop e-mail messages, refer to Configure E-mail Notification Web Page on page 8-6.

Figure 3.18 Edit Web Features HIM Screen



Bit	Description
0	E-mail Cfg (Default = 1 = Enabled)
1 - 15	Not Used

Bit 0 is the right-most bit. In Figure 3.18 it equals "1" (Enabled).

Changes to this parameter take effect immediately. A reset is not required.

Resetting the Adapter

Changes to switch settings and some adapter parameters require that you reset the adapter before the new settings take effect. You can reset the adapter by cycling power to the drive or by using the HIM's Reset Device function located in the drive's DIAGNOSTIC folder.

ATTENTION: Risk of injury or equipment damage exists. If the adapter is transmitting control I/O to the drive, the drive may fault when you reset the adapter. Determine how your drive will respond before resetting the adapter.

Restoring Adapter Parameters to Factory Defaults

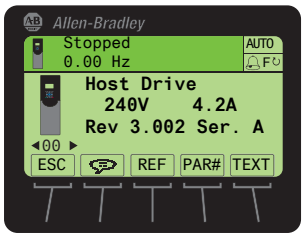
Adapter parameters can be restored in two ways:

- ALL—restores ALL adapter parameters to their factory default values.
- MOST—restores MOST adapter parameters—except the following which are used for network setup:
  - **Parameter 36 - [BOOTP]**
  - **Parameters 38** through **41 - [IP Addr Cfg 1-4]**
  - **Parameters 42** through **45 - [Subnet Cfg 1-4]**
  - **Parameters 46** through **49 - [Gateway Cfg 1-4]**
  - **Parameter 50 - [Net Rate Cfg]**

To restore adapter parameters to their factory default values:

1. Access the Status screen, which is displayed on HIM power up.

Figure 3.19 Status Screen



2. Use the 4 ◂ or ▸ 6 key to scroll to Port 13, which is the port always dedicated to the embedded EtherNet/IP adapter.
3. Press the folder key to display its last-viewed folder.
4. Use the 4 ◂ or ▸ 6 key to scroll to the MEMORY folder.
5. Use the 2 ▾ or 8 ▴ key to select **Set Defaults**.
6. Press the 5 (Enter) key to display the Set Defaults pop-up box.
7. Press the 5 (Enter) key again to display the warning pop-up box to reset parameters to their factory default values.

8. Press the MOST *soft key* to restore MOST parameters to factory defaults or press the ALL *soft key* to restore ALL parameters. Or press the ESC *soft key* to cancel.

   Important: When performing a Set Defaults, the drive may detect a conflict and then not allow this function to occur. If this happens, first resolve the conflict and then repeat this Set Defaults procedure. Common reasons for a conflict include the drive running or a controller in Run mode.

9. Reset the adapter by cycling power to the drive or by using the HIM's Reset Device function located in the drive's DIAGNOSTIC folder.

Viewing the Adapter Status Using Parameters

The following parameters provide information about the status of the adapter. You can view these parameters at any time.

Parameter	Description
34 - [DLs From Net Act]	The number of controller-to-drive Datalinks that are included in the network I/O connection (controller outputs).
35 - [DLs To Net Act]	The number of drive-to-controller Datalinks that are included in the network I/O connection (controller inputs).
37 - [Net Addr Src]	Source from which the adapter node address is taken. This will be either "0" (Switches), "1" (Parameters) which uses the address from **Parameters 38-41 [IP Addr Cfg x]**, or "2" (BOOTP). The source is determined by the settings of the IP address switches (Figure 2.1) on the adapter.
50 - [Net Rate Act]	The data rate used by the adapter.
86 - [Fr Peer Status]	The status of the consumed peer input connection. Values: 0 = Off; 1 = Waiting; 2 = Running; 3 = Faulted
77 - [DLs Fr Peer Act]	The number of peer-to-drive Datalinks that the drive is expecting.
88 - [DLs To Peer Act]	The number of drive-to-peer Datalinks that the drive is expecting.

Flash Updating the Adapter

The adapter can be flash updated over the network or serially through a direct connection from a computer to the drive using a 1203-USB or 1203-SSS serial converter.

When flashing over the network, you can use the Allen-Bradley software tool ControlFLASH, the built-in flash capability of DriveExplorer Lite or Full, or the built-in flash capability of DriveExecutive.

When flashing through a direct serial connection from a computer to a drive, you can use the same Allen-Bradley software tools described above, or you can use HyperTerminal set to the X-modem protocol.

To obtain a flash update for this adapter, go to http://www.ab.com/support/abdrives/webupdate. This site contains all firmware update files and associated Release Notes that describe firmware update enhancements/anomalies, how to determine the existing firmware version, and how to flash update using DriveExplorer, DriveExecutive or ControlFLASH.

Configuring the I/O

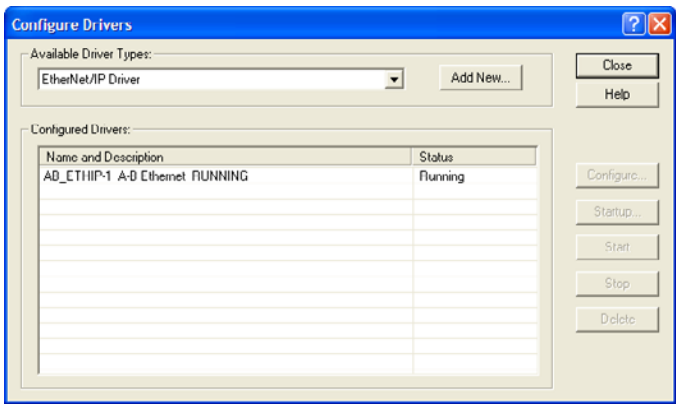
This chapter provides instructions on how to configure a Rockwell Automation controller (ControlLogix, PLC-5, SLC 500 or MicroLogix 1100) to communicate with the adapter and PowerFlex drive.

Using RSLinx Classic

RSLinx Classic, in all its variations (Lite, Gateway, OEM, etc.), is used to provide a communication link between the computer, network, and controller. RSLinx Classic requires its network-specific driver to be configured before communications are established with network devices. To configure the RSLinx driver:

1. Start RSLinx and select **Communications > Configure Drivers** to display the Configure Drivers screen.
2. In the Available Driver Types box, select "EtherNet/IP Driver" and then click **Add New** to display the EtherNet/IP Driver Selection screen.
3. When the Add New RSLinx Driver screen appears, type a name (if desired) and click **OK** to display the "Configure driver:" screen.
4. Depending on your application, select either the browse local or remote subnet option, and click **OK**. The Configure Drivers screen reappears with the new driver in the Configured Drivers list (Figure 4.1).

Figure 4.1 Configure Drivers Screen with a Configured Driver



5. Click **Close** to close the Configure Drivers screen. Leave RSLinx running.

6. Verify that your computer recognizes the drive. Select **Communications > RSWho** and, in the menu tree, click the "+" symbol next to the Ethernet driver.

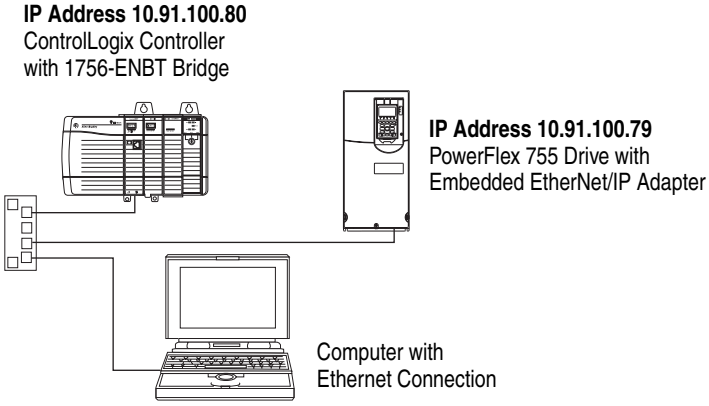
   Note that two other RSLinx drivers (Ethernet devices or Remote Devices via Linx Gateway) may be used. Use one of these drivers if the "EtherNet/IP Driver" cannot see your drive.

ControlLogix Example

Example Network

After the adapter is configured, the drive and adapter will be a single node on the network. This section provides the steps needed to configure a simple EtherNet/IP network (see Figure 4.2). In our example, we will configure a 1756-ENBT (Series A) bridge to communicate with a drive using Logic Command/Status, Reference/Feedback, and 32 Datalinks (16 to read/16 to write) over the network.

Figure 4.2 Example ControlLogix EtherNet/IP Network

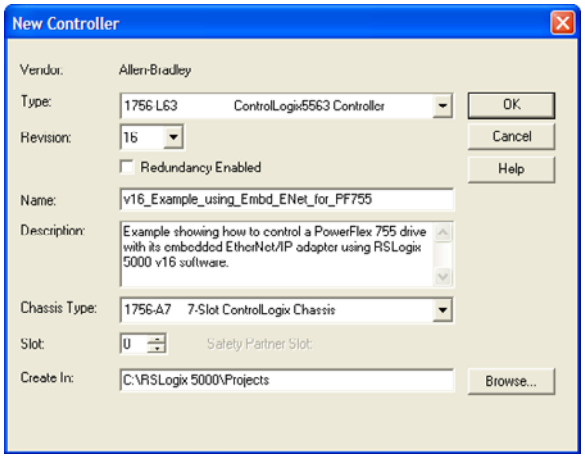


Adding the Bridge to the I/O Configuration

To establish communications between the controller and adapter over the network, you must first add the ControlLogix controller and its bridge to the I/O configuration.

1. Start RSLogix 5000. The RSLogix 5000 window appears. Select **File > New** to display the New Controller screen (Figure 4.3).

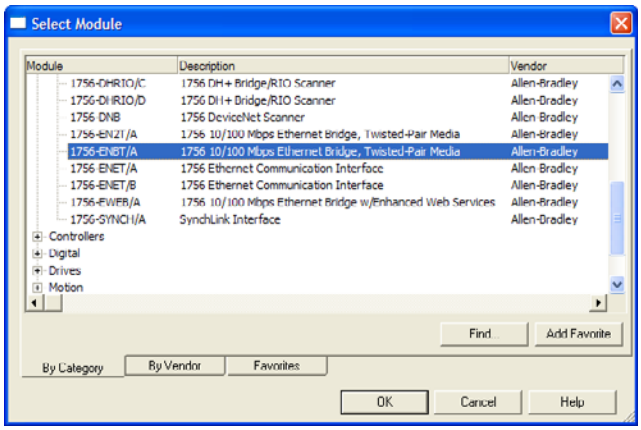
Figure 4.3 New Controller Screen (RSLogix 5000 v16 shown)



Select the appropriate choices for the fields in the screen to match your application. Then click **OK**. The RSLogix 5000 window reappears with the treeview in the left pane.

2. In the treeview, right-click the I/O Configuration folder and select **New Module…** The Select Module screen appears. Expand the Communications group to display all of the available communication modules (Figure 4.4).

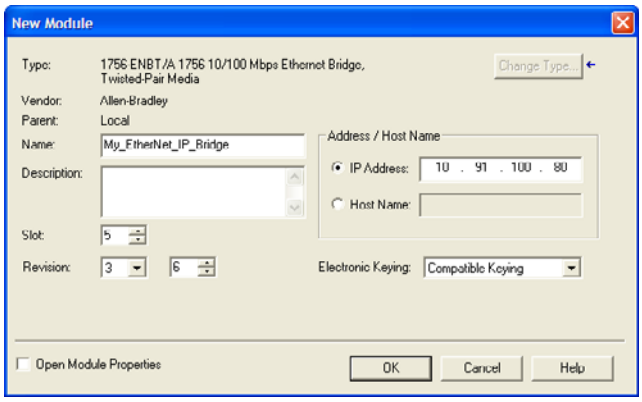
Figure 4.4 Bridge Select Module Screen



3. In the list, select the EtherNet/IP bridge used by your controller. In this example, we use a 1756-ENBT EtherNet/IP Bridge (Series A), so the 1756-ENBT/A option is selected. Then click **OK**. In the Select Major Revision pop-up dialog box, select the major revision of its firmware.

4. Click **OK**. The bridge's New Module screen (Figure 4.5) appears.

Figure 4.5 Bridge New Module Screen

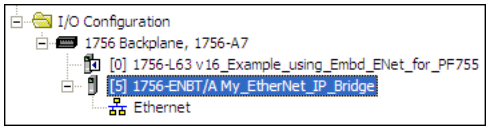


5. Edit the following:

Box	Setting
Name	A name to identify the bridge.
Description	Optional – description of the bridge.
IP Address	The IP address of the EtherNet/IP bridge.
Host Name	Not used.
Slot	The slot of the EtherNet/IP bridge in the rack.
Revision	The minor revision of the firmware in the bridge. (You already set the major revision by selecting the bridge series in Step 3.)
Electronic Keying	**Compatible Module**. The "Compatible Module" setting for Electronic Keying ensures the physical module is consistent with the software configuration before the controller and bridge make a connection. Therefore, ensure that you have set the correct revision in this screen. Refer to the online Help for additional information on this and other Electronic Keying settings.
Open Module Properties	When this box is checked, additional module properties screens will appear to further configure the bridge after clicking **OK**. When unchecked, the bridge's New Module screen will close after clicking **OK**. For this example, uncheck this box.

6. Click **OK**. The bridge is now configured for the EtherNet/IP network. It appears in the I/O Configuration folder. In our example, a 1756-ENBT bridge appears under the I/O Configuration folder (Figure 4.6) with its assigned name.

Figure 4.6 RSLogix 5000: I/O Configuration Folder



There are two ways to add the adapter into the I/O configuration:

- Add-On Drive Profiles (RSLogix 5000 version 16 or higher)
- Generic Profile (RSLogix 5000 all versions)

These are described in the following separate sections. If your version of RSLogix 5000 supports Add-On Drive Profiles, we highly recommend using this method.

Using RSLogix 5000 Add-On Drive Profiles (v16 or Higher)

When compared to using the Generic Profile (all versions), the RSLogix 5000 Add-On Drive Profiles provide these advantages:

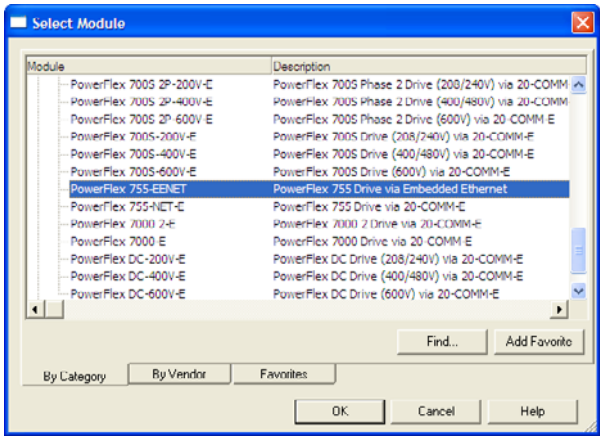
- Profiles for specific drives that provide descriptive controller tags for basic control I/O words (Logic Command/Status and Reference/Feedback) and Datalinks. Additionally, Datalinks automatically take the name of the drive parameter to which they are assigned. These profiles virtually eliminate I/O mismatch errors and substantially reduce drive configuration time.
- New Drive tab eliminates the need for a separate drive software configuration tool.
- Drive configuration settings are saved as part of the RSLogix 5000 v16 project file (.ACD) and also downloaded to the controller.

Adding the Drive/Adapter to the I/O Configuration

To transmit data between the bridge and the drive, you must add the drive as a child device to the parent bridge.

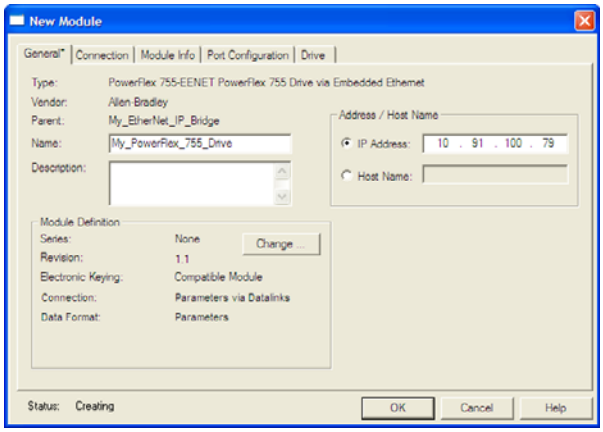
1. In the treeview, right-click on the bridge and select **New Module…** to display the Select Module screen. In our example, we right-click on the 1756-ENBT/A bridge. Expand the Drives group to display all of the available drives with their communication adapters.

Figure 4.7 Drive Select Module Screen



2. From the list (Figure 4.7), select the drive and its connected adapter. For this example, we selected "PowerFlex 755-EENET." Then click **OK**. The drive's New Module screen (Figure 4.8) appears.

Figure 4.8 Drive New Module Screen

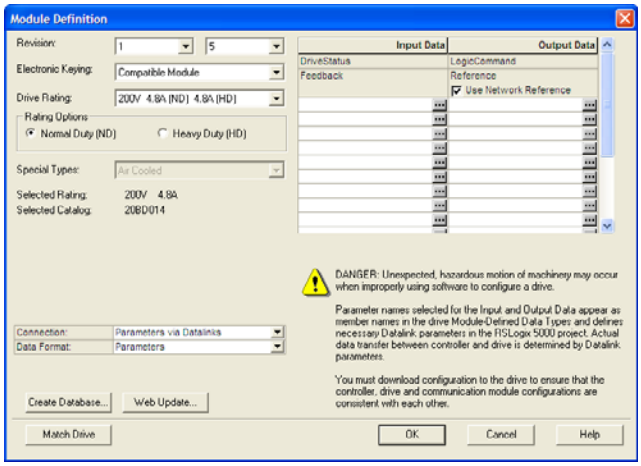


3. On the General tab, edit the following data about the drive/adapter:

Box	Setting
Name	A name to identify the drive.
Description	Optional – description of the drive/adapter.
IP Address	The IP address of the adapter.

4. In the Module Definition section, click **Change…** to launch the Module Definition screen (Figure 4.9) and begin the drive/adapter configuration process.

Figure 4.9 Module Definition Screen



5. In the Module Definition screen, edit the following information:

Box	Setting
Revision	The major and minor revision of the firmware in the drive. If the drive's major and minor revision is not available, the drive database is not installed on your computer. To get the correct database, click either the **Create Database…**, **Web Update…**, or **Match Drive** button on the bottom left of this screen. See the information following this table for descriptions of each button.

Box	Setting
Electronic Keying	**Compatible Module**. The “Compatible Module” setting for Electronic Keying ensures the physical module is consistent with the software configuration before the controller and bridge make a connection. Therefore, ensure that you have set the correct revision in this screen. Refer to the online Help for additional information on this and other Electronic Keying settings. If keying is not required, select “Disable Keying.”
Drive Rating	The voltage and current rating of the drive. If the drive rating is not listed, the drive database is not installed on your computer. To get the correct database, click either the **Create Database…**, **Web Update…**, or **Match Drive** button on the bottom left of this screen. See the information following this table for descriptions of each button.
Rating Options	Selects the drive power output required for the application.
Special Types	Reserved for future use.
Connection	**Parameters via Datalinks.** When selecting “Parameters via Datalinks” (default), the controller tags for the Datalinks use the drive parameter names to which they are assigned. When selecting “Datalinks,” the controller tags for the Datalinks have non-descriptive UserDefinedData[n] names like those used in RSLogix 5000 v15.
Data Format	**Parameters.** When the Connection field is set to “Parameters via Datalinks,” “Parameters” is automatically selected. When the Connection field is set to “Datalinks,” you must select the number of Datalinks required for your application in the “Data Format” field.
Input Data	Assigns selected drive or connected peripheral parameters to be READ by the controller using DL To Net Datalinks. See Steps 5A through 5E below for details.
Output Data	Assigns selected drive or connected peripheral parameters to be WRITTEN to the controller using DL From Net Datalinks. See Steps 5A through 5E below for details.
Use Network Reference Box	Conveniently selects the speed reference for the drive to come from the network. This box is checked by default.

Three buttons at the bottom left of the Module Definition screen are provided for your convenience to perform specific functions:

- **Create Database**: Clicking this button enables you to create a database from an online network device. Thereafter, the database appears in the list for selection on the Module Definition screen.
- **Web Update**: Clicking this button opens the Allen-Bradley Web Updates web site for downloading product-specific firmware update files.
- **Match Drive**: This button is useful when the drive being added to the network matches the drive profile (revision, rating, Datalinks, configuration settings, etc.) of an existing online network drive. Clicking this button conveniently creates a duplicate drive profile from the online drive, and automatically loads this identical information into the Module Definition screen. This eliminates the need to manually enter the information each time a new drive with a matching profile is added to the network.

On the Module Definition screen, notice that the Drive Status, Feedback, Logic Command, and Reference are always used.

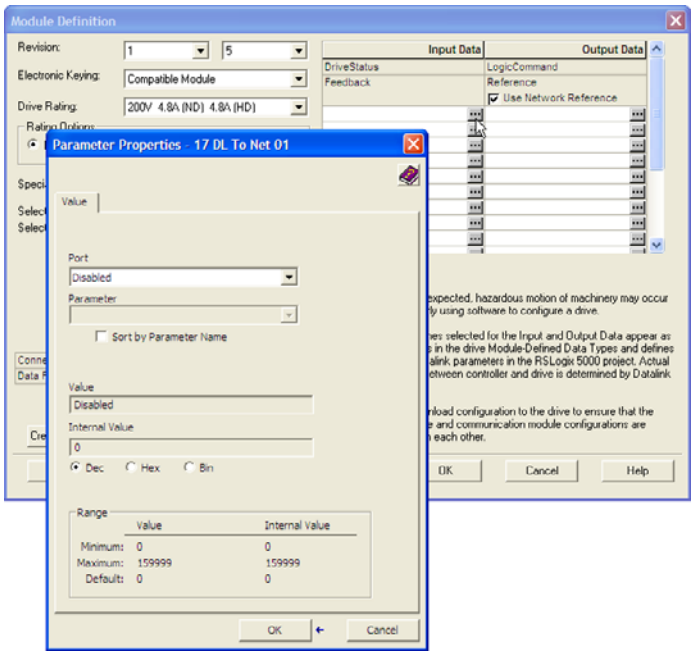
However, when using Datalinks you must still assign adapter **Parameters 01-16 - [DL From Net 01-16]** and **Parameters 17-32**

- [DL To Net 01-16] to point to the appropriate drive or connected peripheral parameters. The procedure to configure the Datalinks on the Module Definition screen for the Input Data and Output Data is the same:

A. Click the ... button in the topmost blank row to display the Parameter Properties screen for the corresponding Datalink.

Important: Always use the Datalink parameters in consecutive numerical order, starting with the first parameter. (For example, use Parameters 01, 02, and 03 to configure 3 Datalinks to write data and/or Parameters 17, 18, 19, 20, and 21 to configure 5 Datalinks to read data.) Otherwise, the network I/O connection will be larger than necessary, which needlessly increases controller response time and memory usage.

Figure 4.10 Example Datalink Parameter Properties Screen



B. In the Port field, click the pull down list and select the port of the device to which this Datalink will be assigned (for this example, Port 0, the PowerFlex 755 drive).

C. In the Parameter field, click the pull down list for the selected device and select the parameter to which this Datalink will point (for this example, drive parameter 370 - [Stop Mode A]).

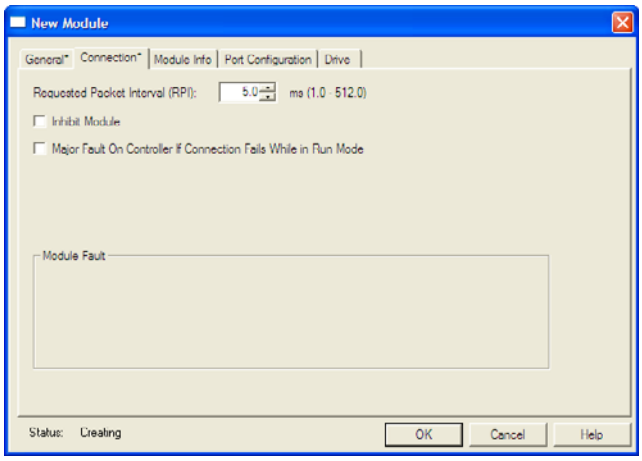
D. Click **OK** to complete configuration of the Datalink. The name of the parameter that this Datalink points to is now shown in the row on the Module Definition screen.

E. Repeat Steps 5A through 5D for each Datalink being configured.

6. Click **OK** on the Module Definition screen to save the drive and adapter configuration and close the screen. The drive's New Module screen reappears.

7. On the New Module screen, click the Connection tab (Figure 4.11).

Figure 4.11 Connection Screen

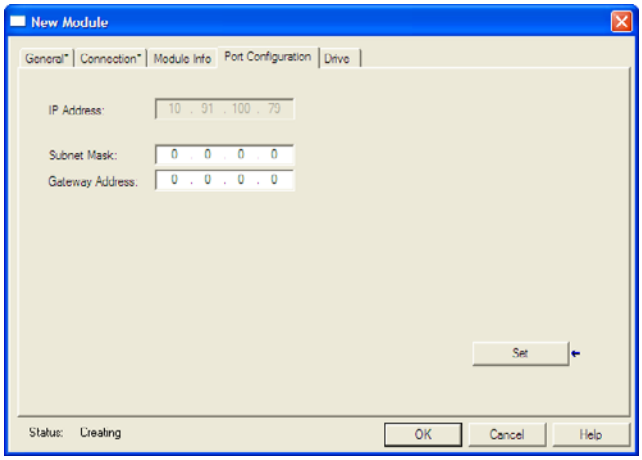


8. In the "Requested Packet Interval (RPI)" box, set the value to 2.0 milliseconds or greater. This value determines the maximum interval that a controller should use to move data to and from the adapter. To conserve bandwidth, use higher values for communicating with low priority devices.

   The "Inhibit Module" box, when checked, inhibits the module from communicating with the RSLogix 5000 project. When the "Major Fault on …" box is checked, a major controller fault will occur when the module's connection fails while the controller is in the Run Mode. For this example, leave the "Inhibit Module" and "Major Fault On …" boxes unchecked.

9. On the New Module screen, click the Port Configuration tab (Figure 4.12).

Figure 4.12 Port Configuration Screen



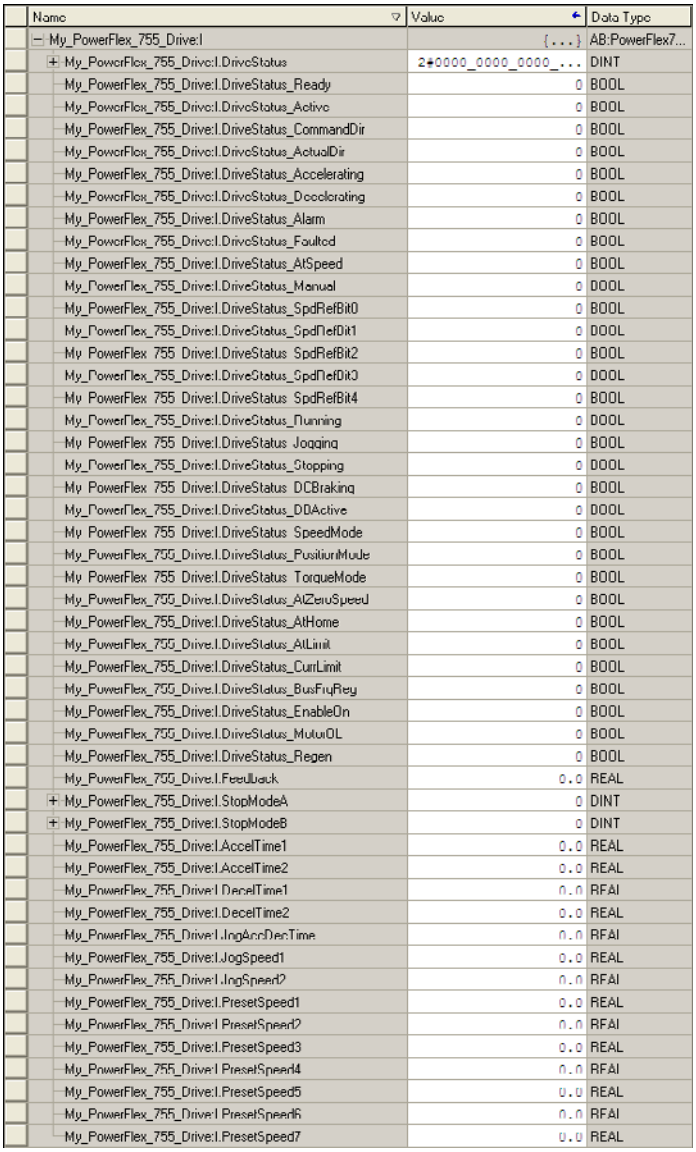
10. In the Port Configuration screen, edit the following information:

Box	Setting
IP Address	The IP address of the adapter that was already set in the General tab. This field is not configurable (grayed out).
Subnet Mask	The Subnet Mask configuration setting of the network. This setting must match the setting of other devices on the network (for example, 255.255.255.0).
Gateway Address	The Gateway Address configuration setting of the network. This setting must match the setting of other devices on the network (for example, 10.91.100.1).
Enable BootP	When this box is checked, BOOTP is enabled in the adapter and will ignore the IP address set in the General tab. When unchecked, the controller uses the set IP address. This is another method to enable/ disable BOOTP in the adapter. For this example, leave this box unchecked.

11. Click **Set** to save the Port Configuration information which sets the corresponding offline Subnet Cfg x and Gateway Cfg x parameters in the adapter.

12. Click **OK** on the New Module screen. The new node ("My_PowerFlex_755_Drive" in this example) now appears under the bridge ("My_EtherNet_IP_Bridge" in this example) in the I/O Configuration folder. If you double-click on the Controller Tags, you will see that module-defined data types and tags have been automatically created (Figure 4.13 and Figure 4.14). Note that all tag names are defined and Datalinks include the assigned drive parameter name. After you save and download the configuration, these tags allow you to access the Input and Output data of the drive via the controller's ladder logic.

Figure 4.13 Controller Input Tags



Name	Value	Data Type
My_PowerFlex_755_Drive:I	{...}	AB:PowerFlex7...
My_PowerFlex_755_Drive:I.DriveStatus	2#0000_0000_0000_...	DINT
My_PowerFlex_755_Drive:I.DriveStatus_Ready	0	BOOL
My_PowerFlex_755_Drive:I.DriveStatus_Active	0	BOOL
My_PowerFlex_755_Drive:I.DriveStatus_CommandDir	0	BOOL
My_PowerFlex_755_Drive:I.DriveStatus_ActualDir	0	BOOL
My_PowerFlex_755_Drive:I.DriveStatus_Accelerating	0	BOOL
My_PowerFlex_755_Drive:I.DriveStatus_Decelerating	0	BOOL
My_PowerFlex_755_Drive:I.DriveStatus_Alarm	0	BOOL
My_PowerFlex_755_Drive:I.DriveStatus_Faulted	0	BOOL
My_PowerFlex_755_Drive:I.DriveStatus_AtSpeed	0	BOOL
My_PowerFlex_755_Drive:I.DriveStatus_Manual	0	BOOL
My_PowerFlex_755_Drive:I.DriveStatus_SpdRefBit0	0	BOOL
My_PowerFlex_755_Drive:I.DriveStatus_SpdRefBit1	0	BOOL
My_PowerFlex_755_Drive:I.DriveStatus_SpdRefBit2	0	BOOL
My_PowerFlex_755_Drive:I.DriveStatus_SpdRefBit3	0	BOOL
My_PowerFlex_755_Drive:I.DriveStatus_SpdRefBit4	0	BOOL
My_PowerFlex_755_Drive:I.DriveStatus_Running	0	BOOL
My_PowerFlex_755_Drive:I.DriveStatus_Jogging	0	BOOL
My_PowerFlex_755_Drive:I.DriveStatus_Stopping	0	BOOL
My_PowerFlex_755_Drive:I.DriveStatus_DCBraking	0	BOOL
My_PowerFlex_755_Drive:I.DriveStatus_DDActive	0	BOOL
My_PowerFlex_755_Drive:I.DriveStatus_SpeedMode	0	BOOL
My_PowerFlex_755_Drive:I.DriveStatus_PositionMode	0	BOOL
My_PowerFlex_755_Drive:I.DriveStatus_TorqueMode	0	BOOL
My_PowerFlex_755_Drive:I.DriveStatus_AtZeroSpeed	0	BOOL
My_PowerFlex_755_Drive:I.DriveStatus_AtHome	0	BOOL
My_PowerFlex_755_Drive:I.DriveStatus_AtLimit	0	BOOL
My_PowerFlex_755_Drive:I.DriveStatus_CurrLimit	0	BOOL
My_PowerFlex_755_Drive:I.DriveStatus_BusFrqReg	0	BOOL
My_PowerFlex_755_Drive:I.DriveStatus_EnableOn	0	BOOL
My_PowerFlex_755_Drive:I.DriveStatus_MotorOL	0	BOOL
My_PowerFlex_755_Drive:I.DriveStatus_Regen	0	BOOL
My_PowerFlex_755_Drive:I.FeedBack	0.0	REAL
My_PowerFlex_755_Drive:I.StopModeA	0	DINT
My_PowerFlex_755_Drive:I.StopModeB	0	DINT
My_PowerFlex_755_Drive:I.AccelTime1	0.0	REAL
My_PowerFlex_755_Drive:I.AccelTime2	0.0	REAL
My_PowerFlex_755_Drive:I.DecelTime1	0.0	REAL
My_PowerFlex_755_Drive:I.DecelTime2	0.0	REAL
My_PowerFlex_755_Drive:I.JogAccDecTime	0.0	REAL
My_PowerFlex_755_Drive:I.JogSpeed1	0.0	REAL
My_PowerFlex_755_Drive:I.JogSpeed2	0.0	REAL
My_PowerFlex_755_Drive:I.PresetSpeed1	0.0	REAL
My_PowerFlex_755_Drive:I.PresetSpeed2	0.0	REAL
My_PowerFlex_755_Drive:I.PresetSpeed3	0.0	REAL
My_PowerFlex_755_Drive:I.PresetSpeed4	0.0	REAL
My_PowerFlex_755_Drive:I.PresetSpeed5	0.0	REAL
My_PowerFlex_755_Drive:I.PresetSpeed6	0.0	REAL
My_PowerFlex_755_Drive:I.PresetSpeed7	0.0	REAL

Figure 4.14 Controller Output Tags

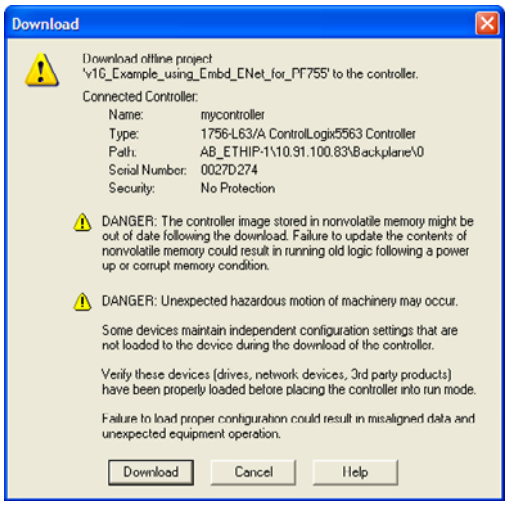
Name	Value	Data Type
My_PowerFlex_755_Drive:O	{...}	AB:PowerFlex7...
My_PowerFlex_755_Drive:O.LogicCommand	2#0000_0000_0000_...	DINT
My_PowerFlex_755_Drive:O.LogicCommand_Stop	0	BOOL
My_PowerFlex_755_Drive:O.LogicCommand_Start	0	BOOL
My_PowerFlex_755_Drive:O.LogicCommand_Jog1	0	BOOL
My_PowerFlex_755_Drive:O.LogicCommand_ClearFaults	0	BOOL
My_PowerFlex_755_Drive:O.LogicCommand_Forward	0	BOOL
My_PowerFlex_755_Drive:O.LogicCommand_Reverse	0	BOOL
My_PowerFlex_755_Drive:O.LogicCommand_Manual	0	BOOL
My_PowerFlex_755_Drive:O.LogicCommand_AccelTime1	0	BOOL
My_PowerFlex_755_Drive:O.LogicCommand_AccelTime2	0	BOOL
My_PowerFlex_755_Drive:O.LogicCommand_DecelTime1	0	BOOL
My_PowerFlex_755_Drive:O.LogicCommand_DecelTime2	0	BOOL
My_PowerFlex_755_Drive:O.LogicCommand_SpdRefSel0	0	BOOL
My_PowerFlex_755_Drive:O.LogicCommand_SpdRefSel1	0	BOOL
My_PowerFlex_755_Drive:O.LogicCommand_SpdRefSel2	0	BOOL
My_PowerFlex_755_Drive:O.LogicCommand_CoastStop	0	BOOL
My_PowerFlex_755_Drive:O.LogicCommand_CLimitStop	0	BOOL
My_PowerFlex_755_Drive:O.LogicCommand_Run	0	BOOL
My_PowerFlex_755_Drive:O.LogicCommand_Jog2	0	BOOL
My_PowerFlex_755_Drive:O.Reference	0.0	REAL
My_PowerFlex_755_Drive:O.StopModeA	0	DINT
My_PowerFlex_755_Drive:O.StopModeB	0	DINT
My_PowerFlex_755_Drive:O.AccelTime1	0.0	REAL
My_PowerFlex_755_Drive:O.AccelTime2	0.0	REAL
My_PowerFlex_755_Drive:O.DecelTime1	0.0	REAL
My_PowerFlex_755_Drive:O.DecelTime2	0.0	REAL
My_PowerFlex_755_Drive:O.JogAccDecTime	0.0	REAL
My_PowerFlex_755_Drive:O.JogSpeed1	0.0	REAL
My_PowerFlex_755_Drive:O.JogSpeed2	0.0	REAL
My_PowerFlex_755_Drive:O.PresetSpeed1	0.0	REAL
My_PowerFlex_755_Drive:O.PresetSpeed2	0.0	REAL
My_PowerFlex_755_Drive:O.PresetSpeed3	0.0	REAL
My_PowerFlex_755_Drive:O.PresetSpeed4	0.0	REAL
My_PowerFlex_755_Drive:O.PresetSpeed5	0.0	REAL
My_PowerFlex_755_Drive:O.PresetSpeed6	0.0	REAL
My_PowerFlex_755_Drive:O.PresetSpeed7	0.0	REAL

Saving the I/O Configuration to the Controller

After adding the bridge and drive/adapter to the I/O configuration, you must download the configuration to the controller. You should also save the configuration to a file on your computer.

1. In the RSLogix 5000 window, select **Communications > Download**. The Download dialog box (Figure 4.15) appears.

Figure 4.15 Download Dialog Box



TIP: If a message box reports that RSLogix 5000 is unable to go online, select **Communications > Who Active** to try to find your controller in the Who Active screen. After finding and selecting the controller, click **Set Project Path** to establish the path. If your controller does not appear, you need to add or configure the EtherNet/IP driver in RSLinx. Refer to the RSLinx online help.

2. Click **Download** to download the configuration to the controller. When the download is successfully completed, RSLogix 5000 goes into the Online mode and the I/O Not Responding box in the upper-left of the window should be flashing green. Also, a yellow warning symbol ⚠ should be displayed on the I/O Configuration folder in the treeview and on the drive profile.

   If the controller was in Run Mode before clicking **Download**, RSLogix 5000 prompts you to change the controller mode back to Remote Run. In this case, choose the appropriate mode for your application. If the controller was in Program Mode before clicking **Download**, this prompt will not appear.

3. Select **File > Save**. If this is the first time you saved the project, the Save As dialog box appears. Navigate to a folder, type a file name, and click **Save** to save the configuration to a file on your computer.

   To ensure that the present project configuration values are saved, RSLogix 5000 prompts you to upload them. Click **Yes** to upload and save them.

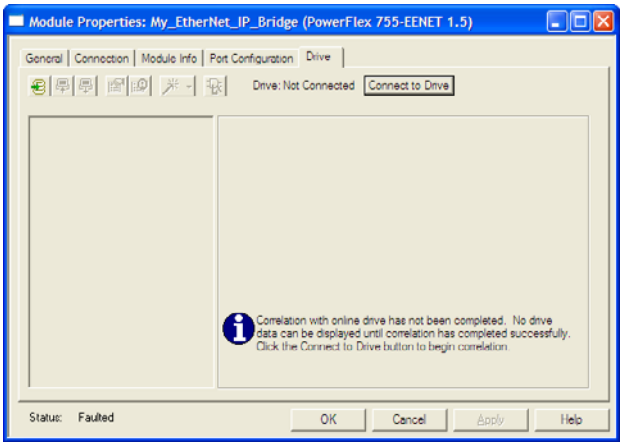
Correlating the Drive with the Controller

You must now correlate the drive settings to the RSLogix 5000 project I/O settings so that they match. This requires loading the project I/O settings into the drive.

1. In the treeview under I/O Configuration, right-click on the drive profile (for this example My_PowerFlex_755_Drive) and select **Properties**.

2. Click the Drive tab and then click **Connect to Drive** (Figure 4.16) to begin the correlation process.

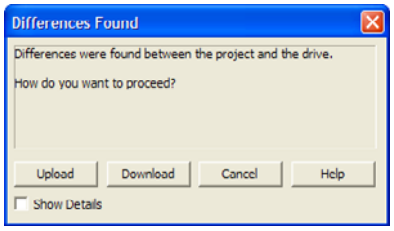
Figure 4.16 Drive Correlation Screen



After the drive configuration data has been verified, a pop-up dialog box will appear to synchronize ports from the online drive to the project to ensure that the correct Datalinks are assigned. Click **OK**.

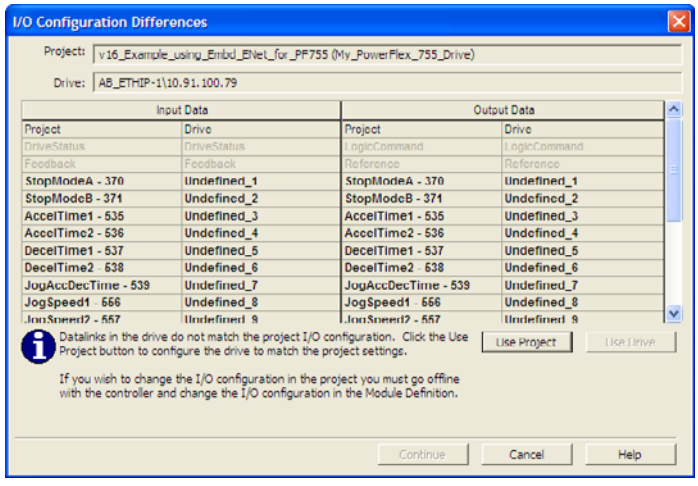
If the Differences Found screen (Figure 4.17) appears — which is typical, click **Download**. This will download the project settings from the controller to the drive and its connected adapter. If **Upload** is clicked, the drive and adapter settings are uploaded to the controller.

Figure 4.17 Differences Found Screen



3. The I/O Configuration Differences screen (Figure 4.18) appears.

Figure 4.18 I/O Configuration Differences Screen



To match the Datalinks in the drive to the project I/O configuration, click **Use Project**. After the datalinks have been matched, the Input Data and Output Data columns are grayed out. Click **Continue**.

A series of download dialog boxes appear, which may take a minute to complete. Thereafter, the I/O OK box in the upper-left of the RSLogix 5000 window should now be solid green and the yellow warning symbols in the treeview under the I/O Configuration folder and drive profile should be gone.

4. Click **OK** to close the Module Properties screen for the drive.

Using the RSLogix 5000 Generic Profile (all versions)

The basic RSLogix 5000 Generic Profile is only recommended when:

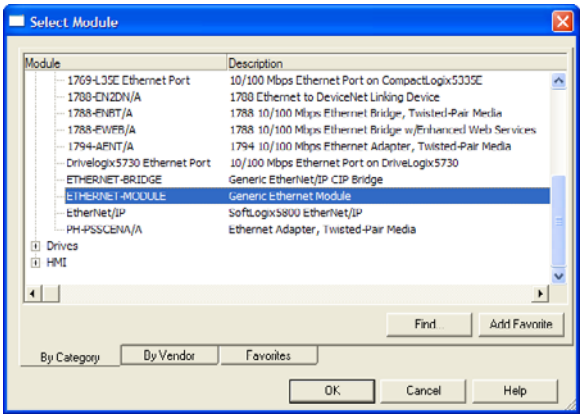
- A specific drive profile in other versions of RSLogix 5000 is unavailable.
- Users are already familiar with a Generic Profile and do not want to convert an existing project to an Add-On Drive Profile (v16 or higher).
- A project must maintain specific revision level control.
- **Version 16 Only** — The controller cannot be taken offline. Version 16 enables the Generic Profile to be added while the controller is online and in the Run mode.

Adding the Drive/Adapter to the I/O Configuration

To transmit data between the bridge and the drive, you must add the drive as a child device to the parent bridge.

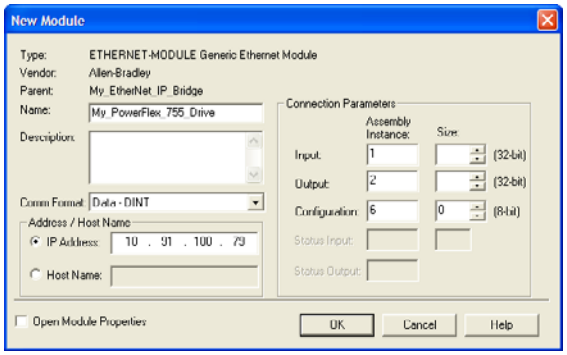
1. In the treeview, right-click on the bridge and select **New Module…** to display the Select Module screen (Figure 4.19). In our example, we right-click on the 1756-ENBT/A bridge. Expand the Communications group to display all of the available communication modules.

Figure 4.19 Select Module Screen



2. Select “ETHERNET-MODULE” from the list (Figure 4.19) to configure the drive and its embedded EtherNet/IP adapter, and then click **OK**. The drive’s New Module screen (Figure 4.20) appears.

Figure 4.20 Drive New Module Screen



3. Edit the following information about the drive/adapter:

Box	Setting
Name	A name to identify the drive and adapter.
Description	Optional – description of the drive/adapter.
Comm Format	**Data - DINT** (This setting formats the data in 32-bit words.)
IP Address	The IP address of the adapter.
Open Module Properties	When this box is checked, the drive’s New Module screen will close when clicking **OK**. When unchecked, additional module properties screens will appear to further configure the drive/adapter when **OK** is clicked. For this example, check this box.

4. Under Connection Parameters, edit the following:

Box	Assembly Instance	Size
Input	**1** (This value is required.)	The value will vary based on the number of **[DL From Net xx]** parameters used for your application (see details below).
Output	**2** (This value is required.)	The value will vary based on the number of **[DL To Net xx]** parameters used for your application (see details below).
Configuration	**6** (This value is required.)	**0** (This value is required.)

Enter the number of 32-bit words that are required for your I/O in the Input Size and Output Size boxes. Since the adapter always uses the 32-bit Logic Status, 32-bit Feedback, and a 32-bit word dedicated for memory allocation of the Generic Ethernet module profile, at least three 32-bit words must be set for the Input Size. The adapter also uses the 32-bit Logic Command and 32-bit Reference, requiring at least two 32-bit words for the Output Size. If any or all of the drive’s sixteen 32-bit Datalinks are used (see Setting a Master-Slave Hierarchy on page 3-8 or Custom Peer I/O Configuration on page 3-11), the Input and Output Size settings must be increased accordingly.

– Input Size: Start with 3 words and add 1 word for each Datalink used to write data. For example, if 3 Datalinks—**[DL From Net xx]** parameters—will be used to write to drive or peripheral

parameters, add 3 words to the required 3 words for a total of 6 words.

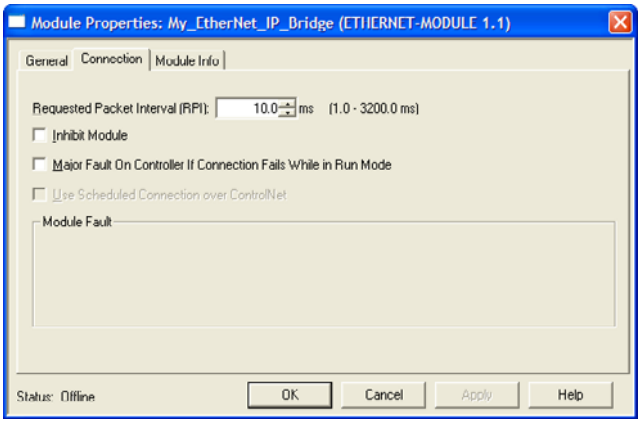
– Output Size: Start with 2 words and add 1 word for each Datalink used to read data. For example, if 7 Datalinks—**[DL To Net xx]** parameters—will be used to read drive or peripheral parameters, add 7 words to the required 2 words for a total of 9 words.

For the example in this manual, all 16 **[DL From Net xx]** and all 16 **[DL To Net xx]** are used, resulting in an Input Size of "19" and an Output Size of "18."

5. After setting the information in the drive's New Module screen, click **OK**. The Module Properties screen appears.

6. Click the Connection tab (Figure 4.21).

Figure 4.21 Connection Screen



7. In the "Requested Packet Interval (RPI)" box, set the value to 2.0 milliseconds or greater. This value determines the maximum interval that a controller should use to move data to and from the adapter. To conserve bandwidth, use higher values for communicating with low priority devices. For this example, leave the "Inhibit Module" and Major Fault …" boxes unchecked.

8. Click **OK**. The new node ("My_PowerFlex_755_Drive" in this example) now appears under the bridge ("My_EtherNet_IP_Bridge" in this example) in the I/O Configuration folder. If you double-click on the Input and Output Controller Tags (Figure 4.22 and Figure 4.23), you will see that module-defined data types and tags have been automatically created. After you save and download the configuration, these tags allow you to access the Input and Output data of the drive via the controller's ladder logic.

Figure 4.22 Input Image Controller Tags

Name	Data Type	Description
My_PowerFlex_755_Drive:I	AB:ETHERNET...	
My_PowerFlex_755_Drive:I.Data	DINT[19]	
My_PowerFlex_755_Drive:I.Data[0]	DINT	Pad Word
My_PowerFlex_755_Drive:I.Data[1]	DINT	Logic Status
My_PowerFlex_755_Drive:I.Data[2]	DINT	Speed Feedback
My_PowerFlex_755_Drive:I.Data[3]	DINT	DL To Net 01
My_PowerFlex_755_Drive:I.Data[4]	DINT	DL To Net 02
My_PowerFlex_755_Drive:I.Data[5]	DINT	DL To Net 03
My_PowerFlex_755_Drive:I.Data[6]	DINT	DL To Net 04
My_PowerFlex_755_Drive:I.Data[7]	DINT	DL To Net 05
My_PowerFlex_755_Drive:I.Data[8]	DINT	DL To Net 06
My_PowerFlex_755_Drive:I.Data[9]	DINT	DL To Net 07
My_PowerFlex_755_Drive:I.Data[10]	DINT	DL To Net 08
My_PowerFlex_755_Drive:I.Data[11]	DINT	DL To Net 09
My_PowerFlex_755_Drive:I.Data[12]	DINT	DL To Net 10
My_PowerFlex_755_Drive:I.Data[13]	DINT	DL To Net 11
My_PowerFlex_755_Drive:I.Data[14]	DINT	DL To Net 12
My_PowerFlex_755_Drive:I.Data[15]	DINT	DL To Net 13
My_PowerFlex_755_Drive:I.Data[16]	DINT	DL To Net 14
My_PowerFlex_755_Drive:I.Data[17]	DINT	DL To Net 15
My_PowerFlex_755_Drive:I.Data[18]	DINT	DL To Net 16

Figure 4.23 Output Image Controller Tags

Name	Data Type	Description
My_PowerFlex_755_Drive:O	AB:ETHERNET...	
My_PowerFlex_755_Drive:O.Data	DINT[18]	
My_PowerFlex_755_Drive:O.Data[0]	DINT	Logic Command
My_PowerFlex_755_Drive:O.Data[1]	DINT	Speed Reference
My_PowerFlex_755_Drive:O.Data[2]	DINT	DL From Net 01
My_PowerFlex_755_Drive:O.Data[3]	DINT	DL From Net 02
My_PowerFlex_755_Drive:O.Data[4]	DINT	DL From Net 03
My_PowerFlex_755_Drive:O.Data[5]	DINT	DL From Net 04
My_PowerFlex_755_Drive:O.Data[6]	DINT	DL From Net 05
My_PowerFlex_755_Drive:O.Data[7]	DINT	DL From Net 06
My_PowerFlex_755_Drive:O.Data[8]	DINT	DL From Net 07
My_PowerFlex_755_Drive:O.Data[9]	DINT	DL From Net 08
My_PowerFlex_755_Drive:O.Data[10]	DINT	DL From Net 09
My_PowerFlex_755_Drive:O.Data[11]	DINT	DL From Net 10
My_PowerFlex_755_Drive:O.Data[12]	DINT	DL From Net 11
My_PowerFlex_755_Drive:O.Data[13]	DINT	DL From Net 12
My_PowerFlex_755_Drive:O.Data[14]	DINT	DL From Net 13
My_PowerFlex_755_Drive:O.Data[15]	DINT	DL From Net 14
My_PowerFlex_755_Drive:O.Data[16]	DINT	DL From Net 15
My_PowerFlex_755_Drive:O.Data[17]	DINT	DL From Net 16

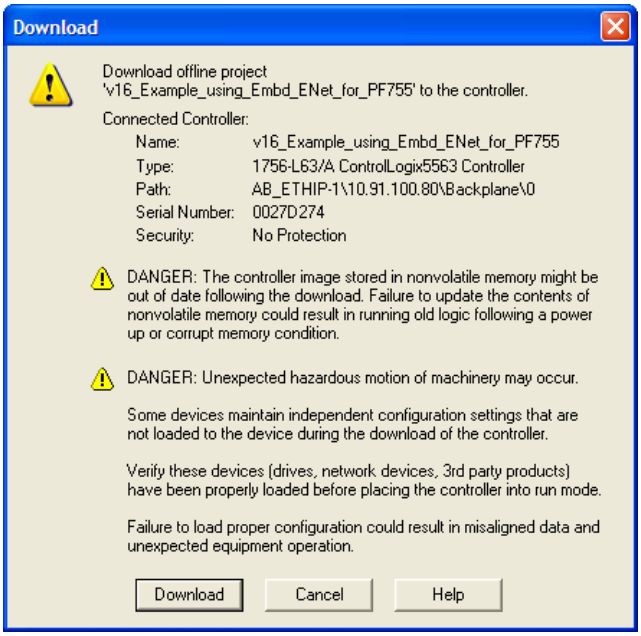
Saving the I/O Configuration to the Controller

After adding the bridge and drive/adapter to the I/O configuration, you must download the configuration to the controller. You should also save the configuration to a file on your computer.

TIP: When using RSLogix 5000 (v16 or higher), you can add the I/O configuration of a Generic Profile while the controller is online and in the Run mode.

1. In the RSLogix 5000 window, select **Communications > Download**. The Download dialog box (Figure 4.24) appears.

Figure 4.24 Download Dialog Box



TIP: If a message box reports that RSLogix 5000 is unable to go online, select **Communications > Who Active** to try to find your controller in the Who Active screen. After finding and selecting the controller, click **Set Project Path** to establish the path. If your controller does not appear, you need to add or configure the EtherNet/IP driver in RSLinx. See Using RSLinx Classic on page 4-1 for details.

2. Click **Download** to download the configuration to the controller. When the download is successfully completed, RSLogix 5000 goes into the Online mode and the I/O OK box in the upper-left of the screen should be solid green.

3. Select **File > Save**. If this is the first time you saved the project, the Save As dialog box appears. Navigate to a folder, type a file name, and click **Save** to save the configuration to a file on your computer.

4. Any Datalinks that were enabled in the controller and adapter during I/O configuration must also be configured in the drive. Each Datalink being used must be assigned to a specific parameter in the drive or connected peripheral (see Setting a Master-Slave Hierarchy on page 3-8 or Custom Peer I/O Configuration on page 3-11). If this is not done, the controller will receive or send placeholder data instead of actual drive or peripheral parameter values.

5. Place the controller in Remote Run or Run Mode.

Limitations When Using PLC-5, SLC 500, and MicroLogix 1100

Controlling I/O with explicit messages is relatively complex compared to normal implicit I/O control.

ControlLogix and CompactLogix controllers with EtherNet/IP provide the easiest and most integrated form of implicit I/O control for a PowerFlex drive. RSLogix 5000 v16 (and higher) programming software for ControlLogix and CompactLogix controllers contains integrated profiles for PowerFlex drives that, with a few clicks of the mouse, automatically create all controller tags and an implicit connection at the specified Requested Packet Interval to control the drive. This connection is monitored at both ends to ensure that the controller and drive are communicating. A watchdog will cause a drive fault if the drive does not respond within approximately 100 milliseconds. Therefore, using a ControlLogix or CompactLogix controller is by far the much preferred method of controlling drives on EtherNet/IP.

If you are not using either of these type of controllers, then PowerFlex drives on EtherNet/IP can be controlled with explicit messages using PLC-5, SLC 500 or MicroLogix 1100 controllers with the following limitations:

- An explicit message is a much slower form of control and is non-deterministic. This means that you cannot guarantee how long the drive will take to start up or stop when the command is given. Therefore, all equipment used in this manner should be subject to a risk assessment, taking into account the mechanical and electrical implementation.
- A timeout value (in seconds) in the drive will issue a drive fault if a message is not received from the controller within the specified time. However, the controller has no way of detecting a loss of communications to the drive until the next cycle of explicit messages. This is another factor in the risk assessment.
- Any additional drives to be controlled will require additional explicit messages for their control, and they need to be carefully sequenced. Most controllers have small communication queues (refer to its User Manual), which need to be carefully managed if messages are not to be lost.
- Each controller has a limited number of communication connections (refer to its User Manual for maximum connections), which will limit the number of drives that can be connected.

In summary, unlike a ControlLogix or CompactLogix controller, programming a PLC-5, SLC 500 or MicroLogix 1100 controller using RSLogix 5 or RSLogix 500 software with explicit messages is a lot more difficult, and produces a far more complex program.

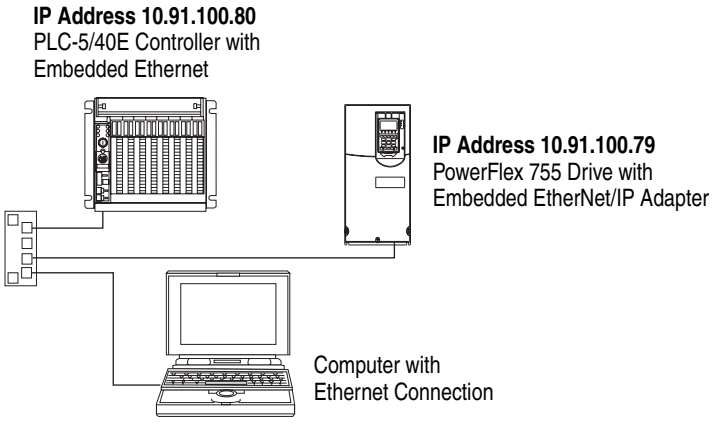
PLC-5 Example

Important: The PLC-5 must be Series E (Rev. D.1 or higher) to support the MultiHop feature that routes messaging to the drive.

Example Network

After the adapter is configured, the connected drive and adapter will be a single node on the network. This section provides the steps needed to configure a simple EtherNet/IP network (see Figure 4.25). In our example, we will configure a PLC-5/40E controller to communicate with a drive using Logic Command/Status, Reference/Feedback, and 32 Datalinks (16 to read/16 to write) over the network.

Figure 4.25 PLC-5 Example EtherNet/IP Network



Configuring Parameters for Network I/O

Since the I/O for the drive is message-based, there is no need to configure any I/O inside the RSLogix 5 (v7 or higher) project until using the I/O as described in Chapter 5.

However, to get the adapter to operate with the I/O created in Chapter 5, we need to configure the adapter to accept the I/O and the drive to point to the appropriate Datalinks.

Since the adapter always uses the 32-bit Logic Status and 32-bit Feedback, at least two 32-bit words must be accounted for in the controller input image. The adapter also uses the 32-bit Logic Command and 32-bit Reference, requiring at least two 32-bit words that must be accounted for in the controller output image. If any or all of the drive's sixteen 32-bit Datalinks are used (see Setting a Master-Slave Hierarchy on page 3-8 or Custom Peer I/O Configuration on page 3-11):

- An additional 32-bit word for each **[DL From Net xx]** Datalink used to write to drive or peripheral parameters must be accounted for in the controller input image.

- An additional 32-bit word for each **[DL To Net xx]** Datalink used to read data must be accounted for in the controller output image.

Adapter Parameter Settings for PLC-5 Example

These adapter settings were used for the example ladder logic program in this section.

Adapter Parameter	Value	Description
01 - [DL From Net 01]	370	Points to drive Par. 370 - [Stop Mode A]
02 - [DL From Net 02]	371	Points to drive Par. 371 - [Stop Mode B]
03 - [DL From Net 03]	535	Points to drive Par. 535 - [Accel Time 1]
04 - [DL From Net 04]	536	Points to drive Par. 536 - [Accel Time 2]
05 - [DL From Net 05]	537	Points to drive Par. 537 - [Decel Time 1]
06 - [DL From Net 06]	538	Points to drive Par. 538 - [Decel Time 2]
07 - [DL From Net 07]	539	Points to drive Par. 539 - [Jog Acc Dec Time]
08 - [DL From Net 08]	556	Points to drive Par. 556 - [Jog Speed 1]
09 - [DL From Net 09]	557	Points to drive Par. 557 - [Jog Speed 2]
10 - [DL From Net 10]	571	Points to drive Par. 571 - [Preset Speed 1]
11 - [DL From Net 11]	572	Points to drive Par. 572 - [Preset Speed 2]
12 - [DL From Net 12]	573	Points to drive Par. 573 - [Preset Speed 3]
13 - [DL From Net 13]	574	Points to drive Par. 574 - [Preset Speed 4]
14 - [DL From Net 14]	575	Points to drive Par. 575 - [Preset Speed 5]
15 - [DL From Net 15]	576	Points to drive Par. 576 - [Preset Speed 6]
16 - [DL From Net 16]	577	Points to drive Par. 577 - [Preset Speed 7]
17 - [DL To Net 01]	370	Points to drive Par. 370 - [Stop Mode A]
18 - [DL To Net 02]	371	Points to drive Par. 371 - [Stop Mode B]
19 - [DL To Net 03]	535	Points to drive Par. 535 - [Accel Time 1]
20 - [DL To Net 04]	536	Points to drive Par. 536 - [Accel Time 2]
21 - [DL To Net 05]	537	Points to drive Par. 537 - [Decel Time 1]
22 - [DL To Net 06]	538	Points to drive Par. 538 - [Decel Time 2]
23 - [DL To Net 07]	539	Points to drive Par. 539 - [Jog Acc Dec Time]
24 - [DL To Net 08]	556	Points to drive Par. 556 - [Jog Speed 1]
25 - [DL To Net 09]	557	Points to drive Par. 557 - [Jog Speed 2]
26 - [DL To Net 10]	571	Points to drive Par. 571 - [Preset Speed 1]
27 - [DL To Net 11]	572	Points to drive Par. 572 - [Preset Speed 2]
28 - [DL To Net 12]	573	Points to drive Par. 573 - [Preset Speed 3]
29 - [DL To Net 13]	574	Points to drive Par. 574 - [Preset Speed 4]
30 - [DL To Net 14]	575	Points to drive Par. 575 - [Preset Speed 5]
31 - [DL To Net 15]	576	Points to drive Par. 576 - [Preset Speed 6]
32 - [DL To Net 16]	577	Points to drive Par. 577 - [Preset Speed 7]

TIP: The **[DL From Net xx]** parameters are inputs into the drive that come from controller outputs (for example, data to write to a drive parameter). The **[DL To Net xx]** parameters are outputs from the drive that go to controller inputs (for example, data to read a drive parameter).

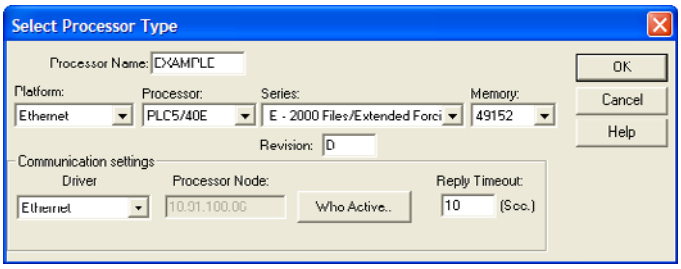
Creating RSLogix 5 (v7 or higher) Project

To transmit (read and write) data between the controller and drive, you must create message instructions that allocate data table addresses in the controller for Logic Command/Status, Reference/Feedback, and Datalinks. Note that three messages need to be configured. The timeout message has to be executed first before the Logic Command, Reference, and DL to Net Datalink messages will work. For more information on N42:3 and N45 target device data table addresses, refer to N-Files on page C-8.

Selecting the Controller

1. Start RSLogix 5. The RSLogix 5 window appears. Select **File > New** to display the Select Processor Type screen (Figure 4.26).

Figure 4.26 PLC-5 Select Processor Type Screen



2. Assign a name for the processor. From the pull-down fields, select the appropriate choices to match your PLC-5 controller and application, and click **OK**. The RSLogix 5 project window appears.

Creating PLC-5 Ladder Logic for the Control Timeout

1. In the RSLogix 5 project window treeview under Program Files double-click on LAD 2.

2. Insert a ladder rung, double-click on the rung to display the rung editor, and enter **MSG MGxx:n**, where:

   xx is an unused data file number (for example, MG10:n), and
   n is an unused element of the data file chosen for xx (for example, MG10:0)

   Then press **Enter**.

3. Insert another separate rung, double-click on the rung to display the rung editor, and enter **BST XIC MGxx:n/DN NXB XIC MGxx:n/ER BND OTU MGxx:n/EN**, where:

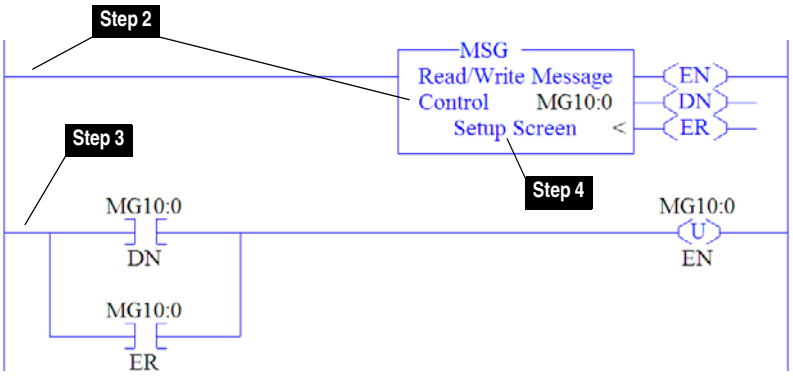
   xx and n must correspond to the assigned data file number and element (for example, MG10:0) for the message created in Step 2.

   Important: The information must be entered with appropriate numbers for “xx” and “n” for your application, and with spaces and forward slashes exactly as shown.

   Then press **Enter**.

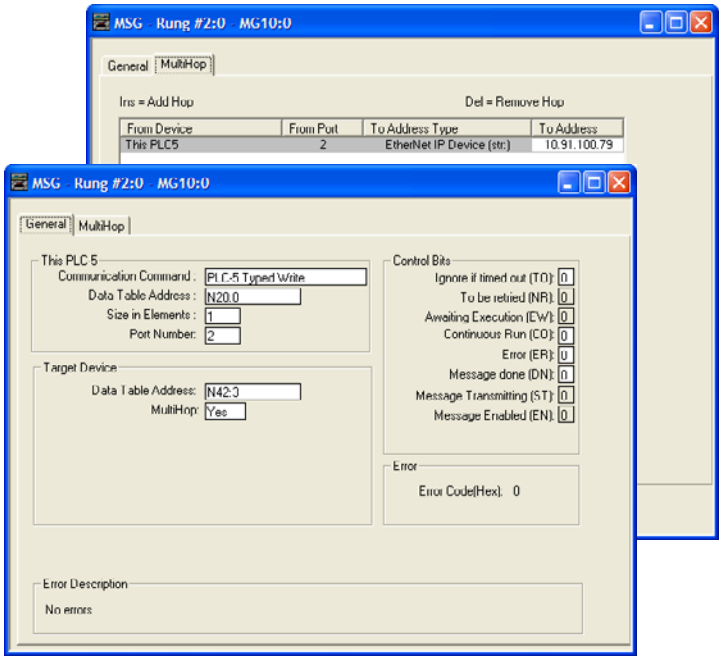
4. In the MSG instruction (Figure 4.27), double-click on Setup Screen to launch the message configuration screen (Figure 4.28).

Figure 4.27 PLC-5 Ladder Logic for the Control Timeout



5. Configure the General tab by entering or verifying the information shown in the screen.

Figure 4.28 PLC-5 Message Configuration Screens for the Control Timeout



General Tab Box	Setting
This PLC-5	
Communication Command	**PLC-5 Typed Write**. The controller type and command type for the controller to write the control timeout value to the drive.
Data Table Address (1)	**N20:0**. An unused controller data table address containing the control timeout value to be written.
Size in Elements (2)	**1**. Number of elements (words) to be transferred. Each element size is a 16-bit integer.
Port Number	**2**. Controller port to which the EtherNet/IP network is connected.
Target Device (data for adapter/drive)	
Data Table Address (3)	**N42:3**. Specific starting address of the destination file in the drive.
MultiHop	**Yes**. Enables communication to allow Ethernet messaging to be routed to the adapter/drive. When “Yes” is selected, a MultiHop tab appears on the message configuration screen.
MultiHop Tab Box	**Setting**
To Address	**10.91.100.79.** The IP address of the adapter connected to the drive.

(1) For details on data table addresses for this example project, refer to Table 5.D on page 5-15.

(2) For details to determine element size for a specific drive, refer to Understanding Controller Data Table Addresses on page 5-14.

(3) For details on setting the control timeout value and its function, see N-Files on page C-8.

Creating PLC-5 Ladder Logic for the Logic Status, Feedback, and DL From Net Datalinks

1. Insert another separate rung, double-click on the rung to display the rung editor, and enter **MSG MGxx:n**, where:

   xx is an unused data file number (for example, MG11:n), and
   n is an unused element of the data file chosen for xx (for example, MG11:0)

   Then press **Enter**.

2. Insert another separate rung, double-click on the rung to display the rung editor, and enter **BST XIC MGxx:n/DN NXB XIC MGxx:n/ER BND OTU MGxx:n/EN**, where:

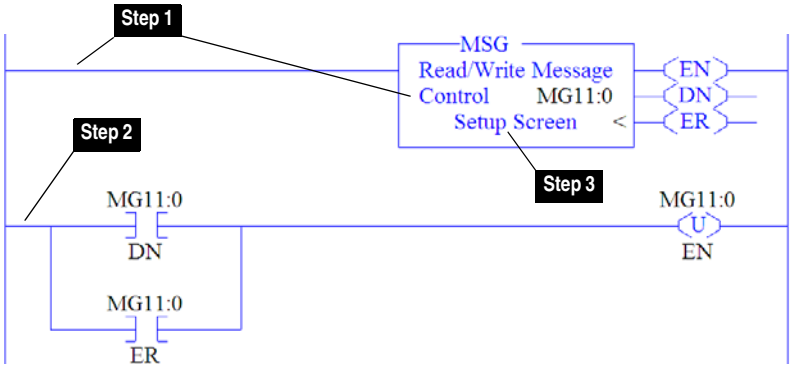
   xx and n must correspond to the assigned data file number and element (for example, MG11:0) for the message created in Step 1.

   Important: The information must be entered with appropriate numbers for “xx” and “n” for your application, and with spaces and forward slashes exactly as shown.

   Then press **Enter**.

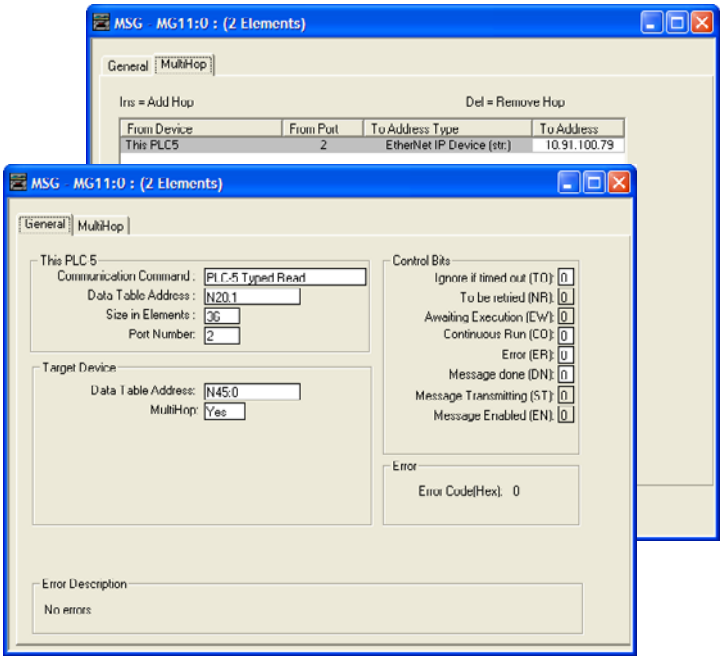
3. In the MSG instruction (Figure 4.29), double-click on Setup Screen to launch the message configuration screen (Figure 4.30).

   Figure 4.29 PLC-5 Ladder Logic for the Logic Status, Feedback, and DL From Net Datalinks



4. Configure the General tab by entering or verifying the information shown in the screen.

Figure 4.30 PLC-5 Message Configuration Screens for the Logic Status Feedback, and DL From Net Datalinks



General Tab Box	Setting
This PLC-5	
Communication Command	**PLC-5 Typed Read**. The controller type and command type for the controller to read data from the drive.
Data Table Address (1)	**N20:1**. An unused controller data table address containing the data to be read from the drive.
Size in Elements (2)	**36**. Number of elements (words) to be transferred. Each element size is a 16-bit integer.
Port Number	**2**. Controller port to which the EtherNet/IP network is connected.
Target Device (data for adapter/drive)	
Data Table Address (3)	**N45:0**. Specific starting address of the source file in the drive.
MultiHop	**Yes**. Enables communication to allow Ethernet messaging to be routed to the adapter/drive. When "Yes" is selected, a MultiHop tab appears on the message configuration screen.
MultiHop Tab Box	**Setting**
To Address	**10.91.100.79.** The IP address of the adapter connected to the drive.

(1) For details on data table addresses for this example project, refer to Table 5.D on page 5-15.

(2) For details to determine element size for a specific drive, refer to Understanding Controller Data Table Addresses on page 5-14.

(3) For N-File details, see N-Files on page C-8.

Creating PLC-5 Ladder Logic for the Logic Command, Reference, and DL To Net Datalinks

1. Insert another separate rung, double-click on the rung to display the rung editor, and enter **MSG MGxx:n**, where:

   xx is an unused data file number (for example, MG12:n), and
   n is an unused element of the data file chosen for xx (for example, MG12:0)

   Then press **Enter**.

2. Insert another separate rung, double-click on the rung to display the rung editor, and enter **BST XIC MGxx:n/DN NXB XIC MGxx:n/ER BND OTU MGxx:n/EN**, where:

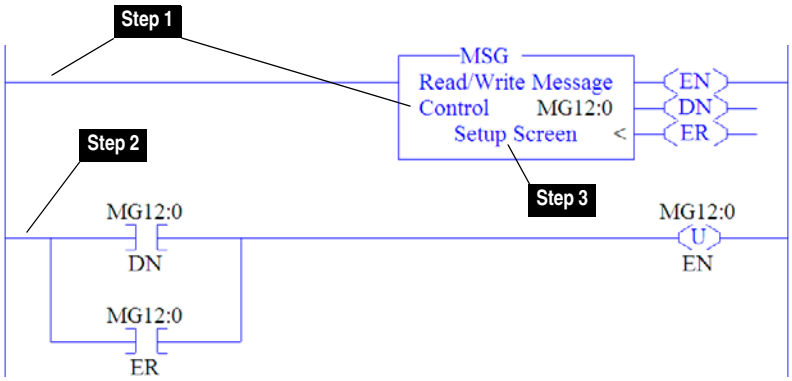
   xx and n must correspond to the assigned data file number and element (for example, MG12:0) for the message created in Step 1.

   Important: The information must be entered with appropriate numbers for "xx" and "n" for your application, and with spaces and forward slashes exactly as shown.

   Then press **Enter**.

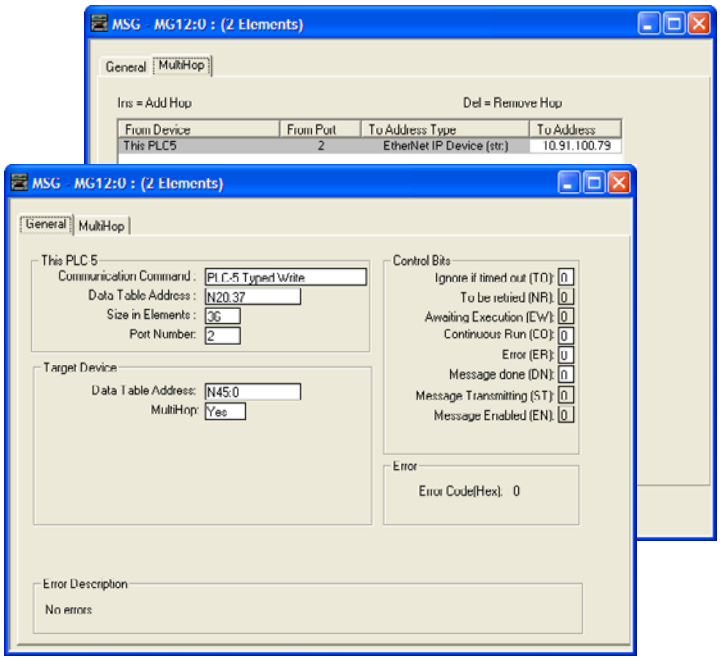
3. In the MSG instruction (Figure 4.31), double-click on Setup Screen to launch the message configuration screen (Figure 4.32).

Figure 4.31 PLC-5 Ladder Logic for the Logic Command, Reference, and DL To Net Datalinks



4. Configure the General tab by entering or verifying the information shown in the screen.

Figure 4.32 PLC-5 Message Configuration Screens for the Logic Command, Reference, and DL To Net Datalinks



General Tab Box	Setting
This PLC-5	
Communication Command	**PLC-5 Typed Write**. The controller type and command type for the controller to write data to the drive.
Data Table Address (1)	**N20:37**. An unused controller data table address containing the data to be written to the drive.
Size in Elements (2)	**36**. Number of elements (words) to be transferred. Each element size is a 16-bit integer.
Port Number	**2**. Controller port to which the EtherNet/IP network is connected.
Target Device (data for adapter/drive)	
Data Table Address (3)	**N45:0**. Specific starting address of the destination file in the drive.
MultiHop	**Yes**. Enables communication to allow Ethernet messaging to be routed to the adapter/drive. When "Yes" is selected, a MultiHop tab appears on the message configuration screen.
MultiHop Tab Box	**Setting**
To Address	**10.91.100.79.** The IP address of the adapter connected to the drive.

(1) For details on data table addresses for this example project, refer to Table 5.D on page 5-15.

(2) For details to determine element size for a specific drive, refer to Understanding Controller Data Table Addresses on page 5-14.

(3) For N-File details, see N-Files on page C-8.

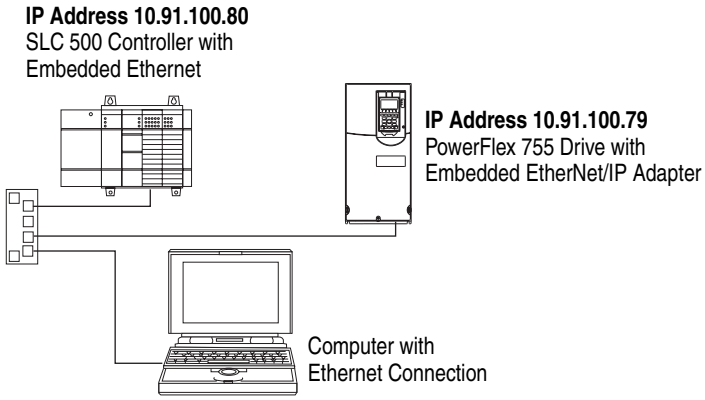
TIP: If the controller is controlling more than one drive, it is recommended to intersperse the control I/O messaging for each drive to conserve network bandwidth and decrease response time. That is, sequence the message instructions for each drive so that its group of messages will occur at a different time than those for another drive.

SLC 500 Example

Example Network

After the adapter is configured, the connected drive and adapter will be a single node on the network. This section provides the steps needed to configure a simple EtherNet/IP network (see Figure 4.33). In our example, we will configure a SLC 500 controller to communicate with a drive using Logic Command/Status, Reference/Feedback, and 32 Datalinks (16 to read/16 to write) over the network.

Figure 4.33 SLC 500 Example EtherNet/IP Network



Configuring Parameters for Network I/O

Since the I/O for the drive is message-based, there is no need to configure any I/O inside the RSLogix 500 (v7 or higher) project until using the I/O as described in Chapter 5.

However, to get the adapter to operate with the I/O created in Chapter 5, we need to configure the adapter to accept the I/O and drive to point to the appropriate Datalinks.

Since the adapter always uses the 32-bit Logic Status and 32-bit Feedback, at least two 32-bit words must be accounted for in the controller input image. The adapter also uses the 32-bit Logic Command and 32-bit Reference, requiring at least two 32-bit words that must be accounted for in the controller output image. If any or all of the drive's sixteen 32-bit Datalinks are used (see Setting a Master-Slave Hierarchy on page 3-8 or Custom Peer I/O Configuration on page 3-11):

- An additional 32-bit word for each **[DL From Net xx]** Datalink used to write to drive or peripheral parameters must be accounted for in the controller input image.
- An additional 32-bit word for each **[DL To Net xx]** Datalink used to read data must be accounted for in the controller output image.

Adapter Parameter Settings for SLC 500 Example

These adapter settings were used for the example ladder logic program in this section.

Adapter Parameter	Value	Description
01 - [DL From Net 01]	370	Points to drive Par. 370 - [Stop Mode A]
02 - [DL From Net 02]	371	Points to drive Par. 371 - [Stop Mode B]
03 - [DL From Net 03]	535	Points to drive Par. 535 - [Accel Time 1]
04 - [DL From Net 04]	536	Points to drive Par. 536 - [Accel Time 2]
05 - [DL From Net 05]	537	Points to drive Par. 537 - [Decel Time 1]
06 - [DL From Net 06]	538	Points to drive Par. 538 - [Decel Time 2]
07 - [DL From Net 07]	539	Points to drive Par. 539 - [Jog Acc Dec Time]
08 - [DL From Net 08]	556	Points to drive Par. 556 - [Jog Speed 1]
09 - [DL From Net 09]	557	Points to drive Par. 557 - [Jog Speed 2]
10 - [DL From Net 10]	571	Points to drive Par. 571 - [Preset Speed 1]
11 - [DL From Net 11]	572	Points to drive Par. 572 - [Preset Speed 2]
12 - [DL From Net 12]	573	Points to drive Par. 573 - [Preset Speed 3]
13 - [DL From Net 13]	574	Points to drive Par. 574 - [Preset Speed 4]
14 - [DL From Net 14]	575	Points to drive Par. 575 - [Preset Speed 5]
15 - [DL From Net 15]	576	Points to drive Par. 576 - [Preset Speed 6]
16 - [DL From Net 16]	577	Points to drive Par. 577 - [Preset Speed 7]
17 - [DL To Net 01]	370	Points to drive Par. 370 - [Stop Mode A]
18 - [DL To Net 02]	371	Points to drive Par. 371 - [Stop Mode B]
19 - [DL To Net 03]	535	Points to drive Par. 535 - [Accel Time 1]
20 - [DL To Net 04]	536	Points to drive Par. 536 - [Accel Time 2]
21 - [DL To Net 05]	537	Points to drive Par. 537 - [Decel Time 1]
22 - [DL To Net 06]	538	Points to drive Par. 538 - [Decel Time 2]
23 - [DL To Net 07]	539	Points to drive Par. 539 - [Jog Acc Dec Time]
24 - [DL To Net 08]	556	Points to drive Par. 556 - [Jog Speed 1]
25 - [DL To Net 09]	557	Points to drive Par. 557 - [Jog Speed 2]
26 - [DL To Net 10]	571	Points to drive Par. 571 - [Preset Speed 1]
27 - [DL To Net 11]	572	Points to drive Par. 572 - [Preset Speed 2]
28 - [DL To Net 12]	573	Points to drive Par. 573 - [Preset Speed 3]
29 - [DL To Net 13]	574	Points to drive Par. 574 - [Preset Speed 4]
30 - [DL To Net 14]	575	Points to drive Par. 575 - [Preset Speed 5]
31 - [DL To Net 15]	576	Points to drive Par. 576 - [Preset Speed 6]
32 - [DL To Net 16]	577	Points to drive Par. 577 - [Preset Speed 7]

TIP: The **[DL From Net xx]** parameters are inputs into the drive that come from controller outputs (for example, data to write to a drive parameter). The **[DL To Net xx]** parameters are outputs from the drive that go to controller inputs (for example, data to read a drive parameter).

Creating RSLogix 500 (v7 or higher) Project

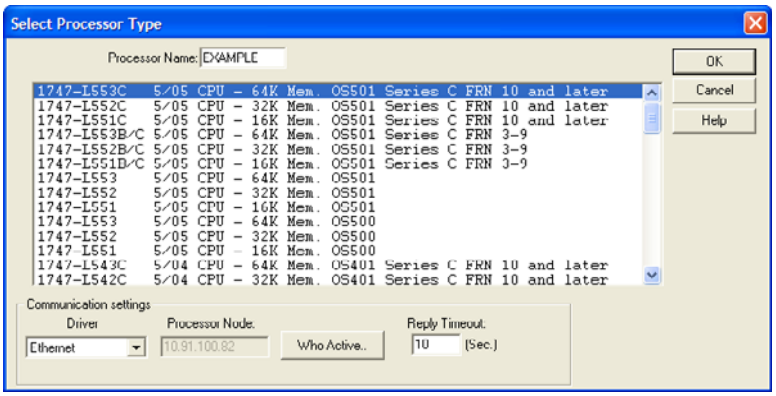
To transmit (read and write) data between the controller and drive, you must create message instructions that allocate data table addresses in the controller for Logic Command/Status, Reference/Feedback, and Datalinks. Note that three messages need to be configured. The timeout

message has to be executed first before the Logic Command, Reference, and DL to Net Datalink messages will work. For more information on N42:3 and N45 target device data table addresses, refer to N-Files on page C-8.

Selecting the Controller

1. Start RSLogix 500. The RSLogix 500 window appears. Select **File > New** to display the Select Processor Type screen (Figure 4.34).

Figure 4.34 SLC 500 Select Processor Type Screen



2. Assign a name for the processor. In the list, select a 1747-L55x type controller. Then select the appropriate choices for the fields in the screen to match your application, and click **OK**. The RSLogix 500 project window appears.

Creating SLC 500 Ladder Logic for the Control Timeout

1. In the RSLogix 500 project window treeview under Program Files double-click on LAD 2.

2. Insert a ladder rung, double-click on the rung to display the rung editor, and enter **MSG WRITE 500CPU LOCAL Nxx:n**, where:

   xx is an unused data file number (for example, N10:n), and
   n is an unused element of the data file chosen for xx (for example, N10:0)

   Then press **Enter**.

3. Insert another separate rung, double-click on the rung to display the rung editor, and enter **BST XIC Nxx:n/13 NXB XIC Nxx:n/12 BND OTU Nxx:n/15**, where:

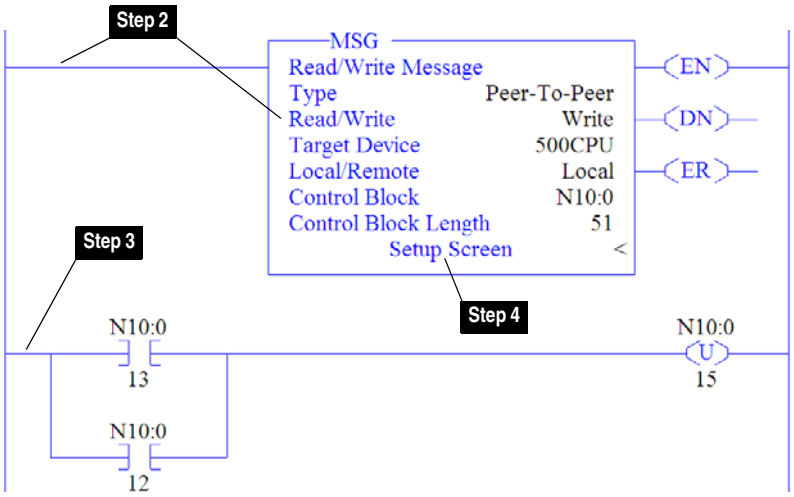
   xx and n must correspond to the assigned data file number and element (for example, N10:0) for the message created in Step 2.

   Important: The information must be entered with appropriate numbers for “xx” and “n” for your application, and with spaces and forward slashes exactly as shown.

   Then press **Enter**.

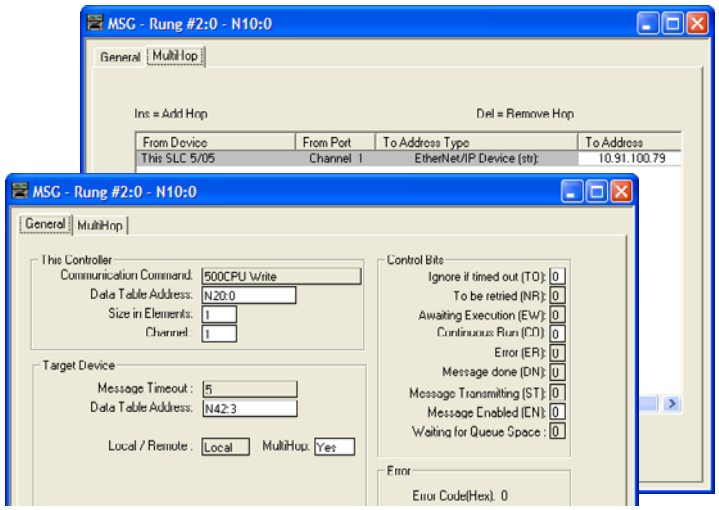
4. In the MSG instruction (Figure 4.35), double-click on Setup Screen to launch the message configuration screen (Figure 4.36).

Figure 4.35 SLC 500 Ladder Logic for the Control Timeout



5. Configure the General tab by entering or verifying the information shown in the screen.

Figure 4.36 SLC 500 Message Configuration Screens for the Control Timeout



General Tab Box	Setting
This Controller	
Communication Command	This setting is unavailable (grayed out) and is established when the message is created in the ladder rung.
Data Table Address (1)	**N20:0**. An unused controller data table address containing the control timeout value to be written.
Size in Elements (2)	**1**. Number of elements (words) to be transferred. Each element size is a 16-bit integer.
Channel	**1**. Controller port to which the EtherNet/IP network is connected.
Target Device (data for adapter/drive)	
Message Timeout	This setting is unavailable (grayed out). Message timeout duration in seconds.
Data Table Address (3)	**N42:3**. Specific starting address of the destination file in the drive.
MultiHop	**Yes**. Enables communication to allow Ethernet messaging to be routed to the adapter/drive. When “Yes” is selected, a MultiHop tab appears on the message configuration screen.
MultiHop Tab Box	**Setting**
To Address	**10.91.100.79.** The IP address of the adapter connected to the drive.

(1) For details on data table addresses for this example project, refer to Table 5.D on page 5-15.

(2) For details to determine element size for a specific drive, refer to Understanding Controller Data Table Addresses on page 5-14.

(3) For details on setting the control timeout value and its function, see N-Files on page C-8.

Creating SLC 500 Ladder Logic for the Logic Status, Feedback, and DL From Net Datalinks

1. Insert another separate rung, double-click on the rung to display the rung editor, and enter **MSG READ 500CPU LOCAL Nxx:n**, where:

   xx is an unused data file number (for example, N11:n), and
   n is an unused element of the data file chosen for xx (for example, N11:0)

   Then press **Enter**.

2. Insert another separate rung, double-click on the rung to display the rung editor, and enter **BST XIC Nxx:n/13 NXB XIC Nxx:n/12 BND OTU Nxx:n/15**, where:

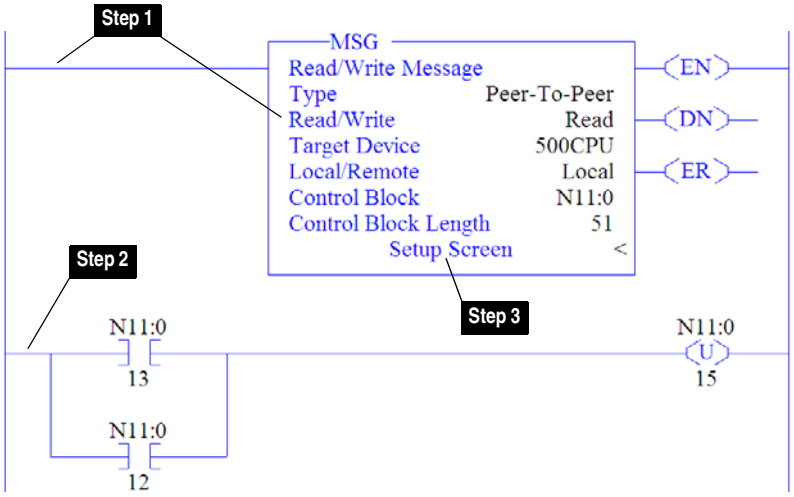
   xx and n must correspond to the assigned data file number and element (for example, N11:0) for the message created in Step 1.

   Important: The information must be entered with appropriate numbers for “xx” and “n” for your application, and with spaces and forward slashes exactly as shown.

   Then press **Enter**.

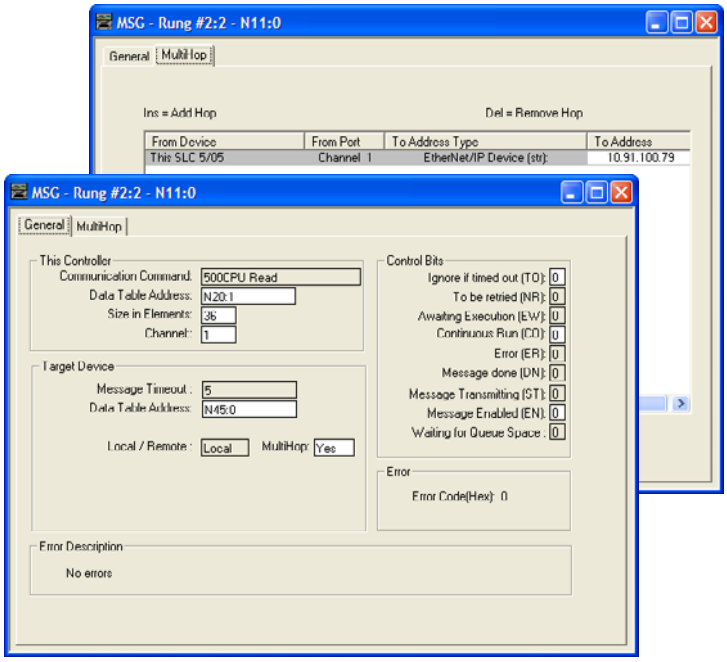
3. In the MSG instruction (Figure 4.37), double-click on Setup Screen to launch the message configuration screen (Figure 4.38).

Figure 4.37 SLC 500 Ladder Logic for the Logic Status, Feedback, and DL From Net Datalinks



4. Configure the General tab by entering or verifying the information shown in the screen.

Figure 4.38 SLC 500 Message Configuration Screens for the Logic Status, Feedback, and DL From Net Datalinks



General Tab Box	Setting
This Controller	
Communication Command	This setting is unavailable (grayed out) and is established when the message is created in the ladder rung.
Data Table Address (1)	**N20:1**. An unused controller data table address containing the data to be read from the drive.
Size in Elements (2)	**36**. Number of elements (words) to be transferred. Each element size is a 16-bit integer.
Channel	**1**. Controller port to which the EtherNet/IP network is connected.
Target Device (data for adapter/drive)	
Message Timeout	This setting is unavailable (grayed out). Message timeout duration in seconds.
Data Table Address (3)	**N45:0**. Specific starting address of the source file in the drive.
MultiHop	**Yes**. Enables communication to allow Ethernet messaging to be routed to the adapter/drive. When “Yes” is selected, a MultiHop tab appears on the message configuration screen.
MultiHop Tab Box	**Setting**
To Address	**10.91.100.79.** The IP address of the adapter connected to the drive.

(1) For details on data table addresses for this example project, refer to Table 5.D on page 5-15.

(2) For details to determine element size for a specific drive, refer to Understanding Controller Data Table Addresses on page 5-14.

(3) For N-File details, see N-Files on page C-8.

Creating SLC 500 Ladder Logic for the Logic Command, Reference, and DL To Net Datalinks

1. Insert another separate rung, double-click on the rung to display the rung editor, and enter **MSG WRITE 500CPU LOCAL Nxx:n**, where:

   xx is an unused data file number (for example, N12:n), and
   n is an unused element of the data file chosen for xx (for example, N12:0)

   Then press **Enter**.

2. Insert another separate rung, double-click on the rung to display the rung editor, and enter **BST XIC Nxx:n/13 NXB XIC Nxx:n/12 BND OTU Nxx:n/15**, where:

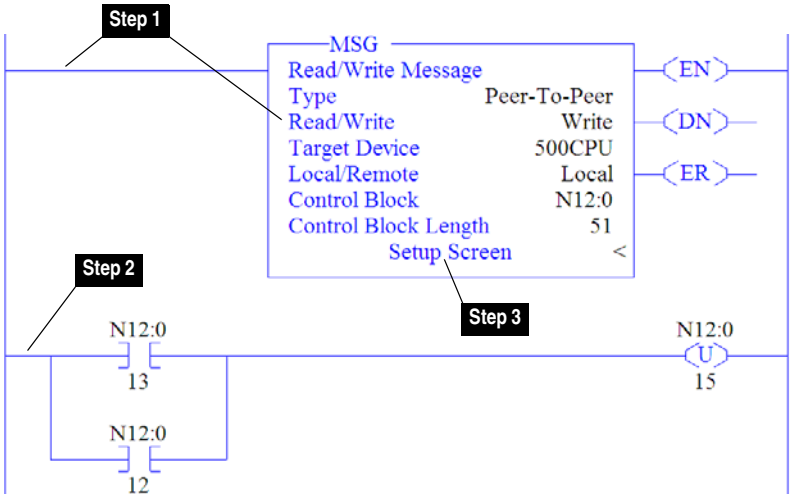
   xx and n must correspond to the assigned data file number and element (for example, N11:0) for the message created in Step 1.

   Important: The information must be entered with appropriate numbers for “xx” and “n” for your application, and with spaces and forward slashes exactly as shown.

   Then press **Enter**.

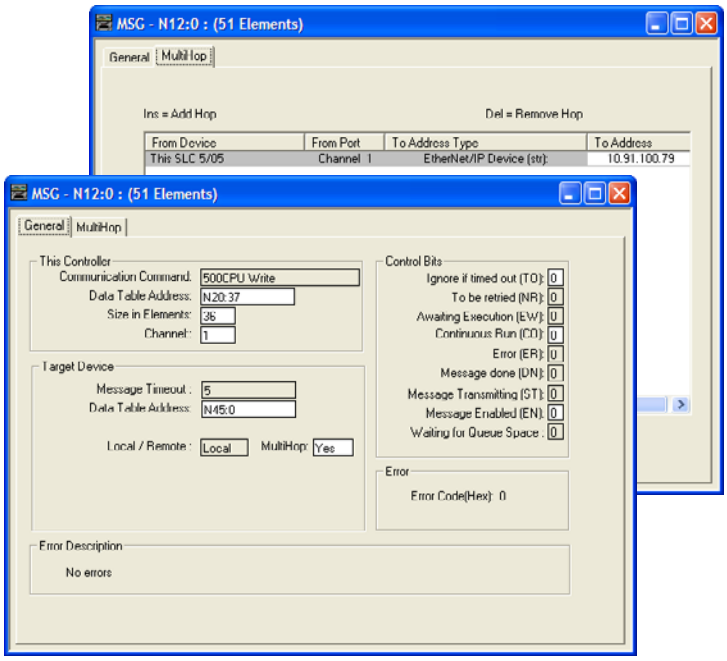
3. In the MSG instruction (Figure 4.39), double-click on Setup Screen to launch the message configuration screen (Figure 4.40).

Figure 4.39 SLC 500 Ladder Logic for the Logic Command, Reference, and DL To Net Datalinks



4. Configure the General tab by entering or verifying the information shown in the screen.

Figure 4.40 SLC 500 Message Configuration Screens for the Logic Command, Reference, and DL To Net Datalinks



General Tab Box	Setting
This Controller	
Communication Command	This setting is unavailable (grayed out) and is established when the message is created in the ladder rung.
Data Table Address (1)	**N20:37**. An unused controller data table address containing the data to be written to the drive.
Size in Elements (2)	**36**. Number of elements (words) to be transferred. Each element size is a 16-bit integer.
Channel	**1**. Controller port to which the EtherNet/IP network is connected.
Target Device (data for adapter/drive)	
Message Timeout	This setting is unavailable (grayed out). Message timeout duration in seconds.
Data Table Address (3)	**N45:0**. Specific starting address of the source file in the drive.
MultiHop	**Yes**. Enables communication to allow Ethernet messaging to be routed to the adapter/drive. When “Yes” is selected, a MultiHop tab appears on the message configuration screen.
MultiHop Tab Box	**Setting**
To Address	**10.91.100.79.** The IP address of the adapter connected to the drive.

(1) For details on data table addresses for this example project, refer to Table 5.D on page 5-15.

(2) For details to determine element size for a specific drive, refer to Understanding Controller Data Table Addresses on page 5-14.

(3) For N-File details, see N-Files on page C-8.

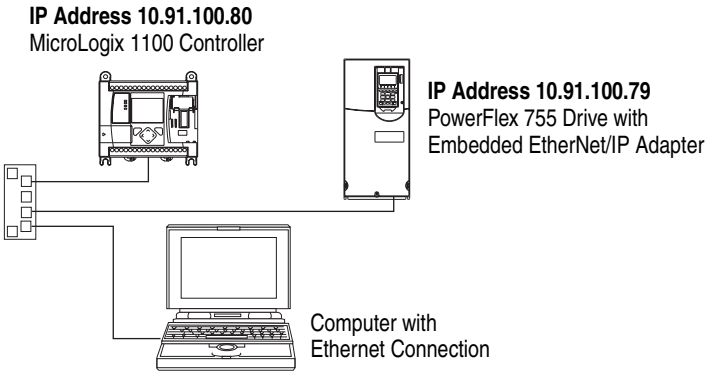
TIP: If the controller is controlling more than one drive, it is recommended to intersperse the control I/O messaging for each drive to conserve network bandwidth and decrease response time. That is, sequence the message instructions for each drive so that its group of messages will occur at a different time than those for another drive.

MicroLogix 1100 Example

Example Network

After the adapter is configured, the connected drive and adapter will be a single node on the network. This section provides the steps needed to configure a simple EtherNet/IP network (see Figure 4.41). In our example, we will configure a MicroLogix 1100 controller to communicate with a drive using Logic Command/Status, Reference/Feedback, and 32 Datalinks (16 to read/16 to write) over the network.

Figure 4.41 MicroLogix 1100 Example EtherNet/IP Network



Configuring Parameters for Network I/O

Since the I/O for the drive is message-based, there is no need to configure any I/O inside the RSLogix 500 (v7 or higher) project until using the I/O as described in Chapter 5.

However, to get the adapter to operate with the I/O created in Chapter 5, we need to configure the adapter to accept the I/O and drive to point to the appropriate Datalinks.

Since the adapter always uses the 32-bit Logic Status and 32-bit Feedback, at least two 32-bit words must be accounted for in the controller input image. The adapter also uses the 32-bit Logic Command and 32-bit Reference, requiring at least two 32-bit words that must be accounted for in the controller output image. If any or all of the drive's sixteen 32-bit Datalinks are used (see Setting a Master-Slave Hierarchy on page 3-8 or Custom Peer I/O Configuration on page 3-11):

- An additional 32-bit word for each **[DL From Net xx]** Datalink used to write to drive or peripheral parameters must be accounted for in the controller input image.
- An additional 32-bit word for each **[DL To Net xx]** Datalink used to read data must be accounted for in the controller output image.

Adapter Parameter Settings for MicroLogix 1100 Example

These adapter settings were used for the example ladder logic program in this section.

Adapter Parameter	Value	Description
01 - [DL From Net 01]	370	Points to drive Par. 370 - [Stop Mode A]
02 - [DL From Net 02]	371	Points to drive Par. 371 - [Stop Mode B]
03 - [DL From Net 03]	535	Points to drive Par. 535 - [Accel Time 1]
04 - [DL From Net 04]	536	Points to drive Par. 536 - [Accel Time 2]
05 - [DL From Net 05]	537	Points to drive Par. 537 - [Decel Time 1]
06 - [DL From Net 06]	538	Points to drive Par. 538 - [Decel Time 2]
07 - [DL From Net 07]	539	Points to drive Par. 539 - [Jog Acc Dec Time]
08 - [DL From Net 08]	556	Points to drive Par. 556 - [Jog Speed 1]
09 - [DL From Net 09]	557	Points to drive Par. 557 - [Jog Speed 2]
10 - [DL From Net 10]	571	Points to drive Par. 571 - [Preset Speed 1]
11 - [DL From Net 11]	572	Points to drive Par. 572 - [Preset Speed 2]
12 - [DL From Net 12]	573	Points to drive Par. 573 - [Preset Speed 3]
13 - [DL From Net 13]	574	Points to drive Par. 574 - [Preset Speed 4]
14 - [DL From Net 14]	575	Points to drive Par. 575 - [Preset Speed 5]
15 - [DL From Net 15]	576	Points to drive Par. 576 - [Preset Speed 6]
16 - [DL From Net 16]	577	Points to drive Par. 577 - [Preset Speed 7]
17 - [DL To Net 01]	370	Points to drive Par. 370 - [Stop Mode A]
18 - [DL To Net 02]	371	Points to drive Par. 371 - [Stop Mode B]
19 - [DL To Net 03]	535	Points to drive Par. 535 - [Accel Time 1]
20 - [DL To Net 04]	536	Points to drive Par. 536 - [Accel Time 2]
21 - [DL To Net 05]	537	Points to drive Par. 537 - [Decel Time 1]
22 - [DL To Net 06]	538	Points to drive Par. 538 - [Decel Time 2]
23 - [DL To Net 07]	539	Points to drive Par. 539 - [Jog Acc Dec Time]
24 - [DL To Net 08]	556	Points to drive Par. 556 - [Jog Speed 1]
25 - [DL To Net 09]	557	Points to drive Par. 557 - [Jog Speed 2]
26 - [DL To Net 10]	571	Points to drive Par. 571 - [Preset Speed 1]
27 - [DL To Net 11]	572	Points to drive Par. 572 - [Preset Speed 2]
28 - [DL To Net 12]	573	Points to drive Par. 573 - [Preset Speed 3]
29 - [DL To Net 13]	574	Points to drive Par. 574 - [Preset Speed 4]
30 - [DL To Net 14]	575	Points to drive Par. 575 - [Preset Speed 5]
31 - [DL To Net 15]	576	Points to drive Par. 576 - [Preset Speed 6]
32 - [DL To Net 16]	577	Points to drive Par. 577 - [Preset Speed 7]

TIP: The **[DL From Net xx]** parameters are inputs into the drive that come from controller outputs (for example, data to write to a drive parameter). The **[DL To Net xx]** parameters are outputs from the drive that go to controller inputs (for example, data to read a drive parameter).

Creating RSLogix 500 (v7 or higher) Project

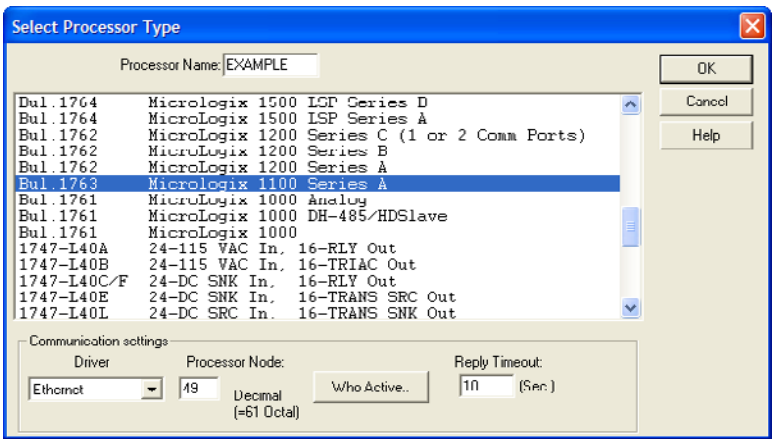
To transmit (read and write) data between the controller and drive, you must create message instructions that allocate data table addresses in the controller for Logic Command/Status, Reference/Feedback, and Datalinks. Note that three messages need to be configured. The timeout

message has to be executed first before the Logic Command, Reference, and DL to Net Datalink messages will work. For more information on N42:3 and N45 target device data table addresses, refer to N-Files on page C-8.

Selecting the Controller

1. Start RSLogix 500. The RSLogix 500 window appears. Select **File > New** to display the Select Processor Type screen (Figure 4.42).

Figure 4.42 MicroLogix 1100 Select Processor Type Screen



2. Assign a name for the processor. In the list, select the MicroLogix 1100. Then select the appropriate choices for the fields in the screen to match your application, and click **OK**. The RSLogix 500 project window appears.

Creating MicroLogix 1100 Ladder Logic for the Control Timeout

1. In the RSLogix 500 project window treeview under Program Files double-click on LAD 2.

2. Insert a ladder rung, double-click on the rung to display the rung editor, and enter **MSG MGxx:n**, where:

   xx is an unused data file number (for example, MG10:n), and
   n is an unused element of the data file chosen for xx (for example, MG10:0)

   Then press **Enter**.

3. Insert another separate rung, double-click on the rung to display the rung editor, and enter **BST XIC MGxx:n/DN NXB XIC MGxx:n/ER BND OTU MGxx:n/EN**, where:

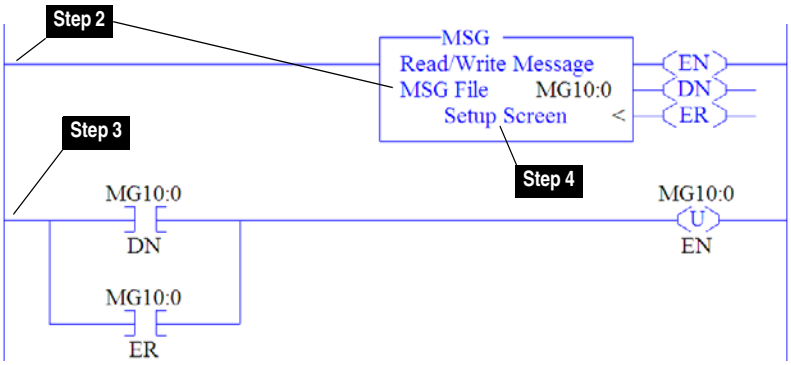
   xx and n must correspond to the assigned data file number and element (for example, MG10:0) for the message created in Step 2.

   Important: The information must be entered with appropriate numbers for “xx” and “n” for your application, and with spaces and forward slashes exactly as shown.

   Then press **Enter**.

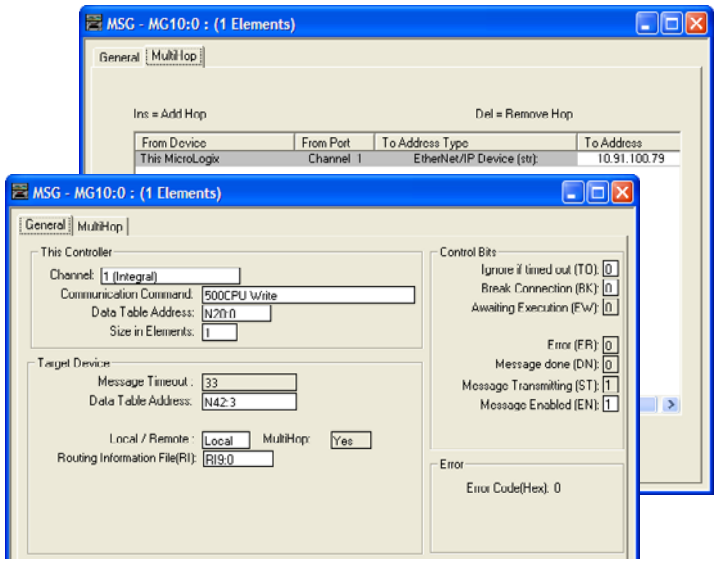
4. In the MSG instruction (Figure 4.43), double-click on Setup Screen to launch the message configuration screen (Figure 4.44).

Figure 4.43 MicroLogix 1100 Ladder Logic for the Control Timeout



5. Configure the General tab by entering or verifying the information shown in the screen.

Figure 4.44 MicroLogix 1100 Message Configuration Screens for the Control Timeout



General Tab Box	Setting
This Controller (data for MicroLogix 1100)	
Channel	**1 (integral)**. Controller port to which the EtherNet/IP network is connected.
Communication Command	**500CPU Write**. The controller type and command type for the controller to read or write data. Since the MicroLogix 1100 is part of the SLC-500 controller family, the “500CPU” controller type was selected. The “Write” command type was selected to write the control timeout value to the drive.
Data Table Address (1)	**N20:0**. An unused controller data table address containing the control timeout value to be written.
Size in Elements (2)	**1**. Number of elements (words) to be transferred. Each element size is a 16-bit integer.
Target Device (data for adapter/drive)	
Message Timeout	**5**. Message timeout duration in seconds.
Data Table Address (3)	**N42:3**. Specific starting address of the destination file in the drive.
Routing Information File	**RI9:0**. An unused routing information file for the controller.
MultiHop Tab Box	**Setting**
To Address	**10.91.100.79.** The IP address of the adapter connected to the drive.

(1) For details on data table addresses for this example project, refer to Table 5.D on page 5-15.

(2) For details to determine element size for a specific drive, refer to Understanding Controller Data Table Addresses on page 5-14.

(3) For details on setting the control timeout value and its function, see N-Files on page C-8.

Creating MicroLogix 1100 Ladder Logic for the Logic Status, Feedback, and DL From Net Datalinks

1. Insert another separate rung, double-click on the rung to display the rung editor, and enter **MSG MGxx:n**, where:

   xx is an unused data file number (for example, MG11:n), and
   n is an unused element of the data file chosen for xx (for example, MG11:0)

   Then press **Enter**.

2. Insert another separate rung, double-click on the rung to display the rung editor, and enter **BST XIC MGxx:n/DN NXB XIC MGxx:n/ER BND OTU MGxx:n/EN**, where:

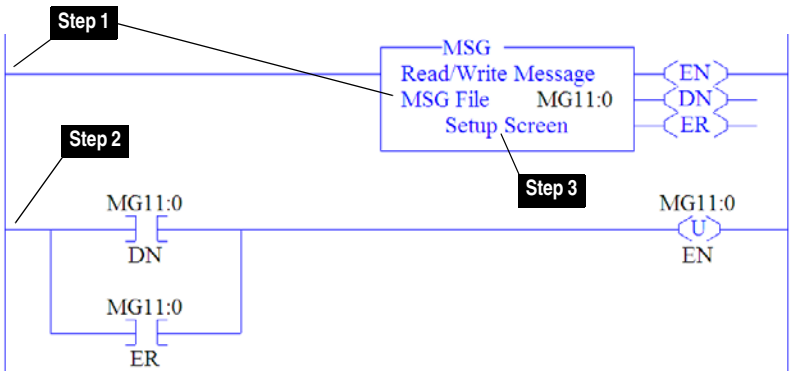
   xx and n must correspond to the assigned data file number and element (for example, MG11:0) for the message created in Step 1.

   Important: The information must be entered with appropriate numbers for “xx” and “n” for your application, and with spaces and forward slashes exactly as shown.

   Then press **Enter**.

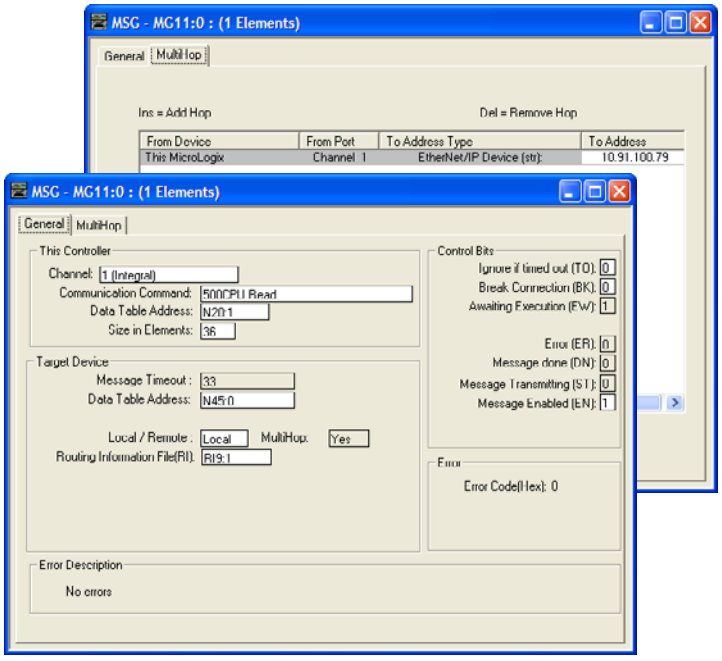
3. In the MSG instruction (Figure 4.45), double-click on Setup Screen to launch the message configuration screen (Figure 4.46).

Figure 4.45 MicroLogix 1100 Ladder Logic for the Logic Status, Feedback, and DL From Net Datalinks



4. Configure the General tab by entering or verifying the information shown in the screen.

Figure 4.46 MicroLogix 1100 Message Configuration Screens for the Logic Status, Feedback, and DL From Net Datalinks



General Tab Box	Setting
This Controller (data for MicroLogix 1100)	
Channel	**1 (integral)**. Controller port to which the EtherNet/IP network is connected.
Communication Command	**500CPU Read**. The controller type and command type for the controller to read or write data. Since the MicroLogix 1100 is part of the SLC-500 controller family, the “500CPU” controller type was selected. The “Read” command type was selected to read data from the drive.
Data Table Address (1)	**N20:1**. An unused controller data table address containing the data to be read from the drive.
Size in Elements (2)	**36**. Number of elements (words) to be transferred. Each element size is a 16-bit integer.
Target Device (data for adapter/drive)	
Message Timeout	**5**. Message timeout duration in seconds.
Data Table Address (3)	**N45:0**. Specific starting address of the source file in the drive.
Routing Information File	**RI9:1**. An unused routing information file for the controller.
MultiHop Tab Box	**Setting**
To Address	**10.91.100.79.** The IP address of the adapter connected to the drive.

(1) For details on data table addresses for this example project, refer to Table 5.D on page 5-15.

(2) For details to determine element size for a specific drive, refer to Understanding Controller Data Table Addresses on page 5-14.

(3) For N-File details, see N-Files on page C-8.

Creating MicroLogix 1100 Ladder Logic for the Logic Command, Reference, and DL To Net Datalinks

1. Insert another separate rung, double-click on the rung to display the rung editor, and enter **MSG MGxx:n**, where:

   xx is an unused data file number (for example, MG12:n), and
   n is an unused element of the data file chosen for xx (for example, MG12:0)

   Then press **Enter**.

2. Insert another separate rung, double-click on the rung to display the rung editor, and enter **BST XIC MGxx:n/DN NXB XIC MGxx:n/ER BND OTU MGxx:n/EN**, where:

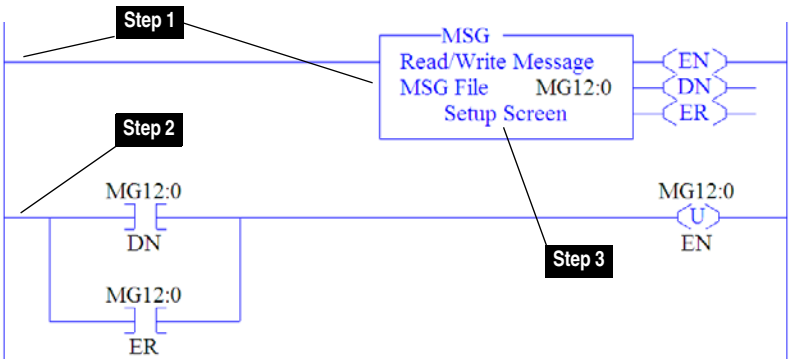
   xx and n must correspond to the assigned data file number and element (for example, MG12:0) for the message created in Step 1.

   Important: The information must be entered with appropriate numbers for "xx" and "n" for your application, and with spaces and forward slashes exactly as shown.

   Then press **Enter**.

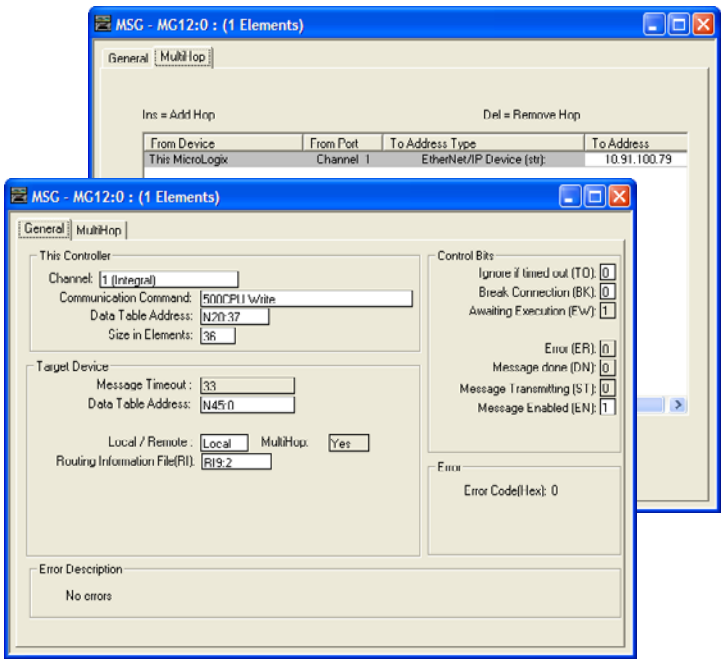
3. In the MSG instruction (Figure 4.47), double-click on Setup Screen to launch the message configuration screen (Figure 4.48).

Figure 4.47 MicroLogix 1100 Ladder Logic for the Logic Command, Reference, and DL To Net Datalinks



4. Configure the General tab by entering or verifying the information shown in the screen.

Figure 4.48 MicroLogix 1100 Message Configuration Screens for the Logic Command, Reference, and DL To Net Datalinks



General Tab Box	Setting
This Controller (data for MicroLogix 1100)	
Channel	**1 (integral)**. Controller port to which the EtherNet/IP network is connected.
Communication Command	**500CPU Write**. The controller type and command type for the controller to read or write data. Since the MicroLogix 1100 is part of the SLC-500 controller family, the “500CPU” controller type was selected. The “Write” command type was selected to write data to the drive.
Data Table Address (1)	**N20:37**. An unused controller data table address containing the data to be written to the drive.
Size in Elements (2)	**36**. Number of elements (words) to be transferred. Each element size is a 16-bit integer.
Target Device (data for adapter/drive)	
Message Timeout	**5**. Message timeout duration in seconds.
Data Table Address (3)	**N45:0**. Specific starting address of the destination file in the drive.
Routing Information File	**RI9:2**. An unused routing information file for the controller.
MultiHop Tab Box	**Setting**
To Address	**10.91.100.79.** The IP address of the adapter connected to the drive.

(1) For details on data table addresses for this example project, refer to Table 5.D on page 5-15.

(2) For details to determine element size for a specific drive, refer to Understanding Controller Data Table Addresses on page 5-14.

(3) For N-File details, see N-Files on page C-8.

TIP: If the controller is controlling more than one drive, it is recommended to intersperse the control I/O messaging for each drive to conserve network bandwidth and decrease response time. That is, sequence the message instructions for each drive so that its group of messages will occur at a different time than those for another drive.

Using the I/O

This chapter provides information and examples that explain how to control, configure, and monitor a PowerFlex 755 drive using the configured I/O.

ATTENTION: Risk of injury or equipment damage exists. The examples in this publication are intended solely for purposes of example. There are many variables and requirements with any application. Rockwell Automation, Inc. does not assume responsibility or liability (to include intellectual property liability) for actual use of the examples shown in this publication.

About I/O Messaging

On CIP-based networks, including EtherNet/IP, I/O connections are used to transfer the data which controls the PowerFlex drive and sets its Reference. I/O can also be used to transfer data to and from Datalinks in PowerFlex 755 drives.

The adapter includes the Logic Command, Logic Status, Reference, Feedback, and memory allocation for the Generic Ethernet module profile (all as 32-bit words) in the controller's I/O image. This basic I/O must always be configured in the EtherNet bridge using RSLogix. Additional I/O, if needed, can be set using up to 16 Datalinks to write data and/or up to 16 Datalinks to read data. When using any combination of these Datalinks, add one 32-bit word for each Datalink to the basic I/O Input Size and/or Output Size.

Chapter 3, Configuring the Adapter, and Chapter 4, Configuring the I/O, discuss how to configure the adapter and controller on the network for the required I/O. The Glossary defines the different options. This chapter discusses how to use I/O after you have configured the adapter and controller.

Understanding the I/O Image

The terms *input* and *output* are defined from the controller's point of view. Therefore, output I/O is data that is produced by the controller and consumed by the adapter. Input I/O is status data that is produced by the adapter and consumed as input by the controller. The I/O image will vary based on:

- How many of the drive's 32-bit Datalinks (**DL From Net 01-16** and **DL To Net 01-16**) are used.
- **ControlLogix Controllers only** — Specific drive profile used in RSLogix 5000 (Add-On Drive Profile in v16 or higher or Generic Profile in all versions of RSLogix 5000).

ControlLogix Controller Image

Since the Add-On Drive Profile in RSLogix 5000 (v16 or higher) provides descriptive controller tags, the I/O image (tag size and location) is automatically configured based on the drive being used. When using the Generic Profile in RSLogix 5000, however, controller tags are not descriptive or defined.

Table 5.A shows the I/O image when using all of the 32-bit Datalinks.

Table 5.A ControlLogix I/O Image for PowerFlex 750-Series Drives (32-bit Logic Command/Status, Reference/Feedback, and Datalinks)

DINT	Output I/O
0	Logic Command
1	Reference
2	DL From Net 01
3	DL From Net 02
4	DL From Net 03
5	DL From Net 04
6	DL From Net 05
7	DL From Net 06
8	DL From Net 07
9	DL From Net 08
10	DL From Net 09
11	DL From Net 10
12	DL From Net 11
13	DL From Net 12
14	DL From Net 13
15	DL From Net 14
16	DL From Net 15
17	DL From Net 16

DINT	Input I/O
0	Pad (1)
1	Logic Status
2	Feedback
3	DL To Net 01
4	DL To Net 02
5	DL To Net 03
6	DL To Net 04
7	DL To Net 05
8	DL To Net 06
9	DL To Net 07
10	DL To Net 08
11	DL To Net 09
12	DL To Net 10
13	DL To Net 11
14	DL To Net 12
15	DL To Net 13
16	DL To Net 14
17	DL To Net 15
18	DL To Net 16

(1) This is only required when the Generic Profile is used. The Add-On Profile automatically accounts for this and, therefore, hides the Pad in the I/O input image.

PLC-5, SLC 500, and MicroLogix 1100 Controller Image

The I/O image for the PLC-5, SLC 500, and MicroLogix 1100 controller changes depending on how many of the drive's sixteen 32-bit Datalinks are used.

TIP: Since PLC-5, SLC 500, and MicroLogix 1100 controllers are 16-bit devices, each 32-bit word for the Logic Command/Status, Reference/Feedback, and any used Datalinks will consume two contiguous words (Least and Most Significant) in the controller's I/O image. Table 5.B shows the I/O for a drive using all 32-bit Datalinks.

Table 5.B PLC-5, SLC 500, and MicroLogix 1100 I/O Image for PowerFlex 750-Series Drives (32-bit Logic Command/Status, Reference/ Feedback, and Datalinks)

Word	Output I/O
0	Logic Command (LSW)
1	Logic Command (MSW)
2	Reference (LSW)
3	Reference (MSW)
4	DL From Net 01 (LSW)
5	DL From Net 01 (MSW)
6	DL From Net 02 (LSW)
7	DL From Net 02 (MSW)
8	DL From Net 03 (LSW)
9	DL From Net 03 (MSW)
10	DL From Net 04 (LSW)
11	DL From Net 04 (MSW)
12	DL From Net 05 (LSW)
13	DL From Net 05 (MSW)
14	DL From Net 06 (LSW)
15	DL From Net 06 (MSW)
16	DL From Net 07 (LSW)
17	DL From Net 07 (MSW)
18	DL From Net 08 (LSW)
19	DL From Net 08 (MSW)
20	DL From Net 09 (LSW)
21	DL From Net 09 (MSW)
22	DL From Net 10 (LSW)
23	DL From Net 10 (MSW)
24	DL From Net 11 (LSW)
25	DL From Net 11 (MSW)
26	DL From Net 12 (LSW)
27	DL From Net 12 (MSW)
28	DL From Net 13 (LSW)
29	DL From Net 13 (MSW)
30	DL From Net 14 (LSW)
31	DL From Net 14 (MSW)
32	DL From Net 15 (LSW)
33	DL From Net 15 (MSW)
34	DL From Net 16 (LSW)
35	DL From Net 16 (MSW)

Word	Input I/O
0	Logic Status (LSW)
1	Logic Status (MSW)
2	Feedback (LSW)
3	Feedback (MSW)
4	DL To Net 01 (LSW)
5	DL To Net 01 (MSW)
6	DL To Net 02 (LSW)
7	DL To Net 02 (MSW)
8	DL To Net 03 (LSW)
9	DL To Net 03 (MSW)
10	DL To Net 04 (LSW)
11	DL To Net 04 (MSW)
12	DL To Net 05 (LSW)
13	DL To Net 05 (MSW)
14	DL To Net 06 (LSW)
15	DL To Net 06 (MSW)
16	DL To Net 07 (LSW)
17	DL To Net 07 (MSW)
18	DL To Net 08 (LSW)
19	DL To Net 08 (MSW)
20	DL To Net 09 (LSW)
21	DL To Net 09 (MSW)
22	DL To Net 10 (LSW)
23	DL To Net 10 (MSW)
24	DL To Net 11 (LSW)
25	DL To Net 11 (MSW)
26	DL To Net 12 (LSW)
27	DL To Net 12 (MSW)
28	DL To Net 13 (LSW)
29	DL To Net 13 (MSW)
30	DL To Net 14 (LSW)
31	DL To Net 14 (MSW)
32	DL To Net 15 (LSW)
33	DL To Net 15 (MSW)
34	DL To Net 16 (LSW)
35	DL To Net 16 (MSW)

Using Logic Command/ Status

The *Logic Command* is a 32-bit word of control data produced by the controller and consumed by the adapter. The *Logic Status* is a 32-bit word of status data produced by the adapter and consumed by the controller.

When using a ControlLogix controller, the Logic Command word is always DINT 0 in the output image and the Logic Status word is always DINT 0 in the input image when using the Add-On Profile or DINT 1 when using the Generic Profile. For a PLC-5, SLC 500 or MicroLogix 1100 controller, the Logic Command word is always words 0 (least significant word) and 1 (most significant word) in the output image and the Logic Status word is always words 0 (least significant word) and 1 (most significant word) in the input image.

This manual contains the bit definitions for compatible products available at the time of publication in Appendix D, Logic Command/ Status Words for PowerFlex 750-Series Drives.

Using Reference/Feedback

The *Reference* is a 32-bit REAL (floating point) piece of control data produced by the controller and consumed by the adapter. The *Feedback* is a 32-bit REAL (floating point) piece of status data produced by the adapter and consumed by the controller.

The Reference and Feedback 32-bit values represent engineering units. For example, a 32-bit REAL Reference value of “30.0” equals a Reference of 30.0 Hz. Note that the commanded maximum speed can never exceed the value of drive Parameter 510 - [Max Fwd Speed]. Table 5.C shows example References and their results for a PowerFlex 755 drive that has its Parameter 37 - [Maximum Freq] set to 130 Hz and Parameter 520 - [Max Fwd Speed] set to 60 Hz.

When using a ControlLogix controller, the 32-bit REAL Reference is always DINT 1 in the output image and the 32-bit REAL Feedback is always DINT 1 in the input image when using the Add-On Profile or DINT 2 when using the Generic Profile. For a PLC-5, SLC 500 or MicroLogix 1100 controller, the 32-bit REAL Reference word is always words 2 (least significant word) and 3 (most significant word) in the output image and the 32-bit REAL Feedback is always words 2 (least significant word) and 3 (most significant word) in the input image. Because the I/O image is integer-based and the Reference and Feedback are floating point, a COP (Copy) instruction or UDDT is required to correctly write values to the Reference and read values from the Feedback. See the ladder logic program examples in Figure 5.8 and Figure 5.9.

TIP: When using the drive-specific add-on profile, the controller tags for Reference and Feedback are automatically and properly formatted. This eliminates the need for data conversion using COP (copy) instructions or a UDDT.

Table 5.C Example Speed Reference/Feedback Scaling for PowerFlex 750-Series Drives

Network Reference Value	Speed Command Value	Output Speed	Network Feedback Value
130.0	130 Hz	60 Hz (2)	60.0
65.0	65 Hz	60 Hz (2)	60.0
32.5	32.5 Hz	32.5 Hz	32.5
0.0	0 Hz	0 Hz	0.0
-32.5 (1)	32.5 Hz	32.5 Hz	32.5

(1) The effects of values less than 0.0 depend on whether the PowerFlex 755 drive uses a bipolar or unipolar direction mode. Refer to the drive User Manual for details.

(2) The drive runs at 60 Hz instead of 130 Hz or 65 Hz because drive Parameter 520 - [Max Fwd Speed] sets 60 Hz as the maximum speed.

Using Datalinks

A Datalink is a mechanism used by PowerFlex drives to transfer data to and from the controller. Datalinks allow a drive parameter value to be changed without using an Explicit Message. When enabled, each Datalink occupies one 32-bit word in a ControlLogix controller or two 16-bit words in a PLC-5, SLC 500 or MicroLogix 1100 controller.

The following rules apply when using PowerFlex 750-Series drive Datalinks:

- The target of a Datalink can be any Host parameter, including those of a peripheral. For example, drive parameter 535 - [Accel Time 1] can be assigned to the embedded adapter and any or all of the Option Cards installed in the drive.
- The data passed through the drive's Datalink mechanism is determined by the settings of adapter **Parameters 01-16 - [DL From Net 01-16]** and **Parameters 17-32 - [DL To Net 01-16]**.
- When a Datalink I/O connection is active, that Datalink is locked and cannot be changed until that I/O connection becomes idle or inactive.
- When you use a Datalink to change a value, the value is NOT written to the Non-Volatile Storage (NVS). The value is stored in volatile memory and lost when the drive loses power. Thus, use Datalinks when you need to change a value of a parameter frequently.

Datalinks for PowerFlex 750-Series peripherals (embedded EtherNet/IP adapter and option modules such as an encoder or a communication module) are locked when the peripheral has an I/O connection with a controller. When a controller has an I/O connection to the drive, the drive does not allow a reset to defaults, configuration download or anything else that could change the makeup of the I/O connection in a running system. The I/O connection with the controller must first be disabled to allow changes to the respective Datalinks.

Depending on the controller being used, the I/O connection can be disabled by:

- Inhibiting the module in RSLogix 5000
- Putting the controller in Program mode
- Placing the scanner in idle mode
- Disconnecting the drive from the network

DeviceLogix Datalinks are also locked while the DeviceLogix program is running. The DeviceLogix program must first be disabled to allow changes to the Datalinks. Set DeviceLogix parameter 53 - [DLX Operation] to “DisableLogic” to disable the logic (the parameter value will then change to “LogicDisabld”).

Example Ladder Logic Program Information

The example ladder logic programs in the sections of this chapter are intended for and operate PowerFlex 750-Series drives.

Functions of the Example Programs

The example programs enable you to:

- Receive Logic Status information from the drive.
- Send a Logic Command to control the drive (for example, start, stop).
- Send a Reference to the drive and receive Feedback from the drive.
- Send/receive Datalink data to/from the drive.

Logic Command/Status Words

These examples use the Logic Command word and Logic Status word for PowerFlex 750-Series drives. Refer to Appendix D, Logic Command/Status Words for PowerFlex 750-Series Drives to view details.

ControlLogix Example

Creating Ladder Logic Using the RSLogix 5000 Add-On Drive Profiles (v16 or higher)

Since the Add-On Drive Profile automatically created descriptive controller tags (Figure 4.13) for the entire I/O image in Chapter 4, you can use these tags to directly control and monitor the drive without creating any ladder logic program. However, if you intend to use Human Machine Interface devices (PanelView, etc.) to operate the drive and view its status, you will need to create descriptive user-defined Program tags (Figure 5.1) and a ladder logic program that will pass the Controller tag data to the Program tags.

Figure 5.1 ControlLogix Program Tags for Integrated Drive Profile Ladder Logic Program Example

Name	Value	Data Type
Status_Reverse	0	BOOL
Status_Ready	0	BOOL
Status_Forward	0	BOOL
Status_Faulted	0	BOOL
Status_At_Speed	0	BOOL
Status_Active	0	BOOL
Speed_Reference	0.0	REAL
Speed_Feedback	0.0	REAL
Command_Stop	0	BOOL
Command_Start	0	BOOL
Command_Jog	0	BOOL
Command_Forward_Reverse	0	BOOL
Command_Clear_Faults	0	BOOL

An example ladder logic program that uses the automatically-created descriptive Controller tags and passes their data to the user-defined Program tags is shown in Figure 5.2 and Figure 5.3. Note that the prefix for the drive Controller tags is determined by the name assigned when configuring the I/O (Chapter 4).

Figure 5.2 ControlLogix Example Ladder Logic Program Using Add-On Drive Profiles for Logic Status/Feedback

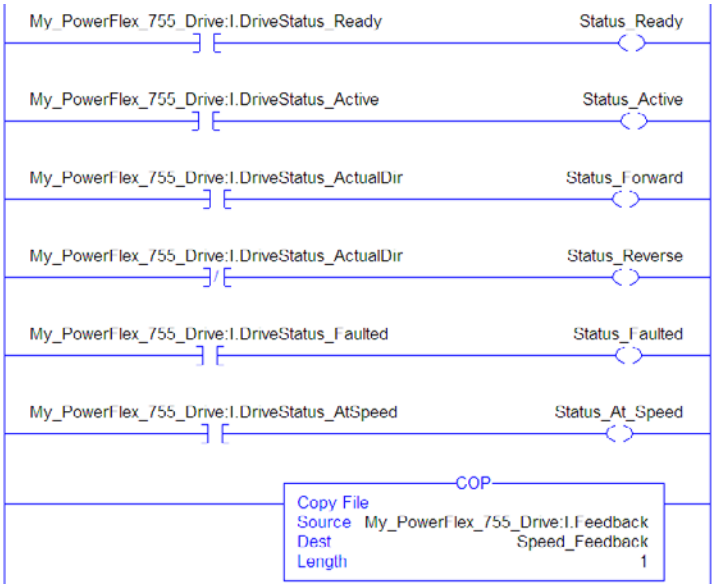
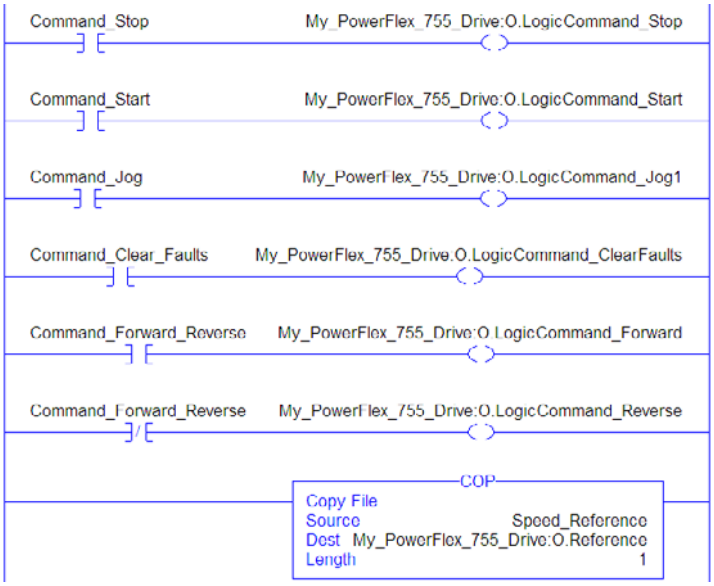


Figure 5.3 ControlLogix Example Ladder Logic Program Using Add-On Drive Profiles for Logic Command/Reference



Creating Ladder Logic Using the RSLogix 5000 Generic Profile (all versions)

Adapter Parameter Settings for ControlLogix Example

These adapter settings were used for the example ladder logic program in this section.

Adapter Parameter	Value	Description
01 - [DL From Net 01]	370	Points to drive Par. 370 - [Stop Mode A]
02 - [DL From Net 02]	371	Points to drive Par. 371 - [Stop Mode B]
03 - [DL From Net 03]	535	Points to drive Par. 535 - [Accel Time 1]
04 - [DL From Net 04]	536	Points to drive Par. 536 - [Accel Time 2]
05 - [DL From Net 05]	537	Points to drive Par. 537 - [Decel Time 1]
06 - [DL From Net 06]	538	Points to drive Par. 538 - [Decel Time 2]
07 - [DL From Net 07]	539	Points to drive Par. 539 - [Jog Acc Dec Time]
08 - [DL From Net 08]	556	Points to drive Par. 556 - [Jog Speed 1]
09 - [DL From Net 09]	557	Points to drive Par. 557 - [Jog Speed 2]
10 - [DL From Net 10]	571	Points to drive Par. 571 - [Preset Speed 1]
11 - [DL From Net 11]	572	Points to drive Par. 572 - [Preset Speed 2]
12 - [DL From Net 12]	573	Points to drive Par. 573 - [Preset Speed 3]
13 - [DL From Net 13]	574	Points to drive Par. 574 - [Preset Speed 4]
14 - [DL From Net 14]	575	Points to drive Par. 575 - [Preset Speed 5]
15 - [DL From Net 15]	576	Points to drive Par. 576 - [Preset Speed 6]
16 - [DL From Net 16]	577	Points to drive Par. 577 - [Preset Speed 7]
17 - [DL To Net 01]	370	Points to drive Par. 370 - [Stop Mode A]
18 - [DL To Net 02]	371	Points to drive Par. 371 - [Stop Mode B]
19 - [DL To Net 03]	535	Points to drive Par. 535 - [Accel Time 1]
20 - [DL To Net 04]	536	Points to drive Par. 536 - [Accel Time 2]

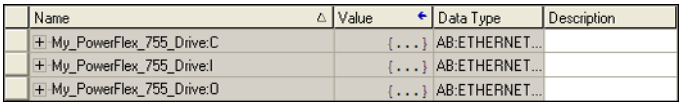
Adapter Parameter	Value	Description
21 - [DL To Net 05]	537	Points to drive Par. 537 - [Decel Time 1]
22 - [DL To Net 06]	538	Points to drive Par. 538 - [Decel Time 2]
23 - [DL To Net 07]	539	Points to drive Par. 539 - [Jog Acc Dec Time]
24 - [DL To Net 08]	556	Points to drive Par. 556 - [Jog Speed 1]
25 - [DL To Net 09]	557	Points to drive Par. 557 - [Jog Speed 2]
26 - [DL To Net 10]	571	Points to drive Par. 571 - [Preset Speed 1]
27 - [DL To Net 11]	572	Points to drive Par. 572 - [Preset Speed 2]
28 - [DL To Net 12]	573	Points to drive Par. 573 - [Preset Speed 3]
29 - [DL To Net 13]	574	Points to drive Par. 574 - [Preset Speed 4]
30 - [DL To Net 14]	575	Points to drive Par. 575 - [Preset Speed 5]
31 - [DL To Net 15]	576	Points to drive Par. 576 - [Preset Speed 6]
32 - [DL To Net 16]	577	Points to drive Par. 577 - [Preset Speed 7]

TIP: The **[DL From Net xx]** parameters are inputs into the drive that come from controller outputs (for example, data to write to a drive parameter). The **[DL To Net xx]** parameters are outputs from the drive that go to controller inputs (for example, data to read a drive parameter).

Controller Tags

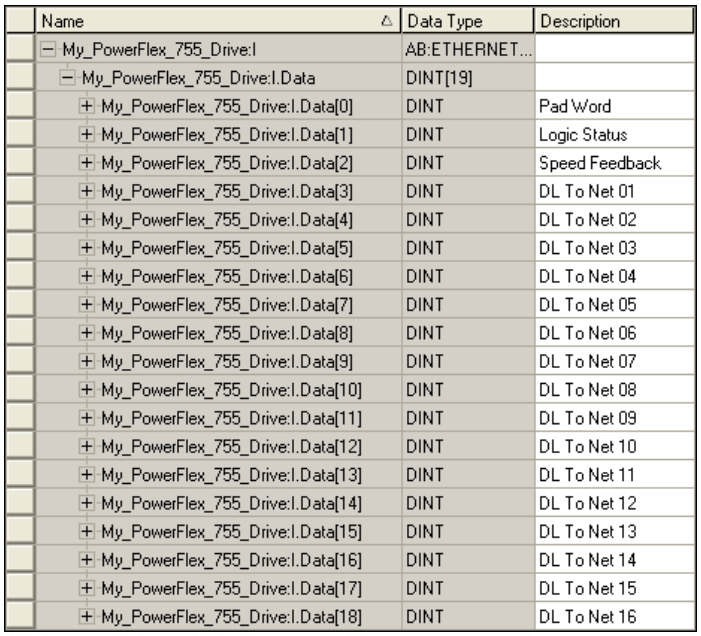
When you add the adapter and drive to the I/O configuration (Chapter 4), RSLogix 5000 automatically creates generic (non-descriptive) controller tags. In this example program, the following controller tags are used.

Figure 5.4 ControlLogix Controller Tags for Generic Drive Profile Example Ladder Logic Program



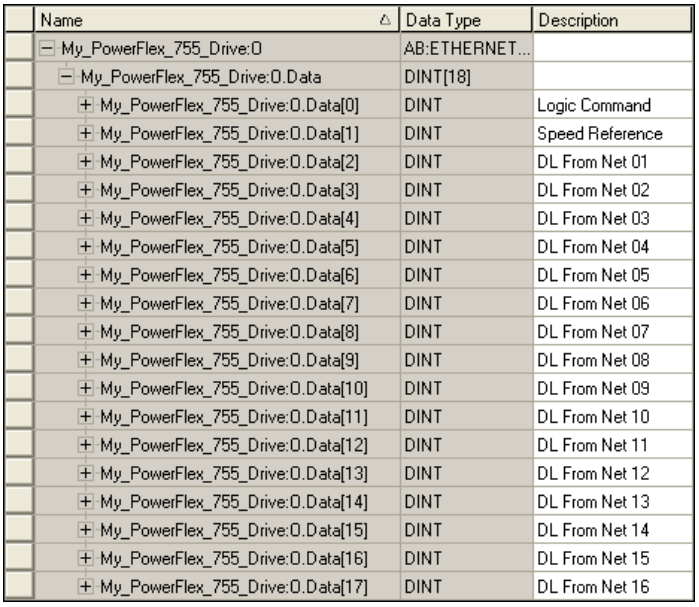
You can expand the Output and Input tags to reveal the output and input configuration. The Input tag for this example requires nineteen 32-bit words of data (see Figure 5.5). The Output tag for this example program requires eighteen 32-bit words of data (see Figure 5.6).

Figure 5.5 ControlLogix Input Image for Generic Drive Profile Example Ladder Logic Program



Name	Data Type	Description
My_PowerFlex_755_Drive:I	AB:ETHERNET...	
My_PowerFlex_755_Drive:I.Data	DINT[19]	
My_PowerFlex_755_Drive:I.Data[0]	DINT	Pad Word
My_PowerFlex_755_Drive:I.Data[1]	DINT	Logic Status
My_PowerFlex_755_Drive:I.Data[2]	DINT	Speed Feedback
My_PowerFlex_755_Drive:I.Data[3]	DINT	DL To Net 01
My_PowerFlex_755_Drive:I.Data[4]	DINT	DL To Net 02
My_PowerFlex_755_Drive:I.Data[5]	DINT	DL To Net 03
My_PowerFlex_755_Drive:I.Data[6]	DINT	DL To Net 04
My_PowerFlex_755_Drive:I.Data[7]	DINT	DL To Net 05
My_PowerFlex_755_Drive:I.Data[8]	DINT	DL To Net 06
My_PowerFlex_755_Drive:I.Data[9]	DINT	DL To Net 07
My_PowerFlex_755_Drive:I.Data[10]	DINT	DL To Net 08
My_PowerFlex_755_Drive:I.Data[11]	DINT	DL To Net 09
My_PowerFlex_755_Drive:I.Data[12]	DINT	DL To Net 10
My_PowerFlex_755_Drive:I.Data[13]	DINT	DL To Net 11
My_PowerFlex_755_Drive:I.Data[14]	DINT	DL To Net 12
My_PowerFlex_755_Drive:I.Data[15]	DINT	DL To Net 13
My_PowerFlex_755_Drive:I.Data[16]	DINT	DL To Net 14
My_PowerFlex_755_Drive:I.Data[17]	DINT	DL To Net 15
My_PowerFlex_755_Drive:I.Data[18]	DINT	DL To Net 16

Figure 5.6 ControlLogix Output Image for Generic Drive Profile Example Ladder Logic Program



Name	Data Type	Description
My_PowerFlex_755_Drive:O	AB:ETHERNET...	
My_PowerFlex_755_Drive:O.Data	DINT[18]	
My_PowerFlex_755_Drive:O.Data[0]	DINT	Logic Command
My_PowerFlex_755_Drive:O.Data[1]	DINT	Speed Reference
My_PowerFlex_755_Drive:O.Data[2]	DINT	DL From Net 01
My_PowerFlex_755_Drive:O.Data[3]	DINT	DL From Net 02
My_PowerFlex_755_Drive:O.Data[4]	DINT	DL From Net 03
My_PowerFlex_755_Drive:O.Data[5]	DINT	DL From Net 04
My_PowerFlex_755_Drive:O.Data[6]	DINT	DL From Net 05
My_PowerFlex_755_Drive:O.Data[7]	DINT	DL From Net 06
My_PowerFlex_755_Drive:O.Data[8]	DINT	DL From Net 07
My_PowerFlex_755_Drive:O.Data[9]	DINT	DL From Net 08
My_PowerFlex_755_Drive:O.Data[10]	DINT	DL From Net 09
My_PowerFlex_755_Drive:O.Data[11]	DINT	DL From Net 10
My_PowerFlex_755_Drive:O.Data[12]	DINT	DL From Net 11
My_PowerFlex_755_Drive:O.Data[13]	DINT	DL From Net 12
My_PowerFlex_755_Drive:O.Data[14]	DINT	DL From Net 13
My_PowerFlex_755_Drive:O.Data[15]	DINT	DL From Net 14
My_PowerFlex_755_Drive:O.Data[16]	DINT	DL From Net 15
My_PowerFlex_755_Drive:O.Data[17]	DINT	DL From Net 16

Program Tags

To use the Controller tags that are automatically created, you need to create the following Program tags for this example program.

Figure 5.7 ControlLogix Program Tags for Generic Drive Profile Example Ladder Logic Program

Name	Value	Data Type	Description
Command_Clear_Faults	0	BOOL	
Command_Forward_Reverse	0	BOOL	
Command_Jog	0	BOOL	
Command_Start	0	BOOL	
Command_Stop	0	BOOL	
Speed_Feedback	0.0	REAL	
Speed_Reference	0.0	REAL	
Status_Active	0	BOOL	
Status_At_Speed	0	BOOL	
Status_Faulted	0	BOOL	
Status_Forward	0	BOOL	
Status_Ready	0	BOOL	
Status_Reverse	0	BOOL	

Figure 5.8 ControlLogix Example Ladder Logic Program Using Generic Drive Profiles for Logic Status/Feedback

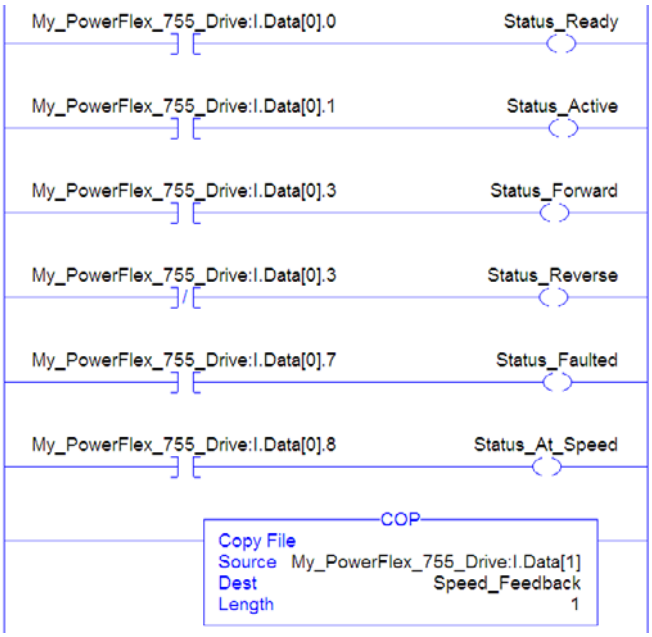
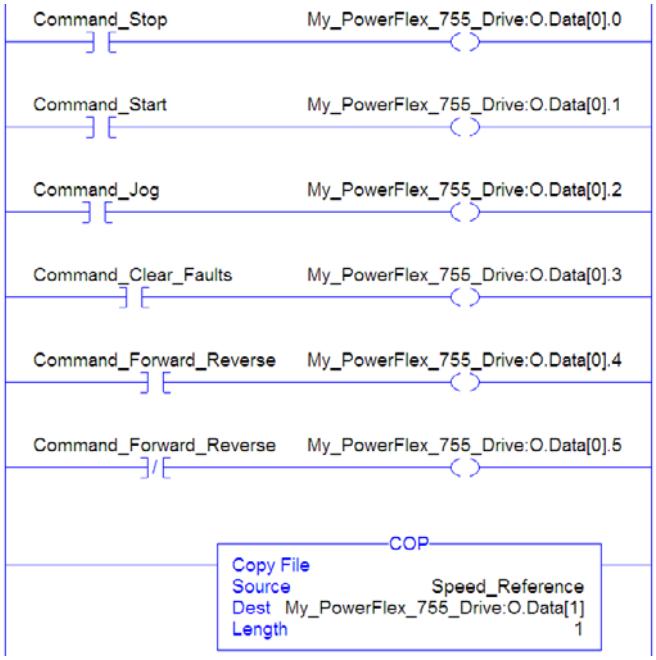


Figure 5.9 ControlLogix Example Ladder Logic Program Using Generic Drive Profiles for Logic Command/Reference



Example Datalink Data

The Datalink data used in the example program is shown in Figure 5.10. Note that to describe the parameters to which the Datalinks are assigned, you may want to add descriptions to the automatically-created generic controller tags or create User Defined Data Types (UDDT).

Figure 5.10 ControlLogix Example Datalinks for Ladder Logic Program Using Generic Drive Profile

Name	Value	Style	Data Type
DL_From_Net	{...}		DL_From_Net
DL_From_Net._01_Stop_Mode_A	1	Decimal	DINT
DL_From_Net._02_Stop_Mode_B	2	Decimal	DINT
DL_From_Net._03_Accel_Time_1	2.5	Float	REAL
DL_From_Net._04_Accel_Time_2	5.0	Float	REAL
DL_From_Net._05_Decel_Time_1	7.5	Float	REAL
DL_From_Net._06_Decel_Time_2	10.0	Float	REAL
DL_From_Net._07_Jog_Acc_Dec_Time	12.5	Float	REAL
DL_From_Net._08_Jog_Speed_1	10.0	Float	REAL
DL_From_Net._09_Jog_Speed_2	15.0	Float	REAL
DL_From_Net._10_Preset_Speed_1	20.0	Float	REAL
DL_From_Net._11_Preset_Speed_2	25.0	Float	REAL
DL_From_Net._12_Preset_Speed_3	30.0	Float	REAL
DL_From_Net._13_Preset_Speed_4	35.0	Float	REAL
DL_From_Net._14_Preset_Speed_5	40.0	Float	REAL
DL_From_Net._15_Preset_Speed_6	45.0	Float	REAL
DL_From_Net._16_Preset_Speed_7	50.0	Float	REAL
DL_To_Net	{...}		DL_To_Net
DL_To_Net._01_Stop_Mode_A	1	Decimal	DINT
DL_To_Net._02_Stop_Mode_B	2	Decimal	DINT
DL_To_Net._03_Accel_Time_1	2.5	Float	REAL
DL_To_Net._04_Accel_Time_2	5.0	Float	REAL
DL_To_Net._05_Decel_Time_1	7.5	Float	REAL
DL_To_Net._06_Decel_Time_2	10.0	Float	REAL
DL_To_Net._07_Jog_Acc_Dec_Time	12.5	Float	REAL
DL_To_Net._08_Jog_Speed_1	10.0	Float	REAL
DL_To_Net._09_Jog_Speed_2	15.0	Float	REAL
DL_To_Net._10_Preset_Speed_1	20.0	Float	REAL
DL_To_Net._11_Preset_Speed_2	25.0	Float	REAL
DL_To_Net._12_Preset_Speed_3	30.0	Float	REAL
DL_To_Net._13_Preset_Speed_4	35.0	Float	REAL
DL_To_Net._14_Preset_Speed_5	40.0	Float	REAL
DL_To_Net._15_Preset_Speed_6	45.0	Float	REAL
DL_To_Net._16_Preset_Speed_7	50.0	Float	REAL

TIP: To determine the Data Type of a parameter, refer to the Data Type column in the chapter containing parameters in the *PowerFlex 750-Series AC Drives User Manual, publication 750-UM001*.

PLC-5, SLC 500, and MicroLogix 1100 Example

Adapter Parameter Settings

For the adapter settings used for the example ladder logic program in this section:

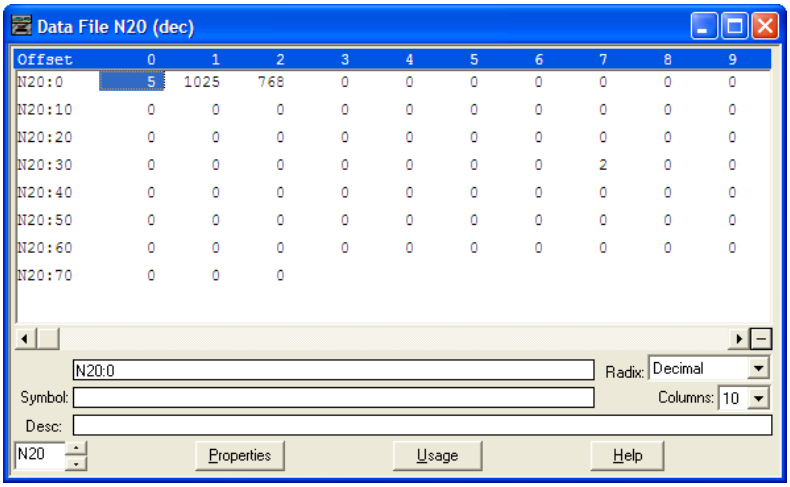
Refer to the Table on...	Controller Type
page 4-22	PLC-5
page 4-30	SLC 500
page 4-38	MicroLogix 1100

Understanding Controller Data Table Addresses

Since PLC-5, SLC 500, and MicroLogix 1100 controllers are 16-bit platforms being used with the 32-bit embedded EtherNet/IP adapter, the data will be transposed from the least significant word (LSW) to the most significant word (MSW) in the controller.

When the I/O was configured (Chapter 4), an available data table file (N20) was used. Figure 5.11 shows the entire data file address structure for this example.

Figure 5.11 Data File Table for Example Ladder Logic Program



Important: The N20:0 data table address in this example is used to set a control timeout value (in seconds) which determines how long it will take the adapter to detect a communication loss. Enter a valid value between 1 - 32767 for N20:0. A value of zero (0) is not valid, since it disables the timeout and all I/O messages (Logic Command/Status, Reference/Feedback, and Datalinks) intended for the drive will not execute.

Table 5.D shows the I/O definitions as they relate to the N20 data table file (Figure 5.11) being used in this example.

For PowerFlex 750-Series drives, which contain both DINT (32-bit format) and REAL (floating point format) data types, you will always read from and write to the LSW data table address in the controller first. Then if the data value exceeds 16 bits, the remaining value will be in the MSW data table address.

Table 5.D PLC-5, SLC 500, and MicroLogix 1100 Data Table Addresses for PowerFlex 750-Series Drives

Data Table Address	Description
N20:0	Control Timeout
N20:1	Logic Status (LSW, see Appendix D)
N20:2	Logic Status (MSW, see Appendix D)
N20:3	Speed Feedback LSW
N20:4	Speed Feedback MSW
N20:5	Value of parameter assigned to adapter Parameter 17 [DL To Net 01] LSW
N20:6	Value of parameter assigned to adapter Parameter 17 [DL To Net 01] MSW
N20:7	Value of parameter assigned to adapter Parameter 18 [DL To Net 02] LSW
N20:8	Value of parameter assigned to adapter Parameter 18 [DL To Net 02] MSW
N20:9	Value of parameter assigned to adapter Parameter 19 [DL To Net 03] LSW
N20:10	Value of parameter assigned to adapter Parameter 19 [DL To Net 03] MSW
N20:11	Value of parameter assigned to adapter Parameter 20 [DL To Net 04] LSW
N20:12	Value of parameter assigned to adapter Parameter 20 [DL To Net 04] MSW
N20:13	Value of parameter assigned to adapter Parameter 21 [DL To Net 05] LSW
N20:14	Value of parameter assigned to adapter Parameter 21 [DL To Net 05] MSW
N20:15	Value of parameter assigned to adapter Parameter 22 [DL To Net 06] LSW
N20:16	Value of parameter assigned to adapter Parameter 22 [DL To Net 06] MSW
N20:17	Value of parameter assigned to adapter Parameter 23 [DL To Net 07] LSW
N20:18	Value of parameter assigned to adapter Parameter 23 [DL To Net 07] MSW
N20:19	Value of parameter assigned to adapter Parameter 24 [DL To Net 08] LSW
N20:20	Value of parameter assigned to adapter Parameter 24 [DL To Net 08] MSW
N20:21	Value of parameter assigned to adapter Parameter 25 [DL To Net 09] LSW
N20:22	Value of parameter assigned to adapter Parameter 25 [DL To Net 09] MSW
N20:23	Value of parameter assigned to adapter Parameter 26 [DL To Net 10] LSW
N20:24	Value of parameter assigned to adapter Parameter 26 [DL To Net 10] MSW
N20:25	Value of parameter assigned to adapter Parameter 27 [DL To Net 11] LSW
N20:26	Value of parameter assigned to adapter Parameter 27 [DL To Net 11] MSW
N20:27	Value of parameter assigned to adapter Parameter 28 [DL To Net 12] LSW
N20:28	Value of parameter assigned to adapter Parameter 28 [DL To Net 12] MSW
N20:29	Value of parameter assigned to adapter Parameter 29 [DL To Net 13] LSW
N20:30	Value of parameter assigned to adapter Parameter 29 [DL To Net 13] MSW
N20:31	Value of parameter assigned to adapter Parameter 30 [DL To Net 14] LSW
N20:32	Value of parameter assigned to adapter Parameter 30 [DL To Net 14] MSW
N20:33	Value of parameter assigned to adapter Parameter 31 [DL To Net 15] LSW
N20:34	Value of parameter assigned to adapter Parameter 31 [DL To Net 15] MSW
N20:35	Value of parameter assigned to adapter Parameter 32 [DL To Net 16] LSW
N20:36	Value of parameter assigned to adapter Parameter 32 [DL To Net 16] MSW
N20:37	Logic Command (LSW, see Appendix D)
N20:38	Logic Command (MSW, see Appendix D)
N20:39	Speed Reference LSW
N20:40	Speed Reference MSW
N20:41	Value of parameter assigned to adapter Parameter 01 [DL From Net 01] LSW
N20:42	Value of parameter assigned to adapter Parameter 01 [DL From Net 01] MSW
N20:43	Value of parameter assigned to adapter Parameter 02 [DL From Net 02] LSW
N20:44	Value of parameter assigned to adapter Parameter 02 [DL From Net 02] MSW
N20:45	Value of parameter assigned to adapter Parameter 03 [DL From Net 03] LSW
N20:46	Value of parameter assigned to adapter Parameter 03 [DL From Net 03] MSW
N20:47	Value of parameter assigned to adapter Parameter 04 [DL From Net 04] LSW
N20:48	Value of parameter assigned to adapter Parameter 04 [DL From Net 04] MSW
N20:49	Value of parameter assigned to adapter Parameter 05 [DL From Net 05] LSW
N20:50	Value of parameter assigned to adapter Parameter 05 [DL From Net 05] MSW

Table 5.D PLC-5, SLC 500, and MicroLogix 1100 Data Table Addresses for PowerFlex 750-Series Drives (Continued)

Data Table Address	Description
N20:51	Value of parameter assigned to adapter Parameter 06 [DL From Net 06] LSW
N20:52	Value of parameter assigned to adapter Parameter 06 [DL From Net 06] MSW
N20:53	Value of parameter assigned to adapter Parameter 07 [DL From Net 07] LSW
N20:54	Value of parameter assigned to adapter Parameter 07 [DL From Net 07] MSW
N20:55	Value of parameter assigned to adapter Parameter 08 [DL From Net 08] LSW
N20:56	Value of parameter assigned to adapter Parameter 08 [DL From Net 08] MSW
N20:57	Value of parameter assigned to adapter Parameter 09 [DL From Net 09] LSW
N20:58	Value of parameter assigned to adapter Parameter 09 [DL From Net 09] MSW
N20:59	Value of parameter assigned to adapter Parameter 10 [DL From Net 10] LSW
N20:60	Value of parameter assigned to adapter Parameter 10 [DL From Net 10] MSW
N20:61	Value of parameter assigned to adapter Parameter 11 [DL From Net 11] LSW
N20:62	Value of parameter assigned to adapter Parameter 11 [DL From Net 11] MSW
N20:63	Value of parameter assigned to adapter Parameter 12 [DL From Net 12] LSW
N20:64	Value of parameter assigned to adapter Parameter 12 [DL From Net 12] MSW
N20:65	Value of parameter assigned to adapter Parameter 13 [DL From Net 13] LSW
N20:66	Value of parameter assigned to adapter Parameter 13 [DL From Net 13] MSW
N20:67	Value of parameter assigned to adapter Parameter 14 [DL From Net 14] LSW
N20:68	Value of parameter assigned to adapter Parameter 14 [DL From Net 14] MSW
N20:69	Value of parameter assigned to adapter Parameter 15 [DL From Net 15] LSW
N20:70	Value of parameter assigned to adapter Parameter 15 [DL From Net 15] MSW
N20:71	Value of parameter assigned to adapter Parameter 16 [DL From Net 16] LSW
N20:72	Value of parameter assigned to adapter Parameter 16 [DL From Net 16] MSW

TIP: Remember that most of the parameters in the drive being read/written with the Datalinks are REAL (floating point) data types. Therefore, use a COP (Copy) instruction to convert the least significant word and most significant word values to a single floating point register (Fx:x).

You can use the controller data table addresses to directly control and monitor the drive without creating any ladder logic program. However, if you intend to use Human Machine Interface devices (PanelView, etc.) to operate the drive and view its status, you may want to create alternate controller data table addresses (Table 5.E and Table 5.F) and a ladder logic program that will pass that data to the data table addresses used for messaging.

Table 5.E Controller and Program Data Table Address Descriptions for Example Logic Status/Feedback Ladder Logic Program

Description	Controller Data Table Address
Drive Ready	N20:1/0
Drive Active	N20:1/1
Actual Direction Forward (XIO)	N20:1/3
Actual Direction Reverse (XIC)	N20:1/3
Drive Faulted	N20:1/7
Drive At Speed	N20:1/8
Speed Feedback	N20:3

Description	Program Data Table Address
Status Ready	B3:1/0
Status Active	B3:1/1
Status Forward	B3:1/3
Status Reverse	B3:1/4
Status Faulted	B3:1/7
Status At Speed	B3:1/8
Speed Feedback	B30:3

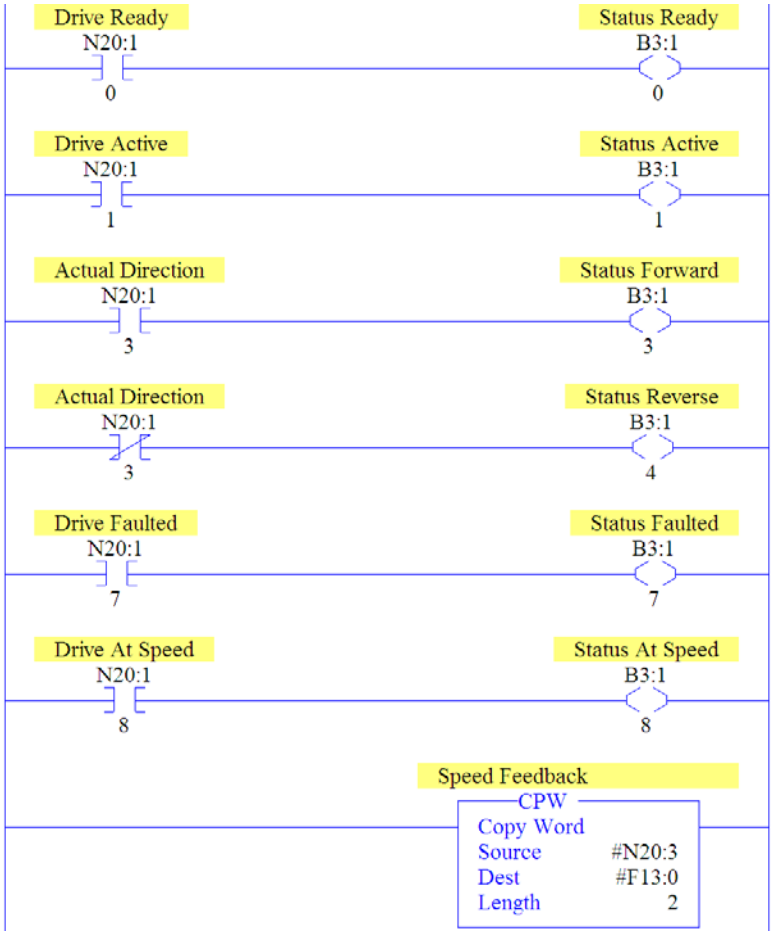
Table 5.F Program and Controller Data Table Address Descriptions for Example Logic Command/Reference Ladder Logic Program

Description	Program Data Table Address
Command Stop	B3:20/0
Command Start	B3:20/1
Command Jog	B3:20/2
Command Clear Faults	B3:20/3
Command Forward Reverse (XIO)	B3:20/4
Command Forward Reverse (XIC)	B3:20/4
Speed Reference	N30:22

Description	Controller Data Table Address
Drive Stop	N20:20/0
Drive Start	N20:20/1
Drive Jog	N20:20/2
Drive Clear Faults	N20:20/3
Drive Forward	N20:20/4
Drive Reverse	N20:20/5
Speed Reference	N20:22

An example ladder logic program that uses these alternate controller data table addresses is shown in Figure 5.12 and Figure 5.13.

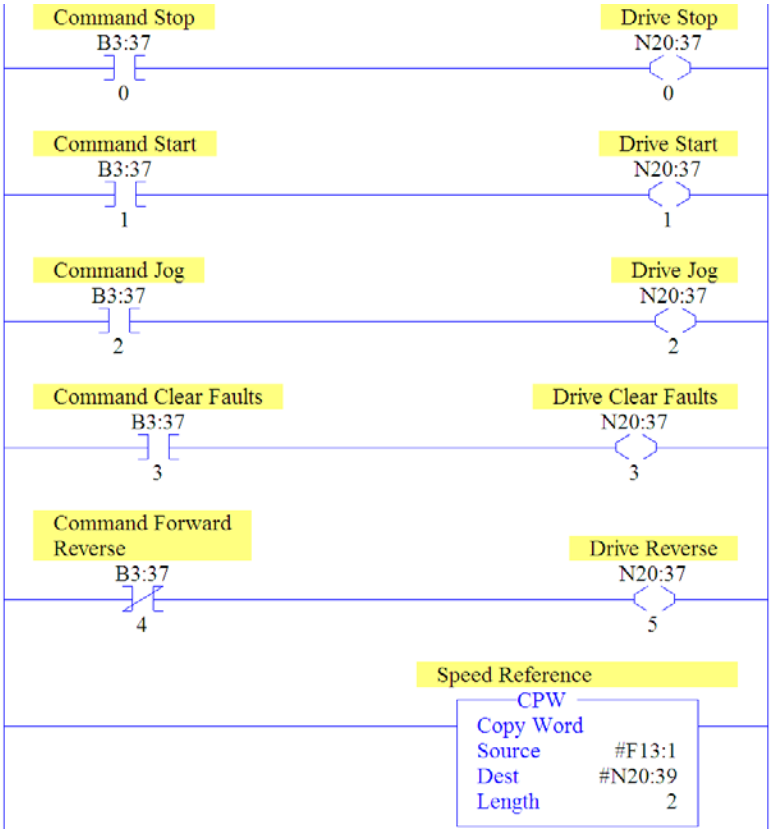
Figure 5.12 PLC-5, SLC 500, and MicroLogix 1100 Example Ladder Logic Program for Logic Status/Feedback



Important: This ladder does not include logic for Datalinks. However, if Datalinks are required and they are a REAL (floating point) data type, a data conversion must be used. For MicroLogix 1100 controllers only, use a CPW (Copy

Word) instruction as shown in the example ladder. For PLC-5 and SLC 500 controllers, use a COP (Copy) instruction.

Figure 5.13 PLC-5, SLC 500, and MicroLogix 1100 Example Ladder Logic Program for Logic Command/Reference



Important: This ladder does not include logic for Datalinks. However, if Datalinks are required and they are a REAL (floating point) data type, a data conversion must be used. For MicroLogix 1100 controllers only, use a CPW (Copy Word) instruction as shown in the example ladder. For PLC-5 and SLC 500 controllers, use a COP (Copy) instruction.

Using Explicit Messaging

This chapter provides information and examples that explain how to use Explicit Messaging to configure and monitor the adapter and connected PowerFlex 750-Series drive.

ATTENTION: Risk of injury or equipment damage exists. The examples in this publication are intended solely for purposes of example. There are many variables and requirements with any application. Rockwell Automation, Inc. does not assume responsibility or liability (to include intellectual property liability) for actual use of the examples shown in this publication.

ATTENTION: Risk of equipment damage exists. If Explicit Messages are programmed to write parameter data to Non-Volatile Storage (NVS) frequently, the NVS will quickly exceed its life cycle and cause the drive to malfunction. Do not create a program that frequently uses Explicit Messages to write parameter data to NVS. Datalinks do not write to NVS and should be used for frequently changed parameters.

Refer to Chapter 5 for information about the I/O Image, using Logic Command/Status, Reference/Feedback, and Datalinks.

About Explicit Messaging

Explicit Messaging is used to transfer data that does not require continuous updates. With Explicit Messaging, you can configure and monitor a slave device's parameters on the network.

Important: When an explicit message is performed, by default no I/O connection is made since it is an "unconnected" message. When timing of the message transaction is important, you can create a dedicated message connection between the controller and drive by checking the "Connected" box on the Communications tab message configuration screen during message setup. These message connections are in addition to the I/O connection. However, the trade off for more message connections is decreased network

performance. If your application cannot tolerate this, do not check the “Connected” box.

TIP: To message to another device in a different drive port, refer to the Instance table in Appendix C:

- DPI Parameter Object section on page C-13 for *Device* parameters.
- Host DPI Parameter Object section on page C-28 for *Host* parameters.

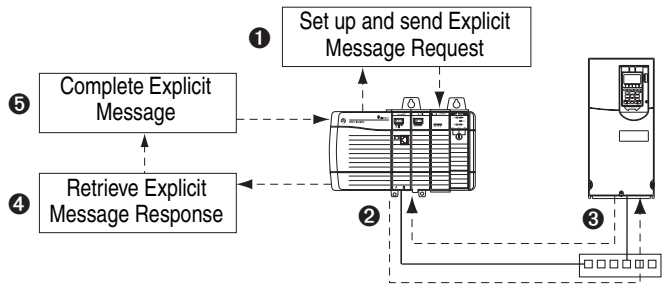
In the Message Configuration screen, set the Instance field to an appropriate value within the range listed for the port in which the device resides.

Performing Explicit Messages

There are five basic events in the Explicit Messaging process. The details of each step will vary depending on the type of controller being used. Refer to the documentation for your controller.

Important: There must be a request message and a response message for all Explicit Messages, whether you are reading or writing data.

Figure 6.1 Explicit Message Process



Event	Description
❶	You format the required data and set up the ladder logic program to send an Explicit Message request to the scanner or bridge module (download).
❷	The scanner or bridge module transmits the Explicit Message Request to the slave device over the network.
❸	The slave device transmits the Explicit Message Response back to the scanner. The data is stored in the scanner buffer.
❹	The controller retrieves the Explicit Message Response from the scanner’s buffer (upload).
❺	The Explicit Message is complete. **Note:** The scanner module may be integrated with the controller (for example, ControlLogix).

For information on the maximum number of Explicit Messages that can be executed at a time, refer to the user manual for the bridge or scanner and/or controller that is being used.

ControlLogix Example

TIP: To display the Message Configuration screen in RSLogix 5000, add a message instruction (MSG), create a new tag for the message (Properties: Base tag type, MESSAGE data type, controller scope), and click the ... button in the message instruction.

For supported classes, instances, and attributes, refer to Appendix C, EtherNet/IP Objects.

ControlLogix Example Ladder Logic Program to Read Single Parameter

A Get Attribute Single message is used to read a single parameter. This read message example reads the value of the 32-bit REAL (floating point) parameter 007 - [Output Current] in a PowerFlex 750-Series drive.

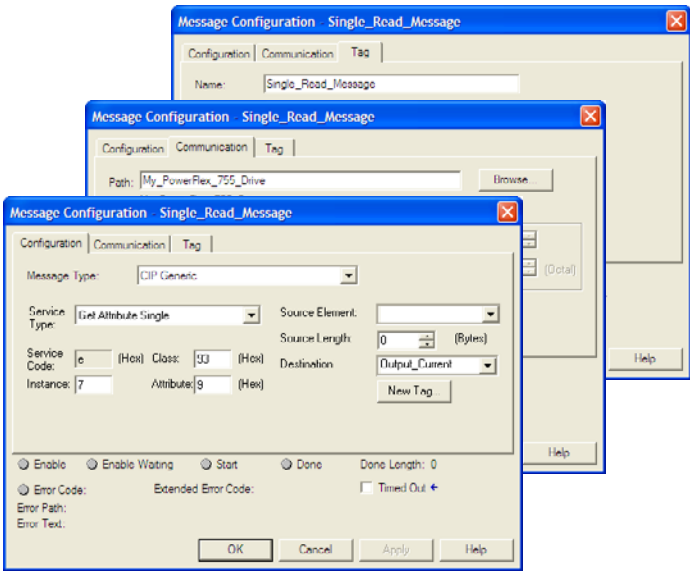
Table 6.A Example Controller Tags for Read Single Parameter Messaging Program

Operand	Controller Tags for Read Single Message	Data Type
XIC	Execute_Single_Read_Message	BOOL
MSG	Single_Read_Message	MESSAGE

Figure 6.2 Example Ladder Logic Explicit Messaging Program for Read Single

ControlLogix – Formatting a Message to Read Single Parameter

Figure 6.3 Get Attribute Single Message Configuration Screens



The following table identifies the data that is required in each box to format a single read message.

Configuration Tab	Example Value	Description
Message Type	CIP Generic	Used to access the Parameter Object in the adapter.
Service Type (1)	Get Attribute Single	This service is used to read a parameter value.
Service Code (1)	e (Hex.)	Code for the requested service.
Class	93 (Hex.)	Class ID for the DPI Parameter Object.
Instance	7 (Dec.)	Instance number is the same as parameter number.
Attribute	9 (Hex.)	Attribute number for the Parameter Value attribute.
Source Element	—	Leave blank (not applicable).
Source Length	0 bytes	Number of bytes of service data to be sent in the message.
Destination	Output_Current (3)	The tag where the data that is read is stored.
Communication Tab	**Example Value**	**Description**
Path (2)	My_PowerFlex_755_Drive	The path is the route that the message will follow.
Tag Tab	**Example Value**	**Description**
Name	Single_Read_Message	The name for the message.

(1) The default setting for Service Type is “Custom,” enabling entry of a Service Code not available from the Service Type pull-down menu. When selecting a Service Type other than “Custom” from the pull-down menu, an appropriate Hex. value is automatically assigned to the Service Code box which grays out (unavailable).

(2) Click **Browse** to find the path, or type in the name of the device listed in the I/O Configuration folder (for this example, My_PowerFlex_755_Drive).

(3) In this example, Output Current is a 32-bit floating point parameter so the Data Type field must be set to “REAL” when creating the controller tag. To read a 32-bit DINT parameter, set the tag Data Type field to “DINT.” For a 16-bit parameter, set the Data Type field to “INT.” Refer to the drive documentation to determine the size of the parameter.

ControlLogix Example Ladder Logic Program to Write Single Parameter

A Set Attribute Single message is used to write to a single parameter. This write message example writes a value to the 32-bit REAL (floating point) parameter 535 - [Accel Time 1] in a PowerFlex 750-Series drive.

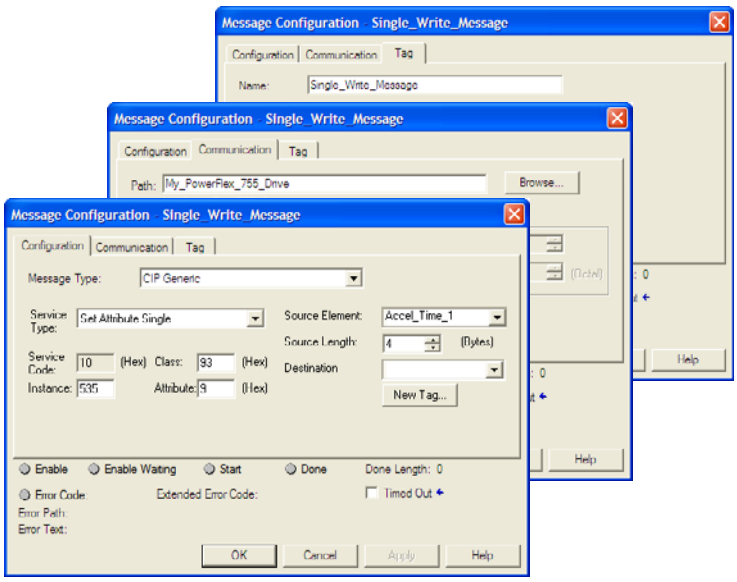
Table 6.B Example Controller Tags for Write Single Parameter Messaging Program

Operand	Controller Tags for Write Single Message	Data Type
XIC	Execute_Single_Write_Message	BOOL
MSG	Single_Write_Message	MESSAGE

Figure 6.4 Example Ladder Logic Explicit Messaging Program for Write Single

ControlLogix – Formatting a Message to Write Single Parameter

Figure 6.5 Set Attribute Single Message Configuration Screens



The following table identifies the data that is required in each box to format a single write message.

Configuration Tab	Example Value	Description
Message Type	CIP Generic	Used to access the Parameter Object in the adapter.
Service Type (1)	Set Attribute Single	This service is used to write a parameter value.
Service Code (1)	10 (Hex.)	Code for the requested service.
Class	93 (Hex.)	Class ID for the DPI Parameter Object.
Instance	535 (Dec.)	Instance number is the same as parameter number.
Attribute (2)	9 or A (Hex.)	Attribute number for the Parameter Value attribute.
Source Element	Accel_Time_1 (4)	Name of the tag for any service data to be sent from the scanner or bridge to the adapter/drive.
Source Length	4 bytes (4)	Number of bytes of service data to be sent in the message.
Destination	—	Leave blank (not applicable).
Communication Tab	**Example Value**	**Description**
Path (3)	My_PowerFlex_755_Drive	The path is the route that the message will follow.
Tag Tab	**Example Value**	**Description**
Name	Single_Write_Message	The name for the message.

(1) The default setting for Service Type is “Custom,” enabling entry of a Service Code not available from the Service Type pull-down menu. When selecting a Service Type other than “Custom” from the pull-down menu, an appropriate Hex. value is automatically assigned to the Service Code box which grays out (unavailable).

(2) Setting the Attribute value to “9” will write the parameter value to the drive’s Non-Volatile Storage (EEPROM) memory, so the parameter value will remain even after the drive is power cycled. **Important:** When set to “9,” be very cautious as the EEPROM may quickly exceed its life cycle and cause the drive to malfunction. Setting the Attribute value to “A” will write the parameter value to temporary memory, so the parameter value will be lost after the drive is power cycled. It is recommended to use the “A” setting when frequent write messages are required.

(3) Click **Browse** to find the path, or type in the name of the device listed in the I/O Configuration folder (for this example, My_PowerFlex_755_Drive).

(4) In this example, Accel Time 1 is a 32-bit floating point parameter so the Data Type field must be set to “REAL” when creating the controller tag. To write to a 32-bit DINT parameter, set the tag Data Type field to “DINT.” For a 16-bit parameter, set the Data Type field to “INT.” Also, the Source Length field on the Message Configuration screen must correspond to the selected Data Type in bytes (for example, 4 bytes for a REAL or DINT, or 2 bytes for an INT). Refer to the drive documentation to determine the size of the parameter.

ControlLogix Example Ladder Logic Program to Read Multiple Parameters

A Scattered Read message is used to read the values of multiple parameters. This read message example reads the values of these five 32-bit REAL (floating point) parameters in a PowerFlex 750-Series drive: 001 - [Output Frequency], 007 - [Output Current], 008 - [Output Voltage], 009 - [Output Power], and 011 - [DC Bus Volts].

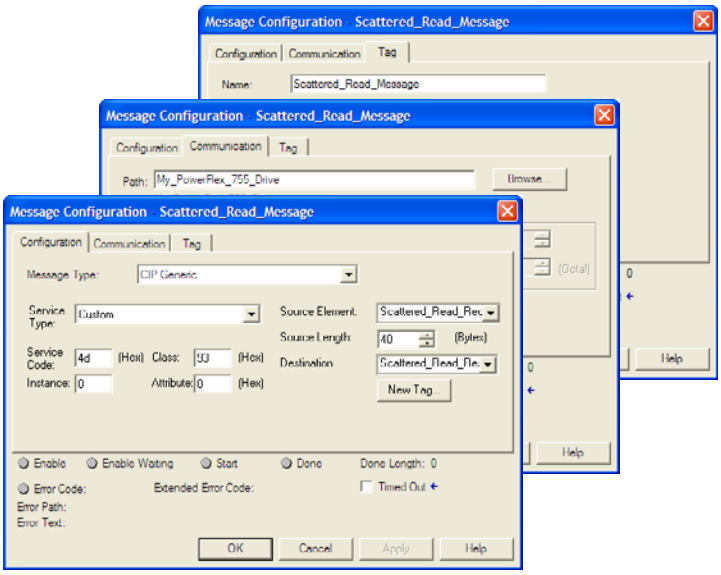
Table 6.C Example Controller Tags for Read Multiple Parameter Messaging Program

Operand	Controller Tags for Read Multiple Message	Data Type
XIC	Execute_Scattered_Read_Message	BOOL
MSG	Scattered_Read_Message	MESSAGE

Figure 6.6 Example Ladder Logic Explicit Messaging Program for Read Multiple

ControlLogix – Formatting a Message to Read Multiple Parameters

Figure 6.7 Scattered Read Message Configuration Screens



The following table identifies the data that is required in each box to format a multiple read message.

Configuration Tab	Example Value	Description
Message Type	CIP Generic	Used to access Parameter Object in the adapter.
Service Type (1)	Custom	Required for scattered messages.
Service Code (1)	4d (Hex.)	Code for the requested service.
Class	93 (Hex.)	Class ID for the DPI Parameter Object.
Instance	0 (Dec.)	Required for scattered messages.
Attribute	0 (Hex.)	Required for scattered messages.
Source Element	Scattered_Read_Request (3)	Name of the tag for any service data to be sent from scanner or bridge to the adapter/drive.
Source Length	40 bytes (3)	Number of bytes of service data to be sent in the message.
Destination	Scattered_Read_Response (4)	The tag where the data that is read is stored.
Communication Tab	**Example Value**	**Description**
Path (2)	My_PowerFlex_755_Drive	The path is the route that the message will follow.
Tag Tab	**Example Value**	**Description**
Name	Scattered_Read_Message	The name for the message.

(1) The default setting for Service Type is “Custom,” enabling entry of a Service Code not available from the Service Type pull-down menu. When selecting a Service Type other than “Custom” from the pull-down menu, an appropriate Hex. value is automatically assigned to the Service Code box which grays out (unavailable). When reading 32-bit REAL (floating point) parameters, as in this example, data conversion using COP (Copy) instructions or UDDT’s is required to correctly show the parameter values.

(2) Click **Browse** to find the path, or type in the name of the device listed in the I/O Configuration folder (for this example, My_PowerFlex_755_Drive).

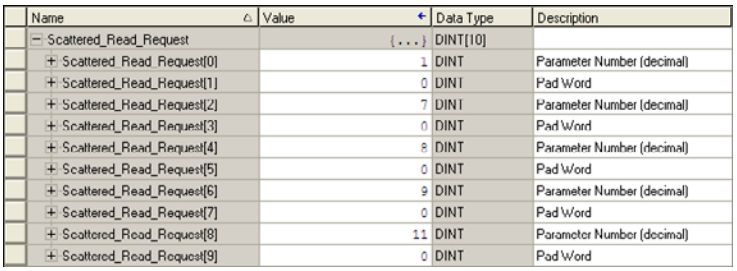
(3) In this example, we are reading five 32-bit REAL (floating point) parameters. Each parameter being read requires two contiguous DINT registers. Therefore, a controller tag was created with its Data Type field set to “DINT[10].” Also, the Source Length field on the Message Configuration screen must correspond to the selected Data Type in bytes (for this example, 40 bytes for a DINT[10] array). Scattered read messages always assume that every parameter being read is a 32-bit parameter, regardless of its actual size. Maximum message length is 256 bytes which can read up to 32 parameters, regardless of their size.

(4) The controller tag for “Scattered_Read_Response” must be the same size as the controller tag for “Scattered_Read_Request” (for this example, 40 bytes), but can be a different data type (for this example, a UDDT to handle conversions to parameter values that are a REAL data type).

ControlLogix Example Scattered Read Request Data

In this example, we use the data structure in Figure 6.8 in the source tag named Scattered Read Request to read these five 32-bit REAL (floating point) parameters in a PowerFlex 750-Series drive: 001 - [Output Frequency], 007 - [Output Current], 008 - [Output Voltage], 009 - [Output Power], and 011 - [DC Bus Volts].

Figure 6.8 Example Scattered Read Request Data

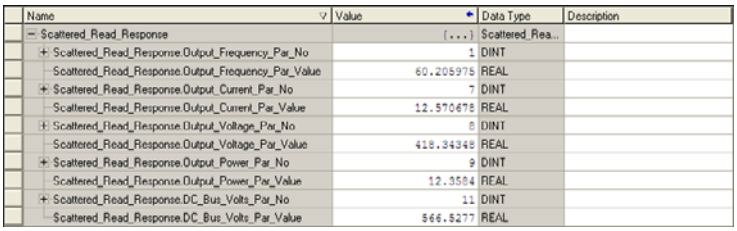


Name	Value	Data Type	Description
Scattered_Read_Request	{...}	DINT[10]	
Scattered_Read_Request[0]	1	DINT	Parameter Number (decimal)
Scattered_Read_Request[1]	0	DINT	Pad Word
Scattered_Read_Request[2]	7	DINT	Parameter Number (decimal)
Scattered_Read_Request[3]	0	DINT	Pad Word
Scattered_Read_Request[4]	8	DINT	Parameter Number (decimal)
Scattered_Read_Request[5]	0	DINT	Pad Word
Scattered_Read_Request[6]	9	DINT	Parameter Number (decimal)
Scattered_Read_Request[7]	0	DINT	Pad Word
Scattered_Read_Request[8]	11	DINT	Parameter Number (decimal)
Scattered_Read_Request[9]	0	DINT	Pad Word

ControlLogix Example Scattered Read Response Data

The Scattered Read Request message reads the multiple parameters and returns their values to the destination tag (Scattered_Read_Response). Figure 6.9 shows the parameter values which, in this example, have been converted using a UDDT for correct presentation. COP (Copy) instructions could have been used for this purpose instead of a UDDT.

Figure 6.9 Example Scattered Read Response Converted Data



Name	Value	Data Type	Description
Scattered_Read_Response	{...}	Scattered_Rea...	
Scattered_Read_Response.Output_Frequency_Par_No	1	DINT	
Scattered_Read_Response.Output_Frequency_Par_Value	60.205975	REAL	
Scattered_Read_Response.Output_Current_Par_No	7	DINT	
Scattered_Read_Response.Output_Current_Par_Value	12.570678	REAL	
Scattered_Read_Response.Output_Voltage_Par_No	8	DINT	
Scattered_Read_Response.Output_Voltage_Par_Value	418.34348	REAL	
Scattered_Read_Response.Output_Power_Par_No	9	DINT	
Scattered_Read_Response.Output_Power_Par_Value	12.3584	REAL	
Scattered_Read_Response.DC_Bus_Volts_Par_No	11	DINT	
Scattered_Read_Response.DC_Bus_Volts_Par_Value	566.5277	REAL	

In this example, the parameters have the following values:

PowerFlex 750-Series Drive Parameter	Read Value
1 - [Output Frequency]	60.205975 Hz
7 - [Output Current]	12.570678 Amp
8 - [Output Voltage]	418.34348 VAC
9 - [Output Power]	12.3534 kW
11 - [DC Bus Volts]	566.5277 VDC

ControlLogix Example Ladder Logic Program to Write Multiple Parameters

A Scattered Write message is used to write to multiple parameters. This write message example writes the following values to these five 32-bit REAL (floating point) parameters in a PowerFlex 750-Series drive:

PowerFlex 750-Series Drive Parameter	Write Value
536 - [Accel Time 2]	11.1 Sec.
538 - [Decel Time 2]	22.2 Sec.
575 - [Preset Speed 5]	33.3 Hz.
576 - [Preset Speed 6]	44.4 Hz.
577 - [Preset Speed 7]	55.5 Hz.

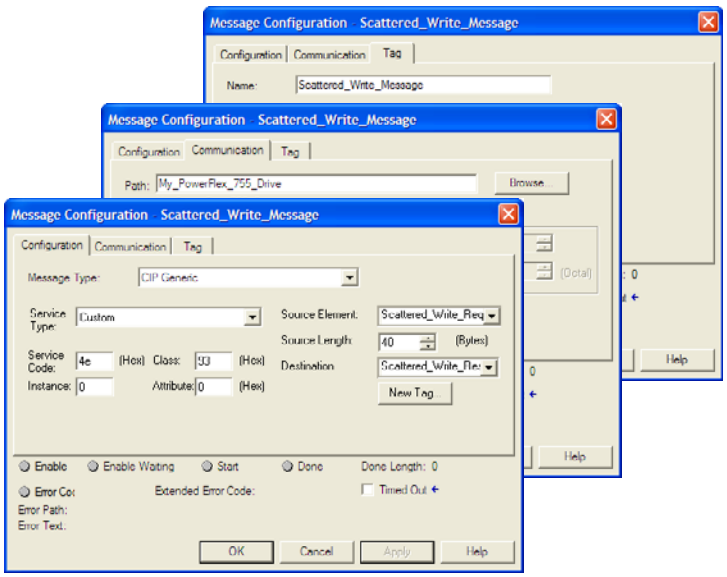
Table 6.D Example Controller Tags for Write Multiple Parameter Messaging Program

Operand	Controller Tags for Write Multiple Message	Data Type
XIC	Execute_Scattered_Write_Message	BOOL
MSG	Scattered_Write_Message	MESSAGE

Figure 6.10 Example Ladder Logic Explicit Messaging Program for Write Multiple

ControlLogix – Formatting a Message to Write Multiple Parameters

Figure 6.11 Scattered Write Multiple Message Configuration Screens



The following table identifies the data that is required in each box to format a multiple write message.

Configuration Tab	Example Value	Description
Message Type	CIP Generic	Used to access Parameter Object in the adapter.
Service Type (1)	Custom	Required for scattered messages.
Service Code (1)	4e (Hex.)	Code for the requested service.
Class	93 (Hex.)	Class ID for the DPI Parameter Object.
Instance	0 (Dec.)	Required for scattered messages.
Attribute	0 (Hex.)	Required for scattered messages.
Source Element	Scattered_Write_Request (3)	Name of the tag for any service data to be sent from scanner or bridge to the adapter/drive.
Source Length	40 bytes (3)	Number of bytes of service data to be sent in the message.
Destination	Scattered_Write_Response (4)	The tag where the data that is read is stored.
Communication Tab	**Example Value**	**Description**
Path (2)	My_PowerFlex_755_Drive	The path is the route that the message will follow.
Tag Tab	**Example Value**	**Description**
Name	Scattered_Write_Message	The name for the message.

(1) The default setting for Service Type is “Custom,” enabling entry of a Service Code not available from the Service Type pull-down menu. When selecting a Service Type other than “Custom” from the pull-down menu, an appropriate Hex. value is automatically assigned to the Service Code box which grays out (unavailable). When writing to 32-bit REAL (floating point) parameters, as in this example, data conversion using COP (Copy) instructions or UDDT’s is required to correctly write the parameter values.

(2) Click **Browse** to find the path, or type in the name of the device listed in the I/O Configuration folder (for this example, My_PowerFlex_755_Drive).

(3) In this example, we are writing to five 32-bit REAL (floating point) parameters. Each parameter being written to requires two contiguous DINT registers. Therefore, a controller tag was created with its Data Type field set to the name of the UDDT of five interleaved DINTs and REALs. Also, the Source Length field on the Message Configuration screen must correspond to the selected Data Type in bytes (for this example, 40 bytes for an array of five scattered read structures. Scattered write messages always assume that every parameter being written to is a 32-bit parameter, regardless of its actual size. Maximum message length is 256 bytes which can write up to 32 parameters, regardless of their size.

(4) The controller tag for “Scattered_Write_Response” must be the same size as the controller tag for “Scattered_Write_Request” (for this example, 40 bytes). An array of DINTs is suggested to be able to read any error codes that are returned.

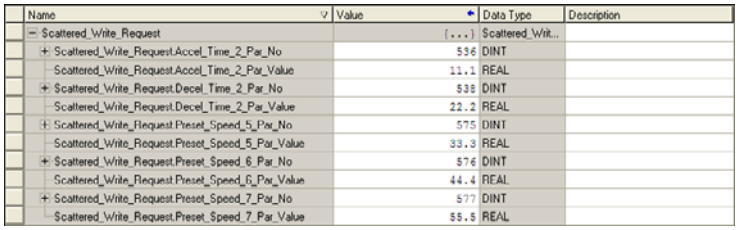
ControlLogix Example Scattered Write Request Data

In this example, we use the data structure in Figure 6.12 in the source tag (Scattered_Write_Request) to write new values to these 32-bit REAL (floating point) parameters:

PowerFlex 750-Series Drive Parameter	Write Value
536 - [Accel Time 2]	11.1 Sec.
538 - [Decel Time 2]	22.2 Sec.
575 - [Preset Speed 5]	33.3 Hz.
576 - [Preset Speed 6]	44.4 Hz.
577 - [Preset Speed 7]	55.5 Hz.

Figure 6.12 shows the parameter values which, in this example, have been converted using a UDDT to correctly write their values. COP (Copy) instructions could have been used for this purpose instead of a UDDT.

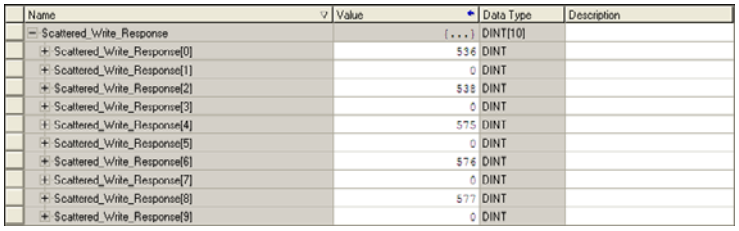
Figure 6.12 Example Scattered Write Request Converted Data



ControlLogix Example Scattered Write Response Data

The results of the message appear in the destination tag named Scattered_Write_Response (Figure 6.13). Values of “0” indicate no errors occurred.

Figure 6.13 Example Scattered Write Response Data



ControlLogix – Explanation of Request and Response Data for Read/Write Multiple Messaging

The data structures in Figure 6.14 use 32-bit words and can accommodate up to 32 parameters in a single message. In the Response Message, a parameter number with Bit 15 set indicates that the associated parameter value field contains an error code.

Figure 6.14 Data Structures for Scattered Read/Write Messages

	Request (Source Data)
DINT 0	Parameter Number
1	Pad
2	Parameter Number
3	Pad
4	Parameter Number
5	Pad
6	Parameter Number
7	Pad
8	Parameter Number
9	Pad
10	Parameter Number
11	Pad
12	Parameter Number
13	Pad
14	Parameter Number
15	Pad
16	Parameter Number
17	Pad
18	Parameter Number
19	Pad
20	Parameter Number
21	Pad
22	Parameter Number
23	Pad
24	Parameter Number
25	Pad
26	Parameter Number
27	Pad
28	Parameter Number
29	Pad
30	Parameter Number
31	Pad
32	Parameter Number
33	Pad
34	Parameter Number
35	Pad
⋮	
62	Parameter Number
63	Pad

	Response (Destination Data)
DINT 0	Parameter Number
1	Parameter Value
2	Parameter Number
3	Parameter Value
4	Parameter Number
5	Parameter Value
6	Parameter Number
7	Parameter Value
8	Parameter Number
9	Parameter Value
10	Parameter Number
11	Parameter Value
12	Parameter Number
13	Parameter Value
14	Parameter Number
15	Parameter Value
16	Parameter Number
17	Parameter Value
18	Parameter Number
19	Parameter Value
20	Parameter Number
21	Parameter Value
22	Parameter Number
23	Parameter Value
24	Parameter Number
25	Parameter Value
26	Parameter Number
27	Parameter Value
28	Parameter Number
29	Parameter Value
30	Parameter Number
31	Parameter Value
32	Parameter Number
33	Parameter Value
34	Parameter Number
35	Parameter Value
⋮	
62	Parameter Number
63	Parameter Value

PLC-5 Example

Important: The PLC-5 must be Series E (Rev. D.1 or higher) to support the MultiHop feature that routes messaging to the drive.

Important: Due to inherent limitations with the PCCC N-File method, only contiguous multiple parameters can be read or written in one explicit message.

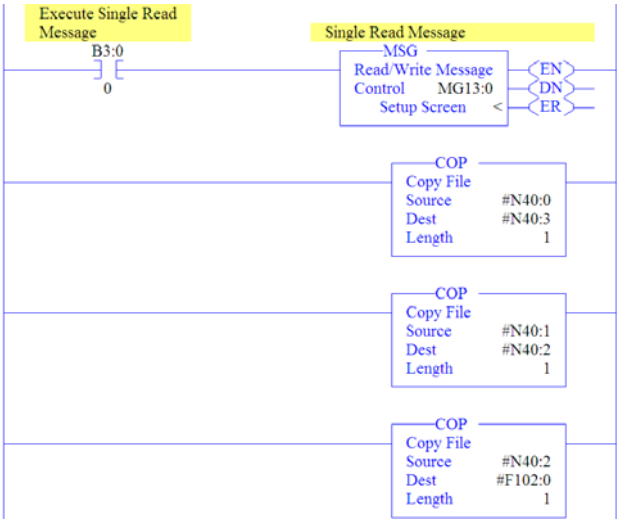
For explicit messaging, the N150 N-Files must be used because they are already mapped to specific parameters in the drive and its connected peripherals. This enables direct access to any parameter.

For PCCC N150 N-File information, refer to page C-9.

PLC-5 Example Ladder Logic Program to Read Single Parameter

A Generic Get Attribute Single message is used to read a single parameter. This read message example reads the value of the 32-bit REAL (floating point) parameter 007 - [Output Current] in a PowerFlex 750-Series drive.

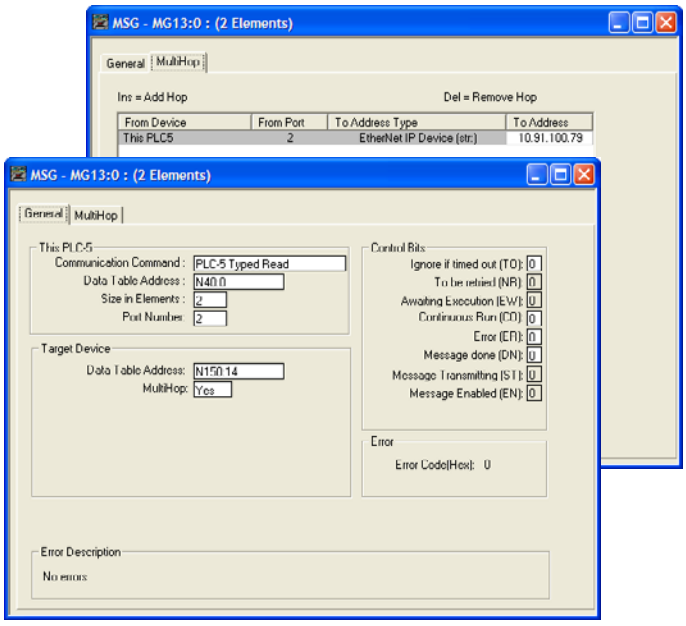
Figure 6.15 Example Ladder Logic Explicit Messaging Program for Read Single



Three COP (Copy) instructions are required to convert the 16-bit integer data table addresses N40:0 (Least Significant Word) and N40:1 (Most Significant Word) to a 32-bit REAL (floating point) data table address F102:0 for correct presentation. The first two COP instructions swap the LSW and MSW, and the third COP instruction correctly presents the 32-bit REAL (floating point) value.

PLC-5 – Formatting a Message to Read Single Parameter

Figure 6.16 Generic Get Attribute Single Message Configuration Screens



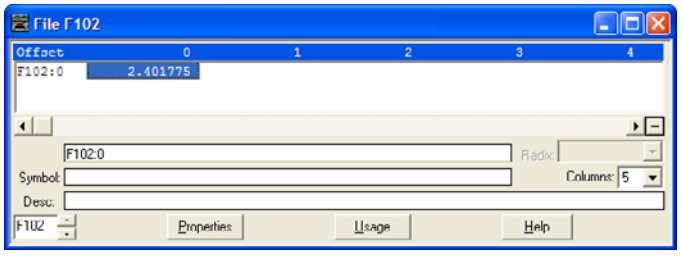
The following table identifies the data that is required in each box to format a single read message.

General Tab	Example Value	Description
Communication Command	PLC-5 Typed Read	Controller type and command type for controller to read data from the drive.
Data Table Address	N40:0	An unused controller data table address containing the message instruction. This address is the starting word of the destination file.
Size in Elements	2	Number of elements (words) to be transferred. Each element size is a 16-bit integer.
Port Number	2	Controller port to which EtherNet/IP network is connected.
Data Table Address	N150:14	Specific starting address of the source file in the drive (refer to page C-9).
MultiHop	Yes	Enables communication to allow EtherNet messaging to be routed to the drive.
MultiHop Tab	**Example Value**	**Description**
To Address	10.91.100.79	IP address of the adapter connected to the drive.

PLC-5 Example Get Attribute Single Response Data

In this example, we use the data table address in Figure 6.17 to store the response value (2.401775 amps) that was read from drive parameter 007 - [Output Current].

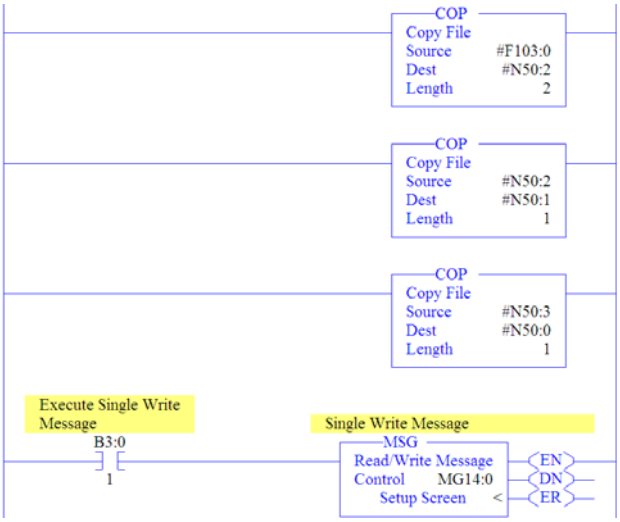
Figure 6.17 Example Get Attribute Single Response Data File



PLC-5 Example Ladder Logic Program to Write Single Parameter

A Generic Set Attribute Single message is used to write to a single parameter. This write message example writes a value to the 32-bit REAL (floating point) parameter 535 - [Accel Time 1] in a PowerFlex 750-Series drive.

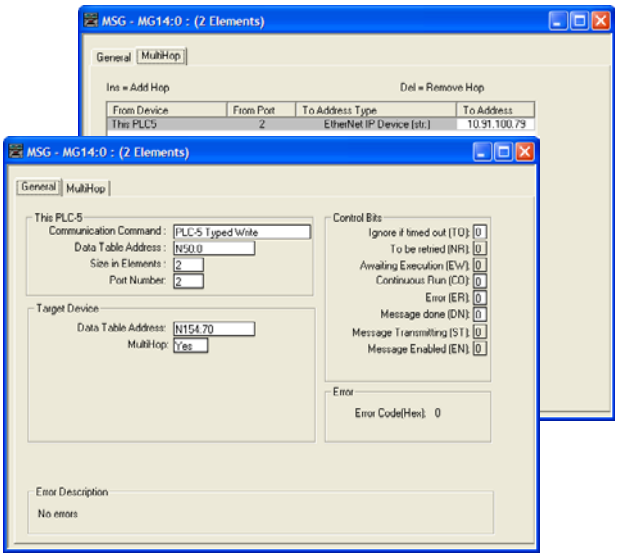
Figure 6.18 Example Ladder Logic Explicit Messaging Program for Write Single



Three COP (Copy) instructions are required to convert the 16-bit integer data table addresses N50:0 (Least Significant Word) and N50:1 (Most Significant Word) to a 32-bit REAL (floating point) data table address F103:0 for correct presentation. The first COP instruction correctly writes the 32-bit REAL (floating point) value. The second and third COP instructions swap the LSW and MSW.

PLC-5 – Formatting a Message to Write Single Parameter

Figure 6.19 Generic Set Attribute Single Message Configuration Screens



The following table identifies the data that is required in each box to format a single write message.

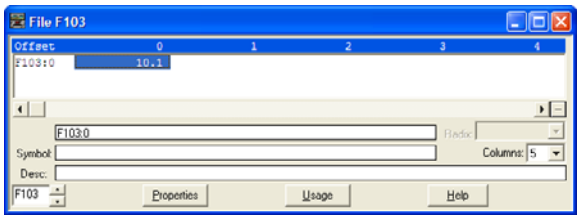
General Tab	Example Value	Description
Communication Command	PLC-5 Typed Write (1)	Controller type and command type for controller to write data to the drive.
Data Table Address	N50:0	An unused controller data table address containing the message instruction. This address is the starting word of the source file.
Size in Elements	2	Number of elements (words) to be transferred. Each element size is a 16-bit integer.
Port Number	2	Controller port to which EtherNet/IP network is connected.
Data Table Address	N154:70	Specific starting address of the destination file in the drive (refer to page C-9).
MultiHop	Yes	Enables communication to allow EtherNet messaging to be routed to the drive.
MultiHop Tab	**Example Value**	**Description**
To Address	10.91.100.79	IP address of the adapter connected to the drive.

(1) **Important:** PCCC N150 N-File write messages are written to the drive's EEPROM. Be very cautious as the EEPROM may quickly exceed its life cycle and cause the drive to malfunction.

PLC-5 Example Set Attribute Single Request Data

In this example, we use the data table address in Figure 6.20 to store the request value (10.1 sec.) that was written to drive parameter 535 - [Accel Time 1].

Figure 6.20 Example Set Attribute SIngle Request Data File



TIP: To verify that the parameter value was successfully written, use the HIM, DriveExplorer or DriveExecutive to access the parameter and view its newly written value.

PLC-5 Reading/Writing Multiple Parameters

You can read or write only contiguous parameters. Scattered read/write messaging is not supported. Also, the range of contiguous parameters must be contained in the same N-File. Two elements (words) are required for each parameter being read or written. For example, to read 5 contiguous parameters, 10 elements (words) must be used.

SLC 500 Example

When using RSLogix 500 v7.10 or lower, explicit messaging must be performed using the PCCC N-File method. For RSLogix 500 v7.20 or higher, the CIP messaging method has been added along with the PCCC N-File method. However, it is recommended to use the CIP method because it is easier to use and understand. For this reason, only instructions for the CIP method are provided. If you must use the PCCC N-File method, refer to the PLC-5 Example on page 6-14.

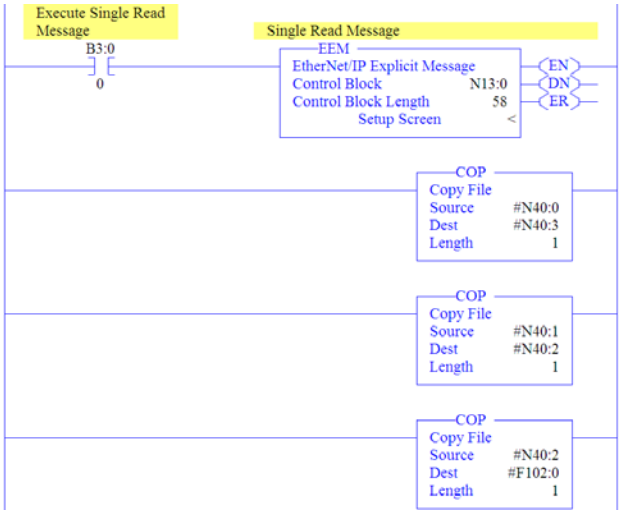
The CIP messaging method provides a Generic Get/Set Attribute Service which can be used to perform single parameter read or write and multiple parameter read or write explicit messages. Also, the Generic Set Attribute Service offers the choice of writing the data to the drive's Non-Volatile Storage (NVS) or the drive's Random Access Memory (RAM). Note that when selecting the data to be written to RAM, the data will be lost if the drive loses power.

For supported classes, instances, and attributes, refer to Appendix C, EtherNet/IP Objects.

SLC 500 Example Ladder Logic Program to Read Single Parameter

A Generic Get Attribute Single message is used to read a single parameter. This read message example reads the value of the 32-bit REAL (floating point) parameter 007 - [Output Current] in a PowerFlex 750-Series drive.

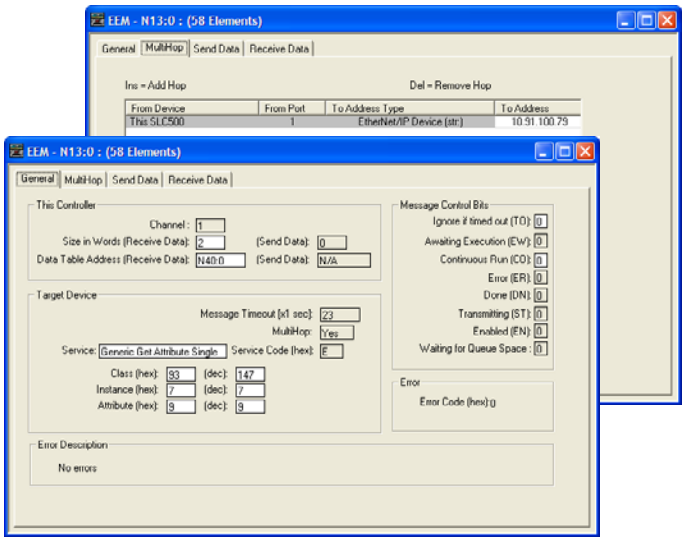
Figure 6.21 Example Ladder Logic Explicit Messaging Program for Read Single



Three COP (Copy) instructions are required to convert the 16-bit integer data table addresses N40:0 (Least Significant Word) and N40:1 (Most Significant Word) to a 32-bit REAL (floating point) data table address F102:0 for correct presentation. The first two COP instructions swap the LSW and MSW, and the third COP instruction correctly presents the 32-bit REAL (floating point) value.

SLC 500 – Formatting a Message to Read Single Parameter

Figure 6.22 Generic Get Attribute Single Message Configuration Screens



The following table identifies the data that is required in each box to format a single read message.

General Tab	Example Value	Description
Size in Words	2 (2)	Number of words to be transferred. Each word size is a 16-bit integer.
Data Table Address	N40:0	An unused controller data table address containing the message instruction. This address is the starting word of the response file.
Service (1)	Generic Get Attribute Single	Code for the requested service.
Class	93 (Hex.)	Class ID for the DPI Parameter Object.
Instance	7 (Dec.)	Instance number is the same as the parameter number.
Attribute	9 (Dec.)	Attribute number for the Parameter Value attribute.
MultiHop Tab	**Example Value**	**Description**
To Address	10.91.100.79	IP address of the adapter connected to the drive.

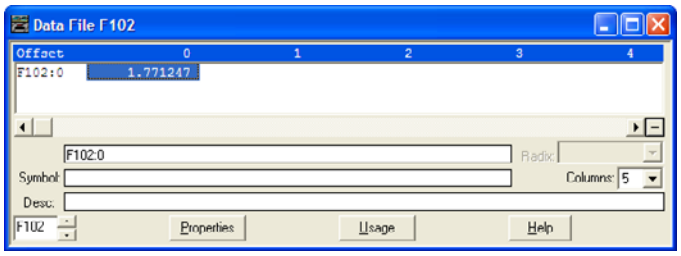
(1) The default setting for Service is “Custom,” enabling entry of a Service Code not available from the Service pull-down menu. When selecting a Service other than “Custom” from the pull-down menu, an appropriate Hex. value is automatically assigned to the Service Code box which grays out (unavailable).

(2) In this example, Output Current is a 32-bit REAL (floating point) parameter. If the parameter being read is a 16-bit parameter, the Size in Words would be set to 1.

SLC 500 Example Get Attribute Single Response Data

In this example, we use the data table address in Figure 6.23 to store the response value (1.771247 amps) that was read from drive parameter 007 - [Output Current].

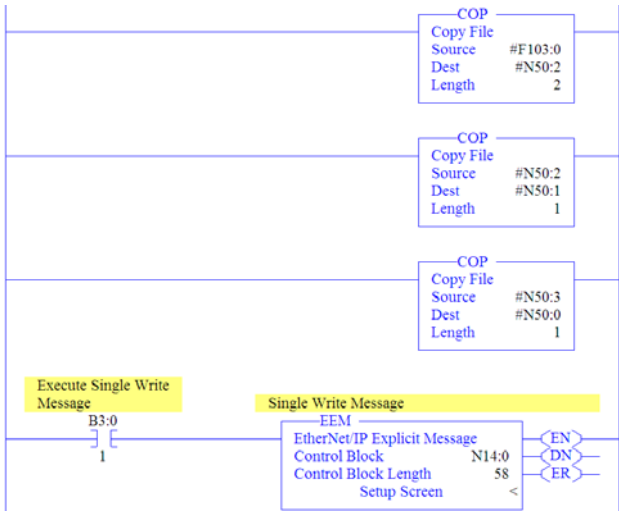
Figure 6.23 Example Get Attribute Single Response Data File



SLC 500 Example Ladder Logic Program to Write Single Parameter

A Generic Set Attribute Single message is used to write to a single parameter. This write message example writes a value to the 32-bit REAL (floating point) parameter 535 - [Accel Time 1] in a PowerFlex 750-Series drive.

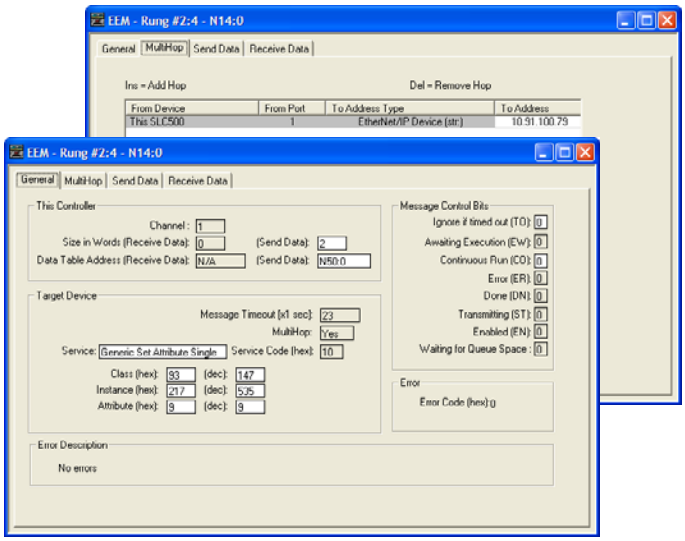
Figure 6.24 Example Ladder Logic Explicit Messaging Program for Write Single



Three COP (Copy) instructions are required to convert the 16-bit integer data table addresses N50:0 (Least Significant Word) and N50:1 (Most Significant Word) to a 32-bit REAL (floating point) data table address F103:0 for correct presentation. The first COP instruction correctly writes the 32-bit REAL (floating point) value. The second and third COP instructions swap the LSW and MSW.

SLC 500 – Formatting a Message to Write Single Parameter

Figure 6.25 Generic Set Attribute Single Message Configuration Screens



The following table identifies the data that is required in each box to format a single write message.

General Tab	Example Value	Description
Size in Words	2 (3)	Number of words to be transferred. Each word size is a 16-bit integer.
Data Table Address	N50:0	An unused controller data table address containing the message instruction. This address is the starting word of the request file.
Service (1)	Generic Set Attribute Single	Code for the requested service.
Class	93 (Hex.)	Class ID for the DPI Parameter Object.
Instance	535 (Dec.)	Instance number is the same as the parameter number.
Attribute (2)	9 or 10 (Dec.)	Attribute number for the Parameter Value attribute.
MultiHop Tab	**Example Value**	**Description**
To Address	10.91.100.79	IP address of the adapter connected to the drive.

(1) The default setting for Service is “Custom,” enabling entry of a Service Code not available from the Service pull-down menu. When selecting a Service other than “Custom” from the pull-down menu, an appropriate Hex. value is automatically assigned to the Service Code box which grays out (unavailable).

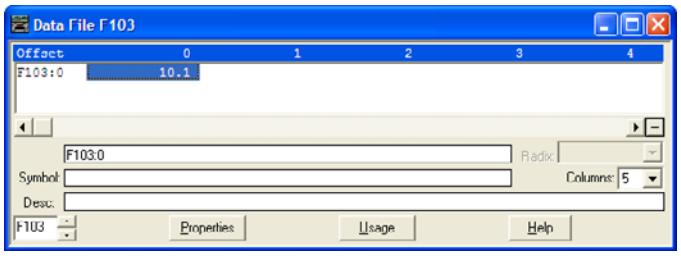
(2) Setting the Attribute value to “9” will write the parameter value to the drive’s Non-Volatile Storage (EEPROM) memory, so the parameter value will remain even after the drive is power cycled. **Important:** When set to “9,” be very cautious as the EEPROM may quickly exceed its life cycle and cause the drive to malfunction. Setting the Attribute value to “10” will write the parameter value to temporary memory, so the parameter value will be lost after the drive is power cycled. It is recommended to use the “10” setting when frequent write messages are required.

(3) In this example, Accel Time 1 is a 32-bit REAL (floating point) parameter. If the parameter being written to is a 16-bit parameter, the Size in Words would be set to 1.

SLC 500 Example Set Attribute Single Request Data

In this example, we use the data table address in Figure 6.26 to store the request value (10.1 sec.) that was written to drive parameter 535 - [Accel Time 1].

Figure 6.26 Example Set Attribute SIngle Request Data File

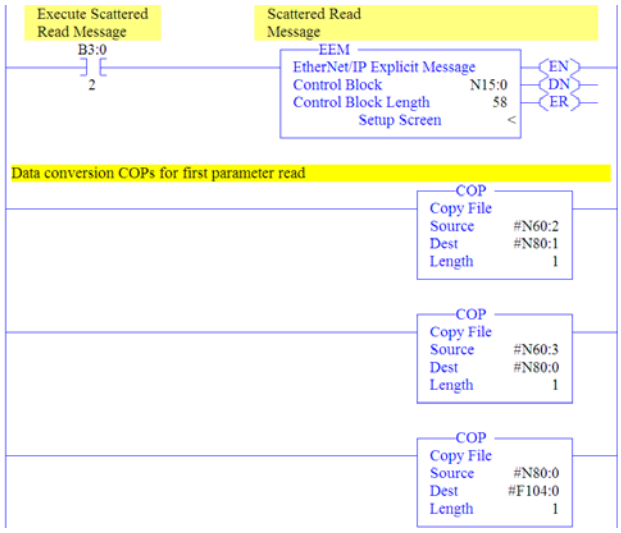


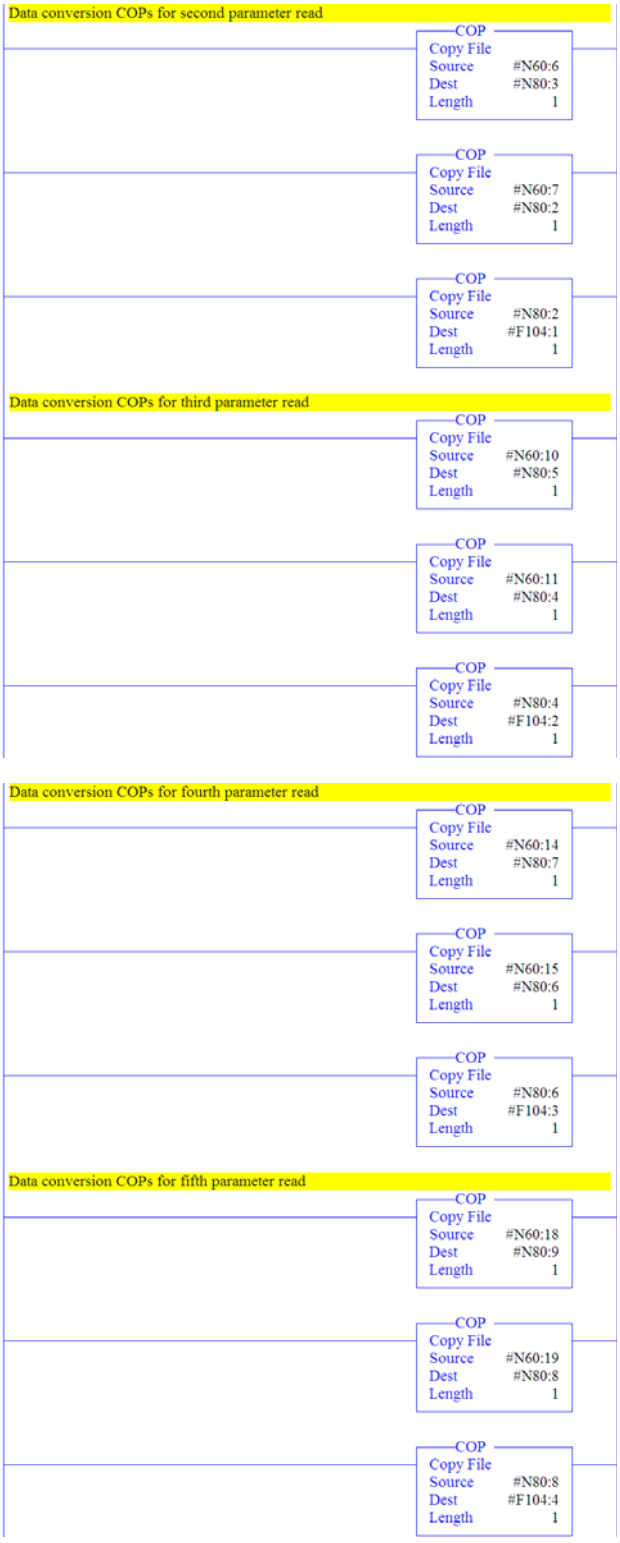
TIP: To verify that the parameter value was successfully written, use the HIM, DriveExplorer or DriveExecutive to access the parameter and view its newly written value.

SLC 500 Example Ladder Logic Program to Read Multiple Parameters

A Custom scattered read message is used to read the values of multiple parameters. This read message example reads the values of these five 32-bit REAL (floating point) parameters in a PowerFlex 750-Series drive: 001 - [Output Frequency], 007 - [Output Current], 008 - [Output Voltage], 009 - [Output Power], and 011 - [DC Bus Volts].

Figure 6.27 Example Ladder Logic Explicit Messaging Program for Read Multiple



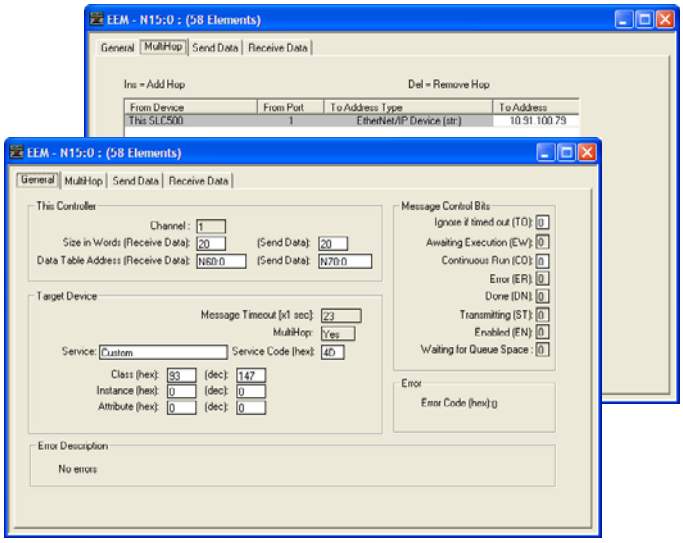


For each parameter being read, three COP (Copy) instructions are required to convert the 16-bit integer data table addresses, for example N60:2 (Least Significant Word) and N60:3 (Most Significant Word) for the first parameter, to a 32-bit REAL (floating point) data table address F104:0 for correct presentation. The first two COP instructions swap

the LSW and MSW, and the third COP instruction correctly presents the 32-bit REAL (floating point) value.

SLC 500 – Formatting a Message to Read Multiple Parameters

Figure 6.28 Custom Scattered Read Message Configuration Screens



The following table identifies the data that is required in each box to format a multiple read message.

General Tab	Example Value	Description
Size in Words		Each word size is a 16-bit integer.
Receive Data	20 (2)	Number of words to be received.
Send Data	20 (2)	Number of words to be sent.
Data Table Address		An unused controller data table address containing the message instruction.
Receive Data	N60:0	This address is the starting word of the response file.
Send Data	N70:0	This address is the starting word of the request file.
Service (1)	Custom	Required for scattered messages.
Service Code	4D (Hex.)	Code for the requested service.
Class	93 (Hex.)	Class ID for the DPI Parameter Object.
Instance	0 (Dec.)	Required for scattered messages.
Attribute	0 (Dec.)	Required for scattered messages.
MultiHop Tab	**Example Value**	**Description**
To Address	10.91.100.79	IP address of the adapter connected to the drive.

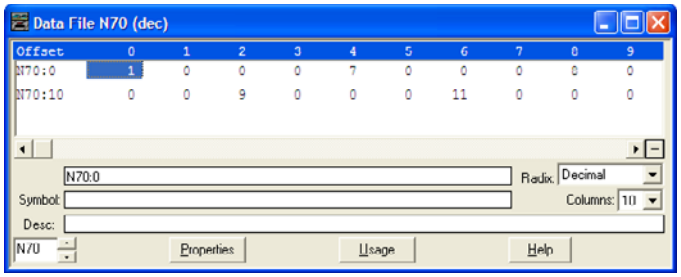
(1) The default setting for Service is “Custom,” enabling entry of a Service Code not available from the Service pull-down menu. When selecting a Service other than “Custom” from the pull-down menu, an appropriate Hex. value is automatically assigned to the Service Code box which grays out (unavailable).

(2) In this example, we are reading five 32-bit REAL (floating point) parameters. Each parameter being read requires four contiguous 16-bit words. Scattered read messages always assume that every parameter being read is a 32-bit parameter, regardless of its actual size. The data structure format is shown in Figure 6.36 on page 6-31. Maximum length is 128 words, which equates to 32 parameters.

SLC 500 Example Scattered Read Request Data

In this example, we use the data table addresses in Figure 6.29 to store the request values to be read from these five 32-bit REAL (floating point) parameters in a PowerFlex 750-Series drive: 001 - [Output Frequency], 007 - [Output Current], 008 - [Output Voltage], 009 - [Output Power], and 011 - [DC Bus Volts].

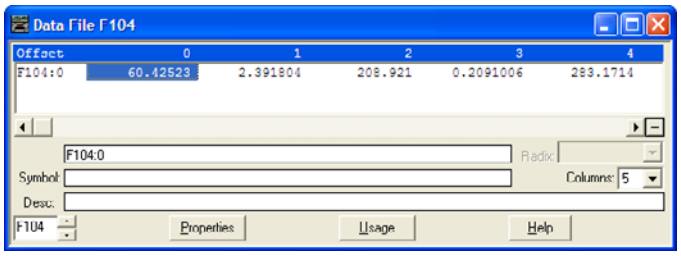
Figure 6.29 Example Scattered Read Request Data File



SLC 500 Example Scattered Read Response Data

In this example, we use the data table addresses in Figure 6.30 to store the response values that were read from the requested drive parameters. These values have been converted using a CPW (Copy Word) instruction for correct presentation.

Figure 6.30 Example Scattered Read Response Data File



In this example, the parameters have the following values:

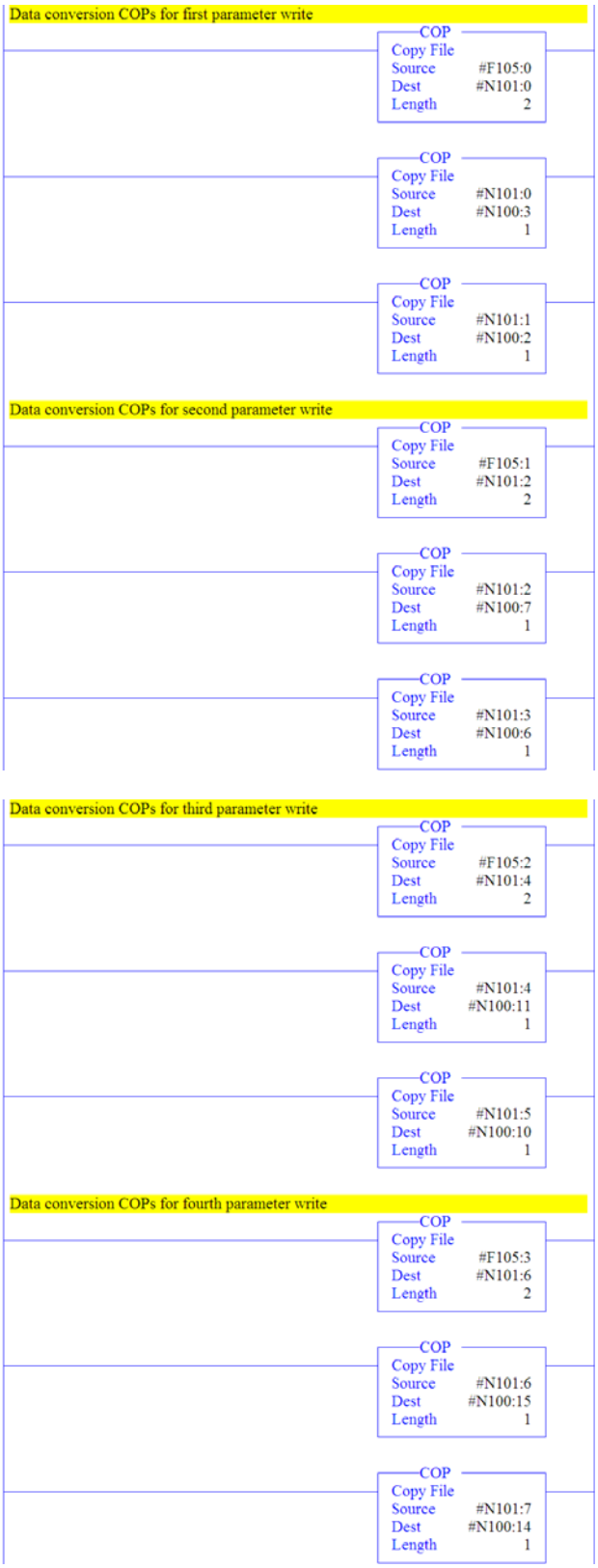
PowerFlex 750-Series Drive Parameter	Address	Read Value
1 - [Output Frequency]	F104:0	60.42523 Hz
7 - [Output Current]	F104:1	2.391804 Amp
8 - [Output Voltage]	F104:2	208.921 VAC
9 - [Output Power]	F104:3	0.2091006 kW
11 - [DC Bus Voltage]	F104:4	283.1714 VDC

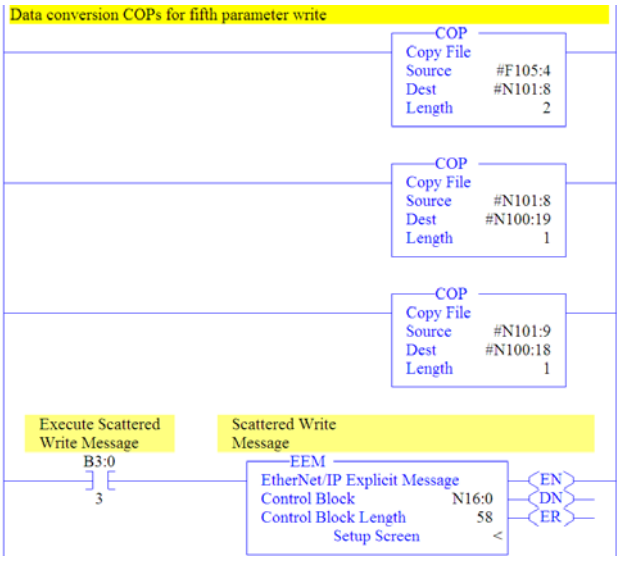
SLC 500 Example Ladder Logic Program to Write Multiple Parameters

A Custom scattered write message is used to write to multiple parameters. This write message example writes the following values to these five 32-bit REAL (floating point) parameters in a PowerFlex 750-Series drive:

PowerFlex 750-Series Drive Parameter	Write Value
536 - [Accel Time 2]	11.1 Sec.
538 - [Decel Time 2]	22.2 Sec.
575 - [Preset Speed 5]	33.3 Hz.
576 - [Preset Speed 6]	44.4 Hz.
577 - [Preset Speed 7]	55.5 Hz.

Figure 6.31 Example Ladder Logic Explicit Messaging Program for Write Multiple

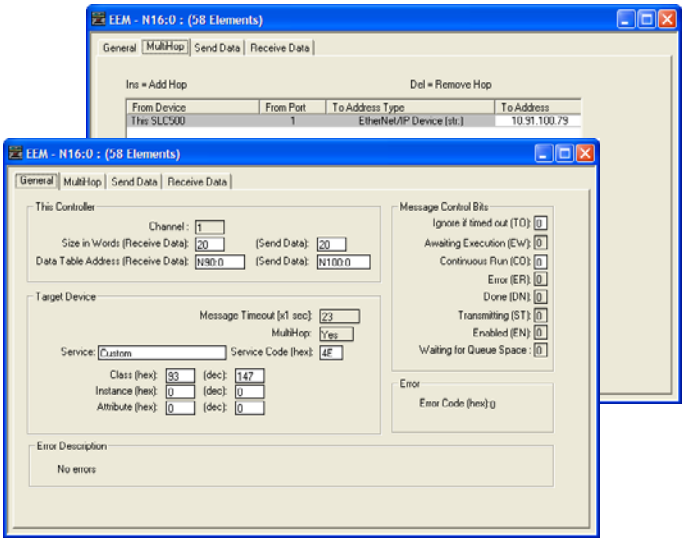




For each parameter being written to, three COP (Copy) instructions are required to convert the 16-bit integer data table addresses, for example N100:2 (Least Significant Word) and N100:3 (Most Significant Word) for the first parameter, to a 32-bit REAL (floating point) data table address F105:0 for correct presentation. The first COP instruction correctly writes the 32-bit REAL (floating point) value. The second and third COP instructions swap the LSW and MSW.

SLC 500 – Formatting a Message to Write Multiple Parameters

Figure 6.32 Custom Scattered Write Multiple Message Configuration Screens



The following table identifies the data that is required in each box to format a multiple write message.

General Tab	Example Value	Description
Size in Words		Each word size is a 16-bit integer.
Receive Data	20 (2)	Number of words to be received.
Send Data	20 (2)	Number of words to be sent.
Data Table Address		An unused controller data table address containing the message instruction.
Receive Data	N90:0	This address is the starting word of the response file.
Send Data	N100:0	This address is the starting word of the request file.
Service (1)	Custom	Required for scattered messages.
Service Code	4E (Hex.) (3)	Code for the requested service.
Class	93 (Hex.)	Class ID for the DPI Parameter Object.
Instance	0 (Dec.)	Required for scattered messages.
Attribute	0 (Dec.)	Required for scattered messages.
MultiHop Tab	**Example Value**	**Description**
To Address	10.91.100.79	IP address of the adapter connected to the drive.

(1) The default setting for Service is “Custom,” enabling entry of a Service Code not available from the Service pull-down menu. When selecting a Service other than “Custom” from the pull-down menu, an appropriate Hex. value is automatically assigned to the Service Code box which grays out (unavailable).

(2) In this example, we are writing to five 32-bit REAL (floating point) parameters. Each parameter being written to requires four contiguous 16-bit words. Scattered write messages always assume that every parameter being written to is a 32-bit parameter, regardless of its actual size. The data structure format is shown in Figure 6.36 on page 6-31. Maximum length is 128 words, which equates to 32 parameters.

(3) Service Code 4E write messages are written to the drive’s Non-Volatile Storage (EEPROM) memory, so the parameter value will remain even after the drive is power cycled. **Important:** Be very cautious as the EEPROM may quickly exceed its life cycle and cause the drive to malfunction.

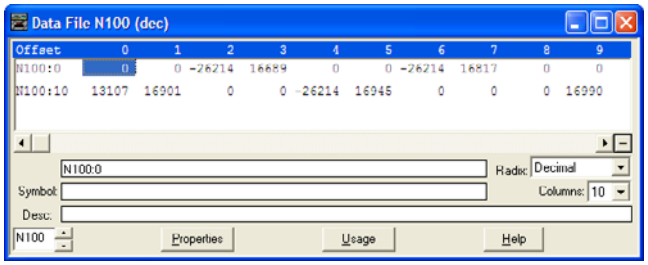
SLC 500 Example Scattered Write Request Data

In this example, we use the F105: data table addresses to store the request values to be written to these 32-bit REAL (floating point) parameters:

PowerFlex 750-Series Drive Parameter	Address	Write Value
536 - [Accel Time 2]	F105:0	11.1 Sec.
538 - [Decel Time 2]	F105:1	22.2 Sec.
575 - [Preset Speed 5]	F105:2	33.3 Hz.
576 - [Preset Speed 6]	F105:3	44.4 Hz.
577 - [Preset Speed 7]	F105:4	55.5 Hz.

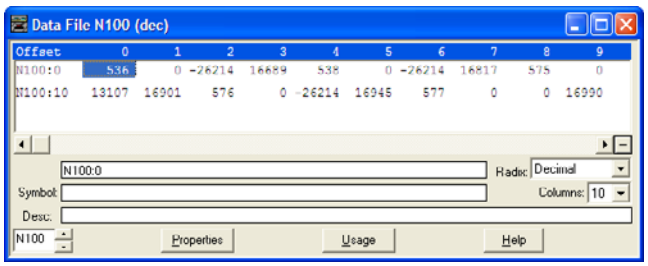
Figure 6.33 shows the parameter values which, in this example, have been converted using a CPW (Copy Word) instruction—one for each value—to correctly write their values. The CPW instruction separates the 32-bit REAL (floating point) value (for example, F105:0 which contains 11.1 seconds) into two 16-bit integers (for example N100:2 and N100:3). While the values in the 16-bit integer registers represent the actual values being written in the 32-bit floating point registers, they will not appear correct, but the message will properly decode them.

Figure 6.33 Example Scattered Write Request Unconverted Data File



To complete message configuration, the numbers of the parameters being written to must now be entered in the appropriate N100: data table registers as shown in Figure 6.34 for this example.

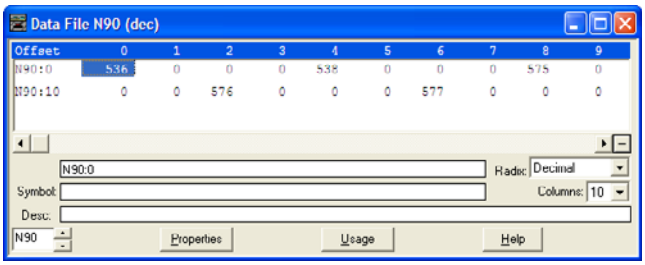
Figure 6.34 Example Scattered Write Request Unconverted Data File with Entered Parameters



SLC 500 Example Scattered Write Response Data

In this example, we use the data table addresses in Figure 6.35 to store the response values that were written to the requested drive parameters. Values of “0” indicate no errors occurred.

Figure 6.35 Example Scattered Write Response Data File



TIP: To verify that the parameter values were successfully written, use the HIM, DriveExplorer or DriveExecutive to access the parameters and view their newly written values.

SLC 500 – Explanation of Request and Response Data for Read/ Write Multiple Messaging

The data structures in Figure 6.36 use 32-bit words and can accommodate up to 32 parameters in a single message. In the Response Message, a parameter number with Bit 15 set indicates that the associated parameter value field contains an error code.

Figure 6.36 Data Structures for Scattered Read/Write Messages

	Request (Source Data)		Response (Destination Data)
Word 0	Parameter Number (LSW)	Word 0	Parameter Number (LSW)
1	Parameter Number (MSW)	1	Parameter Number (MSW)
2	Parameter Value (LSW)	2	Parameter Value (LSW)
3	Parameter Value (MSW)	3	Parameter Value (MSW)
4	Parameter Number (LSW)	4	Parameter Number (LSW)
5	Parameter Number (MSW)	5	Parameter Number (MSW)
6	Parameter Value (LSW)	6	Parameter Value (LSW)
7	Parameter Value (MSW)	7	Parameter Value (MSW)
8	Parameter Number (LSW)	8	Parameter Number (LSW)
9	Parameter Number (MSW)	9	Parameter Number (MSW)
10	Parameter Value (LSW)	10	Parameter Value (LSW)
11	Parameter Value (MSW)	11	Parameter Value (MSW)
12	Parameter Number (LSW)	12	Parameter Number (LSW)
13	Parameter Number (MSW)	13	Parameter Number (MSW)
14	Parameter Value (LSW)	14	Parameter Value (LSW)
15	Parameter Value (MSW)	15	Parameter Value (MSW)
16	Parameter Number (LSW)	16	Parameter Number (LSW)
17	Parameter Number (MSW)	17	Parameter Number (MSW)
18	Parameter Value (LSW)	18	Parameter Value (LSW)
19	Parameter Value (MSW)	19	Parameter Value (MSW)
20	Parameter Number (LSW)	20	Parameter Number (LSW)
21	Parameter Number (MSW)	21	Parameter Number (MSW)
22	Parameter Value (LSW)	22	Parameter Value (LSW)
23	Parameter Value (MSW)	23	Parameter Value (MSW)
24	Parameter Number (LSW)	24	Parameter Number (LSW)
25	Parameter Number (MSW)	25	Parameter Number (MSW)
26	Parameter Value (LSW)	26	Parameter Value (LSW)
27	Parameter Value (MSW)	27	Parameter Value (MSW)
28	Parameter Number (LSW)	28	Parameter Number (LSW)
29	Parameter Number (MSW)	29	Parameter Number (MSW)
30	Parameter Value (LSW)	30	Parameter Value (LSW)
31	Parameter Value (MSW)	31	Parameter Value (MSW)
32	Parameter Number (LSW)	32	Parameter Number (LSW)
33	Parameter Number (MSW)	33	Parameter Number (MSW)
34	Parameter Value (LSW)	34	Parameter Value (LSW)
35	Parameter Value (MSW)	35	Parameter Value (MSW)
⋮		⋮	
124	Parameter Number (LSW)	124	Parameter Number (LSW)
125	Parameter Number (MSW)	125	Parameter Number (MSW)
126	Parameter Value (LSW)	126	Parameter Value (LSW)
127	Parameter Value (MSW)	127	Parameter Value (MSW)

MicroLogix 1100 Example

When using RSLogix 500 v7.10 or lower, explicit messaging must be performed using the PCCC N-File method. For RSLogix 500 v7.20 or higher, the CIP messaging method has been added along with the PCCC N-File method. However, the CIP method is recommended because it is easier to use and understand. For this reason, only instructions for the CIP method are provided.

Important: Due to inherent limitations with the PCCC N-File method, only contiguous multiple parameters can be read or written in one explicit message.

For explicit messaging using the PCCC N-File method, the N150 N-Files must be used because they are already mapped to specific parameters in the drive and its connected peripherals. This enables direct access to any parameter.

For PCCC N150 N-File information, refer to page C-9.

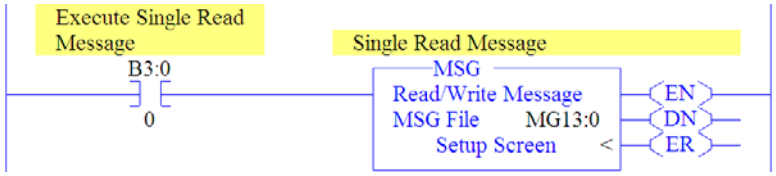
The CIP messaging method provides a Generic Get/Set Attribute Service which can be used to perform single parameter read or write and multiple parameter read or write explicit messages. Also, the Generic Set Attribute Service offers the choice of writing the data to the drive's Non-Volatile Storage (NVS) or the drive's Random Access Memory (RAM). Note that when selecting the data to be written to RAM, the data will be lost if the drive loses power.

For supported classes, instances, and attributes, refer to Appendix C, EtherNet/IP Objects.

MicroLogix 1100 Example Ladder Logic Program to Read Single Parameter

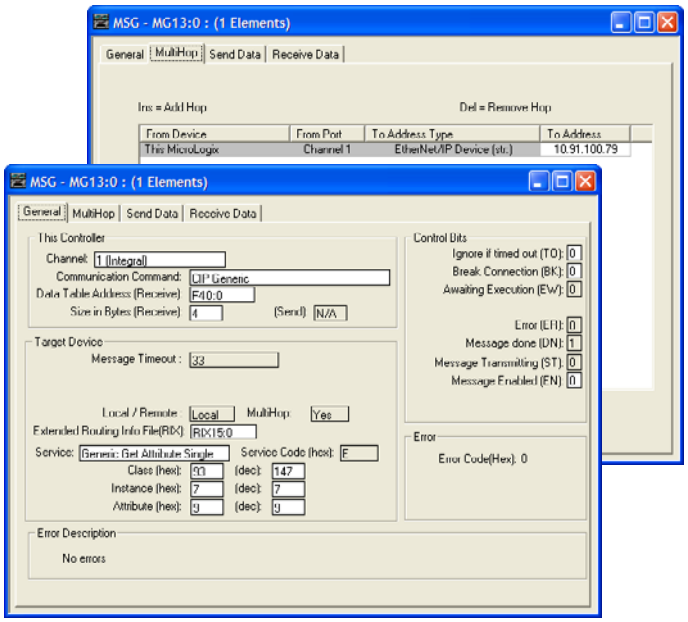
A Generic Get Attribute Single message is used to read a single parameter. This read message example reads the value of the 32-bit REAL (floating point) parameter 007 - [Output Current] in a PowerFlex 750-Series drive.

Figure 6.37 Example Ladder Logic Explicit Messaging Program for Read Single



MicroLogix 1100 – Formatting a Message to Read Single Parameter

Figure 6.38 Generic Get Attribute Single Message Configuration Screens



The following table identifies the data that is required in each box to format a single read message.

General Tab	Example Value	Description
Channel	1	Controller port to which the EtherNet/IP network is connected.
Comm… Command	CIP Generic	Used to access the Parameter Object in the adapter.
Data Table Address	F40:0 (2)	An unused controller data table address containing the message instruction. This address is the starting word of the response file.
Size in Bytes	4 (3)	Number of bytes to be transferred. Each byte size is an 8-bit integer.
Extended Routing…	RIX15:0	An unused routing information file for the controller.
Service (1)	Generic Get Attribute Single	Code for the requested service.
Class	93 (Hex.)	Class ID for the DPI Parameter Object.
Instance	7 (Dec.)	Instance number is the same as the parameter number.
Attribute	9 (Dec.)	Attribute number for the Parameter Value attribute.
MultiHop Tab	**Example Value**	**Description**
To Address	10.91.100.79	IP address of the adapter connected to the drive.

(1) The default setting for Service is “Custom,” enabling entry of a Service Code not available from the Service pull-down menu. When selecting a Service other than “Custom” from the pull-down menu, an appropriate Hex. value is automatically assigned to the Service Code box which grays out (unavailable).

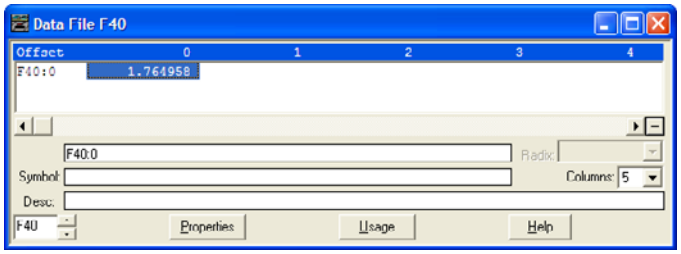
(2) In this example, Output Current is a 32-bit REAL (floating point) parameter. Therefore, set the Data Table Address to “F” type (floating point). If the parameter being read is a 32-bit integer, the Data Table Address type would be set to “L” (long word).

(3) In this example, Output Current is a 32-bit REAL (floating point) parameter. If the parameter being read is a 16-bit parameter, the Size in Bytes would be set to 2.

MicroLogix 1100 Example Get Attribute Single Response Data

In this example, we use the data table address in Figure 6.39 to store the response value (1.77 amps) that was read from drive parameter 007 - [Output Current].

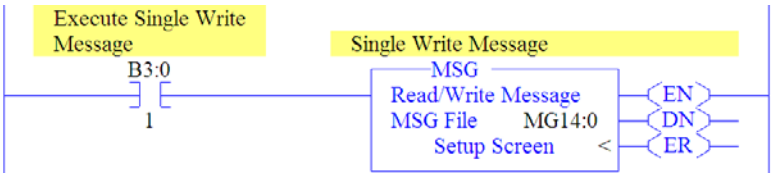
Figure 6.39 Example Get Attribute Single Response Data File



MicroLogix 1100 Example Ladder Logic Program to Write Single Parameter

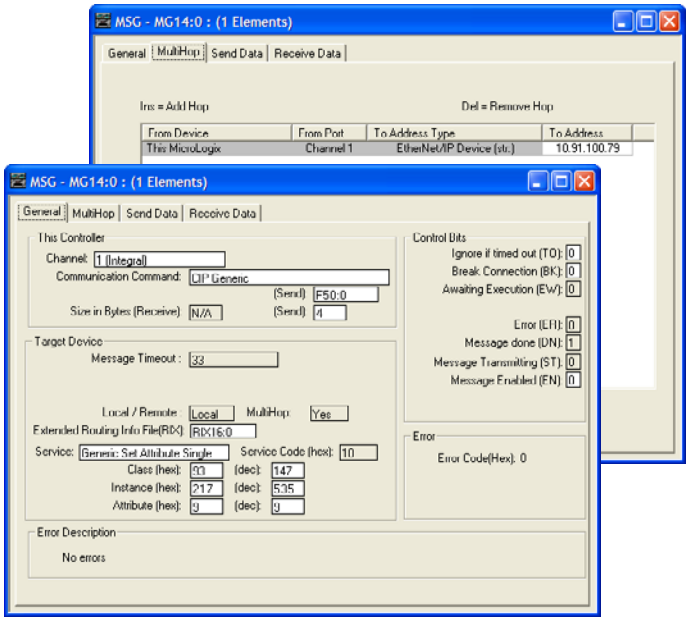
A Generic Set Attribute Single message is used to write to a single parameter. This write message example writes a value to the 32-bit REAL (floating point) parameter 535 - [Accel Time 1] in a PowerFlex 750-Series drive.

Figure 6.40 Example Ladder Logic Explicit Messaging Program for Write Single



MicroLogix 1100 – Formatting a Message to Write Single Parameter

Figure 6.41 Generic Set Attribute Single Message Configuration Screens



The following table identifies the data that is required in each box to format a single write message.

General Tab	Example Value	Description
Channel	1	Controller port to which the EtherNet/IP network is connected.
Comm… Command	CIP Generic	Used to access the Parameter Object in the adapter.
Data Table Address	F50:0 (3)	An unused controller data table address containing the message instruction. This address is the starting word of the request file.
Size in Bytes	4 (4)	Number of bytes to be transferred. Each byte size is an 8-bit integer.
Extended Routing…	RIX16:0	An unused routing information file for the controller.
Service (1)	Generic Set Attribute Single	Code for the requested service.
Class	93 (Hex.)	Class ID for the DPI Parameter Object.
Instance	535 (Dec.)	Instance number is the same as the parameter number.
Attribute (2)	9 or 10 (Dec.)	Attribute number for the Parameter Value attribute.
MultiHop Tab	**Example Value**	**Description**
To Address	10.91.100.79	IP address of the adapter connected to the drive.

(1) The default setting for Service is “Custom,” enabling entry of a Service Code not available from the Service pull-down menu. When selecting a Service other than “Custom” from the pull-down menu, an appropriate Hex. value is automatically assigned to the Service Code box which grays out (unavailable).

(2) Setting the Attribute value to “9” will write the parameter value to the drive’s Non-Volatile Storage (EEPROM) memory, so the parameter value will remain even after the drive is power cycled. **Important:** When set to “9,” be very cautious as the EEPROM may quickly exceed its life cycle and cause the drive to malfunction. Setting the Attribute value to “10” will write the parameter value to temporary memory, so the parameter value will be lost after the drive is power cycled. It is recommended to use the “10” setting when frequent write messages are required.

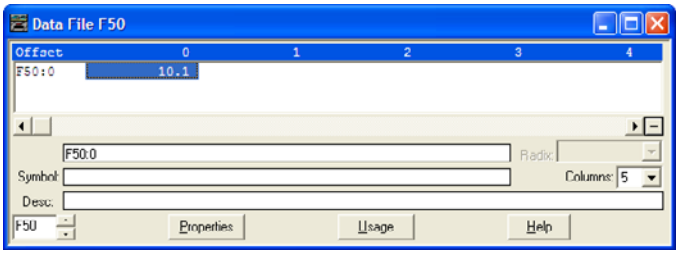
(3) In this example, Accel Time 1 is a 32-bit REAL (floating point) parameter. Therefore, set the Data Table Address to “F” type (floating point). If the parameter being written to is a 32-bit integer, the Data Table Address type would be set to “L” (long word).

(4) In this example, Accel Time 1 is a 32-bit REAL (floating point) parameter. If the parameter being written to is a 16-bit parameter, the Size in Bytes would be set to 2.

MicroLogix 1100 Example Set Attribute Single Request Data

In this example, we use the data table address in Figure 6.42 to store the request value (10.1 sec.) that was written to drive parameter 535 - [Accel Time 1].

Figure 6.42 Example Set Attribute Single Request Data File

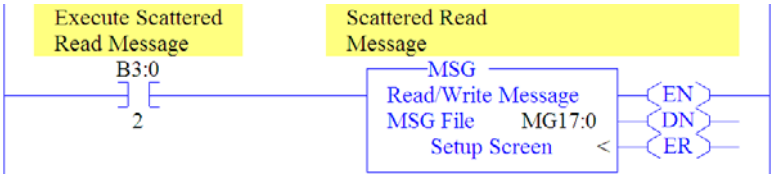


TIP: To verify that the parameter value was successfully written, use the HIM, DriveExplorer or DriveExecutive to access the parameter and view its newly written value.

MicroLogix 1100 Example Ladder Logic Program to Read Multiple Parameters

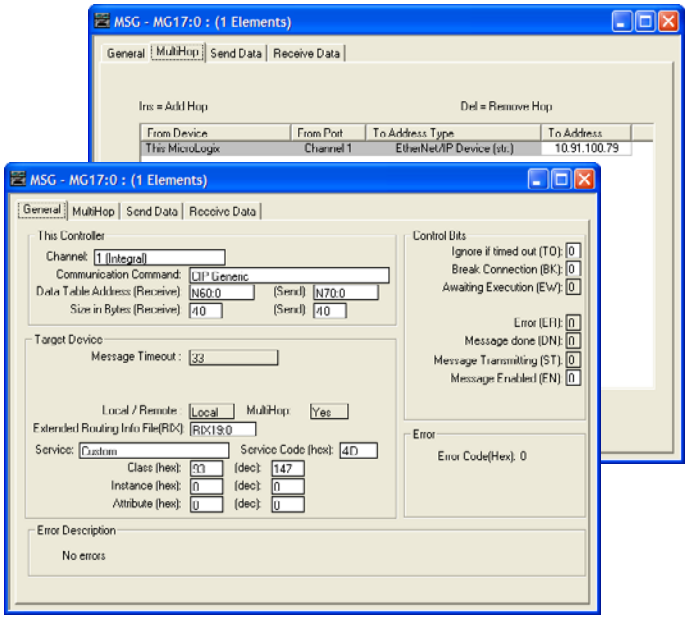
A Custom scattered read message is used to read the values of multiple parameters. This read message example reads the values of these five 32-bit REAL (floating point) parameters in a PowerFlex 750-Series drive: 001 - [Output Frequency], 007 - [Output Current], 008 - [Output Voltage], 009 - [Output Power], and 011 - [DC Bus Volts].

Figure 6.43 Example Ladder Logic Explicit Messaging Program for Read Multiple



MicroLogix 1100 – Formatting a Message to Read Multiple Parameters

Figure 6.44 Custom Scattered Read Message Configuration Screens



The following table identifies the data that is required in each box to format a multiple read message.

General Tab	Example Value	Description
Channel	1	Controller port to which the EtherNet/IP network is connected.
Comm… Command	CIP Generic	Used to access the Parameter Object in the adapter.
Data Table Address		An unused controller data table address containing the message instruction.
Receive	N60:0	This address is the starting word of the response file.
Send	N70:0	This address is the starting word of the request file.
Size in Bytes		Each byte size is an 8-bit integer.
Receive	40 (2)	Number of bytes to be received.
Send	40 (2)	Number of bytes to be sent.
Extended Routing…	RIX19:0	An unused routing information file for the controller.
Service (1)	Custom	Required for scattered messages.
Service Code	4D (Hex.)	Code for the requested service.
Class	93 (Hex.)	Class ID for the DPI Parameter Object.
Instance	0 (Dec.)	Required for scattered messages.
Attribute	0 (Dec.)	Required for scattered messages.
MultiHop Tab	**Example Value**	**Description**
To Address	10.91.100.79	IP address of the adapter connected to the drive.

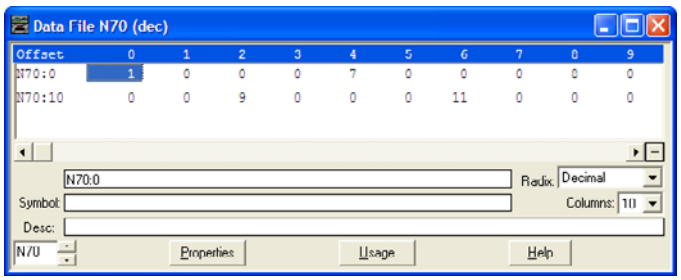
(1) The default setting for Service is “Custom,” enabling entry of a Service Code not available from the Service pull-down menu. When selecting a Service other than “Custom” from the pull-down menu, an appropriate Hex. value is automatically assigned to the Service Code box which grays out (unavailable).

(2) In this example, we are reading five 32-bit REAL (floating point) parameters. Each parameter being read requires four contiguous 16-bit words. Scattered read messages always assume that every parameter being read is a 32-bit parameter, regardless of its actual size. Therefore, the Size in Bytes must be set to 40. The data structure format is shown in Figure 6.52 on page 6-43. Maximum message length is 128 words (256 bytes), which equates to 32 parameters.

MicroLogix 1100 Example Scattered Read Request Data

In this example, we use the data table addresses in Figure 6.45 to store the request values to be read from these five 32-bit REAL (floating point) parameters in a PowerFlex 750-Series drive: 001 - [Output Frequency], 007 - [Output Current], 008 - [Output Voltage], 009 - [Output Power], and 011 - [DC Bus Volts].

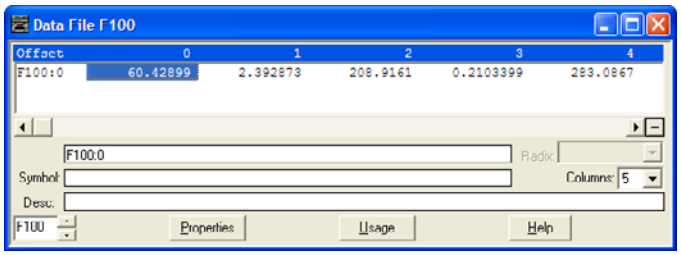
Figure 6.45 Example Scattered Read Request Data File



MicroLogix 1100 Example Scattered Read Response Data

In this example, we use the data table addresses in Figure 6.46 to store the response values that were read from the requested drive parameters. These values have been converted using a CPW (Copy Word) instruction for correct presentation.

Figure 6.46 Example Scattered Read Response Data File



In this example, the parameters have the following values:

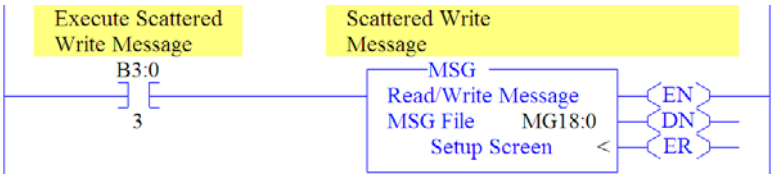
PowerFlex 750-Series Drive Parameter	Address	Read Value
1 - [Output Frequency]	F100:0	60.42899 Hz
7 - [Output Current]	F100:1	2.392873 Amp
8 - [Output Voltage]	F100:2	208.9161 VAC
9 - [Output Power]	F100:3	0.2103399 kW
11 - [DC Bus Voltage]	F100:4	283.0867 VDC

MicroLogix 1100 Example Ladder Logic Program to Write Multiple Parameters

A Custom scattered write message is used to write to multiple parameters. This write message example writes the following values to these five 32-bit REAL (floating point) parameters in a PowerFlex 750-Series drive:

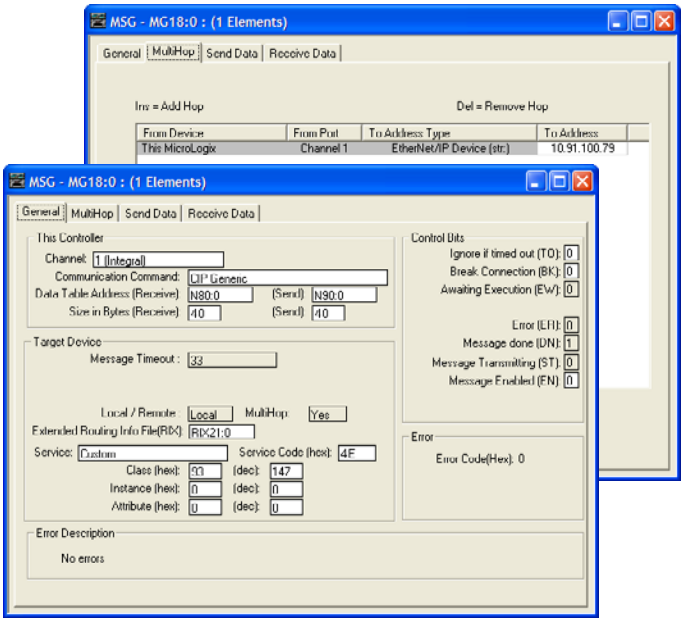
PowerFlex 750-Series Drive Parameter	Write Value
536 - [Accel Time 2]	11.1 Sec.
538 - [Decel Time 2]	22.2 Sec.
575 - [Preset Speed 5]	33.3 Hz.
576 - [Preset Speed 6]	44.4 Hz.
577 - [Preset Speed 7]	55.5 Hz.

Figure 6.47 Example Ladder Logic Explicit Messaging Program for Write Multiple



MicroLogix 1100 – Formatting a Message to Write Multiple Parameters

Figure 6.48 Custom Scattered Write Multiple Message Configuration Screens



The following table identifies the data that is required in each box to format a multiple write message.

General Tab	Example Value	Description
Channel	1	Controller port to which the EtherNet/IP network is connected.
Comm… Command	CIP Generic	Used to access the Parameter Object in the adapter.
Data Table Address		An unused controller data table address containing the message instruction.
Receive	N80:0	This address is the starting word of the response file.
Send	N90:0	This address is the starting word of the request file.
Size in Bytes		Each byte size is an 8-bit integer.
Receive	40 (2)	Number of bytes to be received.
Send	40 (2)	Number of bytes to be sent.
Extended Routing…	RIX21:0	An unused routing information file for the controller.
Service (1)	Custom	Required for scattered messages.
Service Code	4E (Hex.) (3)	Code for the requested service.
Class	93 (Hex.)	Class ID for the DPI Parameter Object.
Instance	0 (Dec.)	Required for scattered messages.
Attribute	0 (Dec.)	Required for scattered messages.
MultiHop Tab	**Example Value**	**Description**
To Address	10.91.100.79	IP address of the adapter connected to the drive.

(1) The default setting for Service is “Custom,” enabling entry of a Service Code not available from the Service pull-down menu. When selecting a Service other than “Custom” from the pull-down menu, an appropriate Hex. value is automatically assigned to the Service Code box which grays out (unavailable).

(2) In this example, we are writing to five 32-bit REAL (floating point) parameters. Each parameter being written to requires four contiguous 16-bit words. Scattered write messages always assume that every parameter being written to is a 32-bit parameter, regardless of its actual size. Therefore, the Size in Bytes must be set to 40. The data structure format is shown in Figure 6.52 on page 6-43. Maximum length is 128 words (256 bytes), which equates to 32 parameters.

(3) Service Code 4E write messages are written to the drive’s Non-Volatile Storage (EEPROM) memory, so the parameter value will remain even after the drive is power cycled. **Important:** Be very cautious as the EEPROM may quickly exceed its life cycle and cause the drive to malfunction.

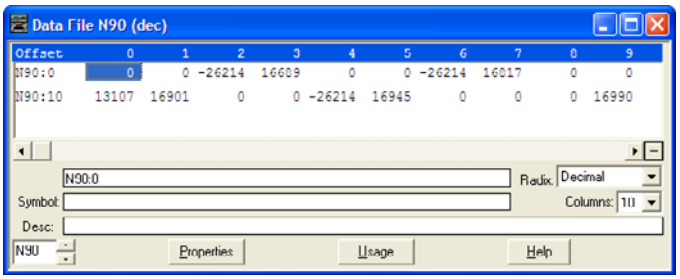
MicroLogix 1100 Example Scattered Write Request Data

In this example, we use the F101: data table addresses to store the request values to be written to these 32-bit REAL (floating point) parameters:

PowerFlex 750-Series Drive Parameter	Address	Write Value
536 - [Accel Time 2]	F101:0	11.1 Sec.
538 - [Decel Time 2]	F101:1	22.2 Sec.
575 - [Preset Speed 5]	F101:2	33.3 Hz.
576 - [Preset Speed 6]	F101:3	44.4 Hz.
577 - [Preset Speed 7]	F101:4	55.5 Hz.

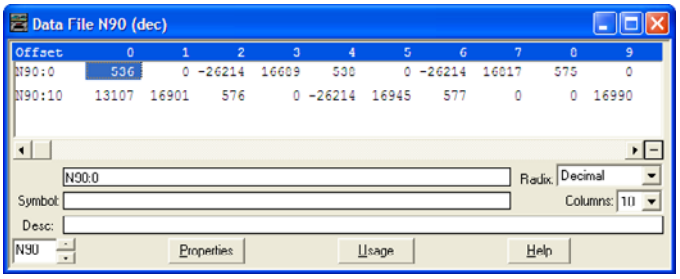
Figure 6.49 shows the parameter values which, in this example, have been converted using a CPW (Copy Word) instruction—one for each value—to correctly write their values. The CPW instruction separates the 32-bit REAL (floating point) value (for example, F101:0 which contains 11.1 seconds) into two 16-bit integers (for example N90:2 and N90:3). While the values in the 16-bit integer registers represent the actual values being written in the 32-bit floating point registers, they will not appear correct, but the message will properly decode them.

Figure 6.49 Example Scattered Write Request Unconverted Data File



To complete message configuration, the numbers of the parameters being written to must now be entered in the appropriate N90: data table registers as shown in Figure 6.50 for this example.

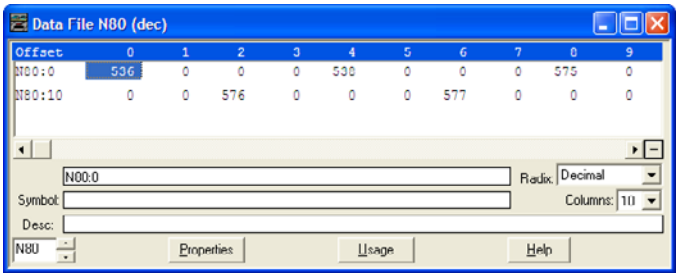
Figure 6.50 Example Scattered Write Request Unconverted Data File with Entered Parameters



MicroLogix 1100 Example Scattered Write Response Data

In this example, we use the data table addresses in Figure 6.51 to store the response values that were written to the requested drive parameters. Values of “0” indicate no errors occurred.

Figure 6.51 Example Scattered Write Response Data File



TIP: To verify that the parameter values were successfully written, use the HIM, DriveExplorer or DriveExecutive to access the parameters and view their newly written values.

MicroLogix 1100 – Explanation of Request and Response Data for Read/Write Multiple Messaging

The data structures in Figure 6.52 use 32-bit words and can accommodate up to 32 parameters in a single message. In the Response Message, a parameter number with Bit 15 set indicates that the associated parameter value field contains an error code.

Figure 6.52 Data Structures for Scattered Read/Write Messages

	Request (Source Data)		Response (Destination Data)
Word 0	Parameter Number (LSW)	Word 0	Parameter Number (LSW)
1	Parameter Number (MSW)	1	Parameter Number (MSW)
2	Parameter Value (LSW)	2	Parameter Value (LSW)
3	Parameter Value (MSW)	3	Parameter Value (MSW)
4	Parameter Number (LSW)	4	Parameter Number (LSW)
5	Parameter Number (MSW)	5	Parameter Number (MSW)
6	Parameter Value (LSW)	6	Parameter Value (LSW)
7	Parameter Value (MSW)	7	Parameter Value (MSW)
8	Parameter Number (LSW)	8	Parameter Number (LSW)
9	Parameter Number (MSW)	9	Parameter Number (MSW)
10	Parameter Value (LSW)	10	Parameter Value (LSW)
11	Parameter Value (MSW)	11	Parameter Value (MSW)
12	Parameter Number (LSW)	12	Parameter Number (LSW)
13	Parameter Number (MSW)	13	Parameter Number (MSW)
14	Parameter Value (LSW)	14	Parameter Value (LSW)
15	Parameter Value (MSW)	15	Parameter Value (MSW)
16	Parameter Number (LSW)	16	Parameter Number (LSW)
17	Parameter Number (MSW)	17	Parameter Number (MSW)
18	Parameter Value (LSW)	18	Parameter Value (LSW)
19	Parameter Value (MSW)	19	Parameter Value (MSW)
20	Parameter Number (LSW)	20	Parameter Number (LSW)
21	Parameter Number (MSW)	21	Parameter Number (MSW)
22	Parameter Value (LSW)	22	Parameter Value (LSW)
23	Parameter Value (MSW)	23	Parameter Value (MSW)
24	Parameter Number (LSW)	24	Parameter Number (LSW)
25	Parameter Number (MSW)	25	Parameter Number (MSW)
26	Parameter Value (LSW)	26	Parameter Value (LSW)
27	Parameter Value (MSW)	27	Parameter Value (MSW)
28	Parameter Number (LSW)	28	Parameter Number (LSW)
29	Parameter Number (MSW)	29	Parameter Number (MSW)
30	Parameter Value (LSW)	30	Parameter Value (LSW)
31	Parameter Value (MSW)	31	Parameter Value (MSW)
32	Parameter Number (LSW)	32	Parameter Number (LSW)
33	Parameter Number (MSW)	33	Parameter Number (MSW)
34	Parameter Value (LSW)	34	Parameter Value (LSW)
35	Parameter Value (MSW)	35	Parameter Value (MSW)
⋮		⋮	
124	Parameter Number (LSW)	124	Parameter Number (LSW)
125	Parameter Number (MSW)	125	Parameter Number (MSW)
126	Parameter Value (LSW)	126	Parameter Value (LSW)
127	Parameter Value (MSW)	127	Parameter Value (MSW)

Notes:

Troubleshooting

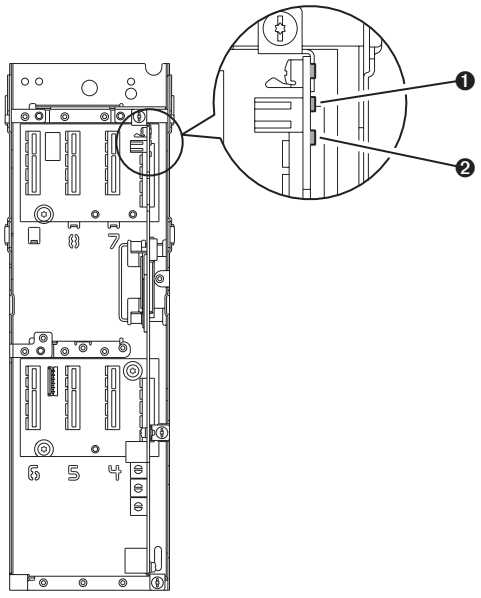
This chapter provides information for diagnosing and troubleshooting potential problems with the adapter and network.

Understanding the Status Indicators

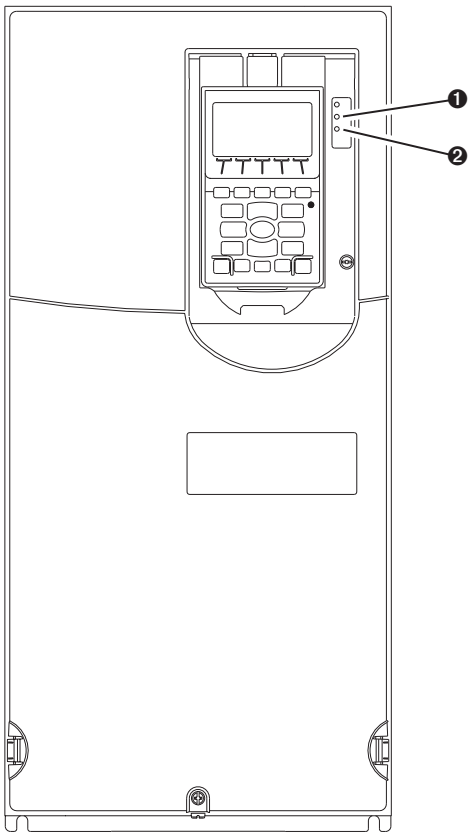
The adapter has two status indicators. They can be viewed with the HIM bezel closed or open (Figure 7.1).

Figure 7.1 Status Indicators

Embedded EtherNet/IP adapter indicators on main control board in drive control pod.



Indicators shown with HIM bezel closed and drive cover installed.



Item	Status Indicator	Description	Page
❶	ENET	EtherNet/IP Connection Status	7-2
❷	LINK	EtherNet/IP Transmit Status	7-2

ENET Status Indicator

Status	Cause	Corrective Actions
Off	The adapter and/or network is not powered, the adapter is not properly connected to the network, or the adapter needs an IP address.	• Apply power to the drive and network. • Securely connect the adapter to the network using an Ethernet cable. • Correctly connect the Ethernet cable to the Ethernet connector. • Set a unique IP address using the adapter switches, a BOOTP server, or by disabling BOOTP and using adapter parameters.
Solid Red	The adapter failed the duplicate IP address detection test.	Configure the adapter to use a unique IP address and cycle power.
Flashing Red	An EtherNet/IP connection has timed out.	• Apply power to the scanner or enable the peer device that will send I/O. • Check the IGMP Snooping/Ethernet Switches for correct operation. • Check the amount of traffic on the network.
Flashing Red/Green	The adapter is performing a self-test.	No action required.
Flashing Green	The adapter is properly connected but is not communicating with any devices on the network.	• Place the controller in RUN mode, or apply power to the peer device that will send I/O. • Program the controller or peer device to recognize and transmit I/O or make a messaging connection to the adapter. • Configure the adapter for the program in the controller or the I/O from the peer device.
Solid Green	The adapter is properly connected and communicating on the network.	No action required.

LINK Status Indicator

Status	Cause	Corrective Actions
Off	The adapter is not powered or is not transmitting on the network.	If ENET indicator is off: • Apply power to the drive. • Securely connect the adapter to the network using an Ethernet cable. • Correctly connect the Ethernet cable to the Ethernet connector. • Set a unique IP address using the adapter switches, a BOOTP server, or by disabling BOOTP and using adapter parameters. If ENET indicator is solid red: • Configure the adapter to use a unique IP address and cycle power. If ENET indicator is flashing red/green or red: • Check the IP address in the adapter and scanner, and verify that the controller can communicate with the adapter. • Ping the adapter. Normal condition if the adapter is idle.
Flashing Green	The adapter is transmitting on the network.	No action required.
Solid Green	The adapter is ready to communicate.	No action required.

Viewing Adapter Diagnostic Items

If you encounter unexpected communications problems, the adapter's diagnostic items may help you or Rockwell Automation personnel troubleshoot the problem. Adapter diagnostic items can be viewed using the enhanced PowerFlex 7-Class HIM, DriveExplorer software (version 6.01 or higher), or DriveExecutive software (version 5.01 or higher). For details on viewing diagnostic items using the HIM, refer to the *Enhanced PowerFlex 7-Class HIM User Manual (publication 20HIM-UM001)*.

Table 7.A Adapter Diagnostic Items

No.	Name	Description
1	Common Logic Cmd	The present value of the Common Logic Command being transmitted to the drive by this adapter.
2	Prod Logic Cmd	The present value of the Product Logic Command being transmitted to the drive by this adapter.
3	Reference	The present value of the Reference being transmitted to the drive by this adapter.
4	Common Logic Sts	The present value of the Common Logic Status being received from the drive by this adapter.
5	Prod Logic Sts	The present value of the Product Logic Status being received from the drive by this adapter.
6	Feedback	The present value of the Feedback being received from the drive by this adapter.
7	Input Size	Size of data transferred from the network to the drive.
8	Output Size	Size of data transferred from the drive to the network.
9	DL Fr Net Avail	The number of From Net Datalinks currently available to the adapter.
10	DL To Net Avail	The number of To Net Datalinks currently available to the adapter.
11	DL Fr Net 01 Val	The present value of respective **DL From Net xx** parameter being transmitted to the drive by this adapter. (If not using a Datalink, its respective value should be zero.)
12	DL Fr Net 02 Val	
13	DL Fr Net 03 Val	
14	DL Fr Net 04 Val	
15	DL Fr Net 05 Val	
16	DL Fr Net 06 Val	
17	DL Fr Net 07 Val	
18	DL Fr Net 08 Val	
19	DL Fr Net 09 Val	
20	DL Fr Net 10 Val	
21	DL Fr Net 11 Val	
22	DL Fr Net 12 Val	
23	DL Fr Net 13 Val	
24	DL Fr Net 14 Val	
25	DL Fr Net 15 Val	
26	DL Fr Net 16 Val	
27	DL To Net 01 Val	The present value of respective **DL To Net xx** parameter being received from the drive by this adapter. (If not using a Datalink, its respective value should be zero.)
28	DL To Net 02 Val	
29	DL To Net 03 Val	
30	DL To Net 04 Val	
31	DL To Net 05 Val	
32	DL To Net 06 Val	
33	DL To Net 07 Val	
34	DL To Net 08 Val	
35	DL To Net 09 Val	
36	DL To Net 10 Val	
37	DL To Net 11 Val	
38	DL To Net 12 Val	
39	DL To Net 13 Val	
40	DL To Net 14 Val	
41	DL To Net 15 Val	
42	DL To Net 16 Val	

Table 7.A Adapter Diagnostic Items (Continued)

No.	Name	Description
43 44 45 46 47 48	HW Addr 1 HW Addr 2 HW Addr 3 HW Addr 4 HW Addr 5 HW Addr 6	Decimal value of each byte in the adapter's Ethernet hardware address. 255 : 255 : 255 : 255 : 255 : 255 [HW Addr 1] [HW Addr 2] [HW Addr 3] [HW Addr 4] HW Addr 5] [HW Addr 6]
49 50 51 52	IP Addr Act 1 IP Addr Act 2 IP Addr Act 3 IP Addr Act 4	Value of each byte in the adapter's current IP address. A value of "0" appears if the adapter does not currently have an IP address. 255 . 255 . 255 . 255 [IP Addr Act 1] [IP Addr Act 2] [IP Addr Act 3] [IP Addr Act 4]
53 54 55 56	Subnet Act 1 Subnet Act 2 Subnet Act 3 Subnet Act 4	Value of each byte in the adapter's current subnet mask. A value of "0" appears if the adapter does not currently have a subnet mask. 255 . 255 . 255 . 255 [Subnet Act 1] [Subnet Act 2] [Subnet Act 3] [Subnet Act 4]
57 58 59 60	Gateway Act 1 Gateway Act 2 Gateway Act 3 Gateway Act 4	Value of each byte in the adapter's current gateway address. A value of "0" appears if the adapter does not currently have a gateway address. 255 . 255 . 255 . 255 [Gateway Act 1] [Gateway Act 2] [Gateway Act 3] [Gateway Act 4]
61	Net Rx Overruns	Number of receive buffer overruns reported by the Ethernet hardware.
62	Net Rx Packets	Number of Ethernet packets that the adapter has received.
63	Net Rx Errors	Number of receive errors reported by the Ethernet hardware.
64	Net Tx Packets	Number of Ethernet packets that the adapter has sent.
65	Net Tx Errors	Number of transmit errors reported by the Ethernet hardware.
66	Last TCP Reset	Last reason that the adapter reset or rejected a TCP/IP connection.
67	Missed IO Pkts	Number of incoming I/O connection packets that the adapter did not receive.
68	Net Addr Sw	The present value of the adapter node address switches.

Viewing and Clearing Events

The adapter maintains an event queue that reports the history of its actions. You can view the event queue using the enhanced PowerFlex 7-Class HIM, DriveExplorer (6.01 or higher) software, or DriveExecutive (5.01 or higher) software. For details on viewing and clearing events using the HIM, refer to the *Enhanced PowerFlex 7-Class HIM User Manual (publication 20HIM-UM001)*.

Many events in the event queue occur under normal operation. If you encounter unexpected communications problems, the events may help you or Allen-Bradley personnel troubleshoot the problem. The following events may appear in the event queue:

Table 7.B Adapter Events

Code	Event	Description
1	No Event	Empty event queue entry.
2	Device Power Up	The adapter was powered up normally.
3	Device Reset	The adapter was manually reset.
4	EEPROM CRC Error	The EEPROM in the adapter is corrupt.
5	App Updated	The application code in the adapter was updated.
6	Boot Updated	The boot code in the adapter was updated.
7-24	Reserved	—
25	DPI Manual Reset	The adapter was reset.
26-28	Reserved	—
29	Net Link Up	An Ethernet link is available for the adapter.
30	Net Link Down	The Ethernet link was removed from the adapter.
31	Net Dup Address	The adapter uses the same IP address as another device on the network.
32	Net Comm Fault	The adapter detected a communications fault on the network.
33	Net Sent Reset	The adapter received a reset from the network.
34	Net IO Close	An I/O connection from the network to the adapter was closed.
35	Net Idle Fault	The adapter is receiving “idle” packets from the network.
36	Net IO Open	An I/O connection from the network to the adapter has been opened.
37	Net IO Timeout	An I/O connection from the network to the adapter has timed out.
38	Net IO Size Err	The adapter received an incorrectly sized I/O packet.
39	PCCC IO Close	The device sending PCCC Control messages to the adapter has set the PCCC Control Timeout to zero.
40	PCCC IO Open	The adapter has begun receiving PCCC Control messages (the PCCC Control Timeout was previously set to a non-zero value).
41	PCCC IO Timeout	The adapter has not received a PCCC Control message for longer than the PCCC Control Timeout.
42	Msg Ctrl Open	The timeout attribute in either the CIP Register or Assembly object was written with a non-zero value, allowing control messages to be sent to the adapter.
43	Msg Ctrl Close	The timeout attribute in either the CIP Register or Assembly object was written with a zero value, disallowing control messages to be sent to the adapter.
44	Msg Ctrl Timeout	The timeout attribute in either the CIP Register or Assembly object elapsed between accesses of those objects.
45	Peer IO Open	The adapter received the first Peer I/O message.
46	Peer IO Timeout	The adapter has not received a Peer I/O message for longer than the Peer I/O Timeout.
47-54	Reserved	—
55	BOOTP Response	The adapter received a response to its BOOTP request.
56	E-mail Failed	The adapter encountered an error attempting to send a requested e-mail message.
57	Option Card Flt	The adapter experienced a generic fault condition (drive only).
58	Module Defaulted	The adapter has been set to defaults.

Notes:

Viewing the Adapter Web Pages

This chapter provides instructions on how to monitor the adapter and connected PowerFlex drive by using the adapter's web interface.

Future enhancements may result in adapter web pages that look different than the examples shown in this chapter.

Accessing the Adapter Web Home Page

After configuring the adapter, you can view its web pages. These pages present information about the adapter, the drive to which it is connected, and the other DPI devices connected to the drive such as a HIM.

By default the adapter web pages are disabled. To enable the adapter web pages, set **Parameter 52 - [Web Enable]** to "1" (Enabled) and then reset the adapter for the change to take effect.

The adapter can be configured to automatically send e-mail messages to desired addresses when selected drive faults occur and/or are cleared, and/or when the adapter takes a communication or idle fault action.

Bit 0 of **Parameter 53 - [Web Features]** can be used to protect the configured settings. For more details, see Configure E-mail Notification Web Page on page 8-6.

Viewing the Web Pages of the Adapter

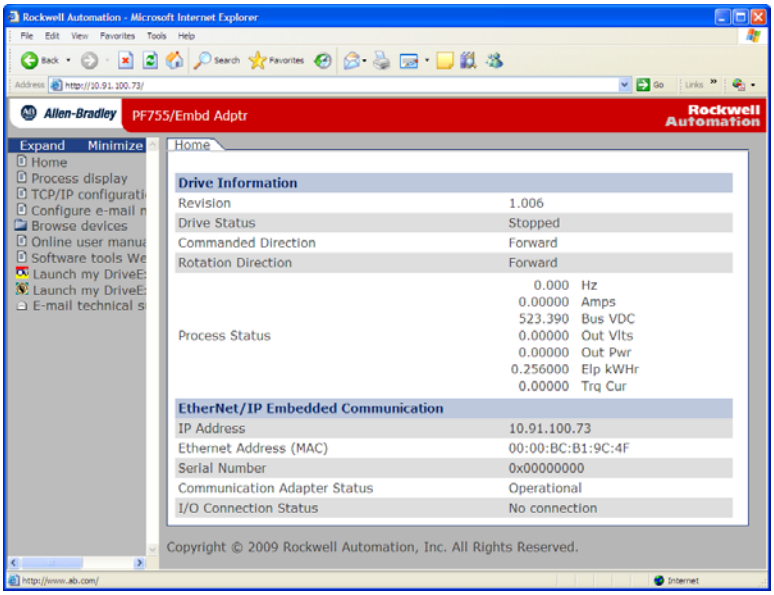
1. On a computer with access to the EtherNet/IP network on which the drive/adapter is installed, launch a web browser such as Microsoft™ Internet Explorer (version 5.0 or greater).

   The computer can access the adapter web pages if it is connected to the same network as the drive/adapter, or if it is connected to a network with access to the drive/adapter's network via a gateway device (for example, a router).

2. In the Address box, type the IP address of the adapter, and then press ENTER. The adapter web Home Page (Figure 8.1) appears.

Important: Clicking the browser's Refresh button always re-displays the Home Page even while viewing another adapter web page.

Figure 8.1 Adapter Web Home Page Example



Title Bar on Adapter Web Pages

The title bar appears on all adapter web pages, including its Home Page. It consists of three elements:

Title Bar Element	Description
Allen-Bradley logo (at far left)	This logo is a hyperlink. Click it to view the ab.com web Home Page.
Adapter Title (middle)	Shows the adapter type or the title configured by the user (via a HIM, DriveExecutive or DriveExplorer).
Rockwell Automation logo (at far right)	This logo is a hyperlink. Click it to view the Rockwell Automation web Home Page.

Navigation Menu on Adapter Web Pages

The navigation menu appears on the left side of all adapter web pages, including its Home page. The navigation menu consists of links and link folders which can be expanded or minimized. The following table shows all navigation menu links and link folders:

Link/Folder	Description
Home link	Click this link to view the adapter's Home Page (Figure 8.1).
Process Display link	Click this link to view the Host's Process Display pop-up window (Figure 8.2)
TCP/IP configuration link	Click this link to view the adapter's TCP/IP Configuration web page showing information about the TCP/IP configuration, such as the adapter's IP address and the number of packets being sent. Figure 8.3 shows an example TCP/IP Configuration web page.
Configure e-mail notification link	Click this link to view the adapter's Configure E-mail Notification web page (Figure 8.4) to configure the adapter to send automatic e-mail messages. An example e-mail message is shown in Figure 8.5.
Browse DPI devices folder	Click this folder to expand and view the Port folders for all present DPI devices, including the drive, adapter, and other DPI devices connected to the drive such as a HIM.
Port x folders	Click a respective Port folder to expand and view its device's various links which take you to related information pages. For Port 00 (PowerFlex 755 Drive) example information pages, see Figure 8.6, Figure 8.7, and Figure 8.8.
Online user manuals link	Click this link to view Rockwell Automation's web page with documentation for drives and other devices.
Software tools Web site link	Click this link to view Allen-Bradley's web page with information about software tools such as DriveExplorer and DriveExecutive.
Launch my DriveExplorer software link	Click this link to launch the DriveExplorer software already installed on your PC.
Launch my DriveExecutive software link	Click this link to launch the DriveExecutive software already installed on your PC.
E-mail technical support link	Click this link to view a new e-mail message window to send a message to Allen-Bradley's Technical Support Team.

Information on Adapter Home Page

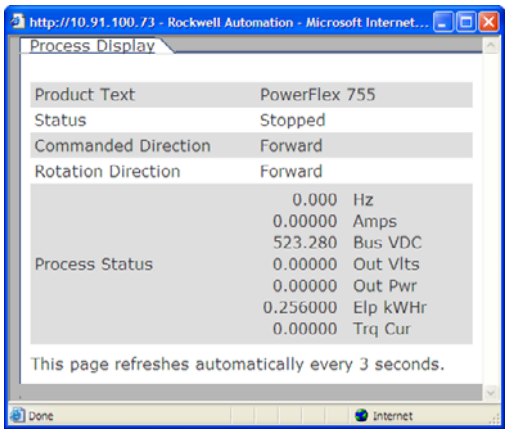
The adapter Home Page displays the following information for the host drive and adapter:

Information for	Description
Host Drive	• Revision • Status • Commanded Direction • Rotation Direction • Process Status
EtherNet/IP Embedded Adapter	• IP Address • Ethernet Address (MAC) • Serial Number • Adapter Status • I/O Connection Status

Process Display Pop-up Window

The Process Display pop-up window dynamically shows the host drive's information. To view this window, click the "Process Display" link in the navigation menu.

Figure 8.2 Example of Process Display Pop-up Window

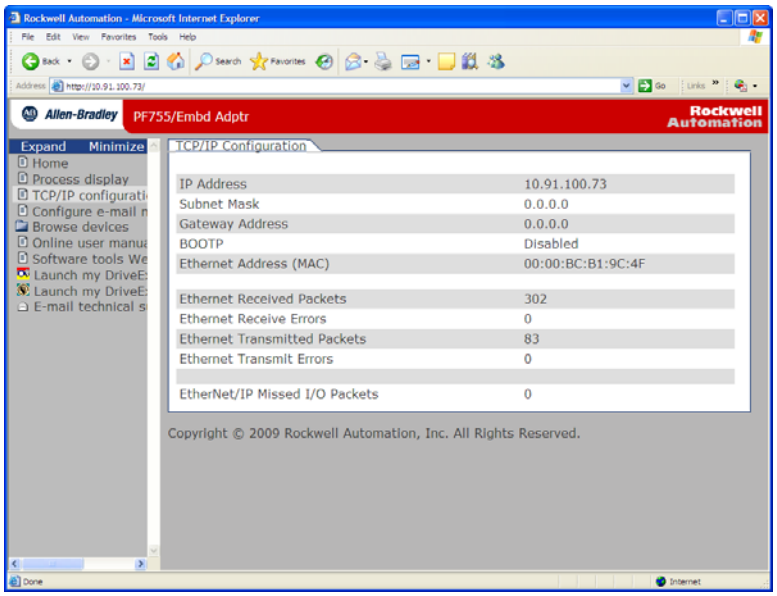


Information	Description
Product Text	Description of host.
Status	Status of host.
Commanded Direction	Commanded direction of host.
Rotation Direction	Rotation direction of host.
Process Status	Line 1: Dynamic value of the host drive feedback parameter, which is not selectable. Lines 2 - 7: Dynamic values of the default displayed host parameters. Each default parameter shown on these lines can be changed by the user (via a HIM, DriveExecutive or DriveExplorer).

TCP/IP Configuration Web Page

The TCP/IP Configuration web page provides information about the adapter's Ethernet settings and network activities.

Figure 8.3 Example of TCP/IP Configuration Web Page



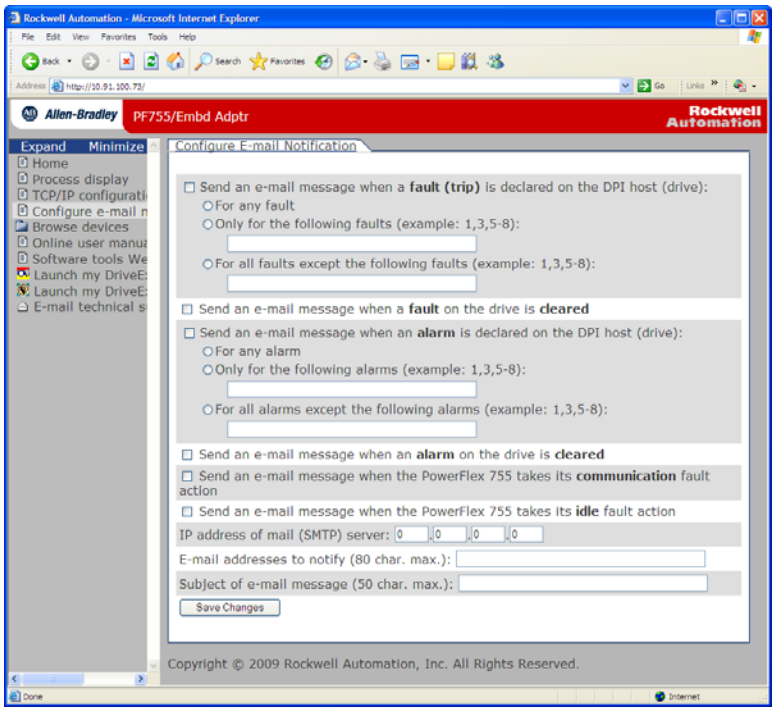
Information	Description
IP Address	IP address of the adapter.
Subnet Mask	Subnet mask for the adapter's network.
Gateway Address	Address for the gateway device on the adapter's network.
BOOTP	Whether BOOTP is being used to configure the adapter's network information.
Ethernet Address (MAC)	Hardware address for the adapter.
Ethernet Received Packets	Number of packets that the adapter has received.
Ethernet Receive Errors	Number of receive errors reported by the hardware.
Ethernet Transmitted Packets	Number of packets that the adapter has sent.
Ethernet Transmit Errors	Number of transmit errors reported by the hardware.
EtherNet/IP Missed I/O Packets	Number of I/O connection packets that the adapter did not receive.

Configure E-mail Notification Web Page

The Configure E-mail Notification web page contains selections and data fields for configuring the adapter to automatically send e-mail messages to desired addresses when selected types of events occur.

By default, settings are not protected. After configuration, settings can be protected by using **Parameter 53 - [Web Features]** to set E-mail Cfg Bit 0 value to "0" (Disabled). To change a protected configuration, it must first be unprotected by setting the E-mail Cfg Bit 0 value back to "1" (Enabled).

Figure 8.4 Example of Configure E-mail Notification Web Page



Configuring E-mail Notification

1. Click the desired DPI host check boxes corresponding to what you want to occur that will send e-mail notification.
   - If you only want e-mail notification when specific faults or alarms occur, click this radio button and enter the fault or alarm numbers in the box.
   - If you only want e-mail notification when all faults or alarms except specific faults or alarms occur, click this radio button and enter the fault or alarm numbers in the box.
2. Click the communication fault and/or idle fault check box if you want e-mail notification when these faults occur.

3. Enter the following information in their respective boxes:

Information	Description
"IP address of..."	Type in the address of the mail server that will be used to deliver the e-mail messages. (When the IP address is unknown, read the TIP shown below this table to determine the mail server address.)
"E-mail addresses to notify..."	Type in addresses to where you want e-mail messages to be sent. Multiple addresses can be used, but they must be separated by commas (comma delimited).
"Subject of e-mail message..."	Type in the desired subject text for the e-mail message.

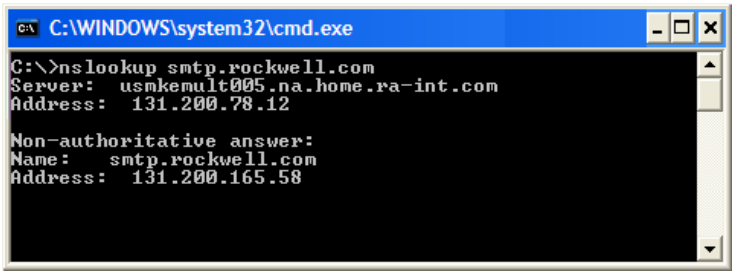
TIP: If the IP address of the e-mail server is unknown, you can contact your IT department or use the DOS window to enter a command to find its IP address.

A. On the Windows task bar, click **Start > Run** to display the Run window.

B. In the Run window Open field, type "cmd" and click **OK** to display the DOS window.

C. On the c:\ > command line, type "nslookup [name of e-mail server]." The entry "c:\ > nslookup smtp.company.com" is an example.

D. Press **ENTER** to display the e-mail server IP address (see example below). The second (bottom) IP address shown in the DOS window (for this example, 131.200.165.58) should be typed into the E-mail Notification Web Page (Figure 8.4).



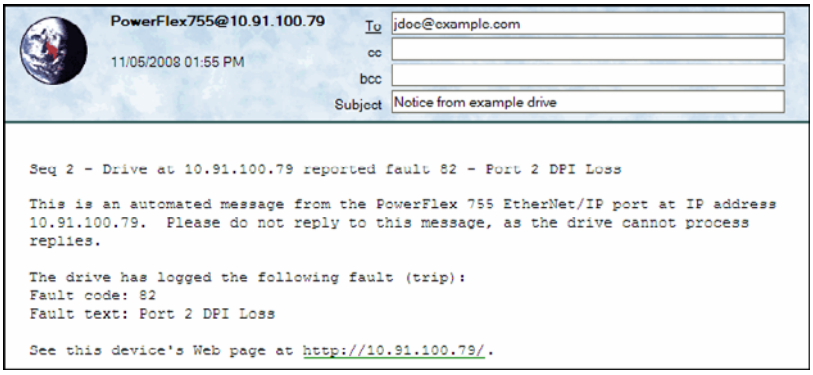
4. Click **Save changes**.

Important: After configuring E-mail Notification, it is recommended to protect the settings. Otherwise the configuration can be changed anytime the web page is accessed with a browser.

Use **Parameter 53 - [Web Features]** to set E-mail Cfg Bit 0 value to "0" (Disabled) to protect the settings.

Figure 8.5 shows an example e-mail message automatically sent by the adapter in response to selected events.

Figure 8.5 Example of E-mail Message Sent by Adapter



TIP: To stop e-mail messages, uncheck all of the "Send an e-mail message when…" boxes.

Disabling the adapter web pages by setting **Parameter 52 - [Web Enable]** to "0" (Disabled) will NOT stop the adapter from sending e-mail messages.

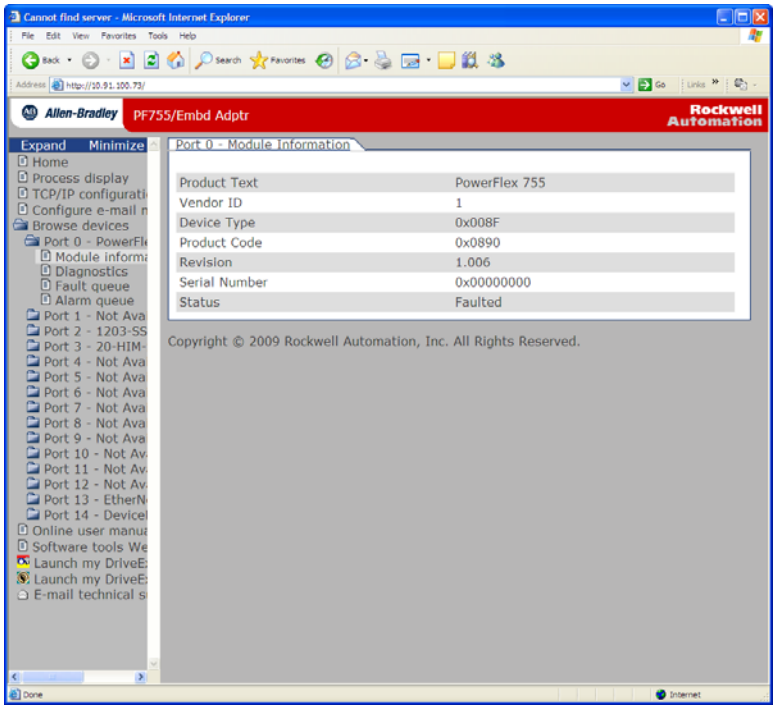
DPI Device Information Pages

DPI device information pages consist of these five pages:

- Module information page
- Diagnostic items page
- Fault queue page
- Alarm queue page
- Event queue page

Figure 8.6 shows an example module information page for the Port 00 device (host drive). Figure 8.7, Figure 8.8, and Figure 8.9 respectively show example diagnostic items, fault queue, and alarm queue pages for this device. Figure 8.10 shows an example event queue page for the Port 13 device (the embedded EtherNet/IP adapter).

Figure 8.6 Example of Port 00 (PowerFlex 755 Drive) Module Information Page



Information	Description
Product Text	Text identifying the device
Vendor ID	1 = Allen-Bradley
Device Type	0x008F (143 decimal)
Product Code	Code for the product name and its rating
Revision	Firmware revision used by the device
Serial Number	Serial number of the device
Status	Operating status of the device (for example, faulted)

Figure 8.7 Example of Port 00 (PowerFlex 755 Drive) Diagnostic Items Page

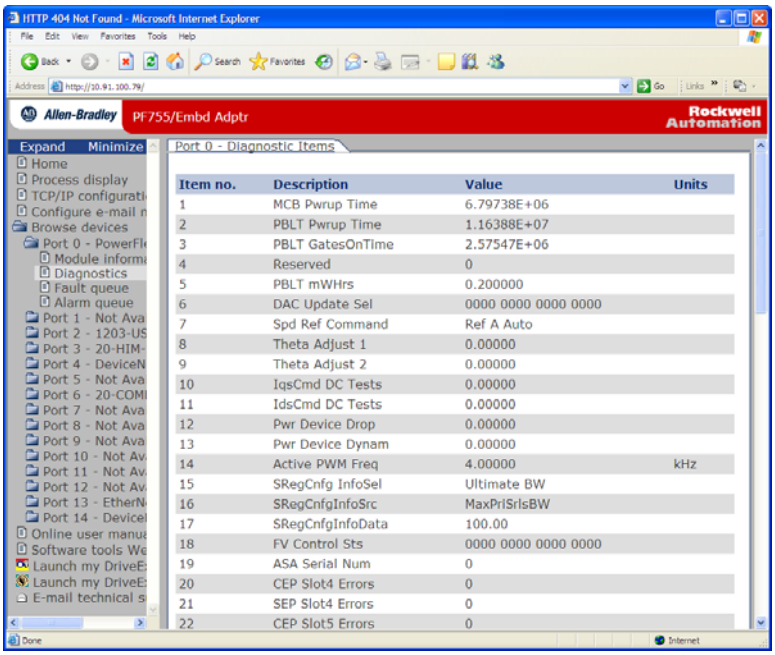


Figure 8.8 Example of Port 00 (PowerFlex 755 Drive) Fault Queue Page

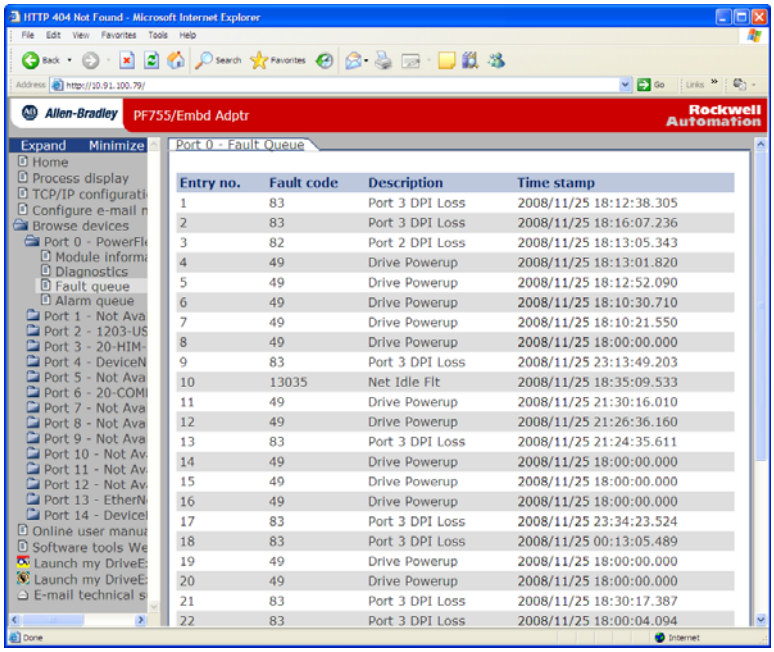


Figure 8.9 Example of Port 00 (PowerFlex 755 Drive) Alarm Queue Page

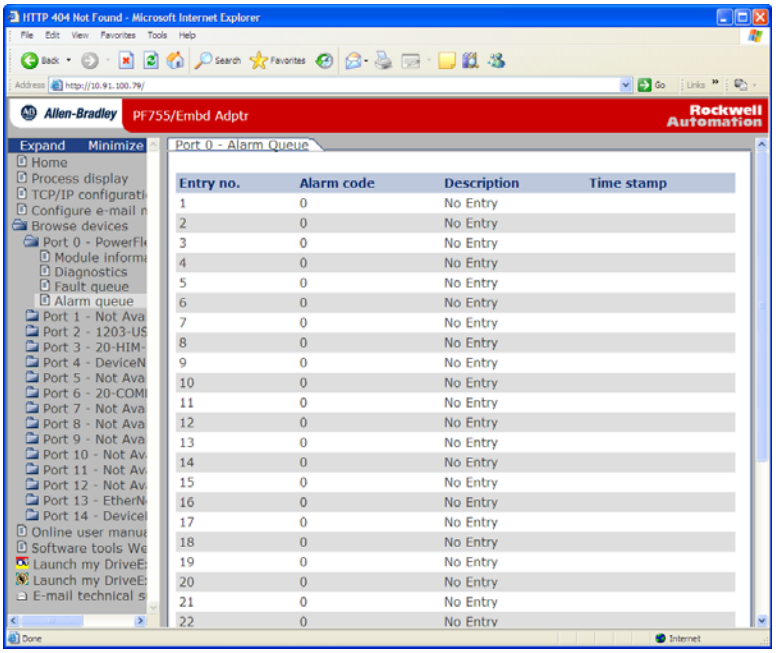
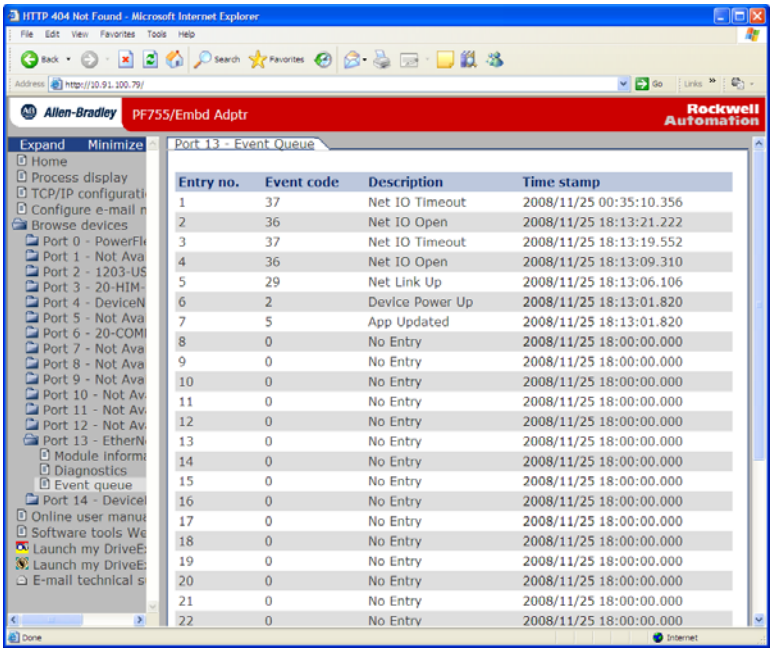


Figure 8.10 shows an example event queue page for the Port 13 device (embedded EtherNet/IP adapter).

Figure 8.10 Example of Port 13 (Embedded EtherNet/IP Adapter) Event Queue Page



Notes:

Specifications

Appendix A presents the specifications for the adapter.

Communications

Network	
Protocol	EtherNet/IP
Data Rates	10Mbps Full Duplex, 10Mbps Half Duplex, 100Mbps Full Duplex, or 100Mbps Half Duplex
Connection Limits	30 TCP connections 16 simultaneous CIP connections including 1 exclusive-owner I/O connection
Requested Packet Interval (RPI)	2 ms minimum
Packet Rate	Up to 400 total I/O packets per second (200 in and 200 out)
Drive	
Protocol	Embedded

Regulatory Compliance

UL	UL508C
cUL	CAN / CSA C22.2 No. 14-M91
CE	EN50178 and EN61800-3
CTick	EN61800-3

NOTE: This is a product of category C2 according to IEC 61800-3. In a domestic environment this product may cause radio interference in which case supplementary mitigation measures may be required.

Notes:

Adapter Parameters

Appendix B provides information about the adapter parameters.

About Parameter Numbers

The parameters in the adapter are numbered consecutively. However, depending on which configuration tool you use, they may have different numbers.

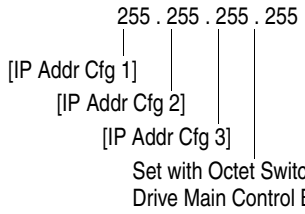
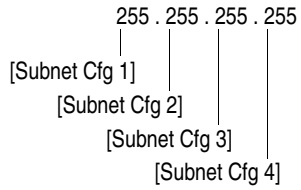
Configuration Tool	Numbering Scheme
• HIM • DriveExplorer • DriveExecutive	The adapter parameters begin with parameter 01. For example, **Parameter 01 - [DL From Net 01]** is parameter 01 as indicated by this manual.
• Explicit Messaging	Refer to Chapter 6, Using Explicit Messaging and Appendix C, EtherNet/IP Objects for details.

How Parameters Are Organized

The embedded EtherNet/IP adapter parameters are displayed in a **Numbered List** view order.

Parameter List

Parameter No.	Name and Description	Details
01	**[DL From Net 01]**	Default: 0
02	**[DL From Net 02]**	Default: 0
03	**[DL From Net 03]**	Default: 0
04	**[DL From Net 04]**	Default: 0
05	**[DL From Net 05]**	Default: 0
06	**[DL From Net 06]**	Default: 0
07	**[DL From Net 07]**	Default: 0
08	**[DL From Net 08]**	Default: 0
09	**[DL From Net 09]**	Default: 0
10	**[DL From Net 10]**	Default: 0
11	**[DL From Net 11]**	Default: 0
12	**[DL From Net 12]**	Default: 0
13	**[DL From Net 13]**	Default: 0
14	**[DL From Net 14]**	Default: 0
15	**[DL From Net 15]**	Default: 0
16	**[DL From Net 16]** Sets the port number and parameter number to which the selected Datalinks should connect. Each selected port/parameter will be written with data received from the network. These are parameters written by the controller (outputs from the controller). If setting the value manually, the parameter value = (10000 * port number) + (destination parameter number). For example, suppose you want to use **Parameter 01 - [DL From Net 01]** to write to Parameter 01 of an optional encoder card plugged into drive Port 5. The value for **Parameter 01 - [DL From Net 01]** would be 50001 [(10000 * 5) + 1].	Default: 0 Minimum: 0 Maximum: 159999 Type: Read/Write Reset Required: No
17	**[DL To Net 01]**	Default: 0
18	**[DL To Net 02]**	Default: 0
19	**[DL To Net 03]**	Default: 0
20	**[DL To Net 04]**	Default: 0
21	**[DL To Net 05]**	Default: 0
22	**[DL To Net 06]**	Default: 0
23	**[DL To Net 07]**	Default: 0
24	**[DL To Net 08]**	Default: 0
25	**[DL To Net 09]**	Default: 0
26	**[DL To Net 10]**	Default: 0
27	**[DL To Net 11]**	Default: 0
28	**[DL To Net 12]**	Default: 0
29	**[DL To Net 13]**	Default: 0
30	**[DL To Net 14]**	Default: 0
31	**[DL To Net 15]**	Default: 0
32	**[DL To Net 16]** Sets the port number and parameter number to which the selected Datalinks should connect. Each selected port/parameter will be read and their values transmitted over the network to the controller. These are parameters read by the controller (inputs to the controller). If setting the value manually, the parameter value = (10000 * port number) + (origination parameter number). For example, suppose you want to use **Parameter 17 - [DL To Net 01]** to read Parameter 01 of an optional I/O card plugged into drive Port 4. The value for **Parameter 17 - [DL To Net 01]** would be 40001 [(10000 * 4) + 1].	Default: 0 Minimum: 0 Maximum: 159999 Type: Read/Write Reset Required: No

Parameter No.	Name and Description	Details
33	**[Port Number]** Displays the drive port to which the embedded EtherNet/IP adapter is dedicated. This is always Port 13.	Default: 13 Value: 13 Type: Read Only
34	**[DLs From Net Act]** Displays the number of actual controller-to-drive Datalinks that the drive is using based on the I/O connection opened by the controller.	Default: 0 Minimum: 0 Maximum: 16 Type: Read Only
35	**[DLs To Net Act]** Displays the number of actual drive-to-controller Datalinks that the controller is using based on the I/O connection opened by the controller.	Default: 0 Minimum: 0 Maximum: 16 Type: Read Only
36	**[BOOTP]** Configures the adapter to use BOOTP so that you can set its IP address, subnet mask, and gateway address with a BOOTP server. When this parameter is disabled, you must use the adapter parameters to set these addressing functions.	Default: 1 = Enabled Values: 0 = Disabled 1 = Enabled Type: Read/Write Reset Required: Yes
37	**[Net Addr Src]** Displays the source from which the adapter node address is taken. This will be switches, **Parameters 40-43 [IP Addr Cfg x]**, or BOOTP. It is determined by the settings of the octet switches on the adapter. See Setting the IP Address Switches on page 2-2 for details.	Default: 0 = Switches Values: 0 = Switches 1 = Parameters 2 = BOOTP Type: Read Only
38 39 40 41	**[IP Addr Cfg 1]** **[IP Addr Cfg 2]** **[IP Addr Cfg 3]** **[IP Addr Cfg 4]** Sets the bytes in the IP address.  **Important:** To set the IP address using these parameters, **Parameter 36 - [BOOTP]** must be set to “0” (Disabled).	Default: 0 Default: 0 Default: 0 Default: 0 Minimum: 0 Maximum: 255 Type: Read/Write Reset Required: Yes
42 43 44 45	**[Subnet Cfg 1]** **[Subnet Cfg 2]** **[Subnet Cfg 3]** **[Subnet Cfg 4]** Sets the bytes of the subnet mask.  **Important:** To set the subnet mask using these parameters, **Parameter 36 - [BOOTP]** must be set to “0” (Disabled).	Default: 0 Default: 0 Default: 0 Default: 0 Minimum: 0 Maximum: 255 Type: Read/Write Reset Required: Yes

Parameter No.	Name and Description	Details
46 47 48 49	**[Gateway Cfg 1]** **[Gateway Cfg 2]** **[Gateway Cfg 3]** **[Gateway Cfg 4]** Sets the bytes of the gateway address. 255 . 255 . 255 . 255 [Gateway Cfg 1] [Gateway Cfg 2] [Gateway Cfg 3] [Gateway Cfg 4] **Important:** To set the gateway address using these parameters, **Parameter 36 - [BOOTP]** must be set to "0" (Disabled).	Default: 0 Default: 0 Default: 0 Default: 0 Minimum: 0 Maximum: 255 Type: Read/Write Reset Required: Yes
50	**[Net Rate Cfg]** Sets the network data rate at which the adapter communicates. (Updates **Parameter 51 - [Net Rate Act]** after a reset.)	Default: 0 = Autodetect Values: 0 = Autodetect 1 = 10Mbps Full 2 = 10Mbps Half 3 = 100Mbps Full 4 = 100Mbps Half Type: Read/Write Reset Required: Yes
51	**[Net Rate Act]** Displays the actual network data rate used by the adapter.	Default: 0 = No Link Values: 0 = No Link 1 = 10Mbps Full 2 = 10Mbps Half 3 = 100Mbps Full 4 = 100Mbps Half 5 = Dup IP Addr Type: Read Only
52	**[Web Enable]** Enables/disables the adapter web page features.	Default: 0 = Disabled Values: 0 = Disabled 1 = Enabled Type: Read/Write Reset Required: Yes
53	**[Web Features]** Enables/disables the Web-configurable e-mail notification feature.	Default: xxxx … xxxx xx01 Bit Values: 0 = Disabled 1 = Enabled Type: Read/Write Reset Required: No

Bit Definition	Not Used	Not Used	Not Used	Not Used	…	Not Used	Not Used	Not Used	Not Used	Not Used	Not Used	Not Used	E-mail Cfg
Default	x	x	x	x	…	x	x	x	x	x	x	x	1
Bit	15	14	13	12	…	7	6	5	4	3	2	1	0

Parameter No.	Name and Description	Details
54	**[Comm Flt Action]** Sets the action that the adapter and drive will take if the adapter detects that I/O communications have been disrupted. This setting is effective only if I/O that controls the drive is transmitted through the adapter.	Default: 0 = Fault Values: 0 = Fault 1 = Stop 2 = Zero Data 3 = Hold Last 4 = Send Flt Cfg Type: Read/Write Reset Required: No
	ATTENTION: Risk of injury or equipment damage exists. **Parameter 54 - [Comm Flt Action]** lets you determine the action of the adapter and connected drive if I/O communications are disrupted. By default, this parameter faults the drive. You can set this parameter so that the drive continues to run. Precautions should be taken to ensure that the setting of this parameter does not create a risk of injury or equipment damage. When commissioning the drive, verify that your system responds correctly to various situations (for example, a disconnected cable).	
55	**[Idle Flt Action]** Sets the action that the adapter and drive will take if the adapter detects that the controller is in program mode or faulted. This setting is effective only if I/O that controls the drive is transmitted through the adapter.	Default: 0 = Fault Values: 0 = Fault 1 = Stop 2 = Zero Data 3 = Hold Last 4 = Send Flt Cfg Type: Read/Write Reset Required: No
	ATTENTION: Risk of injury or equipment damage exists. **Parameter 55 - [Idle Flt Action]** lets you determine the action of the adapter and connected drive when the controller is idle. By default, this parameter faults the drive. You can set this parameter so that the drive continues to run. Precautions should be taken to ensure that the setting of this parameter does not create a risk of injury or equipment damage. When commissioning the drive, verify that your system responds correctly to various situations (for example, a controller in idle state).	
56	**[Peer Flt Action]** Sets the action that the adapter and drive will take if the adapter detects that Peer I/O communications have been disrupted. This setting is effective only if I/O is transmitted through the adapter.	Default: 0 = Fault Values: 0 = Fault 1 = Stop 2 = Zero Data 3 = Hold Last 4 = Send Flt Cfg Type: Read/Write Reset Required: No
	ATTENTION: Risk of injury or equipment damage exists. **Parameter 56 - [Peer Flt Action]** lets you determine the action of the adapter and connected drive if the adapter is unable to communicate with the designated peer. By default, this parameter faults the drive. You can set this parameter so that the drive continues to run. Precautions should be taken to ensure that the setting of this parameter does not create a risk of injury or equipment damage. When commissioning the drive, verify that your system responds correctly to various situations (for example, a disconnected cable).	

Parameter No.	Name and Description	Details	
57	**[Msg Flt Action]** Sets the action that the adapter and drive will take if the adapter detects that explicit messaging, only when used for drive control via PCCC and the CIP Register Object, has been disrupted.	Default: Values: Type: Reset Required:	0 = Fault 0 = Fault 1 = Stop 2 = Zero Data 3 = Hold Last 4 = Send Flt Cfg Read/Write No
	ATTENTION: Risk of injury or equipment damage exists. **Parameter 57 - [Msg Flt Action]** lets you determine the action of the adapter and connected drive if explicit messaging for drive control is disrupted. By default, this parameter faults the drive. You can set this parameter so that the drive continues to run. Precautions should be taken to ensure that the setting of this parameter does not create a risk of injury or equipment damage. When commissioning the drive, verify that your system responds correctly to various situations (for example, a disconnected cable).		
58	**[Flt Cfg Logic]** Sets the Logic Command data that is sent to the drive if any of the following is true: • **Parameter 54 - [Comm Flt Action]** is set to "4" (Send Flt Cfg) and I/O communications are disrupted. • **Parameter 55 - [Idle Flt Action]** is set to "4" (Send Flt Cfg) and the controller is idle. • **Parameter 56 - [Peer Flt Action]** is set to "4" (Send Flt Cfg) and Peer I/O communications are disrupted. • **Parameter 57 - [Msg Flt Action]** is set to "4" (Send Flt Cfg) and explicit messaging for drive control is disrupted. The bit definitions in the Logic Command word for PowerFlex 750-Series drives are shown in Appendix D.	Default: Minimum: Maximum: Type: Reset Required:	0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 1111 1111 1111 1111 1111 1111 1111 1111 Read/Write No
59	**[Flt Cfg Ref]** Sets the Reference data that is sent to the drive if any of the following is true: • **Parameter 54 - [Comm Flt Action]** is set to "4" (Send Flt Cfg) and I/O communications are disrupted. • **Parameter 55 - [Idle Flt Action]** is set to "4" (Send Flt Cfg) and the controller is idle. • **Parameter 56- [Peer Flt Action]** is set to "4" (Send Flt Cfg) and Peer I/O communications are disrupted. • **Parameter 57 - [Msg Flt Action]** is set to "4" (Send Flt Cfg) and explicit messaging for drive control is disrupted.	Default: Minimum: Maximum: Type: Reset Required:	0 -3.40282×10^{38} 3.40282×10^{38} Read/Write No

Parameter No.	Name and Description	Details	
60	**[Flt Cfg DL 01]**	Default:	0
61	**[Flt Cfg DL 02]**	Default:	0
62	**[Flt Cfg DL 03]**	Default:	0
63	**[Flt Cfg DL 04]**	Default:	0
64	**[Flt Cfg DL 05]**	Default:	0
65	**[Flt Cfg DL 06]**	Default:	0
66	**[Flt Cfg DL 07]**	Default:	0
67	**[Flt Cfg DL 08]**	Default:	0
68	**[Flt Cfg DL 09]**	Default:	0
69	**[Flt Cfg DL 10]**	Default:	0
70	**[Flt Cfg DL 11]**	Default:	0
71	**[Flt Cfg DL 12]**	Default:	0
72	**[Flt Cfg DL 13]**	Default:	0
73	**[Flt Cfg DL 14]**	Default:	0
74	**[Flt Cfg DL 15]**	Default:	0
75	**[Flt Cfg DL 16]** Sets the data that is sent to the Datalink in the drive if any of the following is true: • **Parameter 54 - [Comm Flt Action]** is set to "4" (Send Flt Cfg) and I/O communications are disrupted. • **Parameter 55 - [Idle Flt Action]** is set to "4" (Send Flt Cfg) and the controller is idle. • **Parameter 56 - [Peer Flt Action]** is set to "4" (Send Flt Cfg) and Peer I/O communications are disrupted. • **Parameter 57 - [Msg Flt Action]** is set to "4" (Send Flt Cfg) and explicit messaging for drive control is disrupted.	Default: Minimum: Maximum: Type: Reset Required:	0 0 4294967295 Read/Write No
76	**[DLs Fr Peer Cfg]** Sets the number of network-to-drive Datalinks (parameters) that are used for peer I/O. The Datalinks being used are allocated from the end of the list. For example, if this parameter's value is set to "3," Datalinks 14-16 are allocated for the 3 selected Datalinks. The Datalinks allocated for peer I/O cannot overlap with other assigned DL From Net 01-16 parameters.	Default: Minimum: Maximum: Type: Reset Required:	0 0 16 Read/Write Yes
77	**[DLs Fr Peer Act]** Displays the value of **Parameter 76 - [DLs Fr Peer Cfg]** at the time the drive was reset. This is the number of actual peer-to-drive Datalinks that the drive is expecting.	Default: Minimum: Maximum: Type:	0 0 16 Read Only
78	**[Logic Src Cfg]** Controls which of the peer-to-drive Datalinks contain the Logic Command for the drive.	Default: Minimum: Maximum: Type: Reset Required:	0 0 16 Read/Write No
79	**[Ref Src Cfg]** Controls which of the peer-to-drive Datalinks contain the Reference for the drive.	Default: Minimum: Maximum: Type: Reset Required:	0 0 16 Read/Write No

Parameter No.	Name and Description	Details	
80	**[Fr Peer Timeout]** Sets the timeout for a Peer I/O connection. If the time is reached without the adapter receiving (consuming) a message, the adapter will respond with the action specified in **Parameter 56 - [Peer Flt Action]**. In an adapter receiving (consuming) Peer I/O, the value of this parameter must be greater than the product of the value of **Parameter 89 - [To Peer Period]** in the adapter transmitting (producing) Peer I/O multiplied by the value of **Parameter 90 - [To Peer Skip]** in the adapter transmitting (producing) Peer I/O.	Default: Minimum: Maximum: Type: Reset Required:	10.00 Seconds 0.01 Seconds 10.00 Seconds Read/Write No
81 82 83 84	**[Fr Peer Addr 1]** **[Fr Peer Addr 2]** **[Fr Peer Addr 3]** **[Fr Peer Addr 4]** Sets the bytes in the IP address that specifies the device from which the adapter receives (consumes) Peer I/O data. 255 . 255 . 255 . 255 [Peer Inp Addr 1] [Peer Inp Addr 2] [Peer Inp Addr 3] [Peer Inp Addr 4] **Important:** The Peer Inp Addr must be on the same subnet as the embedded EtherNet/IP adapter. Refer to IP Addresses on page G-4 for more information. Changes to these parameters are ignored when **Parameter 85 - [Fr Peer Enable]** is “1” (On).	Default: Default: Default: Default: Minimum: Maximum: Type: Reset Required:	0 0 0 0 0 255 Read/Write No
85	**[Fr Peer Enable]** Controls whether Peer I/O input is operating. A value of “0” (Off) turns off Peer I/O input. A value of “1” (Cmd/Ref) overrides the settings in **Parameters 76 - [DLs Fr Peer Cfg], 78 - [Logic Src Cfg],** and **79 - [Ref Src Cfg]** and automatically uses peer Datalink 01 as the drive's present Logic Command and peer Datalink 02 as the drive's Reference. A value of “2” (Custom) enables peer I/O input using the Datalink count and settings provided by the user.	Default: Values: Type: Reset Required:	0 = Off 0 = Off 1 = Cmd/Ref 2 = Custom Read/Write No
86	**[Fr Peer Status]** Displays the status of the consumed Peer I/O input connection.	Default: Values: Type:	0 = Off 0 = Off 1 = Waiting 2 = Running 3 = Faulted Read Only

Parameter No.	Name and Description	Details	
87	**[DLs To Peer Cfg]** Sets the number of drive-to-network Datalinks (parameters) that are used for Peer I/O. The Datalinks being used are allocated from the end of the list. For example, if this parameter's value is set to "3", Datalinks 14-16 are allocated for the 3 selected Datalinks. The Datalinks allocated for this cannot overlap with other assigned DL To Net 01-16 parameters.	Default: Minimum: Maximum: Type: Reset Required:	0 0 16 Read/Write Yes
88	**[DLs To Peer Act]** Displays the value of **Parameter 87 - [DLs To Peer Cfg]** at the time the drive was reset. This is the number of actual drive-to-peer Datalinks that the drive is expecting.	Default: Minimum: Maximum: Type:	0 0 16 Read Only
89	**[To Peer Period]** Sets the minimum time that an adapter will wait when transmitting data to a peer. **Important:** Changes to this parameter are ignored when **Parameter 91 - [To Peer Enable]** is "0" (Off).	Default: Minimum: Maximum: Type: Reset Required:	10.00 Seconds 0.01 Seconds 10.00 Seconds Read/Write No
90	**[To Peer Skip]** Sets the maximum time that an adapter will wait when transmitting data to a peer. The value of **Parameter 89 - [To Peer Period]** is multiplied by the value of this parameter to set the time. **Important:** Changes to this parameter are ignored when **Parameter 91 - [To Peer Enable]** is "0" (Off).	Default: Minimum: Maximum: Type: Reset Required:	1 1 16 Read/Write No
91	**[To Peer Enable]** Controls whether Peer I/O output is operating. A value of "0" (Off) turns off Peer I/O output. A value of "1" (Cmd/Ref) overrides the settings in **Parameters 31 - [DL To Net 15], 32 - [DL To Net 16], 76 - [DLs Fr Peer Cfg], and 77 - [DLs Fr Peer Act]**, and automatically sends the drive's present Logic Command (as Datalink 01) and Reference (as Datalink 02). A value of "2" (Custom) enables Peer I/O output using the Datalink count and settings provided by the user.	Default: Values: Type: Reset Required:	0 = Off 0 = Off 1 = Cmd/Ref 2 = Custom Read/Write No

Notes:

EtherNet/IP Objects

Appendix C presents information about the EtherNet/IP objects that can be accessed using Explicit Messages. For information on the format of Explicit Messages and example ladder logic programs, refer to Chapter 6, Using Explicit Messaging.

TIP: Refer to the EtherNet/IP specification for more information about EtherNet/IP objects. Information about the EtherNet/IP specification is available on the ODVA web site (http://www.odva.org).

Supported Data Types

Data Type	Description
BOOL	8-bit value -- low bit is true or false
BOOL[x]	Array of n bits
CONTAINER	32-bit parameter value - sign extended if necessary
DINT	32-bit signed integer
INT	16-bit signed integer
LWORD	64-bit unsigned integer
REAL	32-bit floating point
SHORT_STRING	Struct of: USINT length indicator (L); USINT[L] characters
SINT	8-bit signed integer
STRINGN	Struct of: UINT character length indicator (W); UINT length indicator (L); USINT[W x L] string data
STRING[x]	Array of n characters
STRUCT	Structure name only - no size in addition to elements
TCHAR	8 or 16-bit character
UDINT	32-bit unsigned integer
UINT	16-bit unsigned integer
USINT	8-bit unsigned integer

Identity Object

Class Code

Hexadecimal	Decimal
0x01	1

Services

Service Code	Implemented for:		Service Name
	Class	Instance	
0x05	No	Yes	Reset
0x0E	Yes	Yes	Get_Attribute_Single
0x01	Yes	Yes	Get_Attributes_All

Instances

The number of instances depends on the number of components in the device connected to the adapter. This number of components can be read in Instance 0, Attribute 2.

Instance	Description
0	Class
1	Host
2 - 15	Peripherals on Ports 1 - 14

Class Attributes

Attribute ID	Access Rule	Name	Data Type	Description
2	Get	Max Instance	UINT	Total number of instances

Instance Attributes

Attribute ID	Access Rule	Name	Data Type	Description
1	Get	Vendor ID	UINT	1 = Allen-Bradley
2	Get	Device Type	UINT	123
3	Get	Product Code	UINT	Number identifying product name and rating
4	Get	Revision: Major Minor	STRUCT of: USINT USINT	 Value varies Value varies
5	Get	Status	UINT	Bit 0 = Owned Bit 8 = Minor recoverable fault Bit 10 = Major recoverable fault
6	Get	Serial Number	UDINT	Unique 32-bit number
7	Get	Product Name	SHORT_STRING	Product name and rating

Assembly Object

Class Code

Hexadecimal	Decimal
0x04	4

Services

Service Code	Implemented for:		Service Name
	Class	Instance	
0x0E	Yes	Yes	Get_Attribute_Single
0x10	Yes	Yes	Set_Attribute_Single

Instances

Instance	Description
1	All I/O data being read from the DPI device (read-only)
2	All I/O data written to the DPI device (read/write)

Class Attributes

Attribute ID	Access Rule	Name	Data Type	Description
1	Get	Revision	UINT	2
2	Get	Max Instance	UINT	2
100	Set	Control Timeout	UINT	Control timeout in seconds

Instance Attributes

Attribute ID	Access Rule	Name	Data Type	Description
1	Get	Number of Members	UINT	1
2	Get	Member List	ARRAY of STRUCT: UINT UINT Packed EPATH	 Size of member data Size of member path Member path
3	Conditional (1)	Data	Array of Bits	Data to be transferred
4	Get	Size	UINT	Size of assembly data in bits

(1) For instance 1, access rule for the data attribute is Get. For instance 2, it is Get/Set.

Important: Setting an assembly object attribute can be done only when the Control Timeout (class attribute 100) has been set to a non-zero value.

Register Object

Class Code

Hexadecimal	Decimal
0x07	7

Services

	Implemented for:		
Service Code	Class	Instance	Service Name
0x0E	Yes	Yes	Get_Attribute_Single
0x10	Yes	Yes	Set_Attribute_Single

Instances

Instance	Description
1	All I/O data being read from the embedded adapter (read-only)
2	All I/O data written to the embedded adapter (read/write)
3	Logic Status and Feedback data (read-only)
4	Logic Command and Reference data (read/write)
5	DL To Net 01 (input data from embedded adapter to scanner) (read only)
6	DL From Net 01 (output data from scanner to embedded adapter) (read/write)
⋮	⋮
35	DL To Net 16 (output data from scanner to embedded adapter) (read/write)
36	DL From Net 16 (input data from embedded adapter to scanner) (read only)
37	Logic Status and Feedback data (read-only)
38	Masked Logic Command [1] (read/write)
39	Logic Status data (read-only)
40	Logic Command data (read/write)
41	Feedback data (read-only)
42	Reference data (read/write)

[1] The mask command DWORD is set to the value of the first DWORD of the data where there are ones in the second DWORD of the data. Only the bits of the Logic Command that have the corresponding mask bit set are applied.

Class Attributes

Attribute ID	Access Rule	Description
1	Read	Revision
2	Read	Maximum Instance
3	Read	Number of Instances
100	Read/Write	Timeout

Instance Attributes

Attribute ID	Access Rule	Name	Data Type	Description
1	Get	Bad Flag	BOOL	If set to 1, then attribute 4 may contain invalid data. 0 = good 1 = bad
2	Get	Direction	BOOL	Direction of data transfer 0 = Producer Register (drive to network) 1 = Consumer Register (network to drive)
3	Get	Size	UINT	Size of register data in bits
4	Conditional [1]	Data	ARRAY of BITS	Data to be transferred

[1] The access rule of Set is optional if attribute 2, Direction = 1. If Direction = 0, the access rule is Get.

PCCC Object

Class Code

Hexadecimal	Decimal
0x67	103

Services

	Implemented for:		
Service Code	Class	Instance	Service Name
0x4B	No	Yes	Execute_PCCC
0x4C	No	Yes	Execute_DH+

Instances

Supports Instance 1.

Class Attributes

Not supported.

Instance Attributes

Not supported.

Message Structure for Execute_PCCC

Request		
Name	Data Type	Description
Length	USINT	Length of requestor ID
Vendor	UINT	Vendor number of requestor
Serial Number	UDINT	ASA serial number of requestor
Other	Product Specific	Identifier of user, task, etc. on the requestor
CMD	USINT	Command byte
STS	USINT	0
TNSW	UINT	Transport word
FNC	USINT	Function code. Not used for all CMDs.
PCCC_ params	ARRAY of USINT	CMD/FNC specific parameters

Response		
Name	Data Type	Description
Length	USINT	Length of requestor ID
Vendor	UINT	Vendor number of requestor
Serial Number	UDINT	ASA serial number of requestor
Other	Product Specific	Identifier of user, task, etc. on the requestor
CMD	USINT	Command byte
STS	USINT	Status byte
TNSW	UINT	Transport word. Same value as the request.
EXT_STS	USINT	Extended status. Not used for all CMDs.
PCCC_ results	ARRAY of USINT	CMD/FNC specific result data

Message Structure for Execute_DH+

Request		
Name	**Data Type**	**Description**
DLink	UINT	Destination Link ID
DSta	USINT	Destination Station number
DUser	USINT	Destination “User” number
SLink	UINT	Source Link ID
SSta	USINT	Source Station number
SUser	USINT	Source User number
CMD	USINT	Command byte
STS	USINT	0
TNSW	UINT	Transport word
FNC	USINT	Function code; not used for all CMDs
PCCC_ params	ARRAY of USINT	CMD/FNC specific parameters

Response		
Name	**Data Type**	**Description**
DLink	UINT	Destination Link ID
DSta	USINT	Destination Station number
DUser	USINT	Destination “User” number
SLink	UINT	Source Link ID
SSta	USINT	Source Station number
SUser	USINT	Source User number
CMD	USINT	Command byte
STS	USINT	Status byte
TNSW	UINT	Transport word. Same value as the request.
EXT_STS	USINT	Extended Status; not used for all CMDs
PCCC_ results	ARRAY of USINT	CMD/FNC specific result data

The embedded EtherNet/IP adapter supports the following PCCC command types:

CMD	FNC	Description
0x06	0x03	Identify host and some status
0F	67	PLC-5 typed write
0F	68	PLC-5 typed read
0F	95	Encapsulate other protocol
0F	A2	SLC 500 protected typed read with 3 address fields
0F	AA	SLC 500 protected typed write with 3 address fields
0F	A1	SLC 500 protected typed read with 2 address fields
0F	A9	SLC 500 protected typed write with 2 address fields
0F	00	Word range read
0F	01	Word range write

For more information regarding PCCC commands, see *DF1 Protocol and Command Set Manual (Allen-Bradley publication 1770-6.5.16).*

N-Files

N-File	Description	
N42	This N-file lets you read and write some values configuring the port.	
N42:3	Time-out (read/write): Time (in seconds) allowed between messages to the N45 file. If the adapter does not receive a message in the specified time, it performs the fault action configured in its [Comm Flt Action] parameter. A valid setting is between 1 and 32767 seconds.	
N42:7	Adapter Port Number (read only): Drive Port 13 in which the adapter resides.	
N42:8	Peer Adapters (read only): Bit field of devices with peer messaging capabilities.	
N45	This N-file lets you read and write control I/O messages. You can write control I/O messages only when all of the following conditions are true: • The adapter is not receiving I/O from a scanner. For example, there is no scanner on the network, the scanner is in idle (program) mode, the scanner is faulted, or the adapter is not mapped to the scanner. • The adapter is not receiving Peer I/O from another adapter. • The value of N42:3 is set to a non-zero value.	
	Write	*Read*
N45:0	Logic Command (least significant)	Logic Status (least significant)
N45:1	Logic Command (most significant)	Logic Status (most significant)
N45:2	Reference (least significant)	Feedback (least significant)
N45:3	Reference (most significant)	Feedback (most significant)
N45:4	DL From Net 01 (least significant)	DL To Net 01 (least significant)
N45:5	DL From Net 01 (most significant)	DL To Net 01 (most significant)
N45:6	DL From Net 02 (least significant)	DL To Net 02 (least significant)
N45:7	DL From Net 02 (most significant)	DL To Net 02 (most significant)
N45:8	DL From Net 03 (least significant)	DL To Net 03 (least significant)
N45:9	DL From Net 03 (most significant)	DL To Net 03 (most significant)
N45:10	DL From Net 04 (least significant)	DL To Net 04 (least significant)
N45:11	DL From Net 04 (most significant)	DL To Net 04 (most significant)
N45:12	DL From Net 05 (least significant)	DL To Net 05 (least significant)
N45:13	DL From Net 05 (most significant)	DL To Net 05 (most significant)
N45:14	DL From Net 06 (least significant)	DL To Net 06 (least significant)
N45:15	DL From Net 06 (most significant)	DL To Net 06 (most significant)
N45:16	DL From Net 07 (least significant)	DL To Net 07 (least significant)
N45:17	DL From Net 07 (most significant)	DL To Net 07 (most significant)
N45:18	DL From Net 08 (least significant)	DL To Net 08 (least significant)
N45:19	DL From Net 08 (most significant)	DL To Net 08 (most significant)
N45:20	DL From Net 09 (least significant)	DL To Net 09 (least significant)
N45:21	DL From Net 09 (most significant)	DL To Net 09 (most significant)
N45:22	DL From Net 10 (least significant)	DL To Net 10 (least significant)
N45:23	DL From Net 10 (most significant)	DL To Net 10 (most significant)
N45:24	DL From Net 11 (least significant)	DL To Net 11 (least significant)
N45:25	DL From Net 11 (most significant)	DL To Net 11 (most significant)
N45:26	DL From Net 12 (least significant)	DL To Net 12 (least significant)
N45:27	DL From Net 12 (most significant)	DL To Net 12 (most significant)
N45:28	DL From Net 13 (least significant)	DL To Net 13 (least significant)
N45:29	DL From Net 13 (most significant)	DL To Net 13 (most significant)
N45:30	DL From Net 14 (least significant)	DL To Net 14 (least significant)
N45:31	DL From Net 14 (most significant)	DL To Net 14 (most significant)
N45:32	DL From Net 15 (least significant)	DL To Net 15 (least significant)
N45:33	DL From Net 15 (most significant)	DL To Net 15 (most significant)
N45:34	DL From Net 16 (least significant)	DL To Net 16 (least significant)
N45:35	DL From Net 16 (most significant)	DL To Net 16 (most significant)

N-File	Description
N150 – N199	These N-files let you read and write parameter values in the PowerFlex 755 drive as 32-bit double words. You can interpret the data in various ways (for example, 32-bit real, 32-bit integer) To read a parameter, you need to send a message with two elements. For example, to read parameter 1, read two elements beginning at N150:2. As another example, to read parameters 2 - 6, read ten elements beginning at N150:4.
N150:0 – 1	Number of parameters in the drive
N150:2 – 249	Drive parameters 1 – 124
N151:0 – 249	Drive parameters 125 – 249
N152:0 – 249	Drive parameters 250 – 374
N153:0 – 249	Drive parameters 375 – 499
⋮	⋮
N199:0 – 249	Drive parameters 6125 – 6249
N201 – N212	These N-files let you read and write parameter values in DPI Peripherals (for example, a HIM or adapter) as 32-bit double words. You can interpret the data in various ways (for example, 32-bit real, 32-bit integer) To read a parameter, you need to send a message with two elements. For example, to read parameter 1 in the peripheral connected to DPI port 1, read two elements beginning at N201:2. As another example, to read parameters 2 – 6 in the peripheral connected to DPI port 5 (the adapter), read ten elements beginning at N209:4.
N201:0 – 1	Number of parameters in the DPI peripheral at DPI port 1
N201:2 – 249	Parameters 1 – 124 in the DPI peripheral at DPI port 1
N202:0 – 249	Parameters 125 – 249 in the DPI peripheral at DPI port 1
N203:0 – 1	Number of parameters in the DPI peripheral at DPI port 2
N203:2 – 249	Parameters 1 – 124 in the DPI peripheral at DPI port 2
N204:0 – 249	Parameters 125 – 249 in the DPI peripheral at DPI port 2
N205:0 – 1	Number of parameters in the DPI peripheral at DPI port 3
N205:2 – 249	Parameters 1 – 124 in the DPI peripheral at DPI port 3
N206:0 – 249	Parameters 125 – 249 in the DPI peripheral at DPI port 3
N207:0 – 1	Number of parameters in the DPI peripheral at DPI port 4
N207:2 – 249	Parameters 1 – 124 in the DPI peripheral at DPI port 4
N208:0 – 249	Parameters 125 – 249 in the DPI peripheral at DPI port 4
N209:0 – 1	Number of parameters in the DPI peripheral at DPI port 5
N209:2 – 249	Parameters 1 – 124 in the DPI peripheral at DPI port 5
N210:0 – 249	Parameters 125 – 249 in the DPI peripheral at DPI port 5
N211:0 – 1	Number of parameters in the DPI peripheral at DPI port 6
N211:2 – 249	Parameters 1 – 124 in the DPI peripheral at DPI port 6
N212:0 – 249	Parameters 125 – 249 in the DPI peripheral at DPI port 6
N213:0 – 1	Number of parameters in the DPI peripheral at DPI port 7
N213:2 – 249	Parameters 1 – 124 in the DPI peripheral at DPI port 7
N214:0 – 249	Parameters 125 – 249 in the DPI peripheral at DPI port 7
N215:0 – 1	Number of parameters in the DPI peripheral at DPI port 8
N215:2 – 249	Parameters 1 – 124 in the DPI peripheral at DPI port 8
N216:0 – 249	Parameters 125 – 249 in the DPI peripheral at DPI port 8
N217:0 – 1	Number of parameters in the DPI peripheral at DPI port 9
N217:2 – 249	Parameters 1 – 124 in the DPI peripheral at DPI port 9
N218:0 – 249	Parameters 125 – 249 in the DPI peripheral at DPI port 9
N219:0 – 1	Number of parameters in the DPI peripheral at DPI port 10
N219:2 – 249	Parameters 1 – 124 in the DPI peripheral at DPI port 10
N220:0 – 249	Parameters 125 – 249 in the DPI peripheral at DPI port 10
N221:0 – 1	Number of parameters in the DPI peripheral at DPI port 11
N221:2 – 249	Parameters 1 – 124 in the DPI peripheral at DPI port 11
N222:0 – 249	Parameters 125 – 249 in the DPI peripheral at DPI port 11
N223:0 – 1	Number of parameters in the DPI peripheral at DPI port 12
N223:2 – 249	Parameters 1 – 124 in the DPI peripheral at DPI port 12
N224:0 – 249	Parameters 125 – 249 in the DPI peripheral at DPI port 12
N225:0 – 1	Number of parameters in the DPI peripheral at DPI port 13
N225:2 – 249	Parameters 1 – 124 in the DPI peripheral at DPI port 13
N226:0 – 249	Parameters 125 – 249 in the DPI peripheral at DPI port 13
N227:0 – 1	Number of parameters in the DPI peripheral at DPI port 14
N227:2 – 249	Parameters 1 – 124 in the DPI peripheral at DPI port 14
N228:0 – 249	Parameters 125 – 249 in the DPI peripheral at DPI port 14

DPI Device Object

Class Code

Hexadecimal	Decimal
0x92	146

Services

Service Code	Implemented for:		Service Name
	Class	Instance	
0x0E	Yes	Yes	Get_Attribute_Single
0x10	Yes	Yes	Set_Attribute_Single

Instances

The number of instances depends on the number of components in the device. The total number of components can be read in Instance 0, Class Attribute 4.

Instances (Hex.)	(Dec.)	Device
0x0000 – 0x3FFF	0 – 16383	Host Drive
0x4000 – 0x43FF	16384 – 17407	Adapter
0x4400 – 0x47FF	17408 – 18431	Port 1
0x4800 – 0x4BFF	18432 – 19455	Port 2
0x4C00 – 0x4FFF	19456 – 20479	Port 3
0x5000 – 0x53FF	20480 – 21503	Port 4
0x5400 – 0x57FF	21504 – 22527	Port 5
0x5800 – 0x5BFF	22528 – 23551	Port 6
0x5C00 – 0x5FFF	23552 – 24575	Port 7
0x6000 – 0x63FF	24576 – 25599	Port 8
0x6400 – 0x67FF	25600 – 26623	Port 9
0x6800 – 0x6BFF	26624 – 27647	Port 10
0x6C00 – 0x6FFF	27648 – 28671	Port 11
0x7000 – 0x73FF	28672 – 29695	Port 12
0x7400 – 0x77FF	29696 – 30719	Port 13
0x7800 – 0x7BFF	30720 – 31743	Port 14

Example	Description
0	Class Attributes (Drive)
1	Drive Component 1
2	Drive Component 2
⋮	⋮
16384	Class Attributes (Adapter)
16385	Adapter Component 1
⋮	⋮

Class Attributes

Attribute ID	Access Rule	Name	Data Type	Description
0	Get	Family Code	USINT	0x00 = DPI Peripheral 0x90 = PowerFlex 755 0xA0 = 20-750 Series Option Card 0xFF = HIM
1	Get	Family Text	STRING[16]	Text identifying the device.
2	Set	Language Code	USINT	0 = English 1 = French 2 = Spanish 3 = Italian 4 = German 5 = Japanese 6 = Portuguese 7 = Mandarin Chinese 8 = Russian 9 = Dutch 10 = Korean

Attribute ID	Access Rule	Name	Data Type	Description
3	Get	Product Series	USINT	1 = A 2 = B …
4	Get	Number of Components	USINT	Number of components (e.g., main control board, I/O boards) in the device.
5	Set	User Definable Text	STRING[16]	Text identifying the device with a user-supplied name.
6	Get	Status Text	STRING[12]	Text describing the status of the device.
7	Get	Configuration Code	USINT	Identification of variations.
8	Get	Configuration Text	STRING[16]	Text identifying a variation of a family device.
9	Get	Brand Code	UINT	0x0001 = Allen-Bradley
11	Get	NVS Checksum	UINT	Checksum of the Non-Volatile Storage in a device.
12	Get	Class Revision	UINT	2 = DPI
13	Get	Character Set Code	USINT	0 = SCANport HIM 1 = ISO 8859-1 (Latin 1) 2 = ISO 8859-2 (Latin 2) 3 = ISO 8859-3 (Latin 3) 4 = ISO 8859-4 (Latin 4) 5 = ISO 8859-5 (Cyrillic) 6 = ISO 8859-6 (Arabic) 7 = ISO 8859-7 (Greek) 8 = ISO 8859-8 (Hebrew) 9 = ISO 8859-9 (Turkish) 10 = ISO 8859-10 (Nordic) 255 = ISO 10646 (Unicode)
14	Get	Product Option Support	BOOL[64]	
15	Get	Languages Supported	STRUCT of: USINT USINT[n]	 Number of Languages Language Codes (see Class Attribute 2)
16	Get	Date of Manufacture	STRUCT of: UINT USINT USINT	 Year Month Day
17	Get	Product Revision	STRUCT of: USINT USINT	 Major Firmware Release Minor Firmware Release
18	Get	Serial Number	UDINT	Value between 0x00000000 and 0xFFFFFFFF
19	Set	Language Selected	USINT	0 = Default (HIM will prompt at start up) 1 = Language was selected (no prompt)
20	Set	Customer-Generated Firmware	STRING[36]	GUID (Globally Unique Identifier) identifying customer firmware flashed into the device.
30	Get	International Status Text	STRINGN	Text describing the status of device with support for Unicode.
31	Get/Set	International User Definable Text	STRINGN	Text identifying the device with a user-supplied name with support for Unicode.
34	Get	Key Information	STRUCT of: UDINT UDINT UINT UINT UINT USINT USINT USINT USINT USINT USINT	 Rating Code Device Serial Number Customization Code Customization Revision Brand Code Family Code Config Code Language Code Major Revision Minor Revision Customer-Generated Firmware UUID
35	Get	NVS CRC	UDINT	A 32-bit CRC of the Non-Volatile Storage in a device.
39	Get	SI Drive Code	UINT	Code identifying the protocol between the device and host.
128	Get	Customization Code	UINT	Code identifying the customized device.
129	Get	Customization Revision Number	UINT	Revision of the customized device.
130	Get	Customization Device Text	STRING[32]	Text identifying the customized device.

Instance Attributes

Attribute ID	Access Rule	Name	Data Type	Description
3	Get	Component Name	STRING[32]	Name of the component
4	Get	Component Firmware Revision	STRUCT of: USINT USINT	 Major Revision Minor Revision
8	Get	Component Serial Number	UDINT	Value between 0x00000000 and 0xFFFFFFFF
9	Get	International Component Name	STRINGN	Name of the component with support for Unicode.

DPI Parameter Object

Class Code

Hexadecimal	Decimal
0x93	147

To access "Host Config" parameters, use the Host DPI Parameter Object (Class Code 0x9F).

Instances

The number of instances depends on the number of parameters in the device. The total number of parameters can be read in Instance 0, Attribute 0.

Instances (Hex.)	(Dec.)	Device
0x0000 – 0x3FFF	0 – 16383	Host Drive
0x4000 – 0x43FF	16384 – 17407	Adapter
0x4400 – 0x47FF	17408 – 18431	Port 1
0x4800 – 0x4BFF	18432 – 19455	Port 2
0x4C00 – 0x4FFF	19456 – 20479	Port 3
0x5000 – 0x53FF	20480 – 21503	Port 4
0x5400 – 0x57FF	21504 – 22527	Port 5
0x5800 – 0x5BFF	22528 – 23551	Port 6
0x5C00 – 0x5FFF	23552 – 24575	Port 7
0x6000 – 0x63FF	24576 – 25599	Port 8
0x6400 – 0x67FF	25600 – 26623	Port 9
0x6800 – 0x6BFF	26624 – 27647	Port 10
0x6C00 – 0x6FFF	27648 – 28671	Port 11
0x7000 – 0x73FF	28672 – 29695	Port 12
0x7400 – 0x77FF	29696 – 30719	Port 13
0x7800 – 0x7BFF	30720 – 31743	Port 14

Example	Description
0	Class Attributes (Drive)
1	Drive Parameter 1 Attributes
2	Drive Parameter 2 Attributes
⋮	⋮
16384	Class Attributes (Adapter)
16385	Adapter Parameter 1 Attributes
⋮	⋮

Class Attributes

Attribute ID	Access Rule	Name	Data Type	Description
0	Get	Number of Instances	UINT	Number of parameters in the device
1	Set	Write Protect Password	UINT	0 = Password disabled n = Password value
2	Set	NVS Command Write	USINT	0 = No Operation 1 = Store values in active memory to NVS 2 = Load values in NVS to active memory 3 = Load default values to active memory 4 = Partial defaults 5 = System defaults
3	Get	NVS Parameter Value Checksum	UINT	Checksum of all parameter values in a user set in NVS
4	Get	NVS Link Value Checksum	UINT	Checksum of parameter links in a user set in NVS
5	Get	First Accessible Parameter	UINT	First parameter available if parameters are protected by passwords. A "0" indicates all parameters are protected.
7	Get	Class Revision	UINT	2 = DPI
8	Get	First Parameter Processing Error	UINT	The first parameter that has been written with a value outside of its range. A "0" indicates no errors.
9	Set	Link Command	USINT	0 = No Operation 1 = Clear All Parameter Links (This does not clear links to function blocks.)

Instance Attributes

Attribute ID	Access Rule	Name	Data Type	Description
6	Get	DPI Offline Read Full	STRUCT of: BOOL[32] CONTAINER CONTAINER CONTAINER STRING[16] STRING[4] UINT UINT UINT UINT UINT UINT UINT USINT USINT UINT UINT CONTAINER UINT UNIT UNIT INT	 Descriptor Offline Minimum value Offline Maximum value Offline Default value Parameter name Offline parameter units Online minimum parameter instance Online maximum parameter instance Online default parameter instance Multiplier parameter instance Divisor parameter instance Base parameter instance Offset parameter instance Formula number Pad byte (always zero) Help instance Pad word (always a value of zero) Parameter value Multiplier Divisor Base Offset
7	Get	DPI Online Read Full	STRUCT of: BOOL[32] CONTAINER(1) CONTAINER CONTAINER CONTAINER UINT UINT STRING[4] UINT UINT UINT INT USINT[3] USINT STRING[16]	 Descriptor (see page C-16) Parameter value Minimum value Maximum value Default value Next parameter Previous parameter Units (for example, Amps, Hz) Multiplier (2) Divisor (2) Base (2) Offset (2) Link (source of the value) (0 = no link) Always zero (0) Parameter name
8	Get	DPI Descriptor	BOOL[32]	Descriptor (see page C-16)
9	Get/Set	DPI Parameter Value	Various	Parameter value in NVS. (3)
10	Get/Set	DPI RAM Parameter Value	Various	Parameter value in temporary memory.
11	Get/Set	DPI Link	USINT[3]	Link (parameter or function block that is the source of the value) (0 = no link)
12	Get	Help Object Instance	UINT	ID for help text for this parameter
13	Get	DPI Read Basic	STRUCT of: BOOL[32] CONTAINER CONTAINER CONTAINER CONTAINER STRING[16] STRING[4]	 Descriptor (see page C-16) Parameter value Minimum value Maximum value Default value Parameter name Units (for example, Amps, Hz)
14	Get	DPI Parameter Name	STRING[16]	Parameter name
15	Get	DPI Parameter Alias	STRING[16]	Customer supplied parameter name.
16	Get	Parameter Processing Error	USINT	0 = No error 1 = Value is less than the minimum 2 = Value is greater than the maximum
18	Get	International DPI Offline Parameter Text	Struct of: STRINGN STRINGN	 International parameter name International offline units

Attribute ID	Access Rule	Name	Data Type	Description
19	Get	International DPI Online Parameter Text	Struct of: STRINGN STRINGN	 International parameter name International online units
20	Get	International DPI Online Read Full	Struct of: BOOL[32] CONTAINER CONTAINER CONTAINER CONTAINER UINT UINT UINT UINT UINT INT USINT[3] USINT BOOL[32] STRINGN STRINGN	 Descriptor Parameter value Online minimum value Online maximum value Online default value Next Previous Multiplier Divisor Base Offset Link Pad word (always zero) Extended descriptor International parameter name International online parameter units
21	Get	DPI Extended Descriptor	UDINT	Extended Descriptor (see page C-17)
22	Get	International DPI Offline Read Full	Struct of: BOOL CONTAINER CONTAINER CONTAINER UINT UINT UINT UINT UINT UINT USINT USINT UINT UINT CONTAINER UINT UINT UINT INT BOOL[32] STRINGN STRINGN	 Descriptor Offline minimum value Offline maximum value Offline default value Online minimum parameter instance Online maximum parameter instance Online default parameter instance Multiplier parameter instance Divisor parameter instance Base parameter instance Offset parameter instance Formula number Pad word (always zero) Help instance Pad word (always a value of zero) Parameter value Multiplier Divisor Base Offset Extended DPI descriptor International DPI parameter name International DPI offline parameter units

(1) A CONTAINER is a 32-bit block of data that contains the data type used by a parameter value. If signed, the value is sign extended. Padding is used in the CONTAINER to ensure that it is always 32-bits.

(2) This value is used in the formulas used to convert the parameter value between display units and internal units. Refer to Formulas for Converting on page C-18.

(3) Do NOT continually write parameter data to NVS. Refer to the attention on page 6-1.

Descriptor Attributes

Bit	Name	Description
0	Data Type (Bit 1)	Right bit is least significant bit (0). 000 = USINT used as an array of Boolean 001 = UINT used as an array of Boolean 010 = USINT (8-bit integer) 011 = UINT (16-bit integer) 100 = UDINT (32-bit integer) 101 = TCHAR ((8-bit (not Unicode) or 16-bits (Unicode)) 110 = REAL (32-bit floating point value) 111 = Use bits 16, 17, 18
1	Data Type (Bit 2)	
2	Data Type (Bit 3)	
3	Sign Type	0 = unsigned 1 = signed
4	Hidden	0 = visible 1 = hidden
5	Not a Link Sink	0 = May be the sink end of a link 1 = May not be the sink end of a link
6	Not Recallable	0 = Recallable from NVS 1 = Not Recallable from NVS
7	ENUM	0 = No ENUM text 1 = ENUM text
8	Writable	0 = Read only 1 = Read/write
9	Not Writable When Enabled	0 = Writable when enabled (e.g., drive running) 1 = Not writable when enabled
10	Instance	0 = Parameter value is not a Reference to another parameter 1 = Parameter value refers to another parameter
11	Uses Bit ENUM Mask	This parameter instance supports the Bit ENUM Mask attribute. For more information, see the definition of the attribute.
12	Decimal Place (Bit 0)	Number of digits to the right of the decimal point. 0000 = 0 1111 = 15
13	Decimal Place (Bit 1)	
14	Decimal Place (Bit 2)	
15	Decimal Place (Bit 3)	
16	Extended Data Type (Bit 4)	Bit 16 is the least significant bit. 000 = Reserved 001 = UDINT used as an array of Boolean 010 = Reserved 011 = Reserved 100 = Reserved 101 = Reserved 110 = Reserved 111 = Reserved
17	Extended Data Type (Bit 5)	
18	Extended Data Type (Bit 6)	
19	Parameter Exists	Used to mark parameters that are not available to network tools.
20	Not Used	Reserved
21	Formula Links	Indicates the Formula Data is derived from other parameters.
22	Access Level (Bit 1)	A 3-bit field used to control access to parameter data.
23	Access Level (Bit 2)	
24	Access Level (Bit 3)	
25	Writable ENUM	ENUM text: 0 = Read Only, 1 = Read/Write
26	Not a Link Source	0 = May be the source end of a link 1 = May not be the source end of a link
27	Enhanced Bit ENUM	Parameter supports enhanced bit ENUMs.
28	Enhanced ENUM	Parameter supports enhanced ENUMs.
29	Uses DPI Limits Object	Parameter uses the DPI Limits Object. • Intelligent offline tools make use of the Limits Object to select limits and units.
30	Extended Descriptor	Parameter uses Extended Descriptor bits, which can be obtained by reading the DPI Extended Descriptor attribute for this parameter.
31	Always Upload/Download	Parameter shall always be included in uploads and downloads.

Extended Descriptor Attributes

Bit	Name	Description
0	Indirect Mode	0 = Analog (selects entire parameters) 1 = Digital (selects individual bits within parameters)
1	Indirect Type 0	Analog input list (Instance 0xFFFF)
2	Indirect Type 1	Digital input list (Instance 0xFFFE)
3	Indirect Type 2	Feedback list (Instance 0xFFFD)
4	Indirect Type 3	Analog output list (Instance 0xFFFC)
5	Indirect Type 4	Digital output list (Instance 0xFFFB)
6	Indirect Type 5	Undefined (Instance 0xFFFA)
7	Indirect Type 6	Undefined (Instance 0xFFF9)
8	Indirect Type 7	Undefined (Instance 0xFFF8)
9	Indirect Type 8	Undefined (Instance 0xFFF7)
10	Indirect Type 9	Undefined (Instance 0xFFF6)
11	Indirect Type 10	Undefined (Instance 0xFFF5)
12	Indirect Type 11	Undefined (Instance 0xFFF4)
13	Indirect Type 12	Undefined (Instance 0xFFF3)
14	Indirect Type 13	Undefined (Instance 0xFFF2)
15	Indirect Type 14	Parameter-specific list
16	FP Max Decimals Bit 0	These four bits are used on REAL parameters only. They indicate the maximum number of decimal places to be displayed for small values. A value of 0 indicates to not limit the number of decimal places used.
17	FP Max Decimals Bit 1	
18	FP Max Decimals Bit 2	
19	FP Max Decimals Bit 1	
20	Extended Parameter Reference	0 = Not an Extended Parameter Reference 1 = Extended Parameter Reference An Extended Parameter Reference contains a reference to another parameter. The value is formatted the same as an analog mode Indirect Selector parameter (SSpppp, where SS = slot number of device to which this Extended Parameter Reference is pointing, and pppp = number of the parameter or diagnostic item to which this Extended Parameter Reference is pointing). Note that an Extended Parameter Reference can only select parameters unlike an Indirect Selector. An Extended Parameter Reference could be used to configure a Datalink or show the source of a Reference (among other uses).
21	Uses Rating Table Object	This parameter has rating-dependent defaults and limits that can be obtained from the Rating Table Object. The Offline Read Full will include the default value for the smallest rating and limits that will accommodate the full range of values allowed in the family of devices using this particular combination of Family Code and Config Code. The Online Read Full will include the rating-dependent default and limit values for this particular combination of Family Code, Config Code, and Rating Code.
22	Writable Referenced Parameter	This bit must be zero unless the parameter is an Extended Parameter Reference. If the parameter is an Extended Parameter Reference, then: 0 = The referenced parameter may be read-only or writable. 1 = The referenced parameter must always be writable (including while running).
23	Disallow Zero	This bit must be zero unless the parameter is an Indirect Selector or Extended Parameter Reference. If the parameter is an Indirect Selector or Extended Parameter Reference, then: 0 = Allow zero 1 = Disallow zero If this bit is cleared (indicating that a value of zero is allowed), the device must support the “Zero Text” parameter attribute so that a software tool or HIM can obtain text from the Zero Text parameter attribute. If this bit is set (indicating that a value of zero is disallowed), a software tool or HIM will not allow the user to enter a value of zero.
24	Datalink Out	This bit is used by offline tools and indicates that this is a Datalink Out parameter. Bit 20 must also be set.
25	Datalink In	This bit is used by offline tools and indicates that this is a Datalink In parameter. Bits 20 and 22 must also be set.

Bit	Name	Description
26	Not Writable While IO Active	This parameter cannot be written if the I/O data being exchanged between the Host and the peripheral is valid.
27	Command Parameter	This parameter commands the drive to take an action, such as “Reset Defaults” or “Autotune,” and then returns to a value of zero. Offline software tools will not allow setting this parameter to anything other than a value of zero. If an offline file contains a Command Parameter with a non-zero value, the offline software tool will change the value to zero. Note that command parameters cannot have values that do not return to zero.
28	Current Value Is Default	This bit identifies a parameter that will not change if a “Reset Defaults” is commanded. For example, if a drive contains a Language parameter that is set to German, setting defaults will leave the parameter set to German. Likewise, if the parameter is set to French, setting defaults will leave the parameter set to French.
29	Use Zero Text	If the “Disallow Zero” bit is set, this bit must be cleared. If the “Disallow Zero” bit is cleared, then: 0 = Use Disabled Text parameter class attribute. 1 = Use Zero Text parameter instance attribute.
30-31	Reserved	Reserved

Formulas for Converting

Display Value = ((Internal Value + Offset) x Multiplier x Base) / (Divisor x $10^{\text{Decimal Places}}$))

Internal Value = ((Display Value x Divisor x $10^{\text{Decimal Places}}$) / (Multiplier x Base)) - Offset

Common Services

Service Code	Implemented for: Class	Implemented for: Instance	Service Name
0x0E	Yes	Yes	Get_Attribute_Single
0x10	Yes	Yes	Set_Attribute_Single

Object Specific Services

Service Code	Implemented for: Class	Implemented for: Instance	Service Name	Allocation Size (in bytes): Par. Number	Allocation Size (in bytes): Par. Value
0x4D	Yes	No	Get_Attributes_Scattered	4	4
0x4E	Yes	No	Set_Attributes_Scattered	4	4

The table below lists the parameters for the Get_Attributes_Scattered and Set_Attributes_Scattered object-specific service:

Name	Data Type	Description
Parameter Number	UDINT	Parameter to read or write
Parameter Value	UDINT	Parameter value to read or write (zero when reading)

DPI Fault Object

Class Code

Hexadecimal	Decimal
0x97	151

Products such as PowerFlex drives use this object for faults. Adapters use this object for events.

Services

	Implemented for:		
Service Code	Class	Instance	Service Name
0x0E	Yes	Yes	Get_Attribute_Single
0x10	Yes	Yes	Set_Attribute_Single

Instances

The number of instances depends on the maximum number of faults or events supported in the queue. The maximum number of faults/events can be read in Instance 0, Attribute 2.

Instances (Hex.)	(Dec.)	Device
0x0000 – 0x3FFF	0 – 16383	Host Drive
0x4000 – 0x43FF	16384 – 17407	Adapter
0x4400 – 0x47FF	17408 – 18431	Port 1
0x4800 – 0x4BFF	18432 – 19455	Port 2
0x4C00 – 0x4FFF	19456 – 20479	Port 3
0x5000 – 0x53FF	20480 – 21503	Port 4
0x5400 – 0x57FF	21504 – 22527	Port 5
0x5800 – 0x5BFF	22528 – 23551	Port 6
0x5C00 – 0x5FFF	23552 – 24575	Port 7
0x6000 – 0x63FF	24576 – 25599	Port 8
0x6400 – 0x67FF	25600 – 26623	Port 9
0x6800 – 0x6BFF	26624 – 27647	Port 10
0x6C00 – 0x6FFF	27648 – 28671	Port 11
0x7000 – 0x73FF	28672 – 29695	Port 12
0x7400 – 0x77FF	29696 – 30719	Port 13
0x7800 – 0x7BFF	30720 – 31743	Port 14

Example	Description
0	Class Attributes (Drive)
1	Most Recent Drive Fault
2	Second Most Recent Drive Fault
⋮	⋮
16384	Class Attributes (Adapter)
16385	Most Recent Adapter Event
⋮	⋮

Class Attributes

Attribute ID	Access Rule	Name	Data Type	Description
1	Get	Class Revision	UINT	Revision of object
2	Get	Number of Instances	UINT	Maximum number of faults/events that the device can record in its queue
3	Set	Fault Command Write	USINT	0 = No Operation 1 = Clear Fault/Event 2 = Clear Fault/Event Queue 3 = Reset Device
4	Get	Fault Trip Instance Read	UINT	Fault that tripped the device. For adapters, this value is always 1 when faulted.

Attribute ID	Access Rule	Name	Data Type	Description
5	Get	Fault Data List	STRUCT of: USINT USINT UINT[n]	Reserved
6	Get	Number of Recorded Faults	UINT	Number of faults/events in the queue. A “0” indicates the fault queue is empty.
7	Get	Fault Parameter Reference	UINT	Reserved

Instance Attributes

Attribute ID	Access Rule	Name	Data Type	Description
0	Get	Full/All Information	STRUCT of UINT STRUCT of: USINT USINT STRING[16] STRUCT of: LWORD BOOL[16] UINT CONTAINER[n]	 Fault code Fault source DPI port DPI Device Object Fault text Fault time stamp Timer value (0 = timer not supported) BOOL[0]: (0 = invalid data, 1 = valid data) BOOL[1]: (0 = elapsed time, 1 = real time) BOOL[2 - 15]: Not used Reserved Reserved
1	Get	Basic Information	STRUCT of: UINT STRUCT of: USINT USINT STRUCT of: LWORD BOOL[16]	 Fault code Fault source DPI port DPI Device Object Fault time stamp Timer value (0 = timer not supported) BOOL[0]: (0 = invalid data, 1 = valid data) BOOL[1]: (0 = elapsed time, 1 = real time) BOOL[2 - 15]: Not used
2	Get	International Fault Text	STRINGN	Text describing the fault with support for Unicode.

DPI Alarm Object

Class Code

Hexadecimal	Decimal
0x98	152

Products such as PowerFlex drives use this object for alarms or warnings. Adapters do not support this object.

Services

	Implemented for:		
Service Code	Class	Instance	Service Name
0x0E	Yes	Yes	Get_Attribute_Single
0x10	Yes	Yes	Set_Attribute_Single

Instances

The number of instances depends on the maximum number of alarms supported by the queue. The maximum number of alarms can be read in Instance 0, Attribute 2.

Instances (Hex.)	(Dec.)	Device
0x0000 – 0x3FFF	0 – 16383	Host Drive
Only host devices can have alarms.		

Example	Description
0	Class Attributes (Drive)
1	Most Recent Alarm
2	Second Most Recent Alarm
⋮	⋮

Class Attributes

Attribute ID	Access Rule	Name	Data Type	Description
1	Get	Class Revision	UINT	Revision of object
2	Get	Number of Instances	UINT	Maximum number of alarms that the device can record in its queue
3	Set	Alarm Command Write	USINT	0 = No Operation 1 = Clear Alarm 2 = Clear Alarm Queue 3 = Reset Device
4	Get	Fault Data List	STRUCT of: USINT USINT UINT[n]	Reserved
5	Get	Number of Recorded Alarms	UINT	Number of alarms in the queue. A “0” indicates the alarm queue is empty.

Instance Attributes

Attribute ID	Access Rule	Name	Data Type	Description
0	Get	Full/All Information	STRUCT of UINT STRUCT of: USINT USINT STRING[16] STRUCT of: LWORD BOOL[16] UINT CONTAINER[n]	 Alarm code Alarm source DPI port DPI Device Object Alarm text Alarm time stamp Timer value (0 = timer not supported) BOOL[0]: (0 = invalid data, 1 = valid data) BOOL[1]: (0 = elapsed time, 1 = real time) BOOL[2 - 15] Reserved Reserved Reserved
1	Get	Basic Information	STRUCT of UINT STRUCT of: USINT USINT STRUCT of: LWORD BOOL[16]	 Alarm code Alarm source DPI port DPI Device Object Alarm time stamp Timer value (0 = timer not supported) BOOL[0]: (0 = invalid data, 1 = valid data) BOOL[1]: (0 = elapsed time, 1 = real time) BOOL[2 - 15] Reserved
2	Get	International Alarm Text	STRINGN	Text describing the alarm with support for Unicode.

DPI Diagnostic Object

Class Code

Hexadecimal	Decimal
0x99	153

Services

Service Code	Implemented for:		Service Name
	Class	Instance	
0x0E	Yes	Yes	Get_Attribute_Single
0x10	Yes	Yes	Set_Attribute_Single

Instances

The number of instances depends on the maximum number of diagnostic items in the device. The total number of diagnostic items can be read in Instance 0, Attribute 2.

Instances (Hex.)	(Dec.)	Device
0x0000 – 0x3FFF	0 – 16383	Host Drive
0x4000 – 0x43FF	16384 – 17407	Adapter
0x4400 – 0x47FF	17408 – 18431	Port 1
0x4800 – 0x4BFF	18432 – 19455	Port 2
0x4C00 – 0x4FFF	19456 – 20479	Port 3
0x5000 – 0x53FF	20480 – 21503	Port 4
0x5400 – 0x57FF	21504 – 22527	Port 5
0x5800 – 0x5BFF	22528 – 23551	Port 6
0x5C00 – 0x5FFF	23552 – 24575	Port 7
0x6000 – 0x63FF	24576 – 25599	Port 8
0x6400 – 0x67FF	25600 – 26623	Port 9
0x6800 – 0x6BFF	26624 – 27647	Port 10
0x6C00 – 0x6FFF	27648 – 28671	Port 11
0x7000 – 0x73FF	28672 – 29695	Port 12
0x7400 – 0x77FF	29696 – 30719	Port 13
0x7800 – 0x7BFF	30720 – 31743	Port 14

Example	Description
0	Class Attributes (Drive)
1	Drive Diagnostic Item 1
2	Drive Diagnostic Item 2
⋮	⋮
16384	Class Attributes (Adapter)
16385	Adapter Diagnostic Item 1
⋮	⋮

Class Attributes

Attribute ID	Access Rule	Name	Data Type	Description
1	Get	Class Revision	UINT	1
2	Get	Number of Instances	UINT	Number of diagnostic items in the device
3	Get	ENUM Offset	UINT	DPI ENUM object instance offset

Instance Attributes

Attribute ID	Access Rule	Name	Data Type	Description
0	Get	Full/All Info	STRUCT of:	
			BOOL[32]	Descriptor (see page C-16)
			CONTAINER (1)	Value
			CONTAINER	Minimum value
			CONTAINER	Maximum value
			CONTAINER	Default value
			UINT	Pad Word
			UINT	Pad Word
			STRING[4]	Units (for example, Amps, Hz)
			UINT	Multiplier (2)
			UINT	Divisor (2)
			UINT	Base (2)
			INT	Offset (2)
			UDINT	Link (source of the value) (0 = no link)
			STRING[16]	Always zero (0); Parameter name
1	Get/Set	Value	Various	Diagnostic item value
2	Get	International Diagnostic Item Text	Struct of:	
			STRINGN	Diagnostic name text
			STRINGN	Diagnostic units text
3	Get	International Full Read All	STRUCT of:	
			BOOL[32]	Descriptor
			CONTAINER	Value
			CONTAINER	Minimum
			CONTAINER	Maximum
			CONTAINER	Default
			UINT	Pad word
			UINT	Pad word
			UINT	Multiplier
			UINT	Divisor
			UINT	Base
			INT	Offset
			UDINT	Pad
			BOOL[32]	Extended descriptor
			STRINGN	Diagnostic name text
			STRINGN	Diagnostic name text

(1) A CONTAINER is a 32-bit block of data that contains the data type used by a value. If signed, the value is sign extended. Padding is used in the CONTAINER to ensure that it is always 32-bits.

(2) This value is used in the formulas used to convert the value between display units and internal units. Refer to Formulas for Converting on page C-18.

DPI Time Object

Class Code

Hexadecimal	Decimal
0x9B	155

Services

Service Code	Implemented for:		Service Name
	Class	Instance	
0x0E	Yes	Yes	Get_Attribute_Single
0x10	Yes	Yes	Set_Attribute_Single

Instances

The number of instances depends on the number of timers in the device. Instance 1 is always reserved for a real time clock although a device may not support it. The total number of timers can be read in Instance 0, Attribute 2.

Instances (Hex.)	(Dec.)	Device
0x0000 – 0x3FFF	0 – 16383	Host Drive
0x4000 – 0x43FF	16384 – 17407	Adapter
0x4400 – 0x47FF	17408 – 18431	Port 1
0x4800 – 0x4BFF	18432 – 19455	Port 2
0x4C00 – 0x4FFF	19456 – 20479	Port 3
0x5000 – 0x53FF	20480 – 21503	Port 4
0x5400 – 0x57FF	21504 – 22527	Port 5
0x5800 – 0x5BFF	22528 – 23551	Port 6
0x5C00 – 0x5FFF	23552 – 24575	Port 7
0x6000 – 0x63FF	24576 – 25599	Port 8
0x6400 – 0x67FF	25600 – 26623	Port 9
0x6800 – 0x6BFF	26624 – 27647	Port 10
0x6C00 – 0x6FFF	27648 – 28671	Port 11
0x7000 – 0x73FF	28672 – 29695	Port 12
0x7400 – 0x77FF	29696 – 30719	Port 13
0x7800 – 0x7BFF	30720 – 31743	Port 14

Example	Description
0	Class Attributes (Drive)
1	Real Time Clock (Predefined) (not always supported)
2	Timer 1
3	Timer 2
⋮	⋮

Class Attributes

Attribute ID	Access Rule	Name	Data Type	Description
1	Get	Class Revision	UINT	Revision of object
2	Get	Number of Instances	UINT	Number of timers in the object, excluding the real time clock that is predefined.
3	Get	First Device Specific Timer	UINT	Instance of the first timer that is not predefined.
4	Set	Time Command Write	USINT	0 = No Operation 1 = Clear all timers (Does not clear the real time clock or read only timers)
5	Get	Number of Supported Time Zones	UINT	Number of time zones described in the Time Zone List attribute.
6	Get	Time Zone List	STRUCT	Identifies a time zone.

Attribute ID	Access Rule	Name	Data Type	Description
7	Get/Set	Active Time Zone ID	UINT	The ID field of the Time Zone List structure for the desired time zone.
8	Get	Active Time Zone Data	Struct of: INT USINT USINT USINT USINT USINT USINT INT USINT USINT USINT USINT USINT USINT	 Standard bias Standard month Standard day of week Standard week Standard hour Standard minute Standard second Daylight offset Daylight month Daylight day of week Daylight week Daylight hour Daylight minute Daylight second
9	Get/Set	Custom Time Zone Data	Struct of: INT USINT USINT USINT USINT USINT USINT INT USINT USINT USINT USINT USINT USINT	 Standard bias Standard month Standard day of week Standard week Standard hour Standard minute Standard second Daylight offset Daylight month Daylight day of week Daylight week Daylight hour Daylight minute Daylight second

Instance Attributes

Attribute ID	Access Rule	Name	Data Type	Description
0	Get	Read Full	STRUCT of: STRING[16] LWORD or STRUCT BOOL[16]	 Name of the timer Elapsed time in milliseconds unless timer is a real time clock (see attribute 2) See Attribute 3
1	Get	Timer Text	STRING[16]	Name of the timer
2	Get/Set	Timer Value	LWORD -or- STRUCT of: UINT USINT USINT USINT USINT USINT USINT	Elapsed time in milliseconds unless the timer is a real time clock. Real Time Clock Data: Milliseconds (0 – 999) Seconds (0 – 59) Minutes (0 – 59) Hours (0 – 23) Days (1 – 31) Months (1 = January, 12 = December) Years (since 1972)
3	Get	Timer Descriptor	BOOL[16]	BOOL[0]: (0 = invalid data, 1 = valid data) BOOL[1]: (0 = elapsed time, 1 = real time) BOOL[2 - 15]: Not used
4	Get	International Read Full	Struct of: STRINGN STRUCT BOOL[16]	 International timer text Timer value Timer descriptor
5	Get	International Timer Text	STRINGN	Name of this timer
6	Get	Clock Status	BOOL[32]	Identifies clock status

Attribute ID	Access Rule	Name	Data Type	Description
8	Get/Set	Number of Leap Seconds	INT	Identifies the current number of Leap Seconds.
9	Get	Clock Options	BOOL[32]	Identifies the optional functionality available in the device's System Clock.
10	Get/Set	Clock Options Enable	BOOL[32]	Identifies which of the clock's options are enabled.

Host DPI Parameter Object

Class Code

Hexadecimal	Decimal
0x9F	159

To access Device parameters, use the DPI Parameter Object (Class Code 0x93).

Instances

The number of instances depends on the number of parameters in the device. The total number of parameters can be read in Instance 0, Attribute 0.

Instances (Hex.)	(Dec.)	Device
0x0000 – 0x3FFF	0 – 16383	Reserved
0x4000 – 0x43FF	16384 – 17407	Adapter
0x4400 – 0x47FF	17408 – 18431	Port 1
0x4800 – 0x4BFF	18432 – 19455	Port 2
0x4C00 – 0x4FFF	19456 – 20479	Port 3
0x5000 – 0x53FF	20480 – 21503	Port 4
0x5400 – 0x57FF	21504 – 22527	Port 5
0x5800 – 0x5BFF	22528 – 23551	Port 6
0x5C00 – 0x5FFF	23552 – 24575	Port 7
0x6000 – 0x63FF	24576 – 25599	Port 8
0x6400 – 0x67FF	25600 – 26623	Port 9
0x6800 – 0x6BFF	26624 – 27647	Port 10
0x6C00 – 0x6FFF	27648 – 28671	Port 11
0x7000 – 0x73FF	28672 – 29695	Port 12
0x7400 – 0x77FF	29696 – 30719	Port 13
0x7800 – 0x7BFF	30720 – 31743	Port 14

Example	Description
16384	Class Attributes (Adapter)
16385	Adapter Parameter 1 Attributes
16386	Adapter Parameter 2 Attributes
⋮	⋮
17408	Class Attributes (HIM)
17409	HIM Parameter 1 Attributes
17410	HIM Parameter 2 Attributes
⋮	⋮

Class Attributes

Attribute ID	Access Rule	Name	Data Type	Description
0	Get	Number of Instances	UINT	Number of parameters in the device
1	Set	Write Protect Password	UINT	0 = Password disabled n = Password
2	Set	NVS Command Write	USINT	0 = No Operation 1 = Store values in active memory to NVS 2 = Load values in NVS to active memory 3 = Load default values to active memory
3	Get	NVS Parameter Value Checksum	UINT	Checksum of all parameter values in a user set in NVS
4	Get	NVS Link Value Checksum	UINT	Checksum of parameter links in a user set in NVS
5	Get	First Accessible Parameter	UINT	First parameter available if parameters are protected by passwords. A “0” indicates all parameters are protected.
7	Get	Class Revision	UINT	2 = DPI
8	Get	First Parameter Processing Error	UINT	The first parameter that has been written with a value outside of its range. A “0” indicates no errors.
9	Set	Link Command	USINT	0 = No Operation 1 = Clear All Parameter Links (This does not clear links to function blocks.)

Instance Attributes

Attribute ID	Access Rule	Name	Data Type	Description
6	Get	DPI Offline Read Full	STRUCT of: BOOL[32] CONTAINER CONTAINER CONTAINER STRING[16] STRING[4] UINT UINT UINT UINT UINT UINT UINT USINT USINT UINT UINT CONTAINER UINT UNIT UNIT INT	 Descriptor Offline Minimum value Offline Maximum value Offline Default value Parameter name Offline parameter units Online minimum parameter instance Online maximum parameter instance Online default parameter instance Multiplier parameter instance Divisor parameter instance Base parameter instance Offset parameter instance Formula number Pad byte (always zero) Help instance Pad word (always a value of zero) Parameter value Multiplier Divisor Base Offset
7	Get	DPI Online Read Full	STRUCT of: BOOL[32] CONTAINER(1) CONTAINER CONTAINER CONTAINER UINT UINT STRING[4] UINT UINT UINT INT USINT[3] USINT STRING[16]	 Descriptor (see page C-31) Parameter value Minimum value Maximum value Default value Next parameter Previous parameter Units (for example, Amps, Hz) Multiplier (2) Divisor (2) Base (2) Offset (2) Link (source of the value) (0 = no link) Always zero (0) Parameter name
8	Get	DPI Descriptor	BOOL[32]	Descriptor (see page C-31)
9	Get/Set	DPI Parameter Value	Various	Parameter value in NVS. (3)
10	Get/Set	DPI RAM Parameter Value	Various	Parameter value in temporary memory.
11	Get/Set	DPI Link	USINT[3]	Link (parameter or function block that is the source of the value) (0 = no link)
12	Get	Help Object Instance	UINT	ID for help text for this parameter
13	Get	DPI Read Basic	STRUCT of: BOOL[32] CONTAINER CONTAINER CONTAINER CONTAINER STRING[16] STRING[4]	 Descriptor (see page C-31) Parameter value Minimum value Maximum value Default value Parameter name Units (for example, Amps, Hz)
14	Get	DPI Parameter Name	STRING[16]	Parameter name
15	Get	DPI Parameter Alias	STRING[16]	Customer supplied parameter name.
16	Get	Parameter Processing Error	USINT	0 = No error 1 = Value is less than the minimum 2 = Value is greater than the maximum
18	Get	International DPI Offline Parameter Text	Struct of: STRINGN STRINGN	 International parameter name International offline units

Attribute ID	Access Rule	Name	Data Type	Description
19	Get	International DPI Online Parameter Text	Struct of: STRINGN STRINGN	 International parameter name International online units
20	Get	International DPI Online Read Full	Struct of: BOOL[32] CONTAINER CONTAINER CONTAINER CONTAINER UINT UINT UINT UINT UINT INT USINT[3] USINT BOOL[32] STRINGN STRINGN	 Descriptor Parameter value Online minimum value Online maximum value Online default value Next Previous Multiplier Divisor Base Offset Link Pad word (always zero) Extended descriptor International parameter name International online parameter units
21	Get	DPI Extended Descriptor	UDINT	Extended Descriptor (see page C-32)
22	Get	International DPI Offline Read Full	Struct of: BOOL CONTAINER CONTAINER CONTAINER UINT UINT UINT UINT UINT UINT USINT USINT UINT UINT CONTAINER UINT UINT UINT INT BOOL[32] STRINGN STRINGN	 Descriptor Offline minimum value Offline maximum value Offline default value Online minimum parameter instance Online maximum parameter instance Online default parameter instance Multiplier parameter instance Divisor parameter instance Base parameter instance Offset parameter instance Formula number Pad word (always zero) Help instance Pad word (always a value of zero) Parameter value Multiplier Divisor Base Offset Extended DPI descriptor International DPI parameter name International DPI offline parameter units

(1) A CONTAINER is a 32-bit block of data that contains the data type used by a parameter value. If signed, the value is sign extended. Padding is used in the CONTAINER to ensure that it is always 32-bits.

(2) This value is used in the formulas used to convert the parameter value between display units and internal units. Refer to Formulas for Converting on page C-33.

(3) Do NOT continually write parameter data to NVS. Refer to the attention on page 6-1.

Descriptor Attributes

Bit	Name	Description
0	Data Type (Bit 1)	Right bit is least significant bit (0). 000 = USINT used as an array of Boolean 001 = UINT used as an array of Boolean 010 = USINT (8-bit integer) 011 = UINT (16-bit integer) 100 = UDINT (32-bit integer) 101 = TCHAR ((8-bit (not Unicode) or 16-bits (Unicode)) 110 = REAL (32-bit floating point value) 111 = Use bits 16, 17, 18
1	Data Type (Bit 2)	
2	Data Type (Bit 3)	
3	Sign Type	0 = unsigned 1 = signed
4	Hidden	0 = visible 1 = hidden
5	Not a Link Sink	0 = May be the sink end of a link 1 = May not be the sink end of a link
6	Not Recallable	0 = Recallable from NVS 1 = Not Recallable from NVS
7	ENUM	0 = No ENUM text 1 = ENUM text
8	Writable	0 = Read only 1 = Read/write
9	Not Writable When Enabled	0 = Writable when enabled (e.g., drive running) 1 = Not writable when enabled
10	Instance	0 = Parameter value is not a Reference to another parameter 1 = Parameter value refers to another parameter
11	Uses Bit ENUM Mask	This parameter instance supports the Bit ENUM Mask attribute. For more information, see the definition of the attribute.
12	Decimal Place (Bit 0)	Number of digits to the right of the decimal point. 0000 = 0 1111 = 15
13	Decimal Place (Bit 1)	
14	Decimal Place (Bit 2)	
15	Decimal Place (Bit 3)	
16	Extended Data Type (Bit 4)	Bit 16 is the least significant bit. 000 = Reserved 001 = UDINT used as an array of Boolean 010 = Reserved 011 = Reserved 100 = Reserved 101 = Reserved 110 = Reserved 111 = Reserved
17	Extended Data Type (Bit 5)	
18	Extended Data Type (Bit 6)	
19	Parameter Exists	Used to mark parameters that are not available to network tools.
20	Not Used	Reserved
21	Formula Links	Indicates the Formula Data is derived from other parameters.
22	Access Level (Bit 1)	A 3-bit field used to control access to parameter data.
23	Access Level (Bit 2)	
24	Access Level (Bit 3)	
25	Writable ENUM	ENUM text: 0 = Read Only, 1 = Read/Write
26	Not a Link Source	0 = May be the source end of a link 1 = May not be the source end of a link
27	Enhanced Bit ENUM	Parameter supports enhanced bit ENUMs.
28	Enhanced ENUM	Parameter supports enhanced ENUMs.
29	Uses DPI Limits Object	Parameter uses the DPI Limits Object. • Intelligent offline tools make use of the Limits Object to select limits and units.
30	Extended Descriptor	Parameter uses Extended Descriptor bits, which can be obtained by reading the DPI Extended Descriptor attribute for this parameter.
31	Always Upload/Download	Parameter shall always be included in uploads and downloads.

Extended Descriptor Attributes

Bit	Name	Description
0	Indirect Mode	0 = Analog (selects entire parameters) 1 = Digital (selects individual bits within parameters)
1	Indirect Type 0	Analog input list (Instance 0xFFFF)
2	Indirect Type 1	Digital input list (Instance 0xFFFE)
3	Indirect Type 2	Feedback list (Instance 0xFFFD)
4	Indirect Type 3	Analog output list (Instance 0xFFFC)
5	Indirect Type 4	Digital output list (Instance 0xFFFB)
6	Indirect Type 5	Undefined (Instance 0xFFFA)
7	Indirect Type 6	Undefined (Instance 0xFFF9)
8	Indirect Type 7	Undefined (Instance 0xFFF8)
9	Indirect Type 8	Undefined (Instance 0xFFF7)
10	Indirect Type 9	Undefined (Instance 0xFFF6)
11	Indirect Type 10	Undefined (Instance 0xFFF5)
12	Indirect Type 11	Undefined (Instance 0xFFF4)
13	Indirect Type 12	Undefined (Instance 0xFFF3)
14	Indirect Type 13	Undefined (Instance 0xFFF2)
15	Indirect Type 14	Parameter-specific list
16	FP Max Decimals Bit 0	These four bits are used on REAL parameters only. They indicate the maximum number of decimal places to be displayed for small values. A value of 0 indicates to not limit the number of decimal places used.
17	FP Max Decimals Bit 1	
18	FP Max Decimals Bit 2	
19	FP Max Decimals Bit 1	
20	Extended Parameter Reference	0 = Not an Extended Parameter Reference 1 = Extended Parameter Reference An Extended Parameter Reference contains a reference to another parameter. The value is formatted the same as an analog mode Indirect Selector parameter (SSpppp, where SS = slot number of device to which this Extended Parameter Reference is pointing, and pppp = number of the parameter or diagnostic item to which this Extended Parameter Reference is pointing). Note that an Extended Parameter Reference can only select parameters unlike an Indirect Selector. An Extended Parameter Reference could be used to configure a Datalink or show the source of a Reference (among other uses).
21	Uses Rating Table Object	This parameter has rating-dependent defaults and limits that can be obtained from the Rating Table Object. The Offline Read Full will include the default value for the smallest rating and limits that will accommodate the full range of values allowed in the family of devices using this particular combination of Family Code and Config Code. The Online Read Full will include the rating-dependent default and limit values for this particular combination of Family Code, Config Code, and Rating Code.
22	Writable Referenced Parameter	This bit must be zero unless the parameter is an Extended Parameter Reference. If the parameter is an Extended Parameter Reference, then: 0 = The referenced parameter may be read-only or writable. 1 = The referenced parameter must always be writable (including while running).
23	Disallow Zero	This bit must be zero unless the parameter is an Indirect Selector or Extended Parameter Reference. If the parameter is an Indirect Selector or Extended Parameter Reference, then: 0 = Allow zero 1 = Disallow zero If this bit is cleared (indicating that a value of zero is allowed), the device must support the “Zero Text” parameter attribute so that a software tool or HIM can obtain text from the Zero Text parameter attribute. If this bit is set (indicating that a value of zero is disallowed), a software tool or HIM will not allow the user to enter a value of zero.
24	Datalink Out	This bit is used by offline tools and indicates that this is a Datalink Out parameter. Bit 20 must also be set.
25	Datalink In	This bit is used by offline tools and indicates that this is a Datalink In parameter. Bits 20 and 22 must also be set.

Bit	Name	Description
26	Not Writable While IO Active	This parameter cannot be written if the I/O data being exchanged between the Host and the peripheral is valid.
27	Command Parameter	This parameter commands the drive to take an action, such as “Reset Defaults” or “Autotune,” and then returns to a value of zero. Offline software tools will not allow setting this parameter to anything other than a value of zero. If an offline file contains a Command Parameter with a non-zero value, the offline software tool will change the value to zero. Note that command parameters cannot have values that do not return to zero.
28	Current Value Is Default	This bit identifies a parameter that will not change if a “Reset Defaults” is commanded. For example, if a drive contains a Language parameter that is set to German, setting defaults will leave the parameter set to German. Likewise, if the parameter is set to French, setting defaults will leave the parameter set to French.
29	Use Zero Text	If the “Disallow Zero” bit is set, this bit must be cleared. If the “Disallow Zero” bit is cleared, then: 0 = Use Disabled Text parameter class attribute. 1 = Use Zero Text parameter instance attribute.
30-31	Reserved	Reserved

Formulas for Converting

Display Value = ((Internal Value + Offset) x Multiplier x Base) / (Divisor x $10^{\text{Decimal Places}}$))

Internal Value = ((Display Value x Divisor x $10^{\text{Decimal Places}}$) / (Multiplier x Base)) - Offset

Common Services

Service Code	Implemented for: Class	Implemented for: Instance	Service Name
0x0E	Yes	Yes	Get_Attribute_Single
0x10	Yes	Yes	Set_Attribute_Single

Object Specific Services

Service Code	Implemented for: Class	Implemented for: Instance	Service Name	Allocation Size (in bytes): Par. Number	Allocation Size (in bytes): Par. Value
0x4D	Yes	No	Get_Attributes_Scattered	4	4
0x4E	Yes	No	Set_Attributes_Scattered	4	4

The table below lists the parameters for the Get_Attributes_Scattered and Set_Attributes_Scattered object-specific service:

Name	Data Type	Description
Parameter Number	UDINT	Parameter to read or write
Parameter Value	UDINT	Parameter value to read or write (zero when reading)

TCP/IP Interface Object

Class Code

Hexadecimal	Decimal
0xF5	245

Services

Service Code	Implemented for:		Service Name
	Class	Instance	
0x0E	Yes	Yes	Get_Attribute_Single
0x10	No	Yes	Set_Attribute_Single

Instances

The adapter supports one instance of the TCP/IP Interface object.

Number	Description
0	Class Attributes
1	Object Attributes

Class Attributes

Attribute ID	Access Rule	Name	Data Type	Description
1	Get	Revision	UINT	The revision of this object

Instance Attributes

Attribute ID	Access Rule	Name	Data Type	Description
1	Get	Status of TCP/IP Network Interface	UDINT	0 = Not configured 1 = Valid configuration 2 to 15 = Reserved
2	Get	Configuration Capability	UDINT	Bit I Value (0 = False, 1 = True) 0 = Supports BOOTP 1 = DNS Client (able to resolve host names by query to DNS server) 2 = DHCP Client (able to obtain network configuration through DHCP) 3 = DHCP-DNS Update (able to send its host name in the DHCP request) 4 = Configuration Settable (able to set the network configuration via TCP/IP object) 5 to 31 = Reserved
3	Set	Configuration Control	UDINT	Bit I Value 1 – 3 = Startup configuration 0 = Use configuration saved in NVS 1 = Obtain configuration via BOOTP 2 = Obtain configuration via DHCP 3 to 15 = Reserved 4 = DNS Enabled (resolves host names by query to DNS server) 5 to 31 = Reserved

Attribute ID	Access Rule	Name	Data Type	Description
4	Get	Physical Link Object	STRUCT of: UINT Padded EPATH	 Path size Path
5	Get	Interface Configuration	STRUCT of: UDINT UDINT UDINT UDINT UDINT STRING	 Adapter's IP address Adapter's subnet mask Adapter's gateway address Primary name server Secondary name server Default domain name
6	Get	Host Name	STRING	Host name when using DHCP

Ethernet Link Object

Class Code

Hexadecimal	Decimal
0xF6	246

Services

Service Code	Implemented for:		Service Name
	Class	Instance	
0x0E	Yes	Yes	Get_Attribute_Single
0x4C	No	Yes	Get_and_Clear

Instances

The adapter supports one instance of the TCP/IP Interface object.

Number	Description
0	Class Attributes
1	Object Attributes

Class Attributes

Attribute ID	Access Rule	Name	Data Type	Description
1	Get	Revision	UINT	The revision of this object

Instance Attributes

Attribute ID	Access Rule	Name	Data Type	Description
1	Get	Interface Speed	UDINT	Speed in megabits per second (Mbs)
2	Get	Interface Flags	UDINT	Bit I Value 0 = Link status (0 = inactive, 1 = active) 1 = Duplex (0 = half duplex, 1 = full duplex) 2 to 31 = Reserved
3	Get	Physical Address	USINT[6]	MAC address (XX-XX-XX-XX-XX-XX) The first octet (USINT[0]) is on the left.
4	Get	Interface Counters	STRUCT of: UDINT UDINT UDINT UDINT UDINT UDINT UDINT UDINT UDINT UDINT UDINT	 Octets received Unicast packets received Non-unicast packets received Inbound packets received but discarded Inbound packets with errors (not discarded) Inbound packets with unknown protocol Octets sent Unicast packets sent Non-unicast packets sent Outbound packets discarded Outbound packets with errors

Attribute ID	Access Rule	Name	Data Type	Description
5	Get	Media Counters	STRUCT of:	RX = Received, TX = Transmitted
			UDINT	RX frames not having integral number of octets long
			UDINT	RX frames not passing FCS check
			UDINT	TX frames having one collision
			UDINT	TX frames having multiple collisions
			UDINT	Number of times of SQE test error message
			UDINT	TX Frames delayed first attempt by busy medium
			UDINT	Collisions detected later than 512 bit-times in trans.
			UDINT	TX frames failing due to excessive collisions
			UDINT	TX frames failing due to intern MAC sublayer TX error
			UDINT	Times of carrier sense condition loss during trans.
			UDINT	RX frames exceeding the maximum frame size
			UDINT	RX frames failing due to intern MAC sublayer RX error

Notes:

Logic Command/Status Words for PowerFlex 750-Series Drives

Appendix D presents the definitions of the Logic Command and Logic Status words that are used for PowerFlex 750-Series drives.

Logic Command Word

Logic Bits																													Command	Description
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
																															x	Normal Stop	0 = Not Normal Stop 1 = Normal Stop
																													x			Start (1)	0 = Not Start 1 = Start
																												x				Jog 1 (2)	0 = Not Jog 1 (Par. 556) 1 = Jog 1
																											x					Clear Fault (3)	0 = Not Clear Fault 1 = Clear Fault
																										x	x					Unipolar Direction	00 = No Command 01 = Forward Command 10 = Reverse Command 11 = Hold Direction Control
																									x							Manual	0 = Not Manual 1 = Manual
																								x								Reserved	
																						x	x									Accel Time	00 = No Command 01 = Use Accel Time 1 (Par. 535) 10 = Use Accel Time 2 (Par. 536) 11 = Use Present Time
																				x	x											Decel Time	00 = No Command 01 = Use Decel Time 1 (Par. 537) 10 = Use Decel Time 2 (Par. 538) 11 = Use Present Time
																			x													Ref Select 1	000 = No Command 001 = Ref A Select (Par. 545) 010 = Ref B Select (Par. 550) 011 = Preset 3 (Par. 573) 100 = Preset 4 (Par. 574) 101 = Preset 5 (Par. 575) 110 = Preset 6 (Par. 576) 111 = Preset 7 (Par. 577)
																		x														Ref Select 2	
																	x															Ref Select 3	
																x																Reserved	
															x																	Coast Stop	0 = Not Coast to Stop 1 = Coast to Stop
														x																		Current Limit Stop	0 = Not Current Limit Stop 1 = Current Limit Stop
													x																			Run (4)	0 = Not Run 1 = Run
												x																				Jog 2 (2)	0 = Not Jog 2 (Par. 557) 1 = Jog 2
											x																					Reserved	
										x																						Reserved	
									x																							Reserved	
								x																								Reserved	
							x																									Reserved	
						x																										Reserved	
					x																											Reserved	
				x																												Reserved	
			x																													Reserved	
		x																														Reserved	
	x																															Reserved	
x																																Reserved	

(1) A Not Stop condition (logic bit 0 = 0) must first be present before a 1 = Start condition will start the drive.

(2) A Not Stop condition (logic bit 0 = 0) must first be present before a 1 = Jog 1/Jog 2 condition will jog the drive. A transition to a "0" will stop the drive.

(3) To perform this command, the value must switch from "0" to "1."

(4) A Not Stop condition (logic bit 0 = 0) must first be present before a 1 = Run condition will run the drive. A transition to a "0" will stop the drive.

Logic Status Word

Logic Bits																													Command	Description
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
																															x	Run Ready	0 = Not Ready to Run 1 = Ready to Run
																														x		Active	0 = Not Active 1 = Active
																													x			Command Direction	0 = Reverse 1 = Forward
																												x				Actual Direction	0 = Reverse 1 = Forward
																											x					Accelerating	0 = Not Accelerating 1 = Accelerating
																										x						Decelerating	0 = Not Decelerating 1 = Decelerating
																									x							Alarm	0 = No Alarm (Par. 959 & 960) 1 = Alarm
																								x								Fault	0 = No Fault (Par. 952 & 953) 1 = Fault
																							x									At Setpt Spd	0 = Not at Setpoint Speed 1 = At Setpoint Speed
																						x										Manual	0 = Manual Mode Not Active 1 = Manual Mode Active
																					x											Spd Ref ID 0	00000 = Reserved 00001 = Auto Ref A (Par. 545) 00010 = Auto Ref B (Par. 550) 00011 = Auto Preset Speed 3 (Par. 573) 00100 = Auto Preset Speed 4 (Par. 574) 00101 = Auto Preset Speed 5 (Par. 575) 00110 = Auto Preset Speed 6 (Par. 576) 00111 = Auto Preset Speed 7 (Par. 577) 01000 = Reserved 01001 = Reserved 01010 = Reserved 01011 = Reserved 01100 = Reserved 01101 = Reserved 01110 = Reserved 01111 = Reserved 10000 = Man Port 0 10001 = Man Port 1 10010 = Man Port 2 10011 = Man Port 3 10100 = Man Port 4 10101 = Man Port 5 10110 = Man Port 6 10111 = Reserved 11000 = Reserved 11001 = Reserved 11010 = Reserved 11011 = Reserved 11100 = Reserved 11101 = Man Port 13 (Emb. ENET) 11110 = Man Port 14 (Drive Logix) 11111 = Alternate Man Ref Sel
																				x												Spd Ref ID 1	
																			x													Spd Ref ID 2	
																		x														Spd Ref ID 3	
																	x															Spd Ref ID 4	
																x																Reserved	
															x																	Running	0 = Not Running 1 = Running
														x																		Jogging	0 = Not Jogging (Par. 556 & 557) 1 = Jogging
													x																			Stopping	0 = Not Stopping 1 = Stopping
												x																				DC Brake	0 = Not DC Brake 1 = DC Brake
											x																					DB Active	0 = Not Dynamic Brake Active 1 = Dynamic Brake Active
										x																						Speed Mode	0 = Not Speed Mode (Par. 309) 1 = Speed Mode
									x																							Position Mode	0 = Not Position Mode (Par. 309) 1 = Position Mode
								x																								Torque Mode	0 = Not Torque Mode (Par. 309) 1 = Torque Mode
							x																									At Zero Speed	0 = Not at Zero Speed 1 =At Zero Speed
						x																										At Home	0 = Not at Home 1 = At Home
					x																											At Limit	0 = Not at Limit 1 = At Limit
				x																												Current Limit	0 = Not at Current Limit 1 = At Current Limit
			x																													Bus Freq Reg	0 = Not Bus Freq Reg 1 = Bus Freq Reg
		x																														Enable On	0 = Not Enable On 1 = Enable On
	x																															Motor Overload	0 = Not Motor Overload 1 = Motor Overload
x																																Regen	0 = Not Regen 1= Regen

A

Adapter

Devices such as drives, controllers, and computers usually require an adapter to provide a communication interface between them and a network such as EtherNet/IP. An adapter reads data on the network and transmits it to the connected device. It also reads data in the device and transmits it to the network.

The embedded EtherNet/IP adapter connects PowerFlex 750-Series drives to an EtherNet/IP network. Adapters are sometimes also called “cards,” “embedded communication options,” “gateways,” “modules,” and “peripherals.”

B

BOOTP (Bootstrap Protocol)

BOOTP lets the adapter configure itself dynamically at boot time if the network has a BOOTP server. The BOOTP server assigns the adapter a preconfigured IP address, a subnet mask, and a gateway address; therefore, you do not have to configure these using the parameters in the adapter. BOOTP can make it easier to administer an Ethernet network. A free version of Rockwell Software’s BOOTP Server can be accessed at http://www.ab.com/networks.

Bridge

A network device that can route messages from one network to another. A bridge also refers to a communications module in a ControlLogix controller that connects the controller to a network. See also Scanner.

C

CIP (Common Industrial Protocol)

CIP is the transport and application layer protocol used for messaging over EtherNet/IP, ControlNet, and DeviceNet networks. The protocol is used for implicit messaging (real-time I/O) and explicit messaging (configuration, data collection, and diagnostics).

ControlFLASH

An Allen-Bradley software tool that lets users electronically update firmware on printed circuit boards. The tool takes advantage of the growing use of flash memory (electronic erasable chips) across industrial control products.

Controller

A controller, also called programmable logic controller, is a solid-state control system that has a user-programmable memory for storage of instructions to implement specific functions such as I/O control, logic, timing, counting, report generation, communication, arithmetic, and data file manipulation. A controller consists of a central processor, input/output interface, and memory. See also Scanner.

D

Data Rate

The speed at which data is transferred on the EtherNet/IP network. You can set the adapter to a data rate of 10Mbps Full-Duplex, 10Mbps Half-Duplex, 100Mbps Full-Duplex, or 100Mbps Half-Duplex. If another device on the network sets or auto-negotiates the data rate, you can set the adapter to automatically detect the data rate.

Datalinks

A Datalink is a type of pointer used by PowerFlex 750-Series drives to transfer data to and from the controller. Datalinks allow specified parameter value(s) to be accessed or changed without using explicit messages. When enabled, each 32-bit Datalink in a PowerFlex 750-Series drive consumes 4 bytes in the input image table and/or 4 bytes in the output image table of the controller.

DriveExplorer Software

A tool for monitoring and configuring Allen-Bradley products and network communication adapters. It can be run on computers running various Microsoft Windows operating systems. DriveExplorer (version 6.xx or higher) can be used to configure this adapter and PowerFlex drive. Information about DriveExplorer software and a free lite version can be accessed at http://www.ab.com/drives/driveexplorer.

DriveTools SP Software

A software suite designed for running on various Microsoft Windows operating systems. This software suite provides a family of tools, including DriveExecutive, that you can use to program, monitor, control, troubleshoot, and maintain Allen-Bradley products. DriveTools SP can be used with PowerFlex drives. Information about DriveTools SP can be accessed at http://www.ab.com/drives/drivetools.

Duplex

Duplex describes the mode of communication. *Full-duplex* communications let a device exchange data in both directions at the same time. *Half-duplex* communications let a device exchange data only in one direction at a time. The duplex used by the adapter depends on the type of duplex that other network devices, such as switches, support.

E

EDS (Electronic Data Sheet) Files

Simple text files that are used by network configuration tools to describe products so that you can easily commission them on a network. EDS files describe a product device type and revision. EDS files for many Allen-Bradley products can be found at http://www.ab.com/networks/eds.

EtherNet/IP Network

EtherNet/IP (Industrial Protocol) is an open producer-consumer communication network based on the Ethernet standard (IEEE 802.3),

TCP/IP, UDP/IP, and CIP. Designed for industrial communications, both I/O and explicit messages can be transmitted over the network. Each device is assigned a unique IP address and transmits data on the network. The number of devices that an EtherNet/IP network can support depends on the class of IP address. For example, a network with a Class C IP address can have 254 nodes.

General information about EtherNet/IP and the EtherNet/IP specification are maintained by the Open DeviceNet Vendor's Association (ODVA). ODVA is online at http://www.odva.org.

Explicit Messaging

Explicit messages are used to transfer data that does not require continuous updates. They are typically used to configure, monitor, and diagnose devices over the network.

F

Fault Action

A fault action determines how the adapter and connected drive act when a communications fault (for example, a cable is disconnected) occurs or when the controller is switched out of run mode. The former uses a communications fault action, and the latter uses an idle fault action.

Fault Configuration

When communications are disrupted (for example, a cable is disconnected), the adapter and PowerFlex drive can respond with a user-defined fault configuration. The user sets the data that is sent to the drive using specific fault configuration parameters in the adapter. When a fault action parameter is set to use the fault configuration data and a fault occurs, the data from these parameters is sent as the Logic Command, Reference, and/or Datalink(s).

Flash Update

The process of updating firmware in a device. The adapter can be flash updated using various Allen-Bradley software tools. Refer to Flash Updating the Adapter on page 3-18 for more information.

G

Gateway

A device on a network that connects an individual network to a system of networks. When a node needs to communicate with a node on another network, a gateway transfers the data between the two networks. You need to configure the address for the gateway device in the adapter if you want the adapter to communicate with devices that are not on its network.

H

Hardware Address

Each Ethernet device has a unique hardware address (sometimes called a MAC address) that is 48 bits. The address appears as six digits separated by colons (for example, xx:xx:xx:xx:xx:xx). Each digit has a

value between 0 and 255 (0x00 and 0xFF). This address is assigned in the hardware and cannot be changed. It is required to identify the device if you are using a BOOTP utility.

HIM (Human Interface Module)

A device that can be used to configure and control a drive. Enhanced PowerFlex 7-Class HIMs (for example, 20-HIM-A6) can be used to configure PowerFlex 750-Series drives and their connected peripherals.

Hold Last

When communication is disrupted (for example, a cable is disconnected), the adapter and PowerFlex drive can respond by holding last. Hold last results in the drive receiving the last data received via the network connection before the disruption. If the drive was running and using the Reference from the adapter, it will continue to run at the same Reference.

Idle Action

An idle action determines how the adapter and connected drive act when the controller is switched out of run mode.

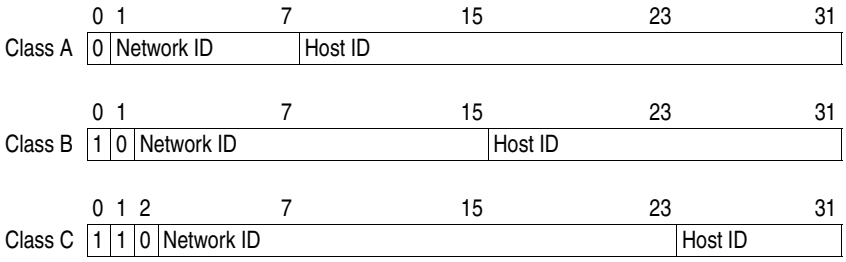
I/O Data

I/O data, sometimes called " implicit messages" or "input/output," is time-critical data such as a Logic Command and Reference. The terms "input" and "output" are defined from the controller's point of view. Output is produced by the controller and consumed by the adapter. Input is produced by the adapter and consumed by the controller.

IP Addresses

A unique IP address identifies each node on an EtherNet/IP network. An IP address consists of 32 bits that are divided into four segments of one byte each. It appears as four decimal integers separated by periods (xxx.xxx.xxx.xxx). Each "xxx" can have a decimal value from 0 to 255. For example, an IP address could be 192.168.0.1.

An IP address has two parts: a network ID and a host ID. The class of network determines the format of the address.



The number of devices on your EtherNet/IP network will vary depending on the number of bytes that are used for the network address. In many cases you are given a network with a Class C address, in which

the first three bytes contain the network address (subnet mask = 255.255.255.0). This leaves 8 bits or 256 addresses on your network. Because two addresses are reserved for special uses (0 is an address for the network usually used by the router, and 255 is an address for broadcast messages to all network devices), you have 254 addresses to use on a Class C address block.

To ensure that each device on the Internet has a unique address, contact your network administrator or Internet Service Provider for unique fixed IP addresses. You can then set the unique IP address for the adapter by using a BOOTP server or by manually configuring parameters in the adapter. The adapter reads the values of these parameters only at power-up.

L

Logic Command/Logic Status

The Logic Command is used to control the PowerFlex 750-Series drive (for example, start, stop, direction). It consists of one 32-bit word of output to the adapter from the network. The definitions of the bits in this word are shown in Appendix D.

The Logic Status is used to monitor the PowerFlex 750-Series drive (for example, operating state, motor direction). It consists of one 32-bit word of input from the adapter to the network. The definitions of the bits in this word are shown in Appendix D.

M

Master-Slave Hierarchy

An adapter configured for a master-slave hierarchy exchanges data with the master device. Usually, a network has one scanner which is the master device, and all other devices (for example, drives connected to EtherNet/IP adapters) are slave devices.

On a network with multiple scanners (called a multimaster hierarchy), each slave device must have a scanner specified as a master.

N

NVS (Non-Volatile Storage)

NVS is the permanent memory of a device. Devices such as the adapter and drive store parameters and other information in NVS so that they are not lost when the device loses power. NVS is sometimes called “EEPROM.”

P

PCCC (Programmable Controller Communications Command)

PCCC is the protocol used by some controllers to communicate with devices on a network. Some software products (for example, DriveExplorer and DriveExecutive) also use PCCC to communicate.

Peer-to-Peer Hierarchy

An adapter that is configured for a peer-to-peer hierarchy can exchange data with a device on the network that is not a scanner. This type of

hierarchy can be set up so that a scanner configures or transmits data to one PowerFlex 750-Series drive which then sends the same configuration or data to other PowerFlex 750-Series drives on the network. To use a peer-to-peer hierarchy, you configure one adapter to transmit data and one or more adapters to receive the data.

Ping

A message that is sent on the network to determine if a node exists.

PowerFlex 750-Series (Architecture Class) Drives

The Allen-Bradley PowerFlex 750-Series drives are part of the PowerFlex 7-Class family of drives.

R

Reference/Feedback

The Reference is used to send a setpoint (for example, speed, frequency, torque) to the drive. It consists of one 32-bit word of output to the adapter from the network.

Feedback is used to monitor the speed of the drive. It consists of one 32-bit word of input from the adapter to the network.

RSLogix 5/500/5000

RSLogix software is a tool for configuring and monitoring controllers to communicate with connected devices. It is a 32-bit application that runs on various Windows operating systems. Information about RSLogix software can be found at http://www.software.rockwell.com/rslogix.

S

Scanner

A scanner is a separate module (of a multi-module controller) or a built-in component (of a single-module controller) that provides communication with adapters connected to a network. See also Controller.

Status Indicators

Status indicators are LEDs that are used to report the status of the adapter, network, and drive. They are on the adapter and can be viewed on the front cover of the drive when the drive is powered.

Subnet Mask

An extension to the IP addressing scheme that lets you use a single network ID for multiple physical networks. A bit mask identifies the part of the address that specifies the network and the part of the address that specifies the unique node on the network. A “1” in the subnet mask indicates the bit is used to specify the network. A “0” in the subnet mask indicates that the bit is used to specify the node.

For example, a subnet mask on a network may appear as follows: 11111111 11111111 11111111 11000000 (255.255.255.192). This

mask indicates that 26 bits are used to identify the network and 6 bits are used to identify devices on each network. Instead of a single physical Class C network with 254 devices, this subnet mask divides it into four networks with up to 62 devices each.

Switches

Network devices that provide virtual connections that help to control collisions and reduce traffic on the network. They are able to reduce network congestion by transmitting packets to an individual port only if they are destined for the connected device. In a control application, in which real time data access is critical, network switches may be required in place of hubs.

T

TCP (Transmission Control Protocol)

EtherNet/IP uses this protocol to transfer Explicit Messaging packets using IP. TCP guarantees delivery of data through the use of retries.

U

UDP (User Datagram Protocol)

EtherNet/IP uses this protocol to transfer I/O packets using IP. UDP provides a simple, but fast capability to send I/O messaging packets between devices. This protocol ensures that adapters transmit the most recent data because it does not use acknowledgements or retries.

UDDT (User-Defined Data Type)

A structure data type that you define during the development of an application (for example, to convert 32-bit REAL parameter data to correctly write and read their values).

Z

Zero Data

When communications are disrupted (for example, a cable is disconnected), the adapter and drive can respond with zero data. Zero data results in the drive receiving zero as values for Logic Command, Reference, and Datalink data. If the drive was running and using the Reference from the adapter, it will stay running but at zero Reference.

Notes:

F

G

H

I

S

T

U

V

W

Z

Notes:

Notes:

U.S. Allen-Bradley Drives Technical Support - Tel: (1) 262.512.8176, Fax: (1) 262.512.2222, Email: support@drives.ra.rockwell.com, Online: www.ab.com/support/abdrives

www.rockwellautomation.com

Power, Control and Information Solutions Headquarters

Americas: Rockwell Automation, 1201 South Second Street, Milwaukee, WI 53204-2496 USA, Tel: (1) 414.382.2000, Fax: (1) 414.382.4444
Europe/Middle East/Africa: Rockwell Automation, Vorstlaan/Boulevard du Souverain 36, 1170 Brussels, Belgium, Tel: (32) 2 663 0600, Fax: (32) 2 663 0640
Asia Pacific: Rockwell Automation, Level 14, Core F, Cyberport 3, 100 Cyberport Road, Hong Kong, Tel: (852) 2887 4788, Fax: (852) 2508 1846

Publication 750COM-UM001A-EN-P – January, 2009