User Manual

Allen-Bradley

PowerFlex 525 Embedded EtherNet/IP Adapter

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.

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.
	**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 consequence.
	**SHOCK HAZARD:** Labels may be on or inside the equipment, for example, a drive or motor, to alert people that dangerous voltage may be present.
	**BURN HAZARD:** Labels may be on or inside the equipment, for example, a drive or motor, to alert people that surfaces may reach dangerous temperatures.
IMPORTANT	Identifies information that is critical for successful application and understanding of the product.

This manual contains new and updated information.

New and Updated Information

This table contains the changes made to this revision.

Topic	Page
Parameter Groups N and M added.	24
Configuring a Master-Slave Hierarchy updated.	31
Enabling Datalinks to Write Data updated.	32
Enabling Datalinks to Read Data updated.	32
Changing the Fault Action table updated.	33
Setting the Fault Configuration Parameters table updated.	33
Description of Drive Rating setting of the Module Definition window updated.	43
Tip for Using ADC with RSLogix 5000/Logix Designer updated.	49
Version of RSLogix 5000 for use with drive generic profile updated.	55
Settings for Connection Parameters table updated.	56
Input Image Controller Tag graphic updated.	58
Output Image Controller Tag graphic updated.	58
I/O Image for PowerFlex 525 Drives table updated.	62
CompactLogix Controller for Drive Generic Profile Ladder Logic Program Example graphic updated.	66
CompactLogix Example Ladder Logic Program Using a Drive Generic Profile for Logic Command/Reference diagram updated.	68
Important table for CompactLogix Examples updated.	71
Example Ladder Logic to Read a Single Parameter diagram updated.	71
Fifth footnote of Scattered Read Message Configuration Screens updated.	76
Tag Tab entry for Scattered Write Multiple Messages Configuration Screens table updated.	78
Drive 0 Control Routine diagram updated.	102
Extra table under Instances for Identity Object removed.	126
Description of Device Type entry in Instances Attribute table for Identity Object updated.	127
Instance Attribute table for Parameter Object updated.	133
Instances table for DPI Device Object updated.	138
Instances table for DPI Parameter Object updated.	141
Instances table for DPI Fault Object updated.	147

Notes:

Appendix D

Logic Command/Status Words: PowerFlex 525 Drives

Glossary

Notes:

Overview

Recommended Documentation

All the recommended documentation listed in this section is available online at http://www.rockwellautomation.com/literature.

The following publications provide additional information:

For…	See…	Publication
EtherNet/IP™	EtherNet/IP Media Planning and Installation Manual(1)	ODVA Pub. 148
	EtherNet/IP Network Infrastructure Guidelines(1)	ODVA Pub. 35
	EtherNet/IP Network Configuration User Manual	ENET-UM001
	Troubleshoot EtherNet/IP Networks	ENET-AT003
	EtherNet/IP Design, Commissioning, and Troubleshooting Quick Reference Drawings	IASIMP-QR023
	Ethernet Design Considerations Reference Manual	ENET-RM002
PowerFlex®520-Series Drives	PowerFlex 525 Adjustable Frequency AC Drive User Manual	520-UM001
HIM (Human Interface Module)	PowerFlex 4-Class HIM (DSI) Quick Reference	22HIM-QR001
RSLinx® Classic	RSLinx Classic Getting Results Guide(2)	LINX-GR001
RSLogix™ 5000	RSLogix 5000 online help(2)	–
CompactLogix™ 5370	CompactLogix 5370 Controllers User Manual (1769-L36ERM)	1769-UM021
MicroLogix™ 1100	MicroLogix 1100 Programmable Controllers User Manual	1763-UM001
MicroLogix™ 1400	MicroLogix 1400 Programmable Controllers User Manual	1766-UM001

(1) For ODVA publications, see the ODVA Ethernet/IP library at http://odva.org/Home/ODVATECHNOLOGIES/EtherNetIP/EtherNetIPLibrary/tabid/76/lng/en-US/Default.aspx

(2) The online help is installed with the software.

Manual Conventions

This manual provides information about the EtherNet/IP adapter embedded on the Main Control Board in PowerFlex 525 drives, and using it for network communication.

The following conventions are used throughout this manual:

- Parameter names are shown in the format axxx [*]. The a represents the parameter group. The xxx represents the parameter number. The * represents the parameter name— for example C141 [EN Rate Cfg].
- 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 Studio 5000™ Engineering and Design Environment combines engineering and design elements into a common environment. The first element in the Studio 5000 environment is the Logix Designer application. The Logix Designer application is the rebranding of RSLogix 5000 software and will continue to be the product to program Logix 5000 controllers for discrete, process, batch, motion, safety, and drive-based solutions. The Studio 5000 environment is the foundation for the future of Rockwell Automation engineering design tools and capabilities. It is the one place for design engineers to develop all the elements of their control system.
- RSLogix 5000 (version 20) was used for the screen captures in this manual. Different versions of the software may differ in appearance and procedures.

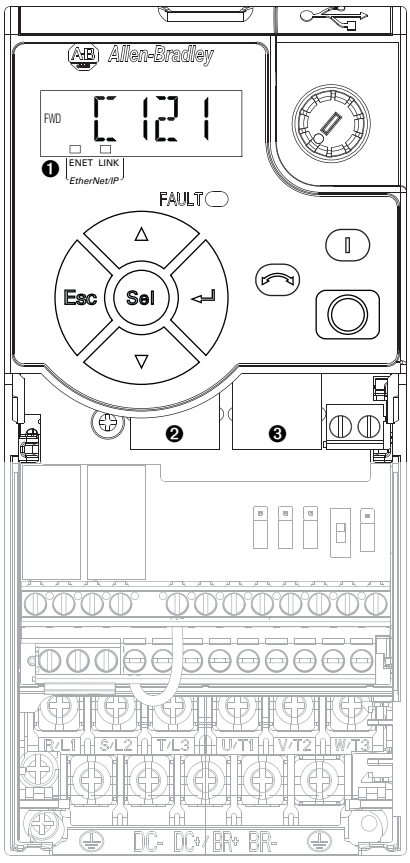
Getting Started

The EtherNet/IP adapter, embedded on the Main Control Board in PowerFlex 525 drives, is used for network communication.

Components

Components of the Embedded EtherNet/IP Adapter

Shown with control module cover removed



Item	Part	Description
❶	ENET status indicator	Two LEDs that indicate the status of the embedded EtherNet/IP adapter and network communications.
	LINK status indicator	
❷	Ethernet port	An RJ-45 connector for the Ethernet cable. It is CAT-5 compliant to ensure reliable data transfer on 100Base-Tx Ethernet connections.
❸	DSI port	An RS485 cable connection for handheld and remote options.

Features

The features of the embedded EtherNet/IP adapter include:

- Drive parameters to set an IP address – or you can use a BOOTP server to configure the IP address.
- Compatibility with various configuration tools to configure the embedded EtherNet/IP adapter and host drive. The tools include the PowerFlex 4/40-class HIM (Human Interface Module 22-HIM-A3 or 22-HIM-C2S), and drive-configuration software such as RSLogix 5000 (version 17 or greater), Logix Designer (version 21 or greater), and Connected Components Workbench (version 3 or greater).
- Status indicators on the drive's LCD display that report the status of the embedded EtherNet/IP adapter and network communications.
- Parameter-configured 16-bit Datalinks in the I/O to meet application requirements (four Datalinks to write data from the network to the drive, and four Datalinks to read data to the network from the drive).
- Explicit Messaging support.
- Master-Slave hierarchy that can be configured to transmit data to and from a controller on the network.
- Multi-drive mode which allows up to five drives to share a single EtherNet/IP node.
- User-defined fault actions to determine how the embedded EtherNet/IP adapter and its host PowerFlex 525 drive respond to:
  - I/O messaging communication disruptions (Comm Flt Action)
  - Controllers in idle mode (Idle Flt Action)
- Automatic Device Configuration (ADC) is an RSLogix 5000 (version 20 or greater), and Logix Designer (version 21 or greater) feature that supports the automatic download of configuration data upon the Logix controller establishing an EtherNet/IP network connection to a PowerFlex 525 drive and its associated peripherals.

Compatible Products

At the time of publication, the embedded EtherNet/IP adapter is compatible with Allen-Bradley PowerFlex 525 drives.

Required Equipment

Equipment Shipped with the Drive

Since the EtherNet/IP adapter is embedded on the Main Control Board in the PowerFlex 525 drive, it is always an integral part of the drive and, therefore, is not shipped with installation instructions.

User-Supplied Equipment

The embedded EtherNet/IP adapter can be configured using the built-in keypad interface on the drive. In addition, you must supply:

❑	Ethernet cable (see the EtherNet/IP Media Planning and Installation Manual, ODVA publication 148 available on the ODVA web site at http://odva.org/Home/ODVATECHNOLOGIES/EtherNetIP/EtherNetIPLibrary/tabid/76/Default.aspx for details)	
❑	Ethernet switch (see the Ethernet Design Considerations Reference Manual, Rockwell Automation publication ENET-RM002 for details)	
❑	Optional configuration tool, such as:	
	–	PowerFlex 22-HIM-A3/-C2S HIM
	–	DHCP/BOOTP Utilities
❑	Controller configuration software, such as:	
	–	RSLinx Classic (version 2.50 or later)
	–	RSLogix 5000 (version 17 or greater) or Logix Designer (version 21 or greater) when using drive-specific Add-On Profile (AOP)
	–	Connected Components Workbench (version 3 or greater)
❑	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, see 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. Drive parameters **C143 [EN Comm Flt Actn] and C144 [EN Idle Flt Actn]** let you determine the action of the embedded EtherNet/IP adapter and drive if I/O communication is disrupted, the controller is idle, or explicit messaging for drive control is disrupted. By default, these parameters fault the drive. You may configure these parameters so that the drive continues to run, however, 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

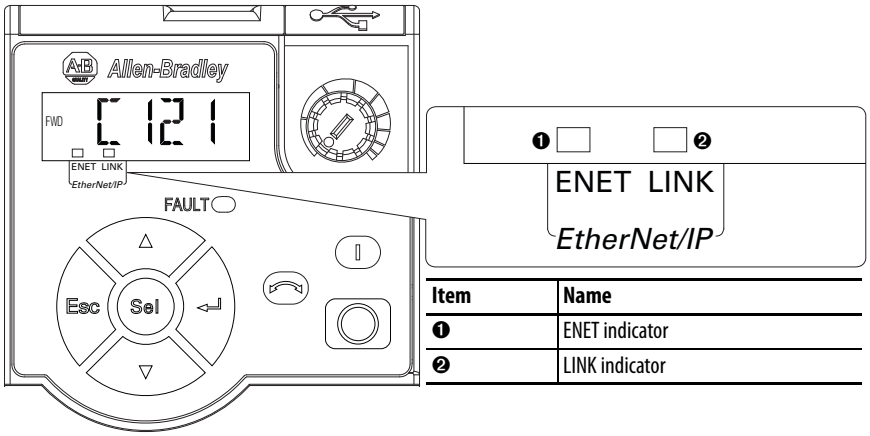
This section is provided to help experienced users quickly start using the embedded EtherNet/IP adapter. If you are unsure how to complete a step, refer to the referenced chapter.

Step	Action	See...
1	**Review the safety precautions for the adapter.**	Throughout this manual
2	**Verify that the PowerFlex drive is properly installed.**	PowerFlex 525 Adjustable Frequency AC Drive User Manual (publication 520-UM001)
3	**Connect the drive to the EtherNet/IP network.** Verify that the PowerFlex drive is not powered. Then, connect the embedded EtherNet/IP adapter to the network using an Ethernet cable.	Chapter 2, Setting Up the Adapter

Step	Action	See...
4	**Apply power to the drive.** a. Replace the control module cover. b. The embedded EtherNet/IP adapter receives power from the drive. Apply power to the drive. The ENET/LINK status indicators on the drive's LCD display should light up and remain steady or flash. If the drive's Fault LED lights up, there is a problem. See Chapter 8, Troubleshooting. c. Configure/verify key drive parameters.	Chapter 2, Setting Up the Adapter
5	**Configure the adapter for your application.** Set drive parameters for the following functions as required by your application: – IP address, subnet mask, and gateway address – Data rate – I/O configuration – Master-Slave hierarchy – Fault actions	Chapter 3, Configuring the Adapter
6	**Configure the controller to communicate with the adapter.** Use a controller configuration tool such as RSLogix 5000/Logix Designer 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 5000/Logix Designer 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.



Item	Name
❶	ENET indicator
❷	LINK indicator

After connecting the embedded EtherNet/IP adapter to the network and applying power to the drive, see Startup Status Indication on page 19 for possible start-up status indications and their descriptions.

Notes:

Setting Up the Adapter

Since the EtherNet/IP adapter is embedded on the Main Control Board in the PowerFlex 525 drive, the only required step is setting its IP address and connecting it to the network.

Preparing for Set-Up

Before setting up the embedded EtherNet/IP adapter:

- Make sure the Ethernet switch is the correct type. A "managed" switch that supports IGMP snooping is usually recommended. An "unmanaged" switch can be used instead if RSLogix 5000 (version 18 or greater) is used and all devices on the network are configured for "unicast" I/O. For more details, see the following documents:
  - EtherNet/IP Media Planning and Installation Manual (ODVA publication 148)
  - EtherNet/IP Network Infrastructure Guidelines (ODVA publication 35)
  - Ethernet Design Considerations Reference Manual (Rockwell Automation publication ENET-RM002)
- Understand IGMP Snooping/Ethernet Switches

  The embedded EtherNet/IP adapter is a multicast device. In most situations, an IGMP snooping (managed) switch is required. If more than one or two embedded EtherNet/IP adapters are connected to the switch, a managed switch is required—otherwise the drive may fault on a Net I/O Timeout network loss. The embedded EtherNet/IP adapter, RSLogix 5000 (version 18 or greater), and a ControlLogix or CompactLogix controller will support unicast. Unicast setup is required when adding the drive to the I/O. When all embedded EtherNet/IP adapters are set up as unicast devices, then an IGMP snooping (managed) switch is not needed.

  Much of EtherNet/IP implicit (I/O) messaging uses IP multicast 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 (managed switches) "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.

- See Appendix A for the number of CIP connections supported by the embedded EtherNet/IP adapter.
- Verify that you have all required equipment. See Required Equipment on page 12.

Setting the IP Address

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

- BOOTP Server – Use BOOTP if you prefer to control the IP addresses of devices using a BOOTP server. The IP address, subnet mask, and gateway addresses will then be provided by the BOOTP server. This is enabled by default.
- Parameters – Use 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 parameters you set.

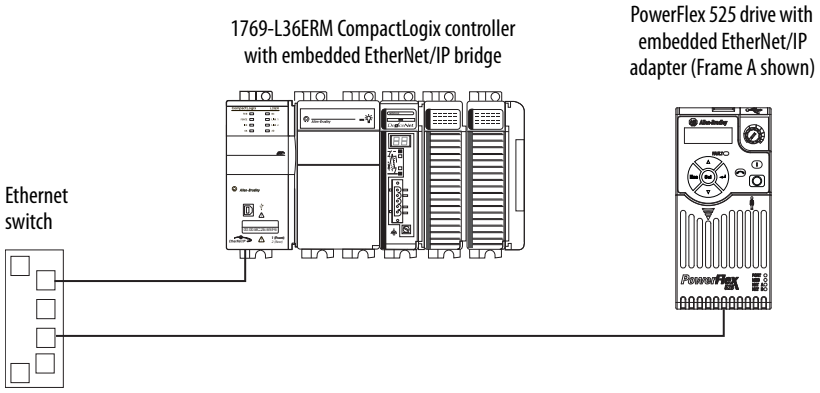
IMPORTANT Regardless of the method used to set the adapter's IP address, each node on the network must have a unique IP address. To change an IP address, you must set the new value and then remove and reapply power to (or reset) the drive.

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 control module cover.

3. Use static control precautions.
4. Connect one end of an Ethernet cable to the EtherNet/IP network.



5. Route the other end of the Ethernet cable through the bottom of the PowerFlex 525 drive, and insert the cable's plug into the embedded EtherNet/IP adapter's mating socket (item 2 in Components of the Embedded EtherNet/IP Adapter on page 11).

Applying Power

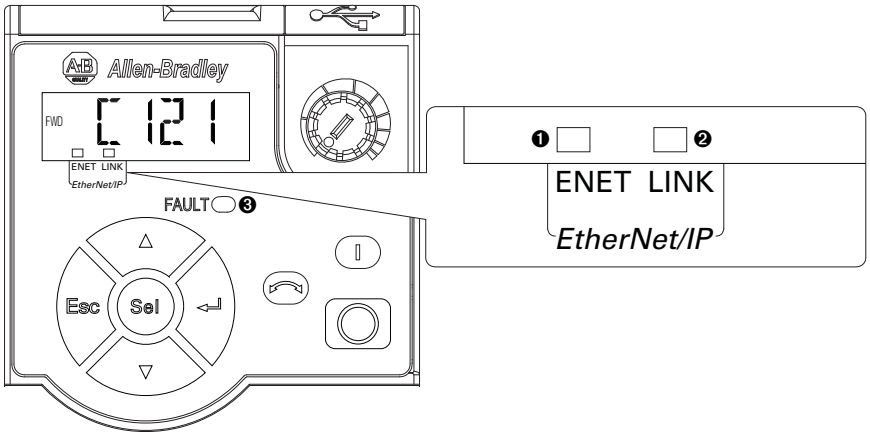
ATTENTION: Risk of equipment damage, injury, or death exists. Unpredictable operation may occur if you fail to verify that parameter settings are compatible with your application. Verify that settings are compatible with your application before applying power to the drive.

Install the control module 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, the "ENET" status indicator on the drive LCD display should remain steady after an initialization. If it does not light up or is flashing, there is a problem. See Chapter 8, Troubleshooting.

Startup Status Indication

After power has been applied, status indicators for the PowerFlex 525 drive and embedded EtherNet/IP adapter can be viewed on the drive LCD display. See Drive and Adapter Status Indicators on page 20 for their location and possible status indications.

Drive and Adapter Status Indicators



Item	Name	State	Description
❶	ENET	Off	Adapter is not connected to the network.
		Steady	Adapter is connected to the network and drive is controlled through Ethernet.
		Flashing	Adapter is connected to the network but drive is not controlled through Ethernet.
❷	LINK	Off	Adapter is not connected to the network
		Steady	Adapter is connected to the network but not transmitting data.
		Flashing	Adapter is connected to the network and transmitting data.
❸	FAULT	Flashing Red	Indicates drive is faulted.

For more details on status indicator operation see Understanding the Status Indicators on page 111.

Configuring/Verifying Key Drive Parameters

The PowerFlex 525 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.

Configuring the drive parameters can be done using the drive's keypad, a HIM, RSLogix 5000/Logix Designer, or Connected Components Workbench. In the following example, the drive will receive the Logic Command and Reference from the network.

1. Set the value of drive parameter **P046 [Start Source 1]** to 5 "EtherNet/IP".
2. Set the value of drive parameter **P047 [Speed Reference1]** to 15 "EtherNet/IP".

TIP The PowerFlex 525 drive supports up to three control functions and three Reference functions.

For more information on how to set different combinations of the control and Reference functions, see the PowerFlex 525 drive user manual, publication 520-UM001.

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. Use either a BOOTP server or 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 to accept an IP address from a BOOTP server. For details, see Using BOOTP on page 26. To set the IP address using parameters, see Setting the IP Address, Subnet Mask, and Gateway Address on page 29.

IMPORTANT New settings for some parameters (for example, parameters **C129 [EN IP Addr Cfg 1]** through **C132 [EN IP Addr Cfg 4]**) are recognized only when power is applied to the adapter. After you change parameter settings, cycle drive power.

Notes:

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, see Appendix B, Adapter Parameters. For definitions of terms in this chapter, see the Glossary.

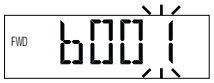
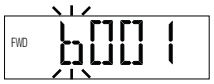
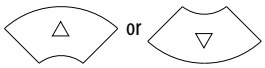
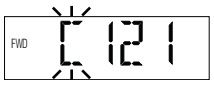
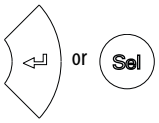
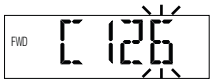
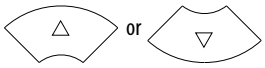
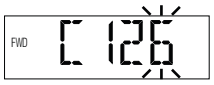
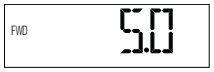
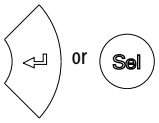
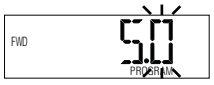
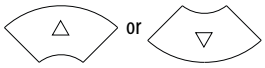
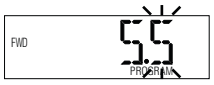
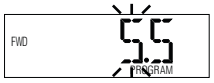
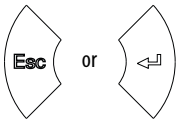
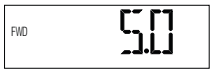
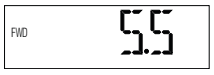
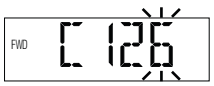
Configuration Tools

The parameters can be configured using the drive keypad interface (see page 24) or a PowerFlex 4-class HIM (Human Interface Module, see page 25).

RSLogix 5000 (version 17 or greater), Logix Designer (version 21 or greater), and Connected Components Workbench (version 3 or greater) can also be used to access the parameters.

Using the Drive Keypad Interface to Access Parameters

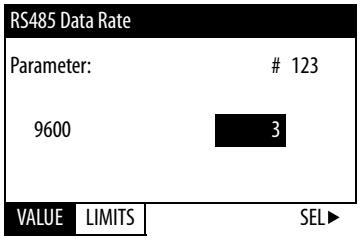
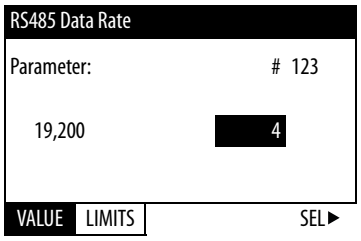
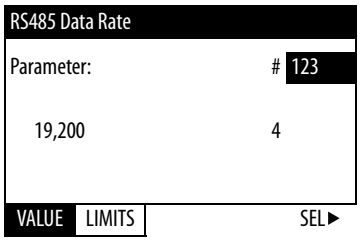
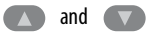
The following is an example of basic integral keypad and display functions. This example provides basic navigation instructions and illustrates how to program a parameter.

Step	Key(s)	Example Display
1. When power is applied, the last user-selected Basic Display Group parameter number is briefly displayed with flashing characters. The display then defaults to that parameter's current value. (Example shows the value of b001 [Output Freq] with the drive stopped.)		
2. Press Esc to display the Basic Display Group parameter number shown on power-up. The parameter number will flash.		
3. Press Esc to enter the parameter group list. The parameter group letter will flash.		
4. Press the Up Arrow or Down Arrow to scroll through the group list (b, P, t, C, L, d, A, f, N, M, and Gx).		
5. Press Enter or Sel to enter a group. The right digit of the last viewed parameter in that group will flash.		
6. Press the Up Arrow or Down Arrow to scroll through the parameter list.		
7. Press Enter to view the value of the parameter. Or Press Esc to return to the parameter list.		
8. Press Enter or Sel to enter Program Mode and edit the value. The right digit will flash and the word Program on the LCD display will light up.		
9. Press the Up Arrow or Down Arrow to change the parameter value.		
10. If desired, press Sel to move from digit to digit or bit to bit. The digit or bit that you can change will flash.		
11. Press Esc to cancel a change and exit Program Mode. Or Press Enter to save a change and exit Program Mode. The digit will stop flashing and the word Program on the LCD display will turn off.		 or 
12. Press Esc to return to the parameter list. Continue to press Esc to back out of the programming menu. If pressing Esc does not change the display, then b001 [Output Freq] is displayed. Press Enter or Sel to enter the group list again.		

Using the PowerFlex 4-Class HIM to Access Parameters

The PowerFlex 4-class HIM can be used to access parameters in the drive (see basic steps shown below). It is recommended that you read through the steps for your HIM before performing the sequence. For additional HIM information, refer to the HIM Quick Reference card, publication 22HIM-QR001.

Step	Key(s)	Example Display
1. Power up the drive. Then connect the HIM to the DSI port of the drive. The Parameters tab for the drive will be displayed.		Parameters Groups Linear List Changed Params DIAG | PARAM | DSEL | MEM | SEL▶
2. Select Groups in the Parameters tab if it is not already selected using the Up Arrow or Down Arrow. Press Enter to select Groups.	▲ and ▼ Enter	Basic Display 001 Output Freq 002 Commanded Freq 003 Output Current 004 Output Voltage 005 DC Bus Voltage B | P | T | C | SEL▶
3. Press Sel repeatedly until the C tab (Communications parameters) is selected.	Sel	Communications 121 Comm Write Mode 122 Cmd Stat Select 123 RS485 Data Rate 124 RS485 Node Addr 125 Comm Loss Action B | P | T | C | SEL▶
4. Press the Up Arrow or Down Arrow to scroll to the Communications parameter you wish to display or modify, then press Enter.	▲ and ▼ Enter	Communications 121 Comm Write Mode 122 Cmd Stat Select 123 RS485 Data Rate 124 RS485 Node Addr 125 Comm Loss Action B | P | T | C | SEL▶ RS485 Data Rate Parameter: # 123 9600 3 VALUE | LIMITS SEL▶

Step	Key(s)	Example Display
5. Press Enter to select the current parameter value and the numeric keys to enter a new value.		 
6 Press Enter to write the new value to the parameter and the parameter number will be selected again.		
7 While the parameter number is selected, you may: a. use the arrow keys to select additional parameters you wish to modify. b. press Enter to go to the parameter number you have input using the keypad. c. press Esc to return to the group parameter list.	  	

Using BOOTP

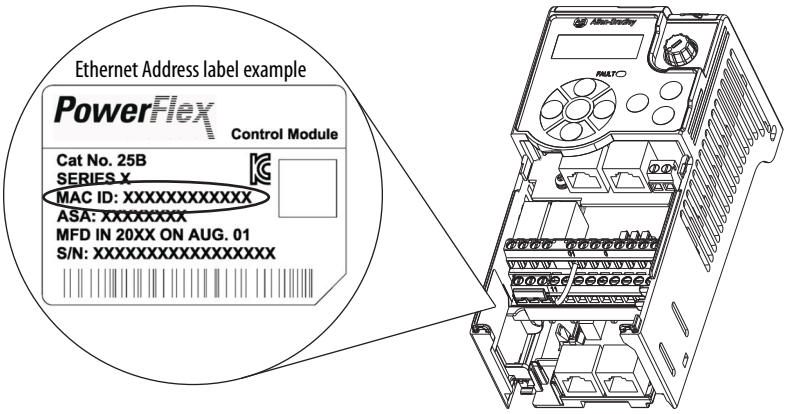
By default, the adapter is configured to accept an IP address, subnet mask, and gateway address from a BOOTP server. You can select from a variety of BOOTP utilities.

The instructions below use the DHCP/BOOTP Utility (version 2.3 or greater), a free stand-alone program from Rockwell Automation that incorporates the functionality of standard DHCP/BOOTP utilities with a graphical interface. It is available from http://www.ab.com/networks/ethernet/bootp.html. See the Readme file and online Help for directions and more information.

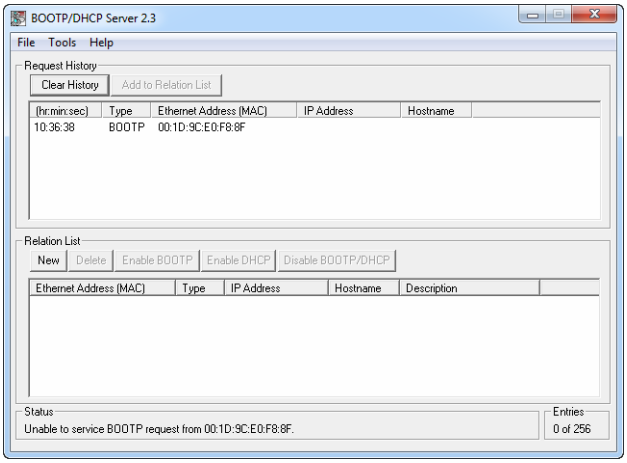
TIP If desired, you can disable BOOTP and configure the IP address, subnet mask, and gateway address using parameters. For details, see Setting the IP Address, Subnet Mask, and Gateway Address on page 29.

Configuring the Adapter Using DHCP/BOOTP Utility

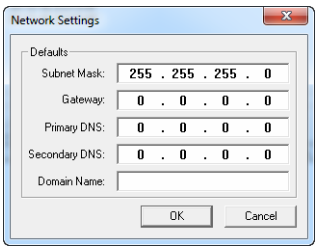
1. Verify and note the adapter's hardware Ethernet Address (MAC), which will be used in Step 6. There are two ways to do this:
   - Use the PowerFlex 525 drive's keypad or a HIM to access the diagnostic parameters of the drive. Scroll to parameters **F687 [HW Addr 1]** through **F692 [HW Addr 6]** to view the adapter's hardware Ethernet Address (MAC). Finally, convert these decimal values to a hex value.
   - Remove the PowerFlex 525 control module front cover and locate the adapter's hardware Ethernet Address (MAC) label.



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



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



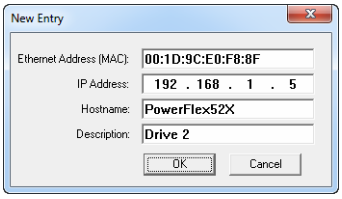
4. Edit the following:

Box	Type
Subnet Mask(1)	The subnet mask for the 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, see the Glossary.

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

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

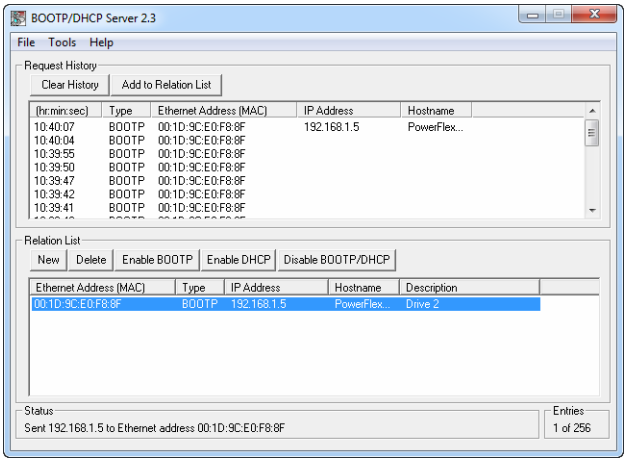


7. Edit the following:

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

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

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



9. To assign this configuration to the adapter, select the device in the Relation List and click **Disable BOOTP/DHCP**. When power is cycled on the drive, the adapter will use the configuration you assigned it and not issue new BOOTP/DHCP 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 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 accept an IP address, subnet mask, and gateway address from a BOOTP server. If you want to set these attributes using parameters instead, you must first disable BOOTP and then set these network address parameters in the drive.

Disabling the BOOTP Feature

1. Set the value of parameter **C128 [EN Addr Sel]** to 1"Parameters".

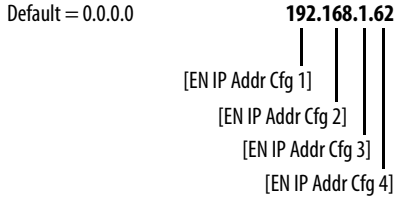
Options	1	"Parameters"
	2	"BOOTP" (Default)

2. Reset the adapter by power cycling the drive.

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

Setting an IP Address Using Parameters

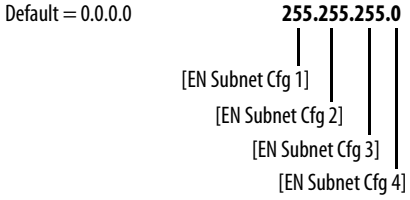
1. Verify that parameter **C128 [EN Addr Sel]** is set to 1"Parameters". This parameter must be set to "Parameters" to configure the IP address using parameters.
2. Set the value of parameters **C129 [EN IP Addr Cfg 1]** through **C132 [EN IP Addr Cfg 4]** to a unique IP address.



3. Reset the adapter by power cycling the drive.

Setting a Subnet Mask Using Parameters

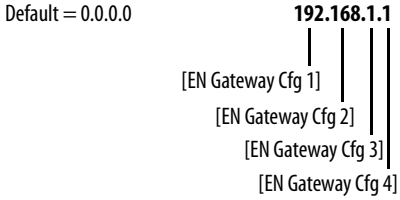
1. Verify that parameter **C128 [EN EN Addr Sel]** is set to 1"Parameters". This parameter must be set to "Parameters" to configure the subnet mask using parameters.
2. Set the value of parameters **C133 [EN Subnet Cfg 1]** through **C136 [EN Subnet Cfg 4]** to the desired value for the subnet mask.



3. Reset the adapter by power cycling the drive.

Setting a Gateway Address Using Parameters

1. Verify that parameter **C128 [EN EN Addr Sel]** is set to 1"Parameters". This parameter must be set to "Parameters" to configure the gateway address using parameters.
2. Set the value of parameters **C137 [EN Gateway Cfg 1]** through **C140 [EN Gateway Cfg 4]** to the desired value for the gateway address.



3. Reset the adapter by power cycling the drive.

Setting the Data Rate

By default, the adapter automatically detects the data (baud) rate and duplex setting used on the network. If you need to set a specific data rate and duplex setting, the value of parameter **C141 [EN Rate Cfg]** determines the Ethernet data rate and duplex setting that the adapter will use to communicate. For definitions of data rate and duplex, see the Glossary.

1. Set the value of parameter **C141 [EN Rate Cfg]** to the data rate at which your network is operating.

Options	0	"Auto detect" (Default)
	1	"10Mbps Full"
	2	"10Mbps Half"
	3	"100Mbps Full"
	4	"100Mbps Half"

TIP Auto detection of data 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 data rate/duplex. If one device has the data rate/duplex hard-coded, the other device must be hard-coded to the same settings.

2. Reset the adapter by power cycling the drive.

Using Master-Slave Hierarchy

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 scanner or bridge.

For a Master-Slave hierarchy, the devices exchanging data must be on the same IP subnet. See "IP Addresses" in the Glossary for information about IP subnets.

Configuring a Master-Slave Hierarchy

The controller I/O image can have anywhere from zero to eight (four In and four Out) additional 16-bit parameters called Datalinks. They are configured using parameters **C153 [EN Data In 1]** through **C156 [EN Data In 4]**, and **C157 [EN Data Out 1]** through **C160 [EN Data Out 4]**. The number of Datalinks actively used is controlled by the connection size in the controller and the in/out parameters. See the respective controller example sections in Chapter 4 for more information on setting the connection size.

When using a ControlLogix or CompactLogix controller and the Generic Profile, or a MicroLogix 1100/1400 controller, configure the Datalink parameters now as described in this section.

TIP When using a ControlLogix or CompactLogix controller and an RSLogix 5000 (version 17 or greater) or Logix Designer (version 21 or greater) drive Add-On Profile, there is no need to configure Datalink parameters at this time. They will be assigned when configuring the drive Add-On Profile (see Adding the Drive to the I/O Configuration on page 40).

Enabling Datalinks To Write Data

IMPORTANT Always use the Datalink parameters in consecutive numerical order, starting with the first parameter. For example, use parameters C153, C154, and C155 to configure three Datalinks to write data. Otherwise, the network I/O connection will be larger than necessary, which needlessly increases controller response time and memory usage.

Parameters **C153 [EN Data In 1]** through **C156 [EN Data In 4]** control which parameters in the drive, adapter, or any other connected peripheral receive values from the network. To configure these parameters, set them to the drive parameter number you want to write them to.

The following steps are required to enable Datalinks to write data:

1. Set the values of only the required number of contiguous controller-to-drive Datalinks 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.

After the above steps are complete, the adapter is ready to receive input data and transfer status data to the master (controller). Next, configure the controller to recognize and transmit I/O to the adapter. See Chapter 4, Configuring the I/O.

Enabling Datalinks To Read Data

IMPORTANT Always use the Datalink parameters in consecutive numerical order, starting with the first parameter. For example, use parameters C157, C158, and C159 to configure three Datalinks to read data. Otherwise, the network I/O connection will be larger than necessary, which needlessly increases controller response time and memory usage.

Parameters **C157 [EN Data Out 1]** through **C160 [EN Data Out 4]** configure which parameters in the drive, adapter, or any other connected peripheral send values to the network. To configure these parameters, set them to the parameter number you wish to read over the network.

The following steps are required to enable Datalinks to read data:

1. Set the values of only the required number of contiguous drive-to-controller Datalinks 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.

After the above steps are complete, the adapter is ready to send output data to the master (controller). Next, configure the controller to recognize and transmit I/O to the adapter. See Chapter 4, Configuring the I/O.

Setting a Fault Action

By default, when communications are disrupted (the network cable is disconnected) and/or the controller is idle (in program mode or faulted), the drive responds by faulting if it is using I/O from the network. You can configure a different response to:

- Disrupted I/O communication by using parameter **C143 [EN Comm Flt Actn]**.
- An idle controller by using parameter **C144 [EN Idle Flt Actn]**.

ATTENTION: Risk of injury or equipment damage exists. Drive parameters C143 [EN Comm Flt Actn] and C144 [EN Idle Flt Actn] respectively let you determine the action of the embedded EtherNet/IP adapter and drive if communications are disrupted or the controller is idle. By default, these parameters fault the drive. You may configure these parameters so that the drive continues to run, however, 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 (a disconnected network cable or controller in idle state).

Changing the Fault Action

Set the values of parameters **C143 [EN Comm Flt Actn]** and **C144 [EN Idle Flt Actn]** to the desired responses:

Value	Action	Description
0	Fault	The drive is faulted and stopped (Default).
1	Stop	The drive is stopped as per P045 [Stop Mode] setting.
2	Zero Data	The Reference and Datalink values transmitted to the drive will be set to "0".
3	Hold Last	The Reference and Datalink values transmitted to the drive will be held at their last value.
4	Send Flt Cfg	The Logic, Reference, and Datalink values will be transmitted to the drive as configured in C145 [EN Flt Cfg Logic], C146 [EN Flt Cfg Ref], and C147 [EN Flt Cfg DL 1] through C150 [EN Flt Cfg DL 4].

Changes to these parameters take effect immediately. A reset is not required. If communication is disrupted and then re-established, the drive will automatically receive commands over the network again.

Setting the Fault Configuration Parameters

When setting parameter **C143 [EN Comm Flt Actn]** and **C144 [EN Idle Flt Actn]** to 4 "Send Flt Cfg," the values in the following parameters are sent to the drive after a communications fault and/or idle fault for drive control fault occurs. You must set these parameters to values required by your application. Changes to these parameters take effect immediately. A reset is not required.

Parameter	Description
C145 [EN Flt Cfg Logic]	A 16-bit integer value sent to the drive for Logic Command.
C146 [EN Flt Cfg Ref]	A 16-bit integer value sent to the drive for Reference.
C147 [EN Flt Cfg DL 1] through C150 [EN Flt Cfg DL 4]	A 16-bit integer value sent to the drive for a Datalink.

Resetting the Adapter

Changes to some parameters require that you reset the adapter before the new settings take effect. You can reset the adapter by cycling power to the drive.

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.

Viewing the Adapter Status Using Parameters

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

Embedded EtherNet/IP Adapter Diagnostic Parameters

Name	Description
F681 [Comm Sts - DSI]	Displays the status of the RS485 serial (DSI) port to the drive.
F682 [Comm Sts - Opt]	Displays the status of the internal communication to the drive.
F683 [Com Sts-Emb Enet]	Displays the status of the embedded EtherNet/IP interface to the drive.
F684 [EN Addr Src]	Displays the network configuration source currently used by the embedded EtherNet/IP interface.
F685 [EN Rate Act]	Displays the network data rate currently used by the embedded EtherNet/IP interface.
F686 [DSI I/O Act]	Displays the drives that are active in Multi-drive mode.
F687 [HW Addr 1] through F692 [HW Addr 6]	Decimal value of each byte in the adapter's Ethernet hardware address.
F693 [EN IP Addr Act 1] through F696 [EN IP Addr Act 4]	Value of each byte in the adapter's present IP address. A value of "0" appears if the adapter does not currently have an IP address.
F697 [EN Subnet Act 1] through F700 [EN Subnet Act 4]	Value of each byte in the adapter's present subnet mask. A value of "0" appears if the adapter does not currently have a subnet mask.
F701 [EN Gateway Act 1] through F704 [EN Gateway Act 4]	Value of each byte in the adapter's present gateway address. A value of "0" appears if the adapter does not currently have a gateway address.
F705 [Drive 0 Logic Cmd] F709 [Drive 1 Logic Cmd] F713 [Drive 2 Logic Cmd] F717 [Drive 3 Logic Cmd] F721 [Drive 4 Logic Cmd]	In Multi-drive mode, this is the logic command being transmitted to drive 0. In Single-drive mode, this is the logic command being used by the drive (whether HS-DSI, EtherNet/IP, or DSI) at the time. If comms control is NOT being used, and the drive is in single-drive mode, then this parameter will show 0.
F706 [Drv 0 Reference] F710 [Drv 1 Reference] F714 [Drv 2 Reference] F718 [Drv 3 Reference] F722 [Drv 4 Reference]	In Multi-drive mode, this is the reference being transmitted to drive 0/1/2/3/4. In Single-drive mode, this is the reference being used by the drive (whether HS-DSI, EtherNet/IP, or DSI) at the time. If comms control is NOT being used, and the drive is in single-drive mode, then this parameter will show 0.
F707 [Drv 0 Logic Sts] F711 [Drv 1 Logic Sts] F715 [Drv 2 Logic Sts] F719 [Drv 3 Logic Sts] F723 [Drv 4 Logic Sts]	In Multi-drive mode, this is the logic status being received from drive 0/1/2/3/4. In Single-drive mode, this is the logic status of the drive at the time.
F708 [Drv 0 Feedback] F712 [Drv 1 Feedback] F716 [Drv 2 Feedback] F720 [Drv 3 Feedback] F724 [Drv 4 Feedback]	In Multi-drive mode, this is the feedback being received from drive 0/1/2/3/4. In Single-drive mode, this is the feedback of the drive at the time.
F725 [EN Rx Overruns]	A count of the number of receive overrun errors reported by the embedded EtherNet/IP interface.
F726 [EN Rx Packets]	A count of the number of receive packets reported by the embedded EtherNet/IP interface.
F727 [EN Rx Errors]	A count of the number of receive errors reported by the embedded EtherNet/IP interface.

Embedded EtherNet/IP Adapter Diagnostic Parameters

Name	Description
F728 [EN Tx Packets]	A count of the number of transmitted packets reported by the embedded EtherNet/IP interface.
F729 [EN Tx Errors]	A count of the number of transmit errors reported by the embedded EtherNet/IP interface.
F730 [EN Missed IO Pkt]	The number of I/O packets missed.
F731 [DSI Errors]	The number of total DSI errors.

Notes:

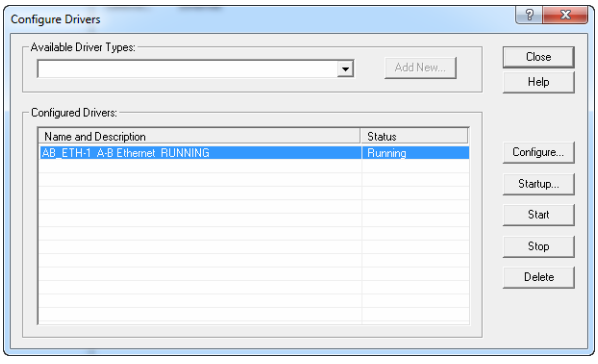
Configuring the I/O

This chapter provides instructions on how to configure a CompactLogix controller to communicate with the embedded EtherNet/IP adapter in the PowerFlex 525 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 window.
2. From the Available Driver Types pull-down box, choose "EtherNet/IP Driver" and then click **Add New...** to display the Add New RSLinx Driver window.
3. Use the default name or type a name and click **OK**. The "Configure driver:" window appears.
4. Depending on your application, select either the browse local or remote subnet option, and click **OK**. The Configure Drivers window reappears with the new driver in the Configured Drivers list.

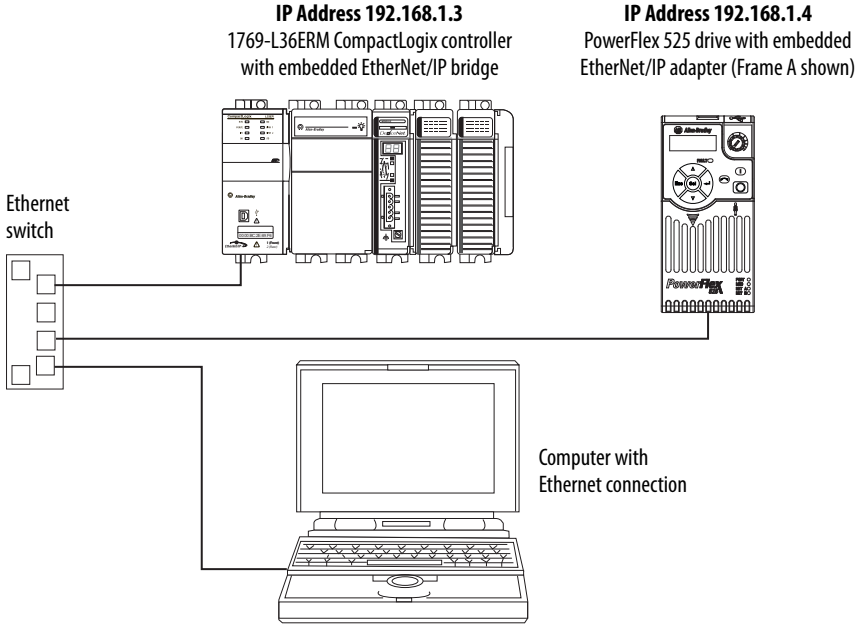


5. Click **Close** to close the Configure Drivers window. Keep RSLinx running.

6. Verify that your computer recognizes the drive. Select **Communications** > **RSWho** and, in the menu tree, click the "+" symbol next to the Ethernet driver.
7. Note that two other RSLinx drivers (Ethernet devices or Remote Devices through Linx Gateway) may be used. Use one of these drivers if the "EtherNet/IP Driver" cannot see your drive.

CompactLogix Example

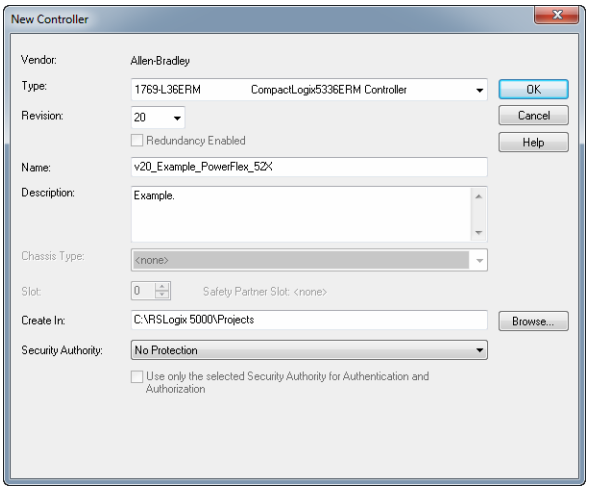
After the adapter is configured, the drive and adapter will be a single node on the network. This section provides the steps needed to configure a simple EtherNet/IP network. In our example, we will configure a 1769-L36ERM CompactLogix controller with embedded EtherNet/IP capability to communicate with a drive using Logic Command/Status, Reference/Feedback, and eight Datalinks (four to read and four to write) over the network.



Adding the Controller to the I/O Configuration

To establish communications between the controller and drive over the network, you must first add the CompactLogix controller and its embedded EtherNet/IP bridge to the I/O configuration.

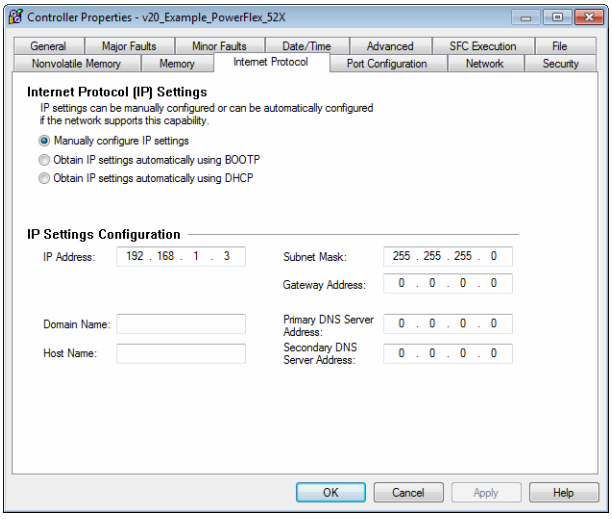
1. Start RSLogix 5000/Logix Designer. The application window appears. Select **File** > **New** to display the New Controller window.



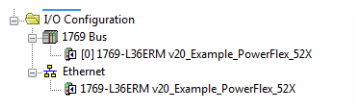
Select the appropriate choices for the fields in the window to match your application. Then click **OK**. The application window reappears with the treeview in the left pane.

Note: If you are using a controller without an embedded EtherNet/IP bridge, you will also need to add the bridge to the I/O configuration. See the user manual for your controller for details.

2. Configure the IP address/Network Settings on your controller or bridge. In this example, the Network Settings are set for a private network.



3. Click **OK**. The controller is now configured for the EtherNet/IP network. It appears in the I/O Configuration folder. In our example, a 1769-L36ERM controller appears under the I/O Configuration folder with its assigned name.



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

- Drive Add-On Profiles (RSLogix 5000 version 17 or greater, Logix Designer version 21 or greater)
- Generic Profile (RSLogix 5000 or Logix Designer, all versions)

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

Using Drive Add-On Profiles with RSLogix 5000 (version 17 or greater) or Logix Designer (version 21 or greater)

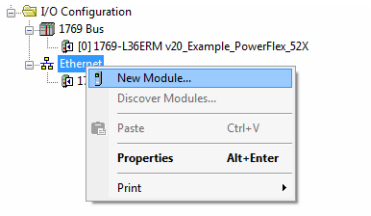
When compared to using the Generic Profile (all versions), the drive Add-On Profiles provide these advantages:

- 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/ Logix Designer project file (.ACD) and also downloaded to the controller.
- Unicast connection (RSLogix 5000 version 18 or greater, or Logix Designer version 21 or greater)
- Drive Add-On Profiles enable I/O to be added online while the controller is in the Run mode.
- Drive Add-On Profiles can be updated anytime. When a new drive is used or to benefit from new updates for Add-On Profiles, you will need the newest Add-On Profile update. Go to www.ab.com/support/abdrives/webupdate to download the latest RSLogix 5000/Logix Designer drive Add-On Profile.

Adding the Drive to the I/O Configuration

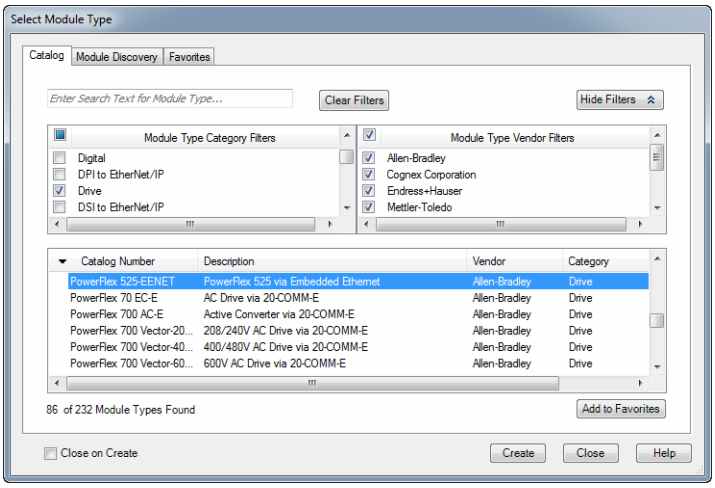
To transmit data between the controller and the drive, you must add the drive as a child device to the parent controller. In this example, RSLogix 5000 version 20 is used with drive Add-On Profile version 1.01.

1. In the treeview, right-click on the Ethernet icon and select **New Module...** to display the Select Module window. Expand the Drives group to display all of the available drives with their communication adapters.

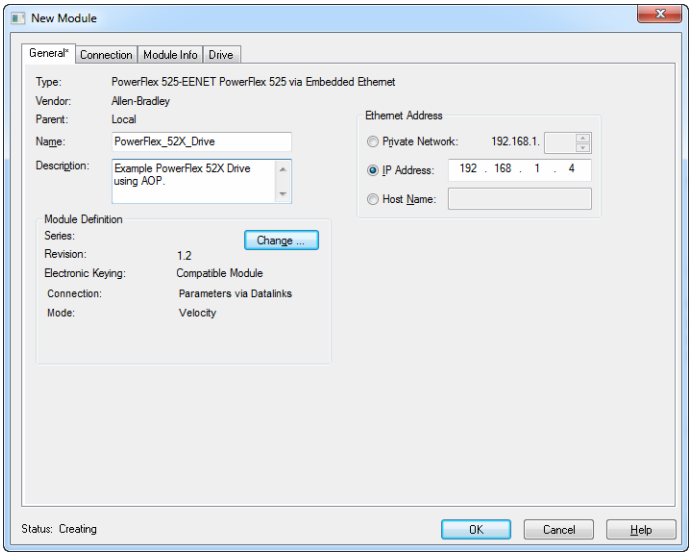


TIP If the PowerFlex drive is not shown, go to www.ab.com/support/abdrives/webupdate and download the latest drive Add-On Profile.

2. In the Select Module Type window, select the drive and its connected adapter from the list. For this example, we selected "PowerFlex 525-EENET." Then click **Create**. The drive's New Module window appears.

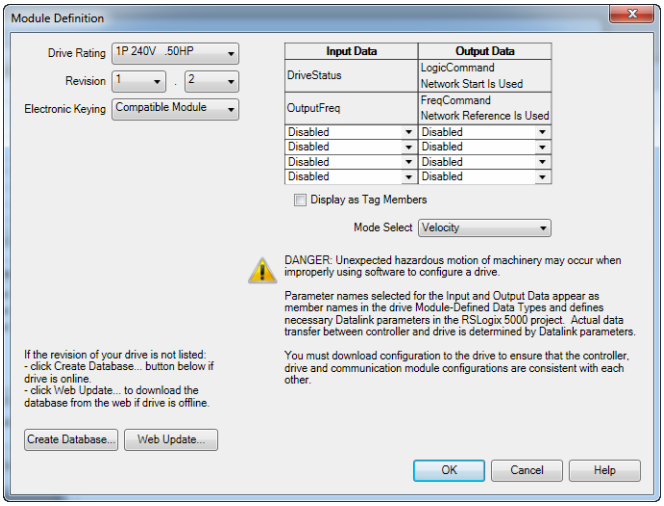


3. On the General tab, edit the following data about the drive:

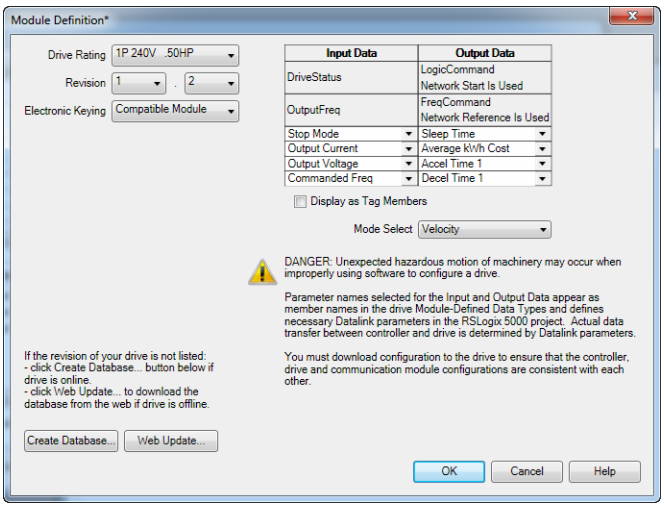


Box	Setting
Name	A name to identify the drive.
Description	Optional – description of the drive.
IP Address	The IP address of the adapter.

4. On the New Module window in the Module Definition section, click **Change...** to launch the Module Definition window and begin the drive configuration process.



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



TIP You may create a database from a network accessible drive using the **Create Database...** button (Recommended).

Box	Setting
Drive Rating	The voltage and horsepower rating of the drive. If the drive rating is not listed, the drive database is not installed on your computer. To get the drive rating, use the **Create Database…**, or **Web Update…** button described above.
Revision	The major and minor revision of the firmware (database) 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 revision, use one of the following buttons at the bottom left of the Module Definition window: • **Create Database…** Creates a database from an online network drive. Clicking this button displays an RSLinx RSWho window. Browse to the online drive (PowerFlex 525), select it, and click OK. The database will be uploaded and stored on the computer. Thereafter, close the Module Definition window and then re-open it to display the new revision. • **Web Update…** When a drive is not available online, opens the Allen-Bradley Drives Web Updates web site to download a specific database file. After downloading the file, close the Module Definition window and then re-open it to display the new revision.
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 window. See the online Help for additional information on this and other Electronic Keying settings. If keying is not required, select "Disable Keying." Drives do not require keying, and so "Disable Keying" is recommended. When using RSLogix 5000 (version 20) or Logix Designer (version 21 or greater) and Automatic Device Configuration (ADC) with Firmware Supervisor flash support to store firmware for the drive, always choose "Exact Match." See the table on page 51 for full details when using ADC.
Input Data	Assigns drive or connected peripheral parameters to be READ by the controller using Datalinks.
Output Data	Assigns drive or connected peripheral parameters to be WRITTEN by the controller using Datalinks.
Mode Select	Sets the I/O configuration to either Velocity or Position mode.

On the Module Definition window, notice that the automatically-assigned controller tags Drive Status, Feedback, Logic Command, and Reference are always used.

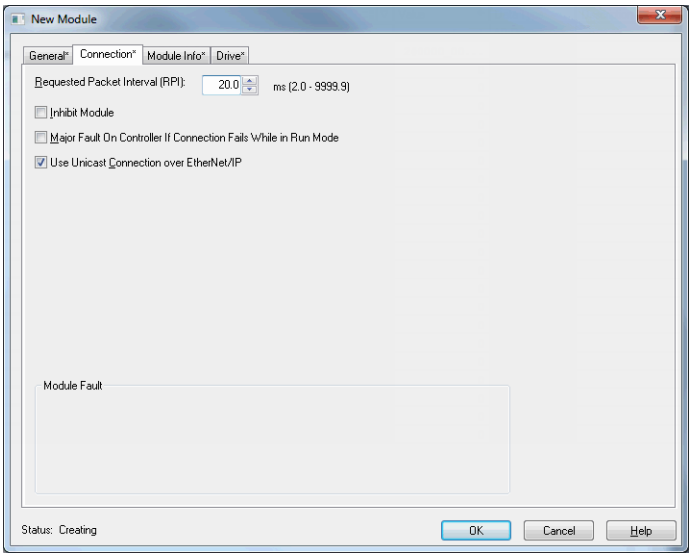
When using Datalinks you must still assign parameters **C153 [EN Data In 1]** through **C156 [EN Data In 4]** and **C157 [EN Data Out 1]** through

C160 [EN Data Out 4] to point to the appropriate drive or connected peripheral parameters. The procedure to configure the Datalinks on the Module Definition window for the Input Data and Output Data is the same:

- Click the ▼ button to assign a parameter to each input and output Datalink you require.

IMPORTANT Always use the Datalink parameters in consecutive numerical order, starting with the first parameter. (For example, use parameters C157, C158, and C159 to configure three Datalinks to write data and/or parameters C153, C154, C155, and C156 to configure four Datalinks to read data.) Otherwise, the network I/O connection will be larger than necessary, which needlessly increases controller response time and memory usage.

6. Click **OK** on the Module Definition window to save the drive configuration and close the window. The drive's New Module window reappears.
7. On the New Module window, click the Connection tab.



8. In the "Requested Packet Interval (RPI)" box, set the value to 5.0 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/Logix Designer 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. Click **OK** on the New Module window.
   The new node ("PowerFlex 525-EENET PowerFlex_52X_Drive" in this example) now appears under the Ethernet icon in the I/O Configuration folder. If you double-click on the Input Controller Tag (Controller Input Tags on page 45) and Output Controller Tag (Controller Output Tags on page 45), you will see that module-defined data types and tags have been automatically created. 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 using the controller's ladder logic.

Controller Input Tags

Name	Value	Data Type	Description
PowerFlex_52X_Drive:I	{...}	AB:PowerFlex5...	
PowerFlex_52X_Drive:I.DriveStatus	2#0000_00...	INT	
PowerFlex_52X_Drive:I.Ready	0	BOOL	
PowerFlex_52X_Drive:I.Active	0	BOOL	
PowerFlex_52X_Drive:I.CommandDir	0	BOOL	
PowerFlex_52X_Drive:I.ActualDir	0	BOOL	
PowerFlex_52X_Drive:I.Accelerating	0	BOOL	
PowerFlex_52X_Drive:I.Decelerating	0	BOOL	
PowerFlex_52X_Drive:I.Faulted	0	BOOL	
PowerFlex_52X_Drive:I.AtReference	0	BOOL	
PowerFlex_52X_Drive:I.CommFreqCnt	0	BOOL	
PowerFlex_52X_Drive:I.CommLogicCnt	0	BOOL	
PowerFlex_52X_Drive:I.ParmsLocked	0	BOOL	
PowerFlex_52X_Drive:I.DigIn1Active	0	BOOL	
PowerFlex_52X_Drive:I.DigIn2Active	0	BOOL	
PowerFlex_52X_Drive:I.DigIn3Active	0	BOOL	
PowerFlex_52X_Drive:I.DigIn4Active	0	BOOL	
PowerFlex_52X_Drive:I.OutputFreq	0	INT	
PowerFlex_52X_Drive:I.StopMode	0	INT	
PowerFlex_52X_Drive:I.OutputCurrent	0	INT	
PowerFlex_52X_Drive:I.OutputVoltage	0	INT	
PowerFlex_52X_Drive:I.CommandedFreq	0	INT	

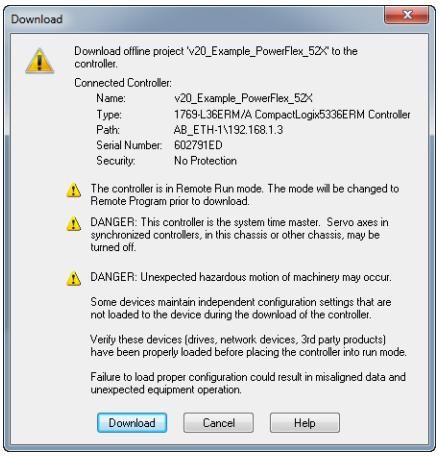
Controller Output Tags

Name	Value	Data Type	Description
PowerFlex_52X_Drive:O	{...}	AB:PowerFlex5...	
PowerFlex_52X_Drive:O.LogicCommand	2#0000_00...	INT	
PowerFlex_52X_Drive:O.Stop	0	BOOL	
PowerFlex_52X_Drive:O.Start	0	BOOL	
PowerFlex_52X_Drive:O.Jog	0	BOOL	
PowerFlex_52X_Drive:O.ClearFaults	0	BOOL	
PowerFlex_52X_Drive:O.Forward	0	BOOL	
PowerFlex_52X_Drive:O.Reverse	0	BOOL	
PowerFlex_52X_Drive:O.ForceKeypadCtrl	0	BOOL	
PowerFlex_52X_Drive:O.MOPIncrement	0	BOOL	
PowerFlex_52X_Drive:O.AccelRate1	0	BOOL	
PowerFlex_52X_Drive:O.AccelRate2	0	BOOL	
PowerFlex_52X_Drive:O.DecelRate1	0	BOOL	
PowerFlex_52X_Drive:O.DecelRate2	0	BOOL	
PowerFlex_52X_Drive:O.FreqSel01	0	BOOL	
PowerFlex_52X_Drive:O.FreqSel02	0	BOOL	
PowerFlex_52X_Drive:O.FreqSel03	0	BOOL	
PowerFlex_52X_Drive:O.MOPDecrement	0	BOOL	
PowerFlex_52X_Drive:O.FreqCommand	0	INT	
PowerFlex_52X_Drive:O.SleepTime	0	INT	
PowerFlex_52X_Drive:O.AveragekWhCost	0	INT	
PowerFlex_52X_Drive:O.AccelTime1	0	INT	
PowerFlex_52X_Drive:O.DecelTime1	0	INT	

Saving the I/O Configuration to the Controller

After adding the controller and drive 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/Logix Designer window, select **Communications** > **Download**. The Download dialog box appears.



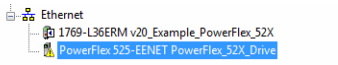
TIP If a message box reports that RSLogix 5000/Logix Designer is unable to go online, select **Communications** > **Who Active** to find your controller in the Who Active screen. After finding and selecting the controller, click **Set Project Path** to establish the path. If your controller does not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

2. Click **Download** to download the configuration to the controller. When the download is successfully completed, RSLogix 5000/Logix Designer goes into 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.
3. If the controller was in Run Mode before clicking **Download**, RSLogix 5000/Logix Designer 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.
4. 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.
5. To ensure that the present project configuration values are saved, RSLogix 5000/Logix Designer prompts you to upload them. Click **Yes** to upload and save them.

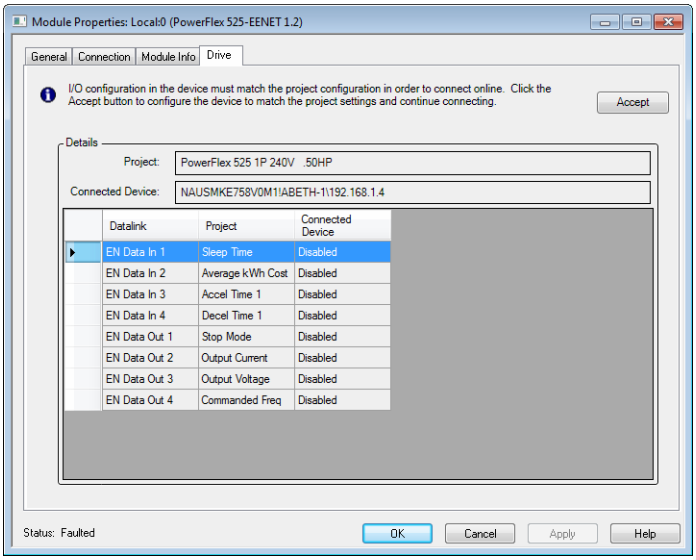
Correlating the Drive with the Controller

You must now correlate the drive settings to the RSLogix 5000/Logix Designer 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 "PowerFlex 525-EENET PowerFlex_52X_Drive") and select Properties.



2. Select the Drive tab to begin the correlation process.

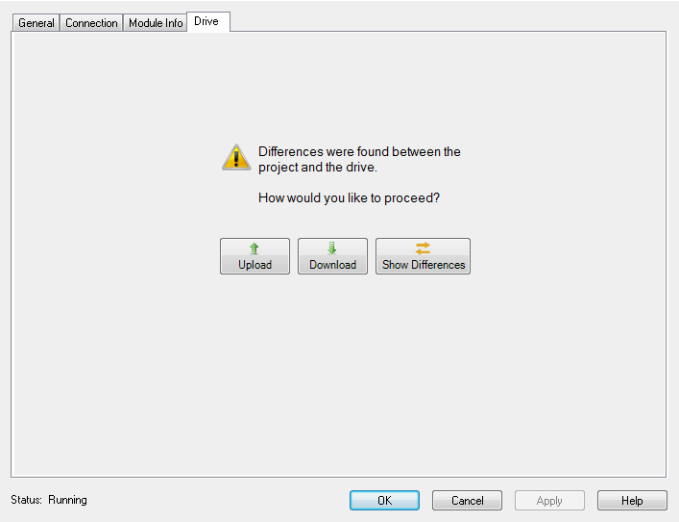


After the drive configuration data has been verified, the Drive tab will display a request to synchronize the configuration with the drive. Click **Accept**.

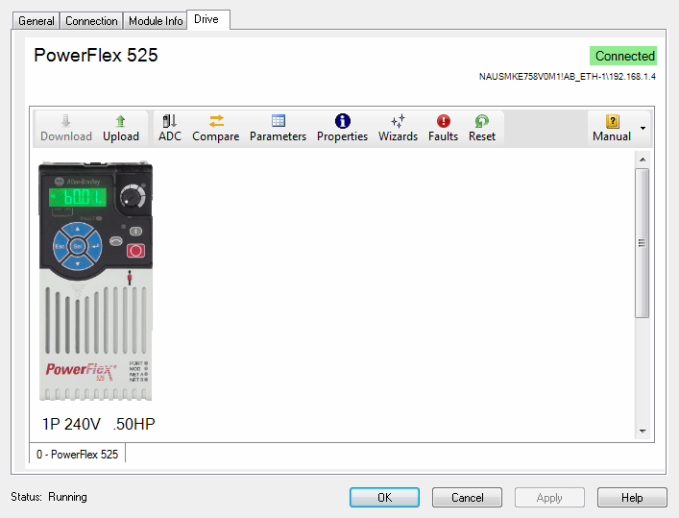
If the Differences Found Screen on page 48 appears—which is typical, click **Download**. This will download the project settings from the controller to the drive. If **Upload** is clicked, the drive settings are uploaded to the controller.

TIP On subsequent connections to the drive (after the initial download), select **Upload**.

Differences Found Screen



3. The Drive tab displays the current status of the drive.



If the download is successful, the Drive tab will show a green Connected indicator in the upper right corner of the window. This tab is extremely useful for configuring drive parameters, accessing start-up wizards and troubleshooting.

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

Using Automatic Device Configuration (ADC) with RSLogix 5000 (version 20) or Logix Designer (version 21 or greater)

Automatic Device Configuration (ADC) is an RSLogix 5000 (version 20) and Logix Designer (version 21 or greater) feature that supports the automatic download of configuration data upon the Logix controller establishing an EtherNet/IP network connection to a PowerFlex 525 drive and its associated peripherals.

IMPORTANT

- ADC is **not** available for DSI peripherals.
- ADC is **not** available in Multi-drive mode.
- ADC is only available when the drive is connected using the embedded EtherNet/IP adapter or future EtherNet/IP DLR adapter to a compatible controller.

An RSLogix 5000 (version 20) or Logix Designer (version 21 or greater) project (.ACD file) contains the configuration settings for any PowerFlex drives in the project. When the project is downloaded to the Logix controller, these settings are also transferred and reside in the controller's memory. Prior to ADC in RSLogix 5000 (version 20) or Logix Designer (version 21 or greater), downloading PowerFlex 525 configuration data was a manual process where the user would open the Drive tab in the respective drive Add-On Profile (AOP) in the application and click on the Download icon. ADC now automates the process and saves the user time. It is particularly beneficial in a drive replacement situation when a production line is down.

TIP Use with Stratix switches such as the Stratix 5700, Stratix 6000, and Stratix 8000 to provide dynamic IP address assignment by port. This eliminates the need for the user to manually enter the IP address, Subnet mask, and Gateway address prior to connecting a replacement drive to the Ethernet network.

ADC can also work in tandem with Firmware Supervisor. If Firmware Supervisor is set up and enabled for a drive ("Exact Match" keying must be used), the drive/peripheral will be automatically flashed (if necessary) prior to any ADC operation.

IMPORTANT **Logix "owns" the configuration of the drive. ADC will be triggered any time the Logix controller detects a configuration signature mismatch when establishing an EtherNet/IP network I/O connection.** The use of other configuration tools, such as a HIM or Connected Components Workbench software should be minimized and restricted to monitor-only operation. Any configuration changes made by these tools will cause a configuration signature mismatch the next time the Logix controller connects to the device and ADC will write over any changes made by the other tool(s). Any drive configuration changes should be made with the drive Add-On Profile.

The drive AOP requires user action to enable ADC. This helps ensure that the user understands ADC operation prior to turning it on. The drive AOPs also have an ADC icon on the Drive tab to show general ADC enable/disable status for the drive:

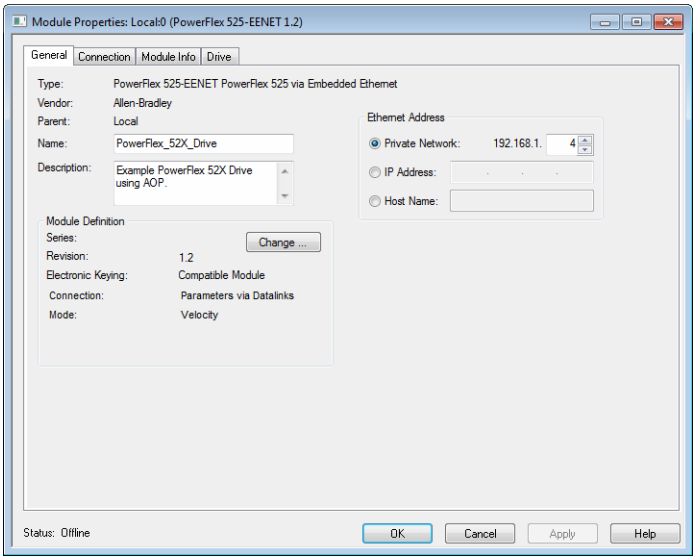
Icon	Meaning
	No ports on the drive have ADC enabled.
	At least one port on the drive has ADC enabled.
	ADC is not supported.

Clicking on the ADC icon will launch the ADC configuration window. This provides a single, convenient location to enable/disable ADC on the drive.

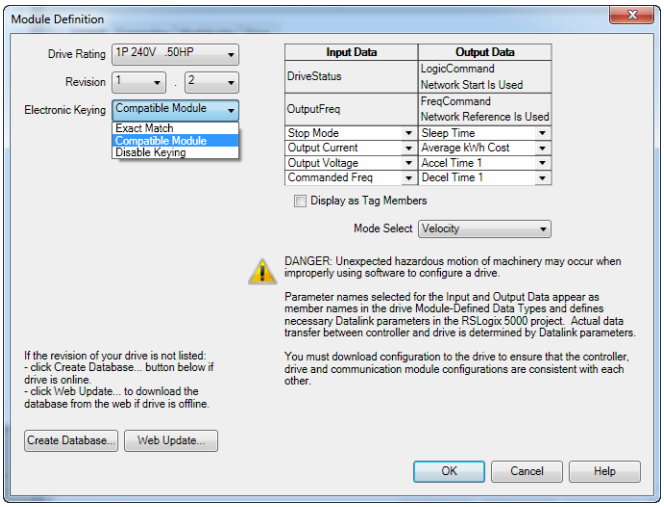
Configuring a PowerFlex 525 Drive for ADC

ADC is configured within the AOP of the PowerFlex 525 drive. Start by creating or opening a PowerFlex 525 drive in the RSLogix 5000/Logix Designer I/O Configuration folder.

1. In the Module Properties window, select the General tab and click **Change...** to open the Module Definition window.



2. Select the appropriate Electronic Keying for your application.



There are three Electronic Keying choices available in the Module Definition window in the drive AOP, but only two are recommended with ADC:

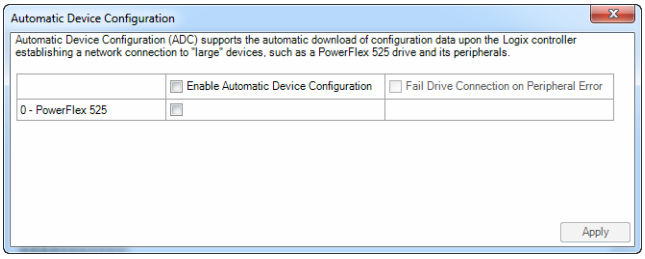
Keying Selection	Recommendation
Exact Match	This selection should only be used if: • Your system design specification requires that a replacement drive/peripheral be identical – down to the Minor revision of firmware (x.xxx). • You will be implementing Firmware Supervisor flash support in addition to ADC. ControlFlash firmware kits for the revision of firmware used for each drive/peripheral must be installed on the PC running RSLogix 5000/Logix Designer. Flash files can be downloaded from: http://www.ab.com/support/abdrives/webupdate
Compatible Module	This selection is the typical ADC selection when Firmware Supervisor is not used. A replacement drive (including peripherals) will need to have the same or higher firmware revision as the original. Since drives with newer firmware are required to be compatible with older firmware, this allows ADC to work without compatibility concerns. Note that if a Series change accompanies a Major firmware change, the replacement drive may or may not be "compatible" with respect to keying.
Disabled	When using ADC, this selection should generally not be used. This selection allows a replacement drive to have any different Major (**x**.xxx) and/or Minor (x.**xxx**) firmware revision. It is up to the user to provide a replacement that has a firmware revision later than or equal to the original drive. If a replacement drive with older firmware is used, the ADC download may fail.

Electronic Keying for HIMs and serial converters are disabled by default. These are typically temporary devices or used for monitoring purposes only and therefore "do not matter" if they are present or not. You still have the option to select these to other Keying selections if desired.

TIP Electronic Keying settings for peripherals can be modified in the Drive tab of the AOP. Select the peripheral tab for the device you wish to modify, then click **Properties**.

Click **OK** when finished.

3. Select the Drive tab and click the ADC icon to open the ADC Settings.



The ADC Settings window provides a single location for ADC configuration of the drive's ports. Global checkboxes at the top of each column checks or unchecks the entire column. Ports can also be turned on/off individually. See the checkbox selection information in Step 3 for additional details.

Click **OK** when finished.

4. Perform the above steps for each additional PowerFlex 525 drive.
5. Save your project and download the project to the Logix controller.

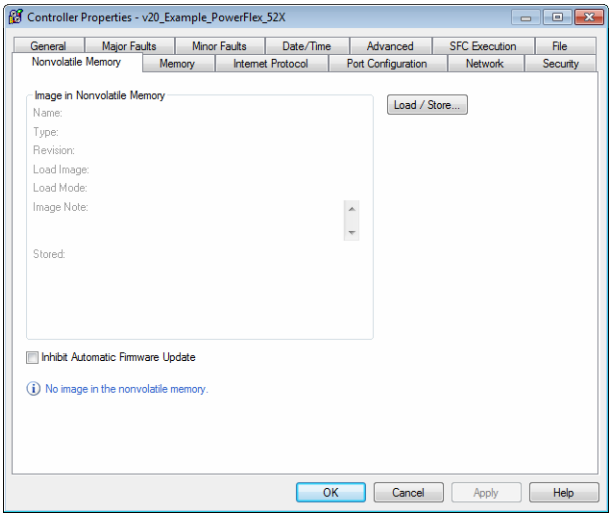
ADC and Logix Memory

Starting in RSLogix 5000 version 16, drive configuration settings have been stored in the project's ACD file which is downloaded and stored in the controller. The majority of Logix controllers have megabytes of memory available, so this typically should not be an issue. You can monitor Logix memory usage in the application in the **Controller Properties > Memory** tab.

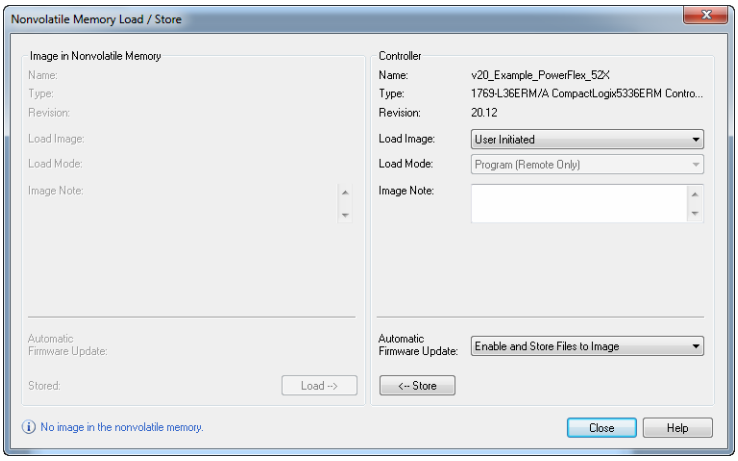
Storing the Drive's and Peripheral's Firmware in the Logix Controller (Firmware Supervisor)

The Logix Firmware Supervisor function has been extended to provide firmware updates for the peripherals connected to the drive. To configure the controller to check and refresh the correct firmware for the drive and peripherals, perform the following steps:

1. Verify that "Exact Match" keying is selected in the drive's and peripherals' properties screens (the drive's is in the General tab; the peripherals' are under the Drive tab, after right-clicking on each peripheral and choosing **Properties**).
2. Verify that ControlFlash firmware kits for each revision of firmware for each device that should be stored in the controller have been installed on the PC running RSLogix 5000/Logix Designer.
3. Verify that a CompactFlash or other storage card has been installed in the controller.
4. Use RSLogix 5000/Logix Designer to go online with the controller in Program mode. Download your program if you have not done so already.
5. In the treeview, right-click on the controller folder at the top of the Controller Organizer and choose **Properties**. On the Controller Properties window, select the Nonvolatile Memory tab.

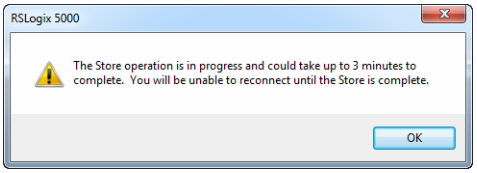


6. Click **Load/Store...**. The Nonvolatile Memory Load/Store window appears.



In the Controller section for the Automatic Firmware Update field, select "Enable and Store Files to Image," and click the **<-- Store** button. You may see two different continue confirmation dialog boxes relating to communication disruptions and erasure of the current contents of the storage card. If okay, click **Yes** on each dialog box.

7. The application will go to the Offline state, and the following dialog box will appear.



Wait a short period of time for the store operation to complete, and then attempt to go online with the controller again.

Monitoring the ADC Progress

The time it takes for the ADC process to complete will vary from seconds to several minutes depending on several factors:

- The number of peripherals enabled for ADC.
- If a configuration signature for the drive/peripheral indicates a configuration download needs to be performed for the given port.
- Whether Firmware Supervisor is enabled and needs to flash the drive and/or any peripherals.

IMPORTANT ADC can automatically reset the drive as part of the configuration process. This is done because some parameters require a reset before they take effect. If a drive is being replaced with an out-of-the-box drive, you will typically see one or more resets during the ADC process.

An operational drive in a running Logix system will have the following status indicator states:

Status Indicator	State	Description
ENET	Off	Adapter is not connected to the network.
	Steady	Adapter is connected to the network and drive is controlled through Ethernet.
	Flashing	Adapter is connected to the network but drive is not controlled through Ethernet.
LINK	Off	Adapter is not connected to the network.
	Steady	Adapter is connected to the network but not transmitting data.
	Flashing	Adapter is connected to the network and transmitting data.
FAULT	Flashing Red	Indicates drive is faulted

See Understanding the Status Indicators on page 111 for more information.

Additional information may also be displayed on the HIM if it is present (flash status, etc.).

If ADC is unsuccessful, RSLogix 5000/Logix Designer can be used to get additional information. When online, the drive at issue should have a yellow triangle ⚠ next to it in the RSLogix 5000/Logix Designer project's I/O Configuration folder. Double-click on the drive to open the drive AOP. The Connection tab will show a Module Fault code and the Drive tab can help you identify issues.

ADC Status Field	Description
Running	Any desired configuration is complete, and the I/O connection is running.
Configuring	ADC is currently updating the configuration of the drive or one of its peripherals. Clicking on the Connection tab will show which device is being updated.
Firmware Updating	ADC is currently updating the firmware of the drive or one of its peripherals. Clicking on the Drive tab will show which device is being updated.
Inhibited	The program has the connection inhibited. You can uninhibit the connection on the Connection tab.
Faulted	A problem is preventing the controller from connecting to the drive (for example, the device at the IP address provided is not a PowerFlex 525 drive). Clicking on the Connection tab will show the cause (Module Fault). Clicking on the Drive tab may also show the faulted ports.

Examples of potential issues/solutions are:

Issue	Solution
"Compatible module" keying selected, but replacement drive or peripheral has an earlier firmware revision than the failed device.	Replace device with a revision that is later than or equal to the failed device. If necessary, use ControlFLASH to flash replacement device first to an acceptable revision level.
Peripheral is required for connection ("Fail Drive Connection on Peripheral Error" was checked), but it is missing.	Add required peripheral or remove peripheral from RSLogix 5000/Logix Designer project for the drive and download project to the controller.
Parameter "out of range" error—ADC wrote a value to a parameter that was out of range (typically would only occur during initial commissioning of a drive system).	Use any available drive software tool to view a linear list of changed parameters to see if the configured value is outside the minimum/maximum value. The drive AOPs are the preferred tool and will highlight any out of range parameter in the Linear List editor. Connected Components Workbench (version 3 or later) may also be used.

Using the RSLogix 5000 (all versions) or Logix Designer (version 21 or greater) Generic Profile

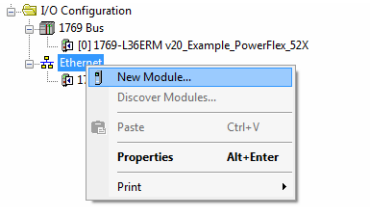
The basic RSLogix 5000/Logix Designer Generic Profile is only recommended when:

- A specific drive profile in other versions of RSLogix 5000/Logix Designer is unavailable.
- Users are already familiar with a Generic Profile and do not want to convert an existing project to a drive Add-On Profile (RSLogix 5000 version 17 or greater and Logix Designer version 21 or greater).
- A project must maintain specific revision level control.
- The controller cannot be taken offline. RSLogix 5000 (all versions) and Logix Designer (version 21 or greater) enables the drive Generic Profile to be added while the controller is online and in the Run mode.

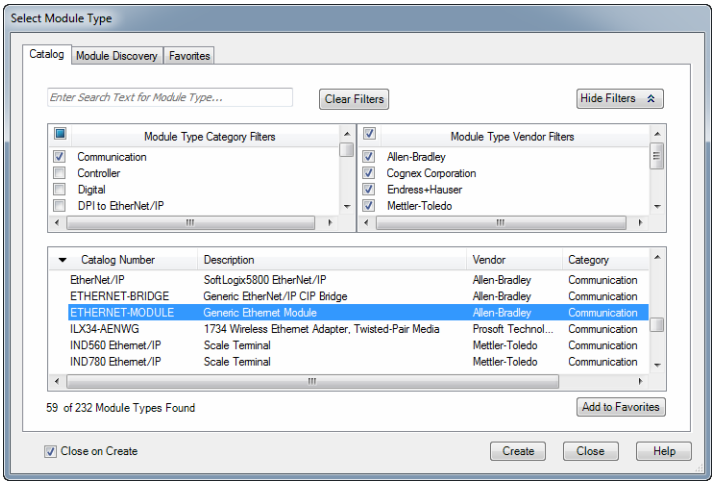
Adding the Drive to the I/O Configuration

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

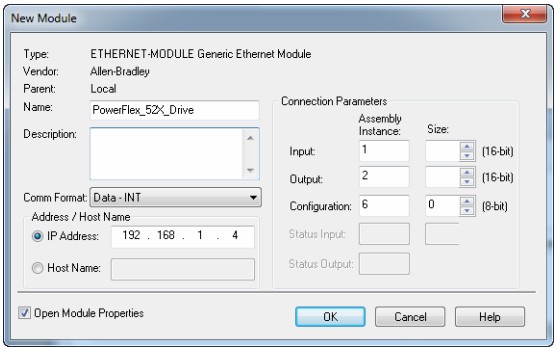
1. In the treeview, right-click on the Ethernet icon and select **New Module...** to display the Select Module Type window. Expand the Communications group to display all of the available communication modules or search for "ETHERNET-MODULE".



2. Select "ETHERNET-MODULE" from the list in the Select Module Type window to configure the drive's embedded EtherNet/IP adapter, and then click **Create**. The drive's New Module window appears.



3. Edit the following information about the drive:



Box	Setting
Name	A name to identify the drive.
Description	Optional – description of the drive.
Comm Format	Data – INT (This setting formats the data in 16-bit words.)
IP Address	The IP address of the drive.
Open Module Properties	When this box is checked, clicking **OK** opens additional module properties screens to further configure the drive. When unchecked, clicking **OK** closes the drive's New Module screen. 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 total number of [**EN Data Out x**] parameters used for your application, either in Single-drive mode (see details below) or Multi-drive mode (see Using Multi-Drive Mode on page 83).
Output	**2** (This value is required.)	The value will vary based on the total number of [**EN Data In x**] parameters used for your application, either in Single-drive mode (see details below) or Multi-drive mode (see Using Multi-Drive Mode on page 83).
Configuration	**6** (This value is required.)	**0** (This value is required.)

Enter the number of 16-bit words that are required for your I/O in the Input Size and Output Size boxes. Since the adapter always uses the 16-bit Logic Status, 16-bit Feedback, and two 16-bit words dedicated for memory allocation of the Generic Ethernet module profile, at least four 16-bit words must be set for the Input Size. The adapter also uses the 16-bit Logic Command and 16-bit Reference, requiring at least two 16-bit words for the Output Size. If any or all of the drive's eight 16-bit Datalinks are used (see Configuring a Master-Slave Hierarchy on page 31), the Input and Output Size settings must be increased accordingly.

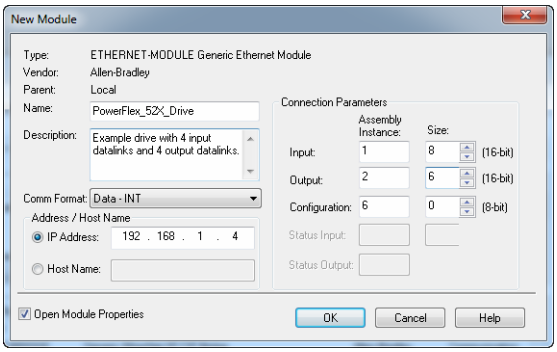
Generic Profile Example of I/O Image

Adapter Word and I/O			
Output Image (Write)	Required	Word 0	Logic Command
		Word 1	Reference
	Optional	Word 2	Datalink 1
		Word 3	Datalink 2
		Word 4	Datalink 3
		Word 5	Datalink 4
Input Image (Read)	Required	Word 0	Padword[1]
		Word 1	Padword
		Word 2	Logic Status
		Word 3	Feedback
	Optional	Word 4	Datalink 1
		Word 5	Datalink 2
		Word 6	Datalink 3
		Word 7	Datalink 4

(1) Padwords only apply when using the generic profile. Padwords are not used when using the full-featured drive Add-On-Profile.

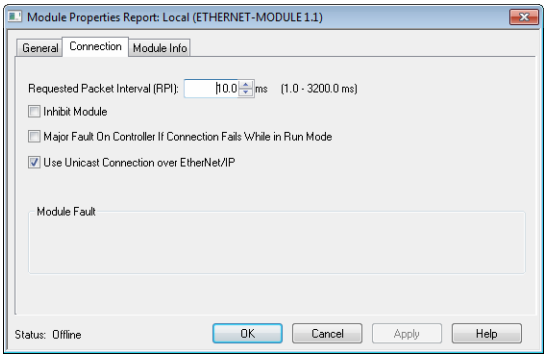
IMPORTANT The Datalink parameters [EN Data Out x] and [EN Data In x] do not actually contain data. These are user-configurable parameters that only contain the parameter number of the parameter whose data will be made available for write/read in the I/O image. See Using Datalinks on page 63.

For the example below, all four [**EN Data Out x**] and all four [**EN Data In x**] parameters are used, resulting in an Input Size of "8" and an Output Size of "6."



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

6. Click the Connection tab.



7. In the "Requested Packet Interval (RPI)" box, set the value to 5.0 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 ("ETHERNET-MODULE_PowerFlex_52X_Drive" in this example) now appears under the Ethernet icon in the I/O Configuration folder. If you double-click on the Input Controller Tag (Input Image Controller Tags on page 58) and Output Controller Tag (Output Image Controller Tags on page 58), 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 using the controller's ladder logic.

Input Image Controller Tags

Name	Value	Data Type	Description
PowerFlex_52X_Drive:C	{...}	AB:ETHERNET_MODULE:C:0	
PowerFlex_52X_Drive:I	{...}	AB:ETHERNET_MODULE_INT_16Bytes:I:0	
PowerFlex_52X_Drive:I.Data	{...}	INT[8]	
PowerFlex_52X_Drive:I.Data[0]	0	INT	Padword
PowerFlex_52X_Drive:I.Data[1]	0	INT	Padword
PowerFlex_52X_Drive:I.Data[2]	0	INT	Logic Status
PowerFlex_52X_Drive:I.Data[3]	0	INT	Speed Feedback
PowerFlex_52X_Drive:I.Data[4]	0	INT	Datalink 1
PowerFlex_52X_Drive:I.Data[5]	0	INT	Datalink 2
PowerFlex_52X_Drive:I.Data[6]	0	INT	Datalink 3
PowerFlex_52X_Drive:I.Data[7]	0	INT	Datalink 4

Output Image Controller Tags

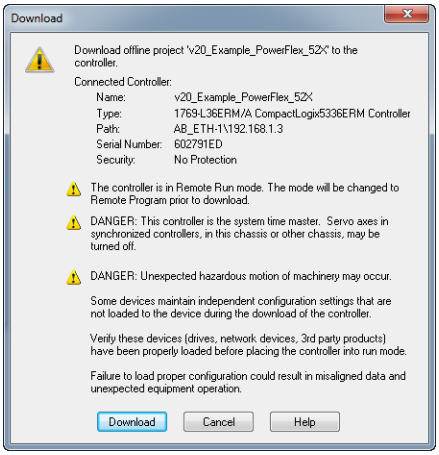
Name	Value	Data Type	Description
PowerFlex_52X_Drive:O	{...}	AB:ETHERNET_MODULE_INT_12Bytes:O:0	
PowerFlex_52X_Drive:O.Data	{...}	INT[6]	
PowerFlex_52X_Drive:O.Data[0]	0	INT	Logic Command
PowerFlex_52X_Drive:O.Data[1]	0	INT	Speed Reference
PowerFlex_52X_Drive:O.Data[2]	0	INT	Datalink 1
PowerFlex_52X_Drive:O.Data[3]	0	INT	Datalink 2
PowerFlex_52X_Drive:O.Data[4]	0	INT	Datalink 3
PowerFlex_52X_Drive:O.Data[5]	0	INT	Datalink 4

Saving the I/O Configuration to the Controller

After adding the bridge and drive 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/Logix Designer, 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/Logix Designer window, select **Communications** > **Download**. The Download dialog box appears.



TIP If a message box reports that RSLogix 5000/Logix Designer is unable to go online, select **Communications** > **Who Active** to find your controller in the Who Active screen. After finding and selecting the controller, click S**et 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 37 for details.

2. Click **Download** to download the configuration to the controller. When the download is successfully completed, RSLogix 5000/Logix Designer goes into the Online mode and the I/O OK box in the upper-left of the screen should be steady 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 drive 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 Configuring a Master-Slave Hierarchy on page 31). 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 in Using MicroLogix 1100/1400

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/Logix Designer 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 MicroLogix 1100/1400 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 embedded EtherNet/IP adapter 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 (see 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 (see 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 MicroLogix 1100/1400 controller using RSLogix 500 software with explicit messages is more difficult, and produces a more complex program.

Using the I/O

This chapter provides information and examples that explain how to control, configure, and monitor a PowerFlex 525 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 525 drives.

The adapter includes the Logic Command, Logic Status, Reference, Feedback, and memory allocation for the Generic Ethernet module profile (all as 16-bit words) in the controller's I/O image. This basic I/O must always be configured in the Ethernet bridge using RSLogix 5000/Logix Designer. Additional I/O, if needed, can be set using up to four Datalinks to write data and/or up to four Datalinks to read data. When using any combination of these Datalinks, add one 16-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 drive 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 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 16-bit Datalinks ([**EN Data In 1...4**] and [**EN Data Out 1...4**]) are used.
- **ControlLogix/CompactLogix Controllers only**—The drive Add-On Profile (AOP) used in RSLogix 5000 (version 17 or greater) or Logix Designer (version 21 or greater), or the Generic Profile (all versions).
- If Multi-drive mode is enabled, and the number of daisy-chained drives that are present.

I/O Controller Image

Since the drive Add-On Profile in RSLogix 5000 (version 17 or greater) and Logix Designer (version 21 or greater) 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/Logix Designer, however, controller tags are not descriptive.

I/O Image for PowerFlex 525 Drives on page 62 shows the I/O image when using all of the 16-bit Datalinks.

I/O Image for PowerFlex 525 Drives
(16-bit Logic Command/Status, Reference/Feedback, and Datalinks)

INT	Output
0	Logic Command
1	Reference
2	Datalink 1
3	Datalink 2
4	Datalink 3
5	Datalink 4

Input Using...			
INT	Drive Add-On Profile	INT	Generic Profile
0	Logic Status	0	Padword
1	Feedback	1	Padword
2	Datalink 1	2	Logic Status
3	Datalink 2	3	Feedback
4	Datalink 3	4	Datalink 1
5	Datalink 4	5	Datalink 2
		6	Datalink 3
		7	Datalink 4

Using Logic Command/Status

The *Logic Command* is a 16-bit word of control data produced by the controller and consumed by the adapter. The *Logic Status* is a 16-bit word of status data produced by the adapter and consumed by the controller.

When using a ControlLogix or CompactLogix controller, the Logic Command word is always INT 0 in the output image and the Logic Status word is always:

- INT 0 in the input image when using the drive Add-On Profile.
- INT 2 when using the Generic Profile.

This manual contains the bit definitions for compatible products available at the time of publication in Appendix D, Logic Command/Status Words: PowerFlex 525 Drives.

Using Reference/Feedback

The *Reference* is a 16-bit word that is produced by the controller and consumed by the adapter. The *Feedback* is a 16-bit word produced by the adapter and consumed by the controller.

When using a ControlLogix or CompactLogix controller, the Reference is always INT 1 in the output image (see I/O Image for PowerFlex 525 Drives on page 62) and the 16-bit Feedback is always:

- INT 1 in the input image when using the drive Add-On Profile.
- INT 3 when using the Generic Profile.

The Reference and Feedback are 16-bit values which represent drive speed in 0.01 Hz.

IMPORTANT There are several parameters in the drive that will override the start source and speed reference command if enabled. For details on these parameters, see the PowerFlex 525 drive's user manual, publication 520-UM001.

Attempting to write a negative value to the Speed Reference will result in the drive ramping to maximum speed due to overflow, the direction of the drive can only be controlled programmatically with the appropriate bits (bits 4 and 5) in the Command Word.

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 read or written to without using an Explicit Message. When enabled, each Datalink occupies one 16-bit word in a ControlLogix, CompactLogix, or MicroLogix controller.

The following rules apply when using PowerFlex 525 drive Datalinks:

- Datalinks cannot be used with Multi-drive mode.
- The target of a Datalink can be any valid parameter, including those of a peripheral. For example, drive parameter **P041 [Accel Time 1]** can be the target of the embedded EtherNet/IP adapter and any or all Option Cards installed in the drive.
- The data passed through the drive's Datalink mechanism is determined by the settings of parameters **C157...C160 [EN Data Out 1...4]** and parameters **C153...C156 [EN Data In 1...4]**.
- When an I/O connection that includes Datalinks is active, those Datalinks being used are 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 525 drive peripherals (embedded EtherNet/IP adapter and Option Modules such as 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/Logix Designer
- Putting the controller in Program mode
- Placing the scanner in idle mode
- Disconnecting the drive from the network

Example Ladder Logic Program Information

The example ladder logic programs in the sections of this chapter are intended for PowerFlex 525 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 525 drives. See Appendix D, Logic Command/Status Words: PowerFlex 525 Drives to view details.

CompactLogix Example

Creating Ladder Logic Using the RSLogix 5000/Logix Designer Drive Add-On Profiles

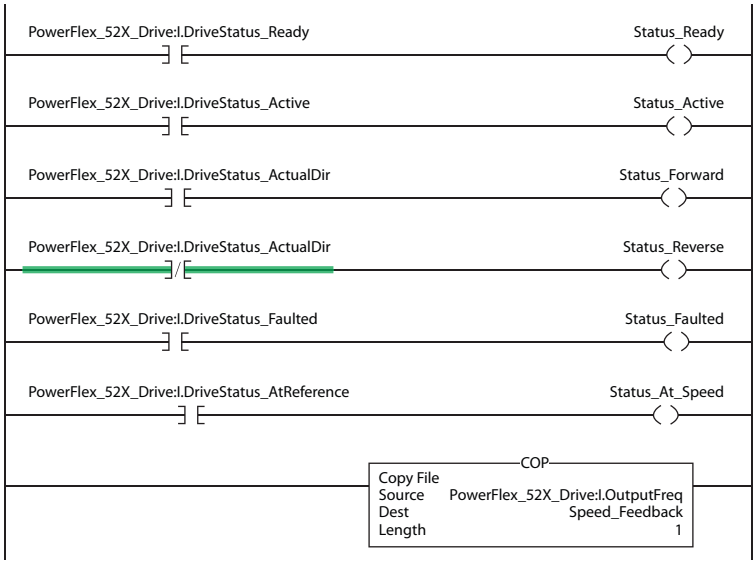
Since the drive Add-On Profile automatically created descriptive controller tags (see Controller Input Tags and Controller Output Tags on page 45) for the entire I/O image in Chapter 4, you can use these tags to directly control and monitor the drive without creating any ladder logic program. However, 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 (see CompactLogix Program Tags for Drive Add-On Profile Ladder Logic Program Example on page 65) and a ladder logic program that will pass the Controller tag data to the Program tags.

CompactLogix Program Tags for Drive Add-On Profile Ladder Logic Program Example

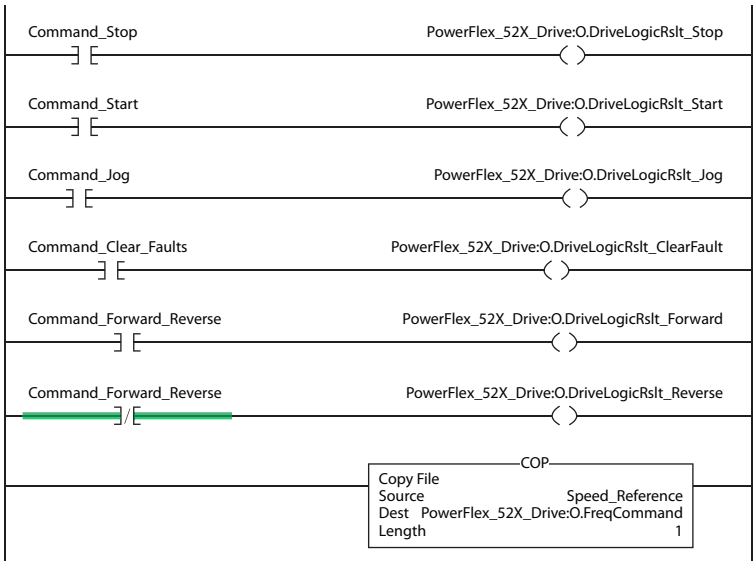
Name	Value	Data Type	Description
+ Speed_Reference	0	INT	
+ Speed_Feedback	0	INT	
Status_Reverse	0	BOOL	
Status_Ready	0	BOOL	
Status_Forward	0	BOOL	
Status_Faulted	0	BOOL	
Status_At_Speed	0	BOOL	
Status_Active	0	BOOL	
Command_Stop	0	BOOL	
Command_Start	0	BOOL	
Command_Jog	0	BOOL	
Command_Forward_Reverse	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 CompactLogix Example Ladder Logic Program Using a Drive Add-On Profile for Logic Status/Feedback on page 65 and CompactLogix Example Ladder Logic Program Using a Drive Add-On Profile for Logic Command/Reference on page 66. Note that the prefix for the drive Controller tags is determined by the name assigned when configuring the I/O (see Chapter 4).

CompactLogix Example Ladder Logic Program Using a Drive Add-On Profile for Logic Status/ Feedback



CompactLogix Example Ladder Logic Program Using a Drive Add-On Profile for Logic Command/ Reference



Creating Ladder Logic Using the RSLogix 5000/Logix Designer Generic Profile

Controller Tags

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

CompactLogix Controller Tags for Drive Generic Profile Ladder Logic Program Example

Name	Value	Data Type	Description
+ PowerFlex_52X_Drive:C	{...}	AB:ETHERNET_MODULE:C:0	
− PowerFlex_52X_Drive:I	{...}	AB:ETHERNET_MODULE_INT_16Bytes:I:0	
− PowerFlex_52X_Drive:I.Data	{...}	INT[8]	
+ PowerFlex_52X_Drive:I.Data[0]	0	INT	Padword
+ PowerFlex_52X_Drive:I.Data[1]	0	INT	Padword
+ PowerFlex_52X_Drive:I.Data[2]	0	INT	Logic Status
+ PowerFlex_52X_Drive:I.Data[3]	0	INT	Speed Feedback
+ PowerFlex_52X_Drive:I.Data[4]	0	INT	Datalink 1
+ PowerFlex_52X_Drive:I.Data[5]	0	INT	Datalink 2
+ PowerFlex_52X_Drive:I.Data[6]	0	INT	Datalink 3
+ PowerFlex_52X_Drive:I.Data[7]	0	INT	Datalink 4
− PowerFlex_52X_Drive:O	{...}	AB:ETHERNET_MODULE_INT_12Bytes:O:0	
− PowerFlex_52X_Drive:O.Data	{...}	INT[6]	
+ PowerFlex_52X_Drive:O.Data[0]	0	INT	Logic Command
+ PowerFlex_52X_Drive:O.Data[1]	0	INT	Speed Reference
+ PowerFlex_52X_Drive:O.Data[2]	0	INT	Datalink 1
+ PowerFlex_52X_Drive:O.Data[3]	0	INT	Datalink 2
+ PowerFlex_52X_Drive:O.Data[4]	0	INT	Datalink 3
+ PowerFlex_52X_Drive:O.Data[5]	0	INT	Datalink 4

You can expand the Input and Output tags to reveal the input and output configuration (see CompactLogix Controller Tags for Drive Generic Profile Ladder Logic Program Example on page 66). The Input tag for this example requires eight16-bit words of data. The Output tag for this example program requires six 16-bit words of data.

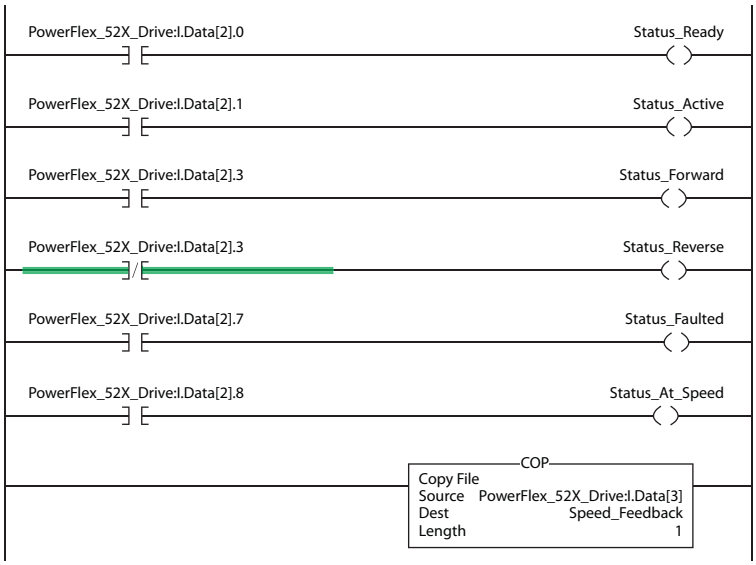
Program Tags

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

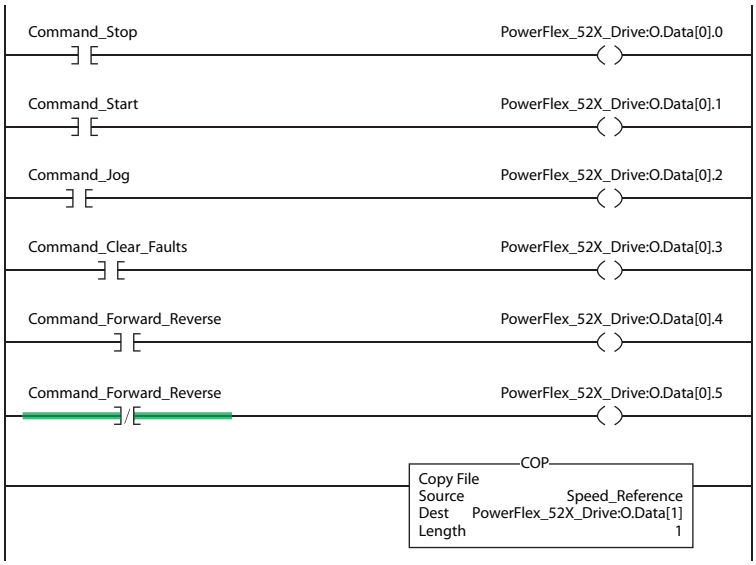
CompactLogix Program Tags for Drive Generic Profile Ladder Logic Program Example

Name	Value	Data Type	Description
+ Speed_Reference	0	INT	
+ Speed_Feedback	0	INT	
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	
Command_Stop	0	BOOL	
Command_Start	0	BOOL	
Command_Jog	0	BOOL	
Command_Forward_Reverse	0	BOOL	
Command_Clear_Faults	0	BOOL	

CompactLogix Example Ladder Logic Program Using a Drive Generic Profile for Logic Status/ Feedback



CompactLogix Example Ladder Logic Program Using a Drive Generic Profile for Logic Command/ Reference



Using Explicit Messaging

This chapter provides information and examples that explain how to use Explicit Messaging to configure and monitor the PowerFlex 525 drive and embedded EtherNet/IP adapter.

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.

ATTENTION: If you need to make frequent parameter changes using Explicit Messages, set parameter C121 [Comm Write Mode] to 1 "RAM only".

See Chapter 5, Using the I/O 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 device's parameters on the network.

IMPORTANT When an explicit message is performed, by default no 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, which is recommended.

IMPORTANT PowerFlex 525 drives have explicit messaging limitations. See Explicit Messaging Class Code Compatibility with PowerFlex 525 Drives on page 70 for more information.

Explicit Messaging Class Code Compatibility with PowerFlex 525 Drives

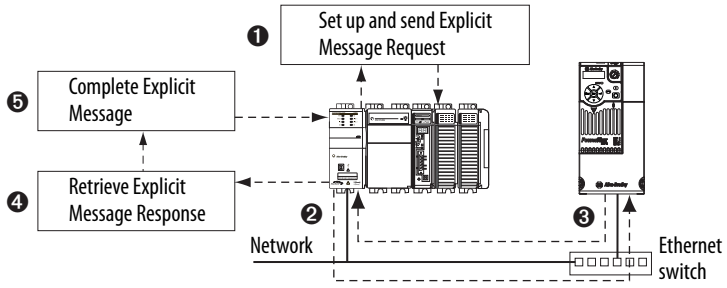
EtherNet/IP Object Class Code	Compatibility	Explicit Messaging Function
Parameter Object 0x0F	Yes	Single parameter reads/write
DPI Parameter Object 0x93	Yes	Single and scattered parameter reads/write

Performing Explicit Messaging

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. See 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.

Explicit Messaging 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.

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

CompactLogix Examples

TIP To display the Message Configuration screen in RSLogix 5000/Logix Designer, 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, see Appendix C, EtherNet/IP Objects.

IMPORTANT The explicit messaging examples in this section can be performed using any software version of RSLogix 5000/Logix Designer.

The read and write messaging examples in this section are for Device parameters which use Class Code 0x93.

The Message Configuration also has a supported Service Type of "Parameter Read" which is Class code 0x0F, Parameter Object.

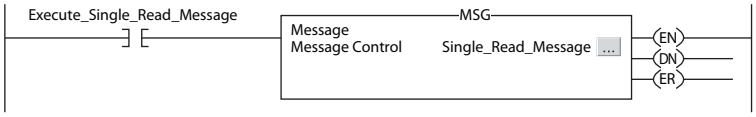
CompactLogix Example Ladder Logic Program to Read a Single Parameter

A Get Attribute Single message is used to read a single parameter. This read message example reads the value of the 16-bit parameter **b003 [Output Current]** in a PowerFlex 525 drive.

Example Controller Tags to Read a Single Parameter

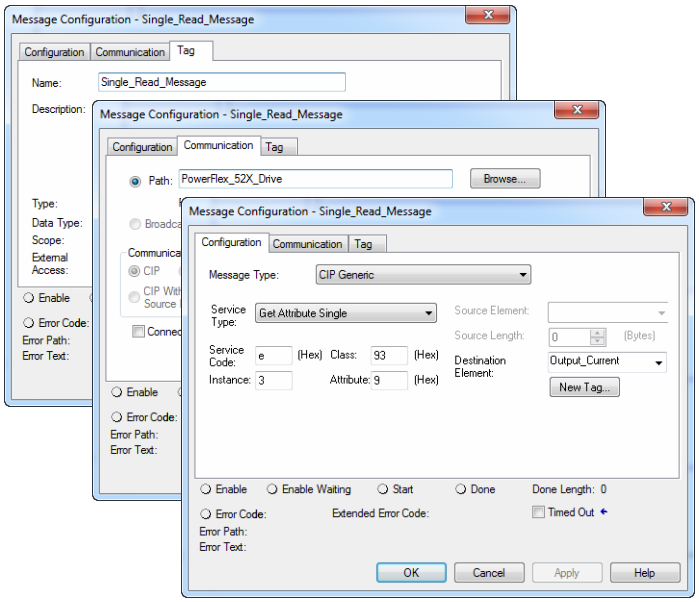
Operation	Controller Tags for Single Read Message	Data Types
XIC	Execute_Single_Read_Message	BOOL
MSG	Single_Read_Message	MESSAGE

Example Ladder Logic to Read a Single Parameter



CompactLogix – Formatting a Message to Read a Single Parameter

Get Attribute Single Message Configuration Screens



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

Configuration Tab	Example Value	Description
Message Type	CIP Generic	Used to access the DPI 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(3)	Class ID for the DPI Parameter Object.
Instance	3 (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(4)	The tag where the data that is read is stored.
Communication Tab	**Example Value**	**Description**
Path(2)	PowerFlex_52X_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 choosing a Service Type other than "Custom" from the pull-down menu, an appropriate Hex. value is automatically assigned to the Service Code box which is dimmed (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, PowerFlex_52X_Drive).

(3) See Explicit Messaging Class Code Compatibility with PowerFlex 525 Drives on page 70 for limitations of PowerFlex 525 drives when using DPI Parameter Object Class code 0x93 for explicit messaging.

(4) In this example, Output Current is a 16-bit parameter requiring the Data Type field to be set to "INT" when creating the controller tag. See the drive documentation to determine the size of the parameter and its data type.

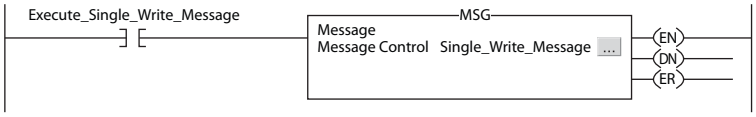
CompactLogix Example Ladder Logic Program to Write a Single Parameter

A Set Attribute Single message is used to write to a single parameter. This write message example writes a value to the 16-bit parameter **P041 [Accel Time 1]** in a PowerFlex 525 drive.

Example Controller Tags to Write a Single Parameter

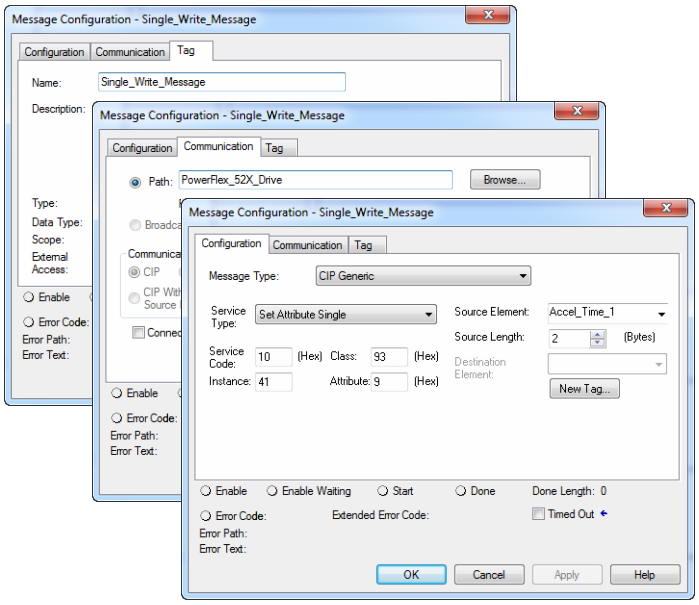
Operation	Controller Tags for Single Write Message	Data Types
XIC	Execute_Single_Write_Message	BOOL
MSG	Single_Write_Message	MESSAGE

Example Ladder Logic to Write a Single Parameter



CompactLogix – Formatting a Messageto Write a Single Parameter

Set Attribute Single Message Configuration Screens



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

Configuration Tab	Example Value	Description
Message Type	CIP Generic	Used to access the DPI Parameter Object in the adapter.
Service Type(1)	Get Attribute Single	This service is used to read a parameter value.
Service Code(1)	10 (Hex.)	Code for the requested service.
Class	93(5)	Class ID for the DPI Parameter Object.
Instance(2)	41 (Dec.)	Instance number is the same as parameter number.
Attribute(3)	9 or A (Hex.)	Attribute number for the Parameter Value attribute.
Source Element	Accel_Time_1(6)	Name of the tag for any service data to be sent from the scanner or bridge to the drive.
Source Length	2 bytes	Number of bytes of service data to be sent in the message.
Destination	–	Leave blank (not applicable).
Communication Tab	**Example Value**	**Description**
Path(4)	PowerFlex_52X_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 choosing a Service Type other than "Custom" from the pull-down menu, an appropriate Hex. value is automatically assigned to the Service Code box which is dimmed (unavailable).

(2) The instance is the parameter number in the drive.

(3) 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. **Important**: If you need to make frequent parameter changes using Explicit Messages, set parameter C121 [Comm Write Mode] to 1 "RAM only".

(4) Click **Browse** to find the path, or type in the name of the device listed in the I/O Configuration folder (for this example, PowerFlex_52X_Drive).

(5) See Explicit Messaging Class Code Compatibility with PowerFlex 525 Drives on page 70 for limitations of PowerFlex 525 drives when using DPI Parameter Object Class code 0x93 for explicit messaging.

(6) In this example, Accel Time 1 is a 16-bit parameter requiring the Data Type field to be set to "INT" when creating the controller tag. Also, the Source Length field on the Message Configuration screen must correspond to the selected Data Type in bytes (for example, 2 bytes for an INT). See the drive documentation to determine the size of the parameter and its data type.

CompactLogix Example Ladder Logic Program to Read Multiple Parameters

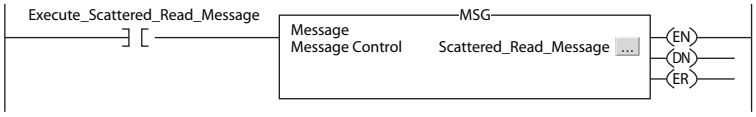
A Scattered Read message is used to read the values of multiple parameters. This read message example reads the values of these five 16-bit parameters in a PowerFlex 525 drive:

- Parameter **b001[Output Freq]**
- Parameter **b003 [Output Current]**
- Parameter **b004 [Output Voltage]**
- Parameter **b005 [DC Bus Voltage]**
- Parameter **b017 [Output Power]**

Example Controller Tags to Read Multiple Parameters

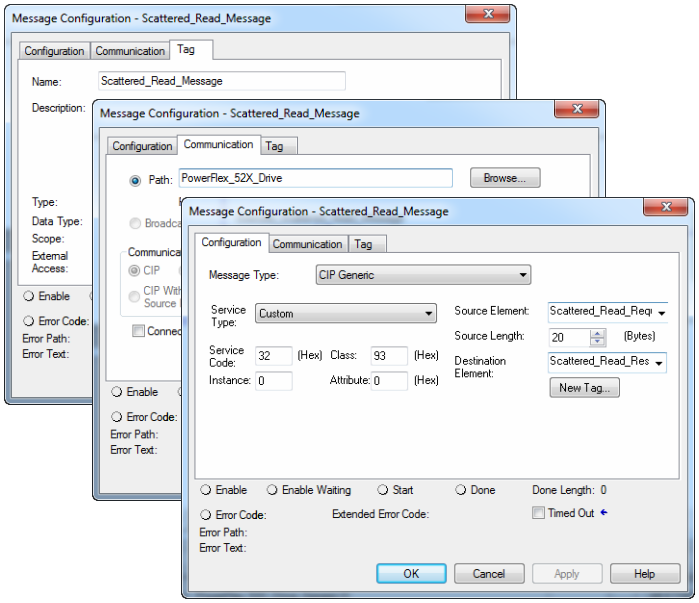
Operation	Controller Tags for Scattered Read Message	Data Types
XIC	Execute_Scattered_Read_Message	BOOL
MSG	Scattered_Read_Message	MESSAGE

Example Ladder Logic to Read Multiple Parameters



CompactLogix – Formatting a Messageto Read Multiple Parameters

Scattered Read Message Configuration Screens



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

Configuration Tab	Example Value	Description
Message Type	CIP Generic	Used to access the DPI Parameter Object in the adapter.
Service Type(1)	Custom	Required for scattered messages.
Service Code(1)	0x32 (Hex.)	Code for the requested service.
Class	93(3)	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(4)	Name of the tag for any service data to be sent from the scanner or bridge to the drive.
Source Length	20 bytes(4)	Number of bytes of service data to be sent in the message.
Destination	Scattered_Read_Response(5)	The tag where the data that is read is stored.
Communication Tab	**Example Value**	**Description**
Path(2)	PowerFlex_52X_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 choosing a Service Type other than "Custom" from the pull-down menu, an appropriate Hex. value is automatically assigned to the Service Code box which is dimmed (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, PowerFlex_52X_Drive).

(3) See Explicit Messaging Class Code Compatibility with PowerFlex 525 Drives on page 70 for limitations of PowerFlex 525 drives when using DPI Parameter Object Class code 0x93 for explicit messaging.

(4) In this example, we are reading five 16-bit parameters. Each parameter being read requires two contiguous INT registers. Therefore, a controller tag was created with its Data Type field set to "INT[10]." Also, the Source Length field on the Message Configuration screen must correspond to the selected Data Type in bytes (for this example, 20 bytes for an INT[10] array). Scattered read messages always assume that every parameter being read is a16-bit parameter, regardless of its actual size. Maximum message length is 256 bytes which can read up to 64 parameters, regardless of their size.

(5) The controller tag for "Scattered_Read_Response" must be the same size as the controller tag for "Scattered_Read_Request" (for this example, 20 bytes), but can be a different data type.

CompactLogix Example Scattered Read Request Data

In this message example, we use the data structure in Figure 101 in the source tag named Scattered Read Request to read these five 16-bit parameters in a PowerFlex 525 drive:

- Parameter **b001 [Output Freq]**
- Parameter **b003 [Output Current]**
- Parameter **b004 [Output Voltage]**
- Parameter **b005 [DC Bus Voltage]**
- Parameter **b017 [Output Power]**

Example Scattered Read Request Data

Name	Value	Data Type	Description
Scattered_Read_Request	{...}	INT[10]	
Scattered_Read_Request[0]	1	INT	Parameter Number
Scattered_Read_Request[1]	0	INT	Pad
Scattered_Read_Request[2]	3	INT	Parameter Number
Scattered_Read_Request[3]	0	INT	Pad
Scattered_Read_Request[4]	4	INT	Parameter Number
Scattered_Read_Request[5]	0	INT	Pad
Scattered_Read_Request[6]	5	INT	Parameter Number
Scattered_Read_Request[7]	0	INT	Pad
Scattered_Read_Request[8]	17	INT	Parameter Number
Scattered_Read_Request[9]	0	INT	Pad

CompactLogix 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). Example Scattered Read Response Converted Data on page 76 shows the parameter values.

Example Scattered Read Response Converted Data

Name	Value	Data Type	Description
Scattered_Read_Response	{...}	INT[10]	
Scattered_Read_Response[0]	1	INT	Parameter Number
Scattered_Read_Response[1]	5000	INT	Value
Scattered_Read_Response[2]	3	INT	Parameter Number
Scattered_Read_Response[3]	1	INT	Parameter Value
Scattered_Read_Response[4]	4	INT	Parameter Number
Scattered_Read_Response[5]	1796	INT	Parameter Value
Scattered_Read_Response[6]	5	INT	Parameter Number
Scattered_Read_Response[7]	349	INT	Parameter Value
Scattered_Read_Response[8]	17	INT	Parameter Number
Scattered_Read_Response[9]	0	INT	Parameter Value

In this message example, the parameters have the following values:

PowerFlex 525 Drive Parameters	Read Value
b001 [Output Freq]	50.00 Hz
b003 [Output Current]	0.01 Amp (No load)
b004 [Output Voltage]	179.6V AC
b005 [DC Bus Voltage]	349V DC
b017 [Output Power]	0 kW (No load)

CompactLogix Example Ladder Logic Program to Write Multiple Parameters

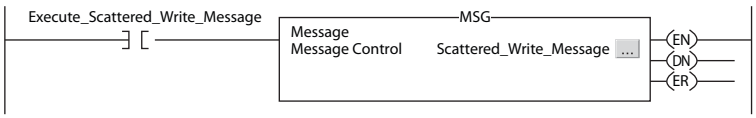
A Scattered Write message is used to write to multiple parameters. This write message example writes the following values to these five 16-bit parameters in a PowerFlex 525 drive:

PowerFlex 525 Drive Parameters	Write Value
A442 [Accel Time 2]	11.10 Sec
A443 [Decel time 2]	22.20 Sec
A415 [Preset Freq 5]	33.30 Hz
A416 [Preset Freq 6]	44.40 Hz
A417 [Preset Freq 7]	55.50 Hz

Example Controller Tags to Write Multiple Parameters

Operation	Controller Tags for Scattered Write Message	Data Types
XIC	Execute_Scattered_Write_Message	BOOL
MSG	Scattered_Write_Message	MESSAGE

Example Ladder Logic to Write Multiple Parameters

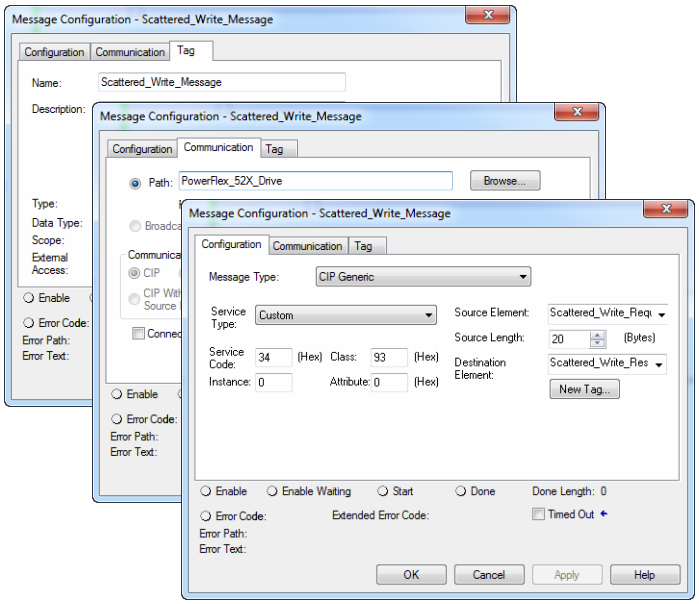


IMPORTANT If the explicit message scattered write must be written continuously, then use a separate explicit message single write for each parameter using DPI Parameter Object Class code 0x93 and attribute A (see CompactLogix – Formatting a Messageto Write a Single Parameter on page 73). Attribute A writes to RAM—not NVS (EEPROM) memory. This example scattered write message using attribute 0 writes to NVS. Over time, continuous writes will exceed the EEPROM life cycle and cause the drive to malfunction.

If you need to make frequent parameter changes using Explicit Messages, set parameter C121 [Comm Write Mode] to 1 “RAM only”.

CompactLogix – Formatting a Message to Write Multiple Parameters

Scattered Write Multiple Message Configuration Screens



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

Configuration Tab	Example Value	Description
Message Type	CIP Generic	Used to access the DPI Parameter Object in the adapter.
Service Type(1)	Custom	Required for scattered messages.
Service Code(1)	0x34 (Hex.)	Code for the requested service.
Class	93(4)	Class ID for the DPI Parameter Object.
Instance	0 (Dec.)	Required for scattered messages.
Attribute(2)	0 (Hex.)	Required for scattered messages.
Source Element	Scattered_Write_Request(5)	Name of the tag for any service data to be sent from the scanner or bridge to the drive.
Source Length	20 bytes(5)	Number of bytes of service data to be sent in the message.
Destination	Scattered_Write_Response(6)	The tag where the data that is read is stored.
Communication Tab	**Example Value**	**Description**
Path(3)	PowerFlex_52X_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 choosing a Service Type other than "Custom" from the pull-down menu, an appropriate Hex. value is automatically assigned to the Service Code box which is dimmed (unavailable).

(2) Scattered writes always write parameter values to the drive's Non-Volatile Storage (EEPROM) memory, so these values 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. **Important**: If you need to make frequent parameter changes using Explicit Messages, set parameter C121 [Comm Write Mode] to 1 "RAM only".

(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, PowerFlex_52X_Drive).

(4) See Explicit Messaging Class Code Compatibility with PowerFlex 525 Drives on page 70 for limitations of PowerFlex 525 drives when using DPI Parameter Object Class code 0x93 for explicit messaging.

(5) In this example, we are writing to five 16-bit parameters. Each parameter being written to requires two contiguous INT registers. Also, the Source Length field on the Message Configuration screen must correspond to the selected Data Type in bytes (for this example, 20 bytes for an array of ten INTs). Scattered write messages always assume that every parameter being written to is a 16-bit parameter, regardless of its actual size. Maximum message length is 256 bytes which can write up to 64 parameters, regardless of their size. For parameter numbering, see DPI Parameter Object on page 141 (Class code 0x93).

(6) The controller tag for "Scattered_Write_Response" must be the same size as the controller tag for "Scattered_Write_Request" (for this example, 20 bytes). An array of INTs is suggested to be able to read any error codes that are returned.

CompactLogix Example Scattered Write Request Data

In this message example, we use the source tag (Scattered_Write_Request) to write new values to these 16-bit parameters:

PowerFlex 525 Drive Parameters	Write Value
A442 [Accel Time 2]	11.10 Sec
A443 [Decel time 2]	22.20 Sec
A415 [Preset Freq 5]	33.30 Hz
A416 [Preset Freq 6]	44.40 Hz
A417 [Preset Freq 7]	55.50 Hz

Example Scattered Write Request Converted Data on page 79 shows the parameter values.

Example Scattered Write Request Converted Data

Name	Value	Data Type	Description
Scattered_Write_Request	{...}	INT[10]	
Scattered_Write_Request[0]	442	INT	Parameter Number
Scattered_Write_Request[1]	1110	INT	Parameter Value
Scattered_Write_Request[2]	443	INT	Parameter Number
Scattered_Write_Request[3]	2220	INT	Parameter Value
Scattered_Write_Request[4]	415	INT	Parameter Number
Scattered_Write_Request[5]	3330	INT	Parameter Value
Scattered_Write_Request[6]	416	INT	Parameter Number
Scattered_Write_Request[7]	4440	INT	Parameter Value
Scattered_Write_Request[8]	417	INT	Parameter Number
Scattered_Write_Request[9]	5550	INT	Parameter Value

CompactLogix Example Scattered Write Response Data

The results of the message appear in the destination tag named Scattered_Write_Response (Example Scattered Write Response Data on page 79). Values of "0" indicate no errors occurred.

Example Scattered Write Response Data

Name	Value	Data Type	Description
Scattered_Write_Response	{...}	INT[10]	
Scattered_Write_Response[0]	442	INT	Parameter Number
Scattered_Write_Response[1]	0	INT	Error Code
Scattered_Write_Response[2]	443	INT	Parameter Number
Scattered_Write_Response[3]	0	INT	Error Code
Scattered_Write_Response[4]	415	INT	Parameter Number
Scattered_Write_Response[5]	0	INT	Error Code
Scattered_Write_Response[6]	416	INT	Parameter Number
Scattered_Write_Response[7]	0	INT	Error Code
Scattered_Write_Response[8]	417	INT	Parameter Number
Scattered_Write_Response[9]	0	INT	Error Code

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

The data structures in Data Structures for Scattered Read Messages on page 80 and Data Structures for Scattered Write Messages on page 81 use 16-bit words and can accommodate up to 64 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 (parameter number in response data will be negative).

The PowerFlex 525 Adjustable Frequency AC Drive User Manual, publication 520-UM001 lists the data type for each parameter.

Data Structures for Scattered Read Messages

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

Data Structures for Scattered Write Messages

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

Notes:

Using Multi-Drive Mode

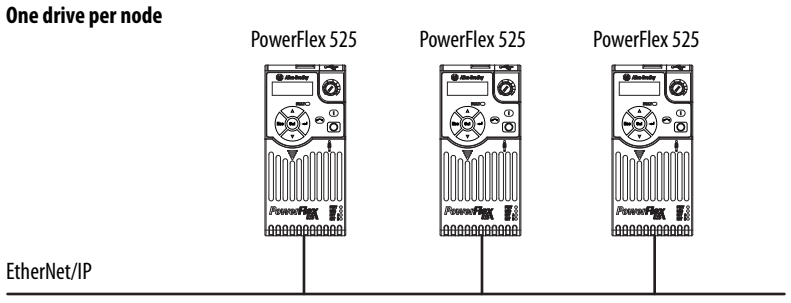
This chapter provides instructions on how to configure a CompactLogix controller to use the PowerFlex 525 drive in Multi-drive mode.

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.

Single-Drive Mode vs. Multi-Drive Mode

Single-drive mode is a typical network installation, where a single EtherNet/IP node consists of a single drive with an EtherNet/IP adapter.

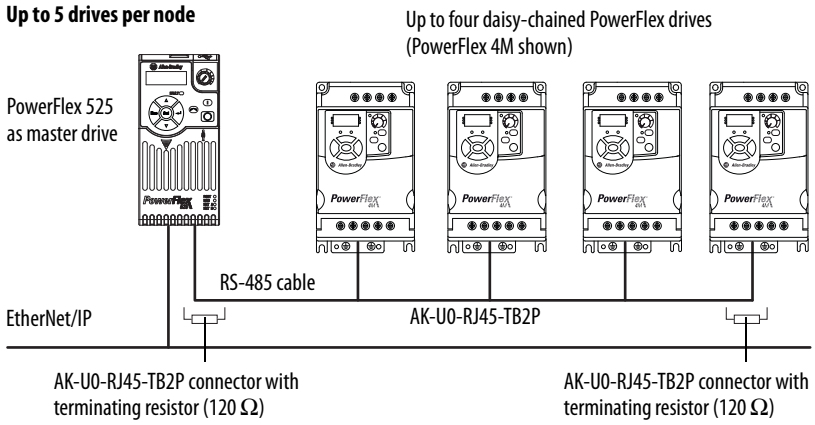
Single-Drive Mode Example for Network



Multi-drive mode is an alternative to the typical network installation, where a single EtherNet/IP node can consist of one to five drives (see Multi-Drive Mode Example for Network on page 84). The first drive must be a PowerFlex 525 drive. The remaining drives can be any PowerFlex drive which supports Multi-drive mode.

IMPORTANT For the examples in the chapter, we will use the PowerFlex 525 as a master drive with four daisy-chained PowerFlex 4M drives.

Multi-Drive Mode Example for Network



Benefits of Multi-drive mode include:

- Lower hardware costs. No need to purchase additional communication adapters for daisy-chained drives.
- Reduces the network node count. For example, in Single-drive mode 30 drives would consume 30 nodes. In Multi-drive mode, 30 drives can be connected in 6 nodes.
- Controller can control, monitor, and read/write parameters for all five drives.

The trade-offs of Multi-drive mode include:

- If the PowerFlex 525 with embedded EtherNet/IP adapter is powered down, then communications with the daisy-chained drives is disrupted and the drives will take the appropriate communications loss action set in each drive.
- Communications throughput to the daisy-chained drives will be slower than if each drive was a separate node on EtherNet/IP (Single-drive mode). This is because the embedded EtherNet/IP adapter must take the EtherNet/IP data for the other drives and sequentially send the respective data to each drive over RS-485. The approximate additional throughput time for Logic Command/Reference to be transmitted and received by each drive is:

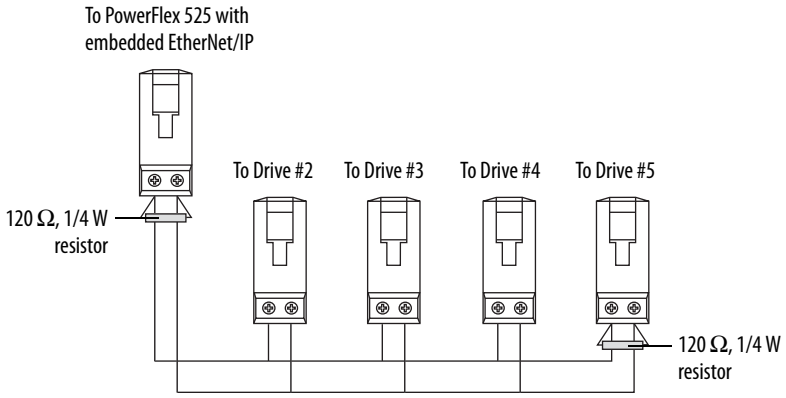
Drive	Additional Throughput Time versus Single-Drive Mode
PowerFlex 525	0 ms
PowerFlex 525 plus 1 drive	+24 ms
PowerFlex 525 plus 2 drives	+48 ms
PowerFlex 525 plus 3 drives	+72 ms
PowerFlex 525 plus 4 drives	+96 ms

- Since the RS-485 ports are used for daisy-chaining the drives, there is no connection for a peripheral device such as a HIM or USB converter module (1203-USB). DSI Splitter cables cannot be used to add a second connection for a peripheral device.

System Wiring

To daisy-chain the drives of the PowerFlex 525, the AK-U0-RJ45-TB2P terminal block connector can be used for easy installation.

The wiring diagram for using AK-U0-RJ45-TB2P terminal block connectors is shown below.



The AK-U0-RJ45-TB2P comes with (5) terminal block connectors and (2) terminating resistors.

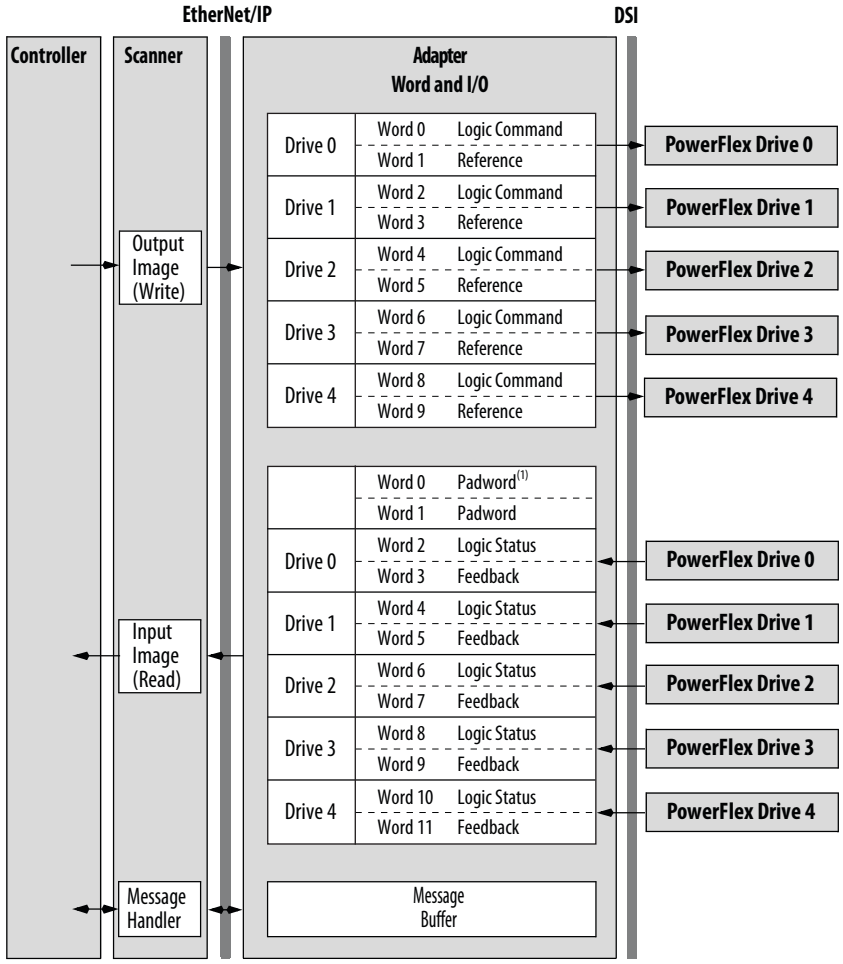
Understanding the I/O Image

The terms *input* and *output* are defined from the scanner's point of view. Therefore, Output I/O is data that is output from the scanner and consumed by the EtherNet/IP adapter. Input I/O is status data that is produced by the adapter and consumed as input by the scanner.

The I/O image table will vary based on the configuration of parameters **C169 [MultiDrv Sel]** and **C175 [DSI I/O Cfg]**. The image table always uses consecutive words starting at word 0.

The Multi-Drive Example of I/O Image on page 86 is an illustration of the Multi-drive I/O image with 16-bit words.

Multi-Drive Example of I/O Image



(1) Padwords only apply when using the generic profile. Padwords are not used when using the full-featured drive Add-On-Profile.

Note: If a daisy-chained drive is disconnected from the RS-485 (DSI) network or powered down, the Logic Status and Feedback words for the affected drive will be set to 0.

Configuring the RS-485 Network

The following parameters must be set in the daisy-chained PowerFlex 4M drives and **not** in the master drive:

Parameter	Value
P106 [Start Source]	5 “Comm Port”
P108 [Speed Reference]	5 “Comm Port”
C302 [Comm Data Rate]	4 “19.2K”
C303 [Comm Node Addr]	1...247 (must be unique)
C306 [Comm Format]	0 “RTU-8-N-1”

Note: The RS-485 Multi-drive network is fixed at 19.2K baud rate, 8 data bits, no parity, and 1 stop bit.

IMPORTANT Parameters [Comm Loss Action] and [Comm Loss Time] in the daisy-chained drives are still used in Multi-drive mode. If the RS-485 cable is disconnected or broken, the disconnected drive(s) will take the corresponding Comm Loss Action(s). On the EtherNet/ IP side, parameters C143 [EN Comm Flt Actn] and C144 [EN Idle Flt Actn] determine the action taken for ALL of the drives on the Multi-drive node.

The following Multi-drive parameters must be set in the master PowerFlex 525 drive:

Parameter	Value
P046 [Start Source 1]	5 "EtherNet/IP"
P047 [Speed Reference1]	15 "EtherNet/IP"
C169 [MultiDrv Sel]	0 "Disabled" 1 "Network Opt" 2 "EtherNet/IP" **Note**: Drive must be power cycled after setting this parameter.
C171 [Drv 1 Addr]	C124 [RS485 Node Addr] in Drive 1
C172 [Drv 2 Addr]	C124 [RS485 Node Addr] in Drive 2
C173 [Drv 3 Addr]	C124 [RS485 Node Addr] in Drive 3
C174 [Drv 4 Addr]	C124 [RS485 Node Addr] in Drive 4
C175 [DSI I/O Cfg]	0 "Drive 0" 1 "Drive 0-1" 2 "Drive 0-2" 3 "Drive 0-3" 4 "Drive 0-4"

IMPORTANT Parameters can be set using a DSI peripheral (22-HIM-A3 or 22-HIM-C2S) only when parameter C169 [MultiDrv Sel] is set to 0 "Disabled".

Using Multi-Drive Add-On Profile

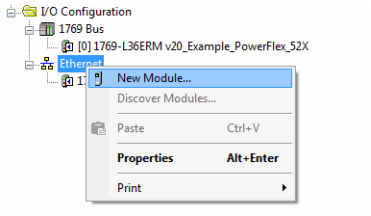
Before using the Multi-drive Add-On Profile, ensure that you have completed the following steps:

- The master and daisy-chained drives are powered, networked, and configured. See System Wiring on page 85.
- The RS-485 network is configured. See Configuring the RS-485 Network on page 86.
- The controller has been added to the I/O configuration. See Adding the Controller to the I/O Configuration on page 38.

Adding the Drive to the I/O Configuration

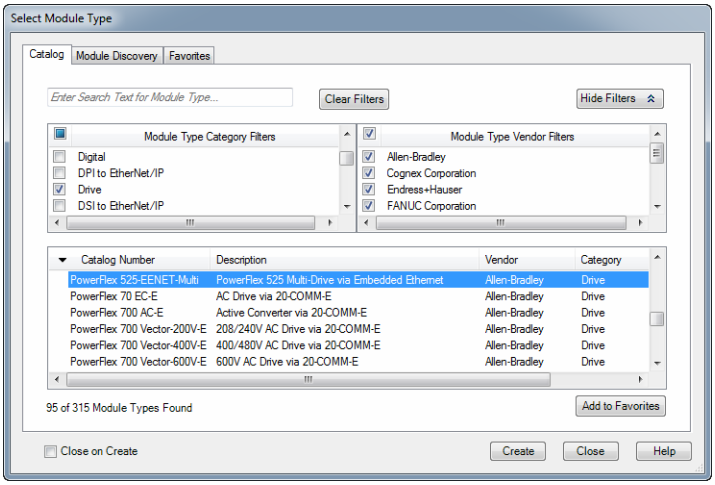
To transmit data between the controller and the drive, you must add the drive as a child device to the parent controller. In this example, RSLogix 5000 version 20 is used with drive Add-On Profile version 1.01 or later.

1. In the treeview, right-click on the Ethernet icon and select **New Module...** to display the Select Module window. Expand the Drives group to display all of the available drives with their communication adapters.

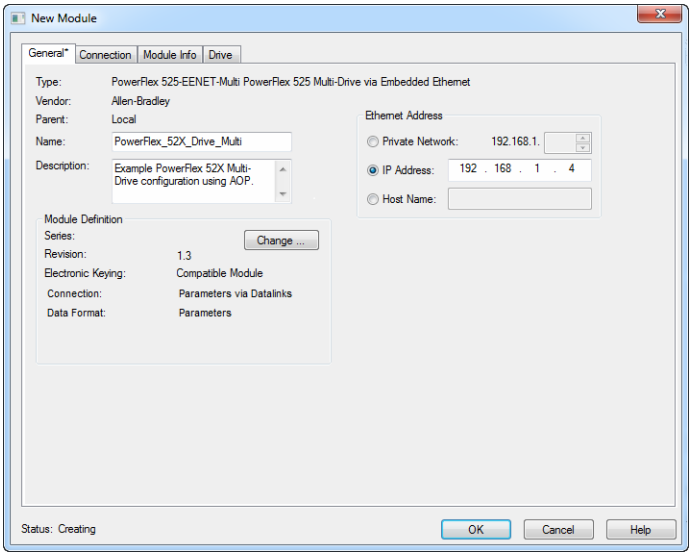


TIP If the PowerFlex drive is not shown, go to www.ab.com/support/abdrives/webupdate and download the latest drive Add-On Profile.

2. In the Select Module Type window, select the drive and its connected adapter from the list. For this example, we selected "PowerFlex 525-EENET-Multi." Then click **Create**. The drive's New Module window appears.

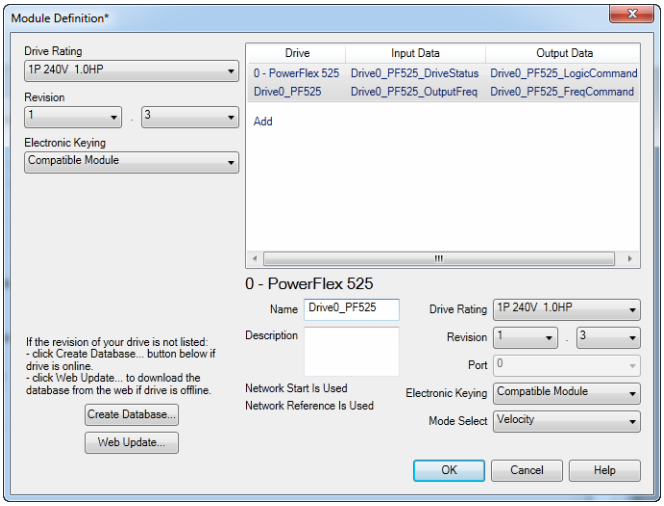


3. On the General tab, edit the following data about the drive:

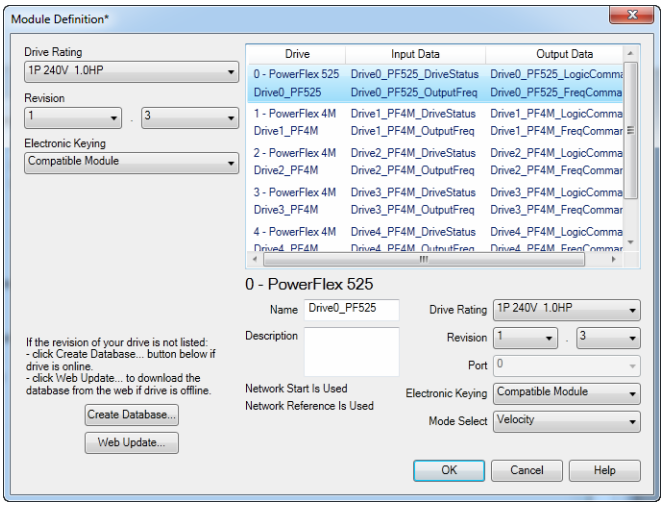


Box	Setting
Name	A name to identify the drive.
Description	Optional – description of the drive/adapter.
IP Address	The IP address of the adapter.

4. On the New Module window in the Module Definition section, click **Change...** to launch the Module Definition window and begin the drive configuration process.



5. In the Module Definition window, edit the following information for the master drive:



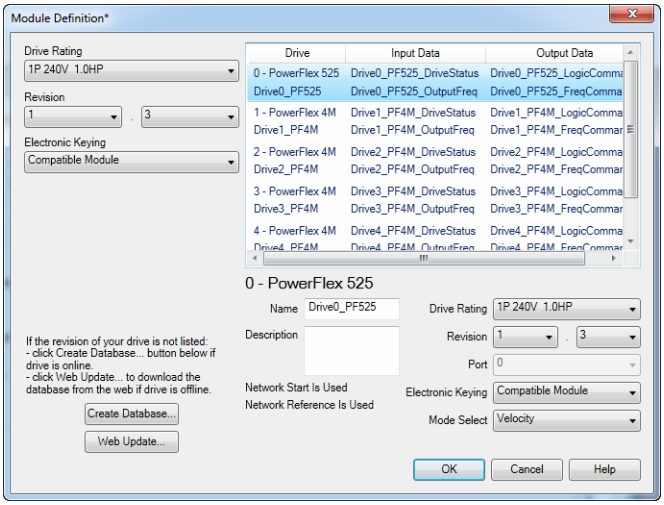
TIP You may create a database from a network accessible drive using the **Create Database...** button (Recommended).

Box	Setting
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 drive rating, use the **Create Database…**, or **Web Update…** button described above.
Revision	The major and minor revision of the firmware (database) 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 revision, use one of the following buttons at the bottom left of the Module Definition window: • **Create Database…** Creates a database from an online network multi-drive. Clicking this button displays an RSLinx RSWho window. Browse to the online drive (PowerFlex 525), select it, and click OK. The database will be uploaded and stored on the computer. Thereafter, close the Module Definition window and then re-open it to display the new revision. • **Web Update…** When a drive is not available online, opens the Allen-Bradley Drives Web Updates web site to download a specific database file. After downloading the file, close the Module Definition window and then re-open it to display the new revision.
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 window. See the online Help for additional information on this and other Electronic Keying settings. If keying is not required, select "Disable Keying." Drives do not require keying, and so "Disable Keying" is recommended.

On the Module Definition window, notice that the automatically-assigned controller tags Drive Status, Feedback, Logic Command, and Reference are always used.

IMPORTANT The Velocity/Positioning mode select is in the lower right of the window when the master PowerFlex 525 drive is selected.

6. Click **Add** to select and define each daisy-chained drive:

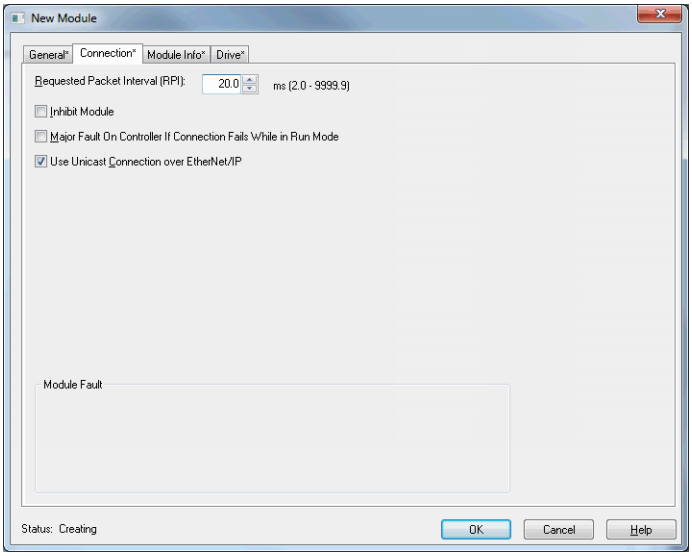


TIP You may create a database from a network accessible drive using the **Create Database...** button (Recommended).

Box	Setting
Name	A name to identify an individual drive.
Description	Optional – description of an individual drive.
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 drive rating, use the **Create Database…**, or **Web Update…** button described above.
Revision	The major and minor revision of the firmware (database) 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 revision, use one of the following buttons at the bottom left of the Module Definition window: • **Create Database…** Creates a database from an online network multi-drive. Clicking this button displays an RSLinx RSWho window. Browse to the online drive (PowerFlex 525), select it, and click OK. The database will be uploaded and stored on the computer. Thereafter, close the Module Definition window and then re-open it to display the new revision. • **Web Update…** When a drive is not available online, opens the Allen-Bradley Drives Web Updates web site to download a specific database file. After downloading the file, close the Module Definition window and then re-open it to display the new revision.
Port	Port assignment for each drive
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 window. See the online Help for additional information on this and other Electronic Keying settings. If keying is not required, select "Disable Keying." Drives do not require keying, and so "Disable Keying" is recommended.
Mode Select	Sets the I/O configuration to either Velocity or Position mode.

7. Click **OK** on the Module Definition window to save the drive configuration and close the window. The drive's New Module window reappears.

8. On the New Module window, click the Connection tab.



9. In the "Requested Packet Interval (RPI)" box, set the value to 5.0 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/Logix Designer 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.

10. Click **OK** on the New Module window.
    The new node ("PowerFlex 525-EENET-Multi PowerFlex_52X_Drive-Multi" in this example) now appears under the Ethernet icon in the I/O Configuration folder. If you double-click on the Input Controller Tag (Controller Input Tags on page 93) and Output Controller Tag (Controller Output Tags on page 94), you will see that module-defined data types and tags have been automatically created. Note that all tag names are defined for each drive. After you save and download the configuration, these tags allow you to access the Input and Output data of the drives using the controller's ladder logic.

Controller Input Tags

Name	Value	Data Type	Description
PowerFlex_52X_Drive_Multi:I	{...}	AB:PowerFlex5...	
PowerFlex_52X_Drive_Multi:I.Drive0_PF525_DriveStatus	2#0000_00...	INT	
PowerFlex_52X_Drive_Multi:I.Drive0_PF525_Ready	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive0_PF525_Active	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive0_PF525_CommandDir	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive0_PF525_ActualDir	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive0_PF525_Accelerating	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive0_PF525_Decelerating	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive0_PF525_Faulted	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive0_PF525_AtReference	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive0_PF525_CommFreqCnt	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive0_PF525_CommLogicCnt	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive0_PF525_ParmsLocked	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive0_PF525_DigIn1Active	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive0_PF525_DigIn2Active	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive0_PF525_DigIn3Active	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive0_PF525_DigIn4Active	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive0_PF525_OutputFreq	0	INT	
PowerFlex_52X_Drive_Multi:I.Drive1_PF4M_DriveStatus	2#0000_00...	INT	
PowerFlex_52X_Drive_Multi:I.Drive1_PF4M_Ready	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive1_PF4M_Active	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive1_PF4M_CommandDir	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive1_PF4M_ActualDir	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive1_PF4M_Accelerating	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive1_PF4M_Decelerating	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive1_PF4M_Alarm	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive1_PF4M_Faulted	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive1_PF4M_AtReference	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive1_PF4M_CommFreqCnt	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive1_PF4M_CommLogicCnt	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive1_PF4M_ParmsLocked	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive1_PF4M_DigIn1Active	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive1_PF4M_DigIn2Active	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive1_PF4M_OutputFreq	0	INT	
PowerFlex_52X_Drive_Multi:I.Drive2_PF4M_DriveStatus	2#0000_00...	INT	
PowerFlex_52X_Drive_Multi:I.Drive2_PF4M_Ready	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive2_PF4M_Active	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive2_PF4M_CommandDir	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive2_PF4M_ActualDir	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive2_PF4M_Accelerating	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive2_PF4M_Decelerating	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive2_PF4M_Alarm	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive2_PF4M_Faulted	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive2_PF4M_AtReference	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive2_PF4M_CommFreqCnt	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive2_PF4M_CommLogicCnt	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive2_PF4M_ParmsLocked	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive2_PF4M_DigIn1Active	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive2_PF4M_DigIn2Active	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive2_PF4M_OutputFreq	0	INT	
PowerFlex_52X_Drive_Multi:I.Drive3_PF4M_DriveStatus	2#0000_00...	INT	
PowerFlex_52X_Drive_Multi:I.Drive3_PF4M_Ready	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive3_PF4M_Active	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive3_PF4M_CommandDir	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive3_PF4M_ActualDir	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive3_PF4M_Accelerating	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive3_PF4M_Decelerating	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive3_PF4M_Alarm	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive3_PF4M_Faulted	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive3_PF4M_AtReference	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive3_PF4M_CommFreqCnt	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive3_PF4M_CommLogicCnt	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive3_PF4M_ParmsLocked	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive3_PF4M_DigIn1Active	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive3_PF4M_DigIn2Active	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive3_PF4M_OutputFreq	0	INT	
PowerFlex_52X_Drive_Multi:I.Drive4_PF4M_DriveStatus	2#0000_00...	INT	
PowerFlex_52X_Drive_Multi:I.Drive4_PF4M_Ready	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive4_PF4M_Active	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive4_PF4M_CommandDir	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive4_PF4M_ActualDir	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive4_PF4M_Accelerating	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive4_PF4M_Decelerating	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive4_PF4M_Alarm	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive4_PF4M_Faulted	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive4_PF4M_AtReference	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive4_PF4M_CommFreqCnt	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive4_PF4M_CommLogicCnt	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive4_PF4M_ParmsLocked	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive4_PF4M_DigIn1Active	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive4_PF4M_DigIn2Active	0	BOOL	
PowerFlex_52X_Drive_Multi:I.Drive4_PF4M_OutputFreq	0	INT	

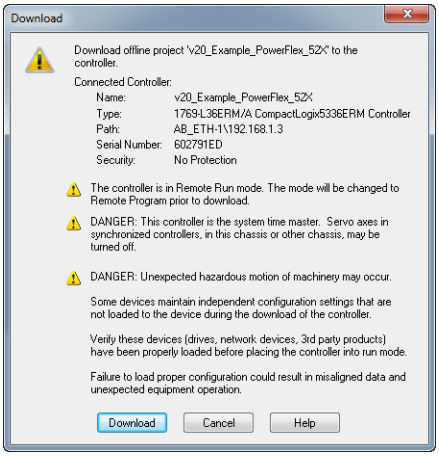
Controller Output Tags

Name	Value	Data Type	Description
PowerFlex_52X_Drive_Multi:O	{...}	AB:PowerFlex5...	
PowerFlex_52X_Drive_Multi:O.Drive0_PF525_LogicCommand	2#0000_00...	INT	
PowerFlex_52X_Drive_Multi:O.Drive0_PF525_Stop	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive0_PF525_Start	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive0_PF525_Jog	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive0_PF525_ClearFaults	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive0_PF525_Forward	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive0_PF525_Reverse	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive0_PF525_ForceKeypadCtrl	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive0_PF525_MOPIncrement	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive0_PF525_AccelRate1	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive0_PF525_AccelRate2	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive0_PF525_DecelRate1	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive0_PF525_DecelRate2	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive0_PF525_FreqSel01	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive0_PF525_FreqSel02	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive0_PF525_FreqSel03	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive0_PF525_MOPDecrement	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive0_PF525_FreqCommand	0	INT	
PowerFlex_52X_Drive_Multi:O.Drive1_PF4M_LogicCommand	2#0000_00...	INT	
PowerFlex_52X_Drive_Multi:O.Drive1_PF4M_Stop	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive1_PF4M_Start	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive1_PF4M_Jog	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive1_PF4M_ClearFaults	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive1_PF4M_Forward	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive1_PF4M_Reverse	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive1_PF4M_RelayOutControl	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive1_PF4M_MOPIncrement	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive1_PF4M_AccelRate1	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive1_PF4M_AccelRate2	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive1_PF4M_DecelRate1	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive1_PF4M_DecelRate2	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive1_PF4M_FreqSel01	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive1_PF4M_FreqSel02	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive1_PF4M_FreqSel03	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive1_PF4M_MOPDecrement	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive1_PF4M_FreqCommand	0	INT	
PowerFlex_52X_Drive_Multi:O.Drive2_PF4M_LogicCommand	2#0000_00...	INT	
PowerFlex_52X_Drive_Multi:O.Drive2_PF4M_Stop	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive2_PF4M_Start	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive2_PF4M_Jog	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive2_PF4M_ClearFaults	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive2_PF4M_Forward	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive2_PF4M_Reverse	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive2_PF4M_RelayOutControl	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive2_PF4M_MOPIncrement	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive2_PF4M_AccelRate1	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive2_PF4M_AccelRate2	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive2_PF4M_DecelRate1	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive2_PF4M_DecelRate2	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive2_PF4M_FreqSel01	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive2_PF4M_FreqSel02	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive2_PF4M_FreqSel03	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive2_PF4M_MOPDecrement	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive2_PF4M_FreqCommand	0	INT	
PowerFlex_52X_Drive_Multi:O.Drive3_PF4M_LogicCommand	2#0000_00...	INT	
PowerFlex_52X_Drive_Multi:O.Drive3_PF4M_Stop	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive3_PF4M_Start	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive3_PF4M_Jog	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive3_PF4M_ClearFaults	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive3_PF4M_Forward	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive3_PF4M_Reverse	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive3_PF4M_RelayOutControl	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive3_PF4M_MOPIncrement	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive3_PF4M_AccelRate1	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive3_PF4M_AccelRate2	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive3_PF4M_DecelRate1	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive3_PF4M_DecelRate2	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive3_PF4M_FreqSel01	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive3_PF4M_FreqSel02	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive3_PF4M_FreqSel03	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive3_PF4M_MOPDecrement	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive3_PF4M_FreqCommand	0	INT	
PowerFlex_52X_Drive_Multi:O.Drive4_PF4M_LogicCommand	2#0000_00...	INT	
PowerFlex_52X_Drive_Multi:O.Drive4_PF4M_Stop	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive4_PF4M_Start	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive4_PF4M_Jog	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive4_PF4M_ClearFaults	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive4_PF4M_Forward	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive4_PF4M_Reverse	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive4_PF4M_RelayOutControl	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive4_PF4M_MOPIncrement	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive4_PF4M_AccelRate1	0	BOOL	
PowerFlex_52X_Drive_Multi:O.Drive4_PF4M_AccelRate2	0	BOOL	

Saving the I/O Configuration to the Controller

After adding the controller and drives 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/Logix Designer window, select **Communications** > **Download**. The Download dialog box appears.



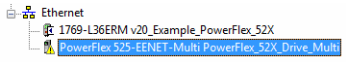
TIP If a message box reports that RSLogix 5000/Logix Designer is unable to go online, select **Communications** > **Who Active** to find your controller in the Who Active screen. After finding and selecting the controller, click **Set Project Path** to establish the path. If your controller does not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

2. Click **Download** to download the configuration to the controller. When the download is successfully completed, RSLogix 5000/Logix Designer goes into 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.
3. If the controller was in Run Mode before clicking **Download**, RSLogix 5000/Logix Designer 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.
4. 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.
5. To ensure that the present project configuration values are saved, RSLogix 5000/Logix Designer prompts you to upload them. Click **Yes** to upload and save them.

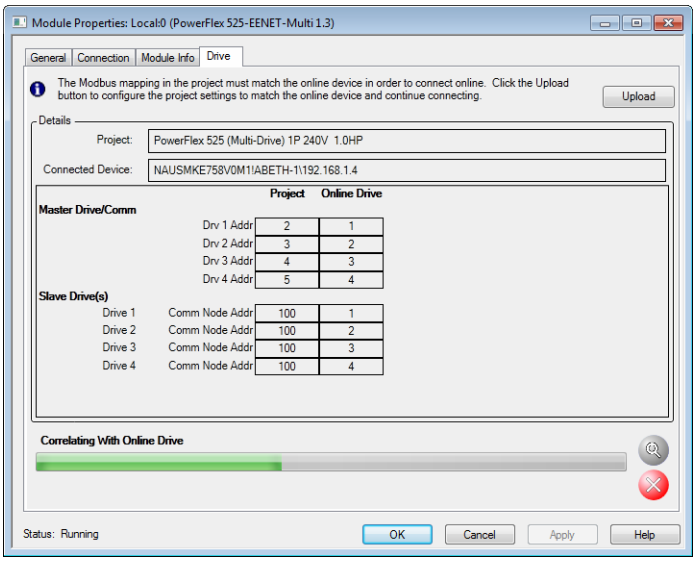
Correlating the Drive with the Controller

You must now correlate the drive settings to the RSLogix 5000/Logix Designer 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 "PowerFlex 525-EENET PowerFlex_52X_Drive") and select Properties.



2. Select the Drive tab to begin the correlation process.

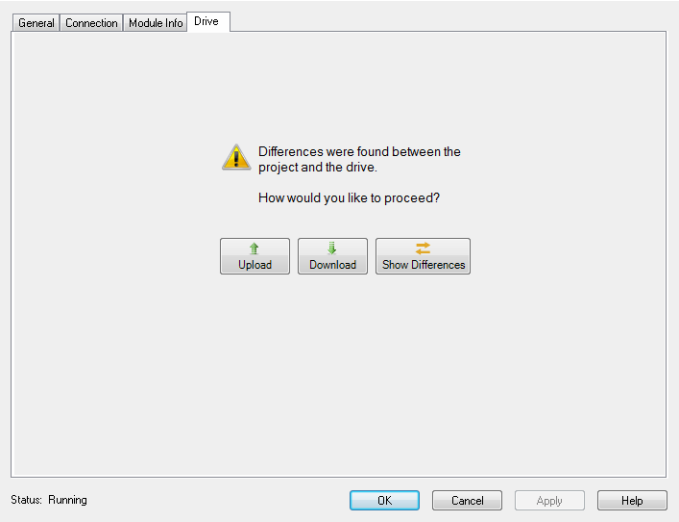


After the drive configuration data has been verified, the Drive tab will display a request to synchronize the configuration with the drive. Click **Upload**. The correlation process will continue. This may take several minutes depending on the number and type of daisy-chained drives.

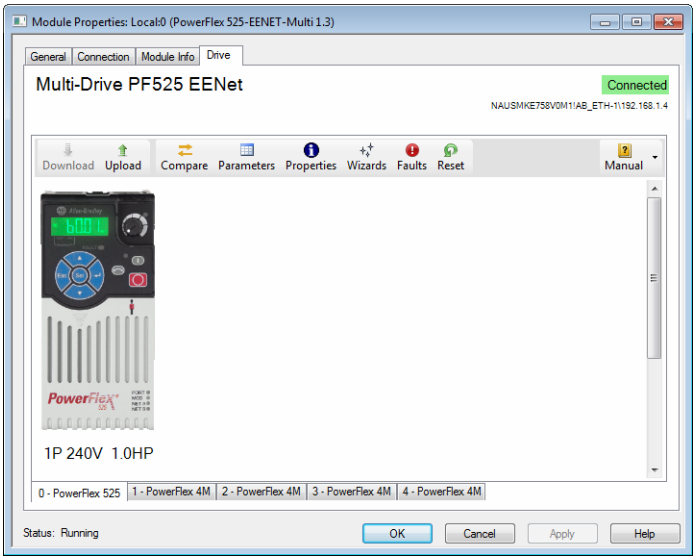
If the Differences Found Screen on page 97 appears—which is typical, click **Download**. This will download the project settings from the controller to the drives. If **Upload** is clicked, the drive settings are uploaded to the controller.

TIP On subsequent connections to the drive (after the initial download), select **Upload**.

Differences Found Screen



3. The Drive tab displays a screen of the drive.



If the download is successful, the Drive tab will show a green Connected indicator in the upper right corner of the window. This tab is extremely useful for configuring drive parameters, accessing start-up wizards and troubleshooting. Note that there is a tab for the master as well as each of the daisy-chained drives.

TIP You may now use the automatically generated tags to create your controller logic.

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

Multi-Drive Ladder Logic Program for Generic Profile

The following is an example of the ladder logic program for the Generic Profile and demonstrates using Multi-drive mode with five drives. See Multi-Drive Mode Example for Network on page 84 for an example of a system layout

diagram. If you have not set-up your drive using the Generic Profile, see Using the RSLogix 5000 (all versions) or Logix Designer (version 21 or greater) Generic Profile on page 55 for instructions.

See Multi-Drive Example of I/O Image on page 86 for the number of 16-bit input and output words to use for your application. In this example, the number of input words is 12 and the number of output words is 10.

Function of the Example Program

The example program provided is for the CompactLogix family, but other Logix-based controllers can also be used similarly. This example program enables you to:

- View status information from the drives such as Ready, Fault, At Speed, and Feedback.
- Control the drives using various Logic Command bits (Stop, Start, etc.) and Reference.
- Perform a single parameter read and write for each drive. The example uses PowerFlex 4M drive parameter **P109 [Accel Time 1]** for both so you can see (read) the change after a write is performed.

The same programming approach can be used with the tags generated by the Multi-drive Add-On Profile. Note that the tags used in this example will be different from those created by the Multi-drive Add-On Profile.

Drive 0 (PowerFlex 525) Settings for the Example Program

- Parameter **C169 [MultiDrv Sel]** is set to 2 “EtherNet/IP”.
- The following parameters are set:

Parameter	Value	Description
P046 [Start Source1]	5	“EtherNet/IP”
P047 [Speed Reference1]	15	“EtherNet/IP”
C175 [DSI I/O Cfg]	4	“Drive 0-4” (5 drives on 1 node)
C171 [Drv 1 Addr](1)	1	Modbus address of Drive 1
C172 [Drv 2 Addr]	2	Modbus address of Drive 2
C173 [Drv 3 Addr]	3	Modbus address of Drive 3
C174 [Drv 4 Addr]	4	Modbus address of Drive 4

(1) The settings for these parameters must match the node address settings in the respective daisy-chained drives.

Drive 1...4 (PowerFlex 4M) Settings for the Example Program

The following parameters are set:

Parameter	Value			
	Drive 1	Drive 2	Drive 3	Drive 4
P106 [Start Source]	5	5	5	5
P108 [Speed Reference]	5	5	5	5
C302 [Comm Data Rate]	4	4	4	4
C303 [Comm Node Addr]	1	2	3	4
C304 [Comm Loss Action]	0	0	0	0
C305 [Comm Loss Time]	5.0 s	5.0 s	5.0 s	5.0 s
C306 [Comm Format]	0	0	0	0

CompactLogix Example Using Generic Profile

The following common Tags are used:

Tag Name	Type	Description
PowerFlex_52X_Drive_Multi_Generic:O	AB:ETHERNET_MODULE_xxx:O:0	Generic EtherNet/IP module I/O tags and configuration
PowerFlex_52X_Drive_Multi_Generic:I	AB:ETHERNET_MODULE_xxx:I:0	
PowerFlex_52X_Drive_Multi_Generic:C	AB:ETHERNET_MODULE:C:0	
Accel_Time_1	INT	–
Drive_Input_Image	INT [12]	Input Image Table
Drive_Output_Image	INT [10]	Output Image Table

The following Tags are used for Drive 0:

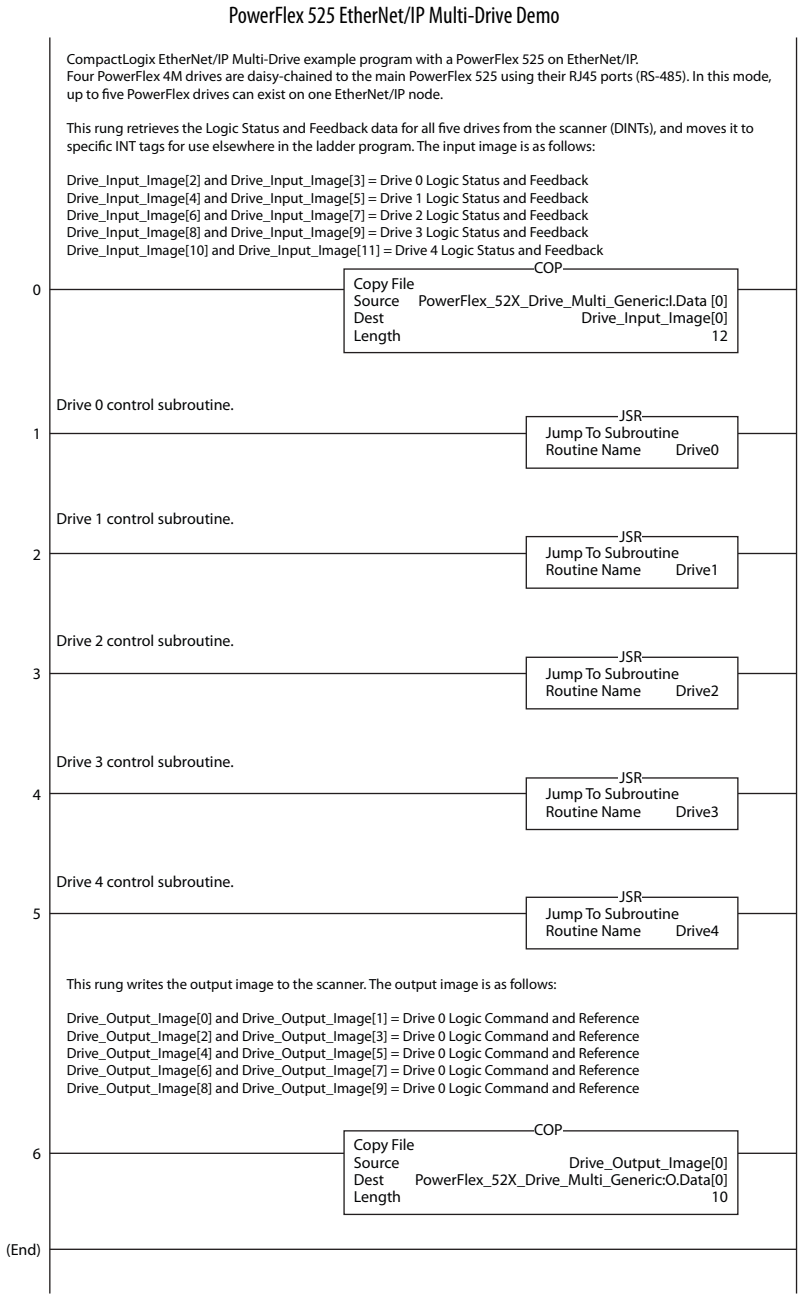
Tag Name	Type	Description
Drive_0_Command_Stop	BOOL	Logic Command bit 0 (STOP)
Drive_0_Command_Start	BOOL	Logic Command bit 1 (START)
Drive_0_Command_Jog	BOOL	Logic Command bit 2 (JOG)
Drive_0_Command_Clear_Faults	BOOL	Logic Command bit 3 (CLEAR FAULTS)
Drive_0_Command_Forward	BOOL	Logic Command bit 4 (FORWARD)
Drive_0_Reference	INT	Speed Reference
Drive_0_Status_Ready	BOOL	Logic Status bit 0 (READY)
Drive_0_Status_Active	BOOL	Logic Status bit 1 (ACTIVE)
Drive_0_Status_Forward	BOOL	Logic Status bit 2 (FORWARD)
Drive_0_Status_Faulted	BOOL	Logic Status bit 7 (FAULT)
Drive_0_Status_At_Reference	BOOL	Logic Status bit 8 (AT SPEED)
Drive_0_Feedback	INT	Speed Feedback
Perform_Parameter_Read_0	BOOL	Initiates the parameter read
Parameter_RD_Value_0	INT	Read value of the parameter
Parameter_RD_Message_0	MESSAGE	Get_Attribute_Single (Read)
Perform_Parameter_Write_0	BOOL	Initiates the parameter value
Parameter_WR_Value_0	INT	Write value to the parameter
Parameter_WR_Message_0	MESSAGE	Set_Attritbute_Single (Write)

The same type of Tags are also used for Drive 1 through Drive 4.

Main Routine

The Main Routine reads the network Input Image from the scanner, calls the various drive control subroutines, and writes the network Output Image to the scanner. See Main Routine on page 100.

Main Routine

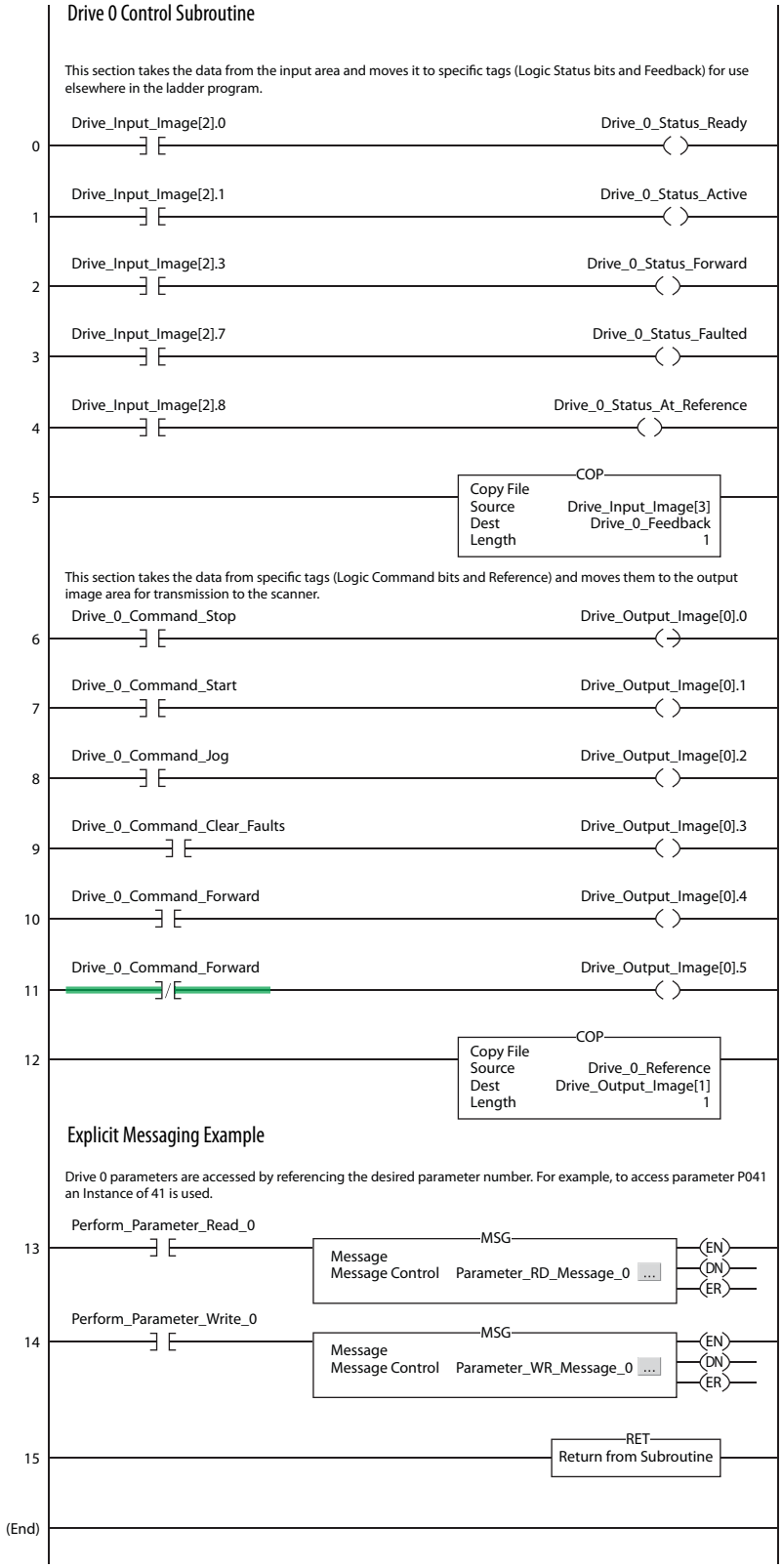


Drive 0...4 Control Routines

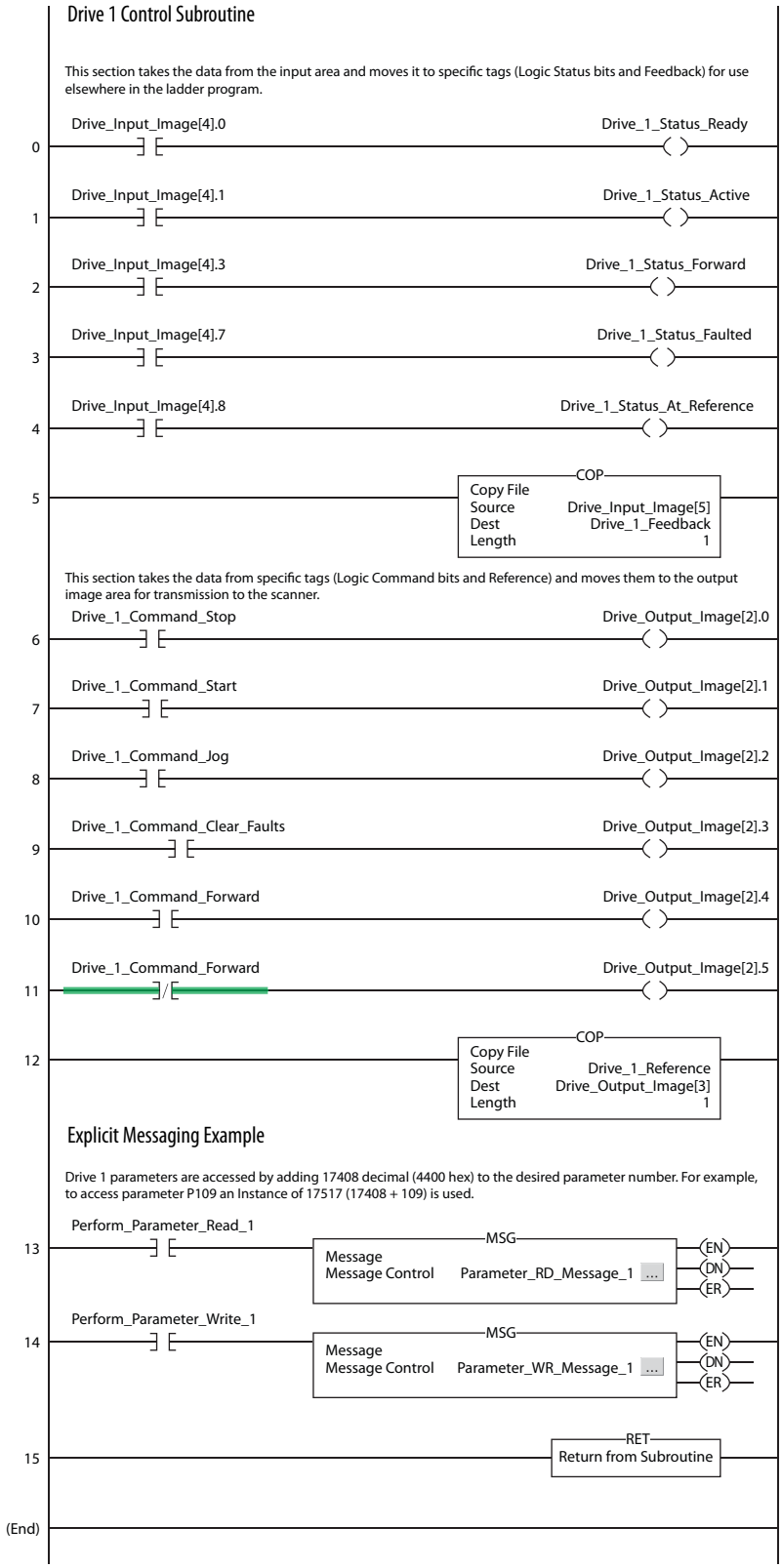
The following Drive Control routines provide status information (Logic Status and Feedback), control (Logic Command and Reference), and parameter read/write for each of the respective drives:

Control Routine	See page...
Drive 0	102
Drive 1	103
Drive 2	104
Drive 3	105
Drive 4	106

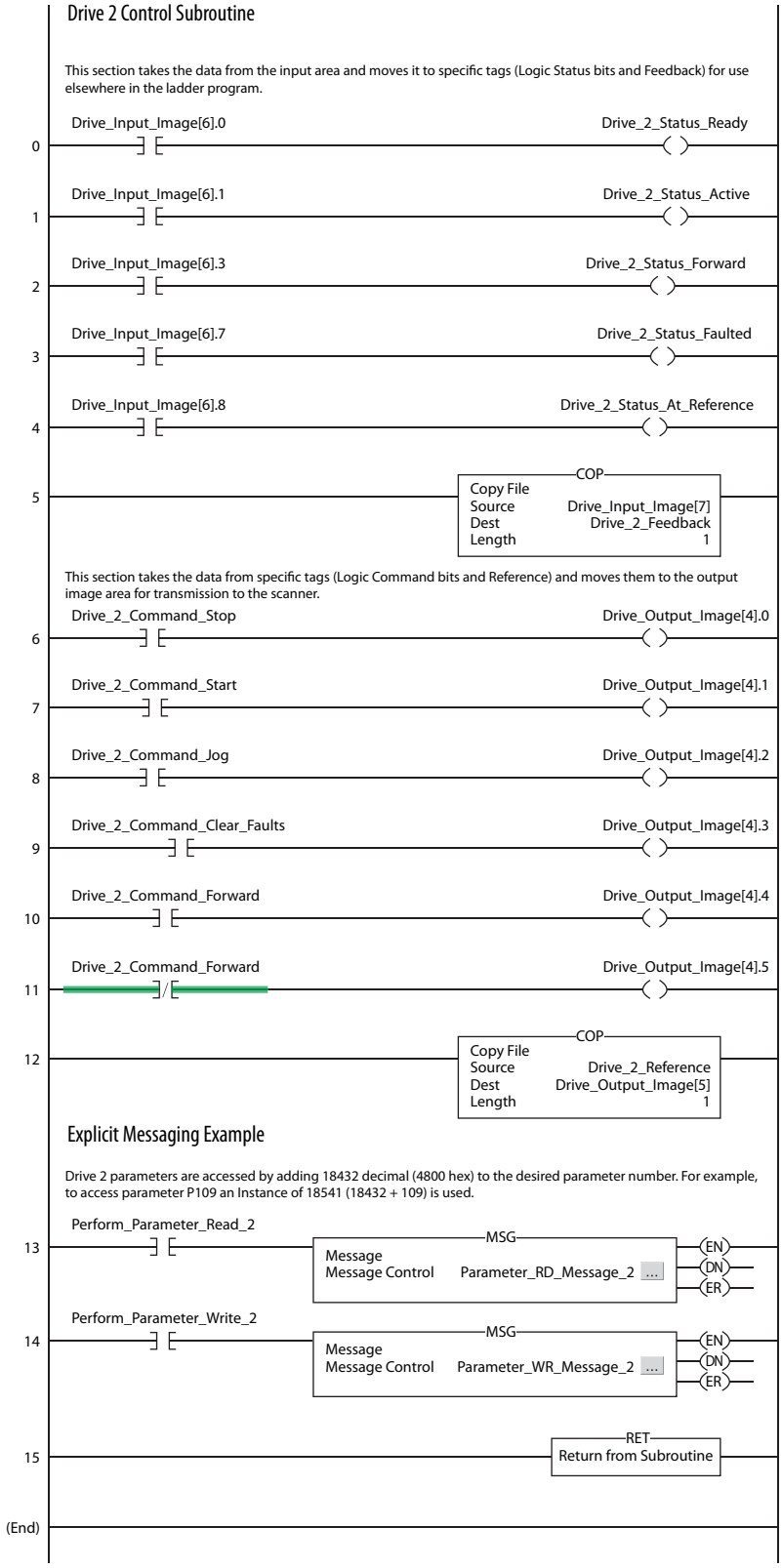
Drive 0 Control Routine



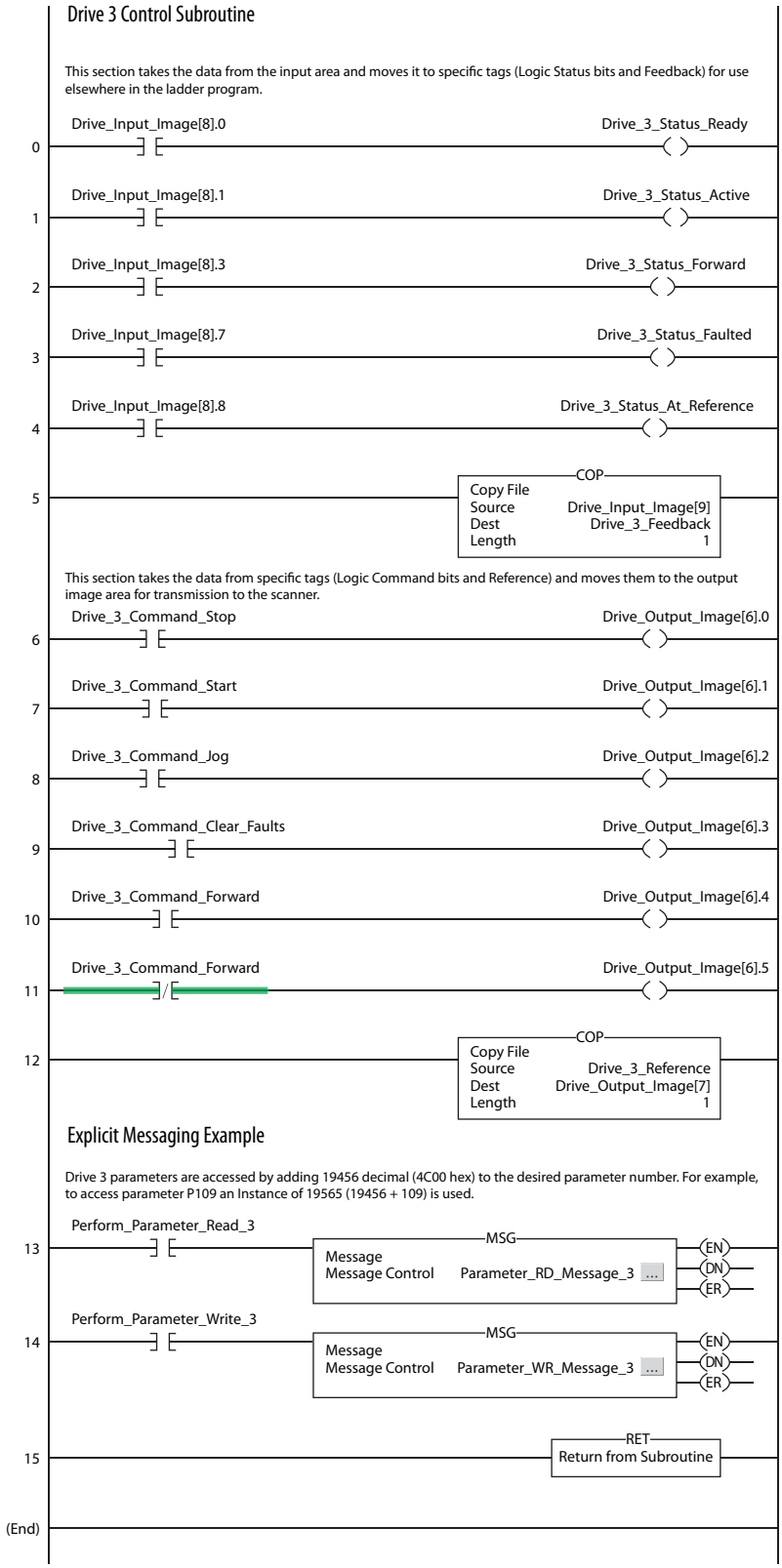
Drive 1 Control Routine



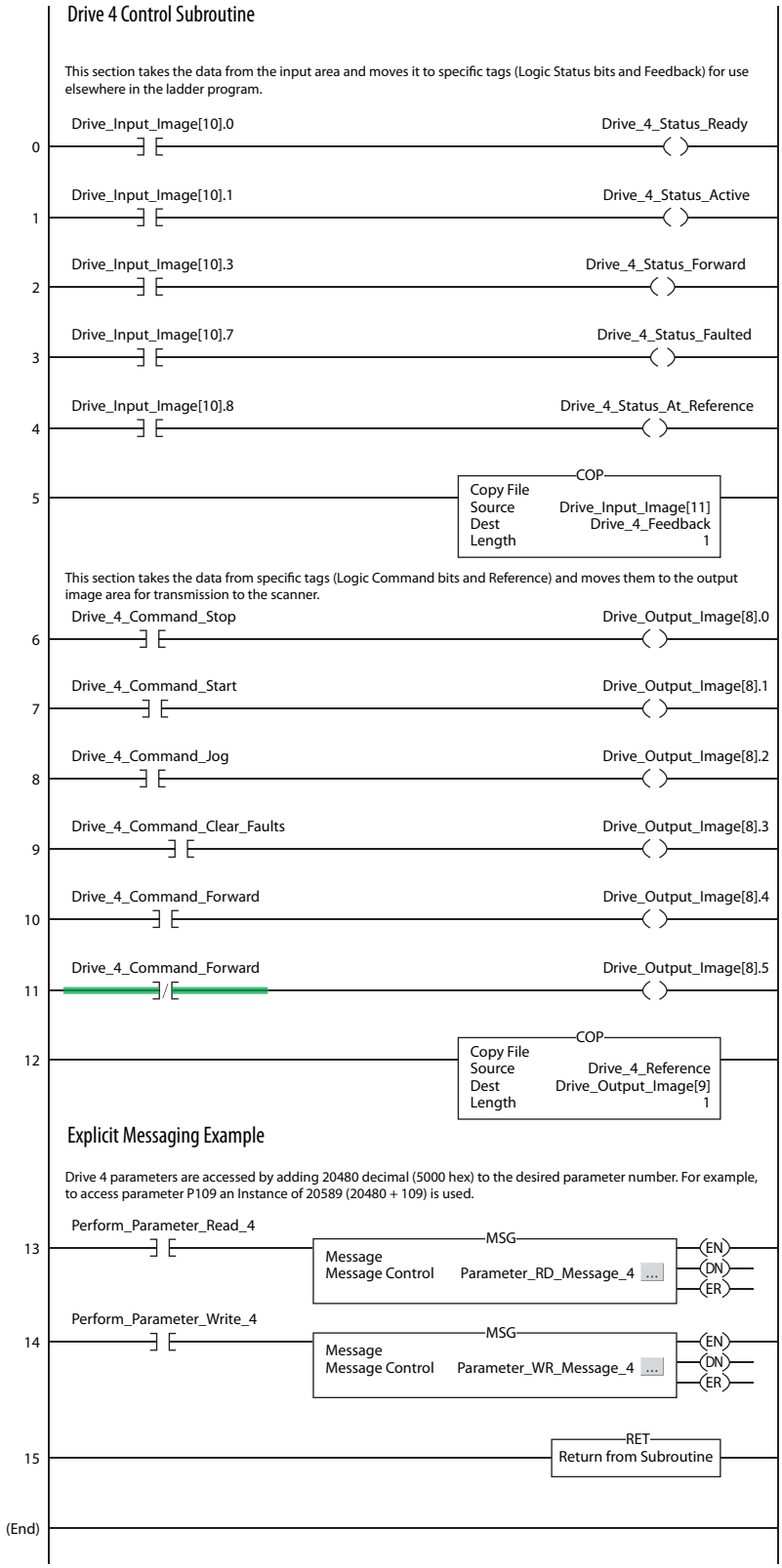
Drive 2 Control Routine



Drive 3 Control Routine



Drive 4 Control Routine



Multi-Drive Mode Explicit Messaging

Parameter addressing for Explicit messaging is different in Multi-drive mode than in Single-drive mode. In Single-drive mode, the Instance value in the message equals the desired parameter number in the drive. In Multi-drive mode, an Instance table is used to account for the parameters in the adapter and up to 5 drives. The parameters in the adapter and each of the drives are offset by 400 hex (1024 decimal):

Instance (Hex.)	Instance (Dec.)	Device	Parameter
0x0000...0x3FFF	0...16383	Drive 0	0...1023
0x4000...0x43FF	16384...17407	Interface(1)	0...1023
0x4400...0x47FF	17408...18431	Drive 1	0...1023
0x4800...0x4BFF	18432...19455	Drive 2	0...1023
0x4C00...0x4FFF	19456...20479	Drive 3	0...1023
0x5000...0x53FF	20480...21503	Drive 4	0...1023
0x5400...0x57FF	21504...22527	Option	0...1023

(1) Interface is the current interface being used to access the information. Note: If using the embedded EtherNet/IP adapter, this instance range contains the same data as Drive 0.

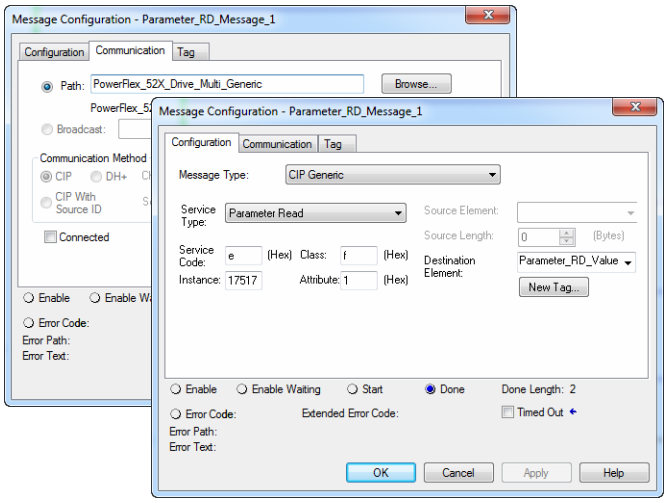
For example, to access [**Accel Time 1**] (parameter P041 in PowerFlex 525 and P109 in PowerFlex 4M) in each of the drives, the following Instances would be used:

- Drive 0 (PowerFlex 525) Instance = 41(0 + 41)
- Drive 1 (PowerFlex 4M) Instance = 17517 (17408 + 109)
- Drive 2 (PowerFlex 4M) Instance = 18541 (18432 + 109)
- Drive 3 (PowerFlex 4M) Instance = 19565 (19456 + 109)
- Drive 4 (PowerFlex 4M) Instance = 20589 (20480 + 109)

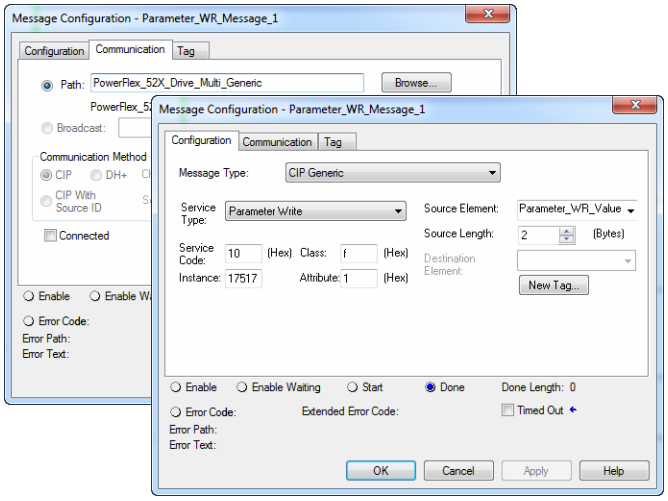
Drive 1 Explicit Message Example

The Explicit message examples in the CompactLogix example program perform a read and a write to PowerFlex 4M parameter P109 [Accel Time 1]. The configuration for the read is shown in Parameter Read Message Configuration on page 108 and the write is shown in Parameter Write Message Configuration on page 108.

Parameter Read Message Configuration



Parameter Write Message Configuration



The Class Code is “f” for the Parameter Object and the Instance Attribute is “1” to select retrieving the parameter value. See Appendix C, Parameter Object for more information. The Instance value is “17517” to access parameter **P109** [**Accel Time 1**] in the first daisy-chained drive.

The Explicit message for Drive 1 to Drive 4 are identical except for the Instance values, see Multi-Drive Mode Explicit Messaging on page 107 for examples.

Additional Information

- When the PowerFlex 525 drive (Drive 0) is powered up, all configured daisy-chained drives must be present before an I/O connection is allowed on EtherNet/IP (before the drives can be controlled).
- If the PowerFlex 525 drive (Drive 0) is powered down, communications with the four daisy-chained drives (Drive 1 to Drive 4) are disrupted and the drives will take their corresponding Comm Loss Actions.

- If any of the daisy-chained drives (Drive 1 to Drive 4) are powered down, the respective Input Image (Logic Status and Feedback) will be set to zero. Status information will not indicate there is a fault at the node, and the I/O connection will not be dropped.

Notes:

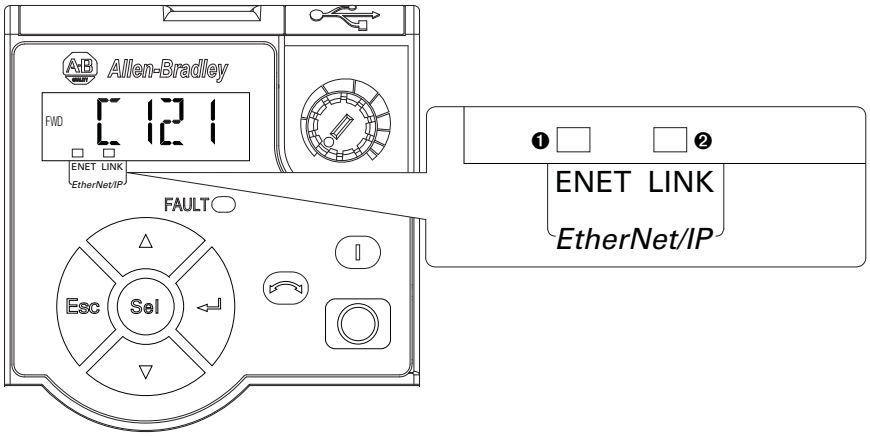
Troubleshooting

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

Understanding the Status Indicators

The embedded EtherNet/IP adapter uses two status indicators located on the drive LCD display.

Status Indicators



Item	Status Indicator	Description	Page
❶	ENET indicator	EtherNet/IP Connection Status	112
❷	LINK indicator	EtherNet/IP Transmit Status	112

ENET Status Indicator

This green color LED indicates the status of the adapter's network connection as shown in the table below.

Status	Cause	Corrective Actions
Off	There is no network link (unplugged Ethernet cable), the network configuration is invalid or a duplicate IP address error has been detected.	• Apply power to the drive. • Securely connect the adapter to the network using an Ethernet cable. Also, make sure the Ethernet cable is correctly connected to the Ethernet connector. • Set a unique IP address using the adapter switches, a BOOTP server, or by disabling BOOTP and using adapter parameters.
Flashing	The adapter is properly connected to the network but drive is not controlled through Ethernet.	• 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.
Steady	The adapter is properly connected to the network and drive is controlled through Ethernet.	No action required.

LINK Status Indicator

This green LED indicates the status of the adapter transmitting on the network as shown in the table below.

Status	Cause	Corrective Actions
Off	There is no network link (unplugged Ethernet cable) or a duplicate IP address error has been detected.	• Apply power to the drive. • Securely connect the adapter to the network using an Ethernet cable. Also, make sure the Ethernet cable is correctly connected to the Ethernet connector.
Flashing	The adapter is properly connected to the network and drive is transmitting data on Ethernet.	No action required.
Steady	The adapter is properly connected to the network but drive is not transmitting data on Ethernet.	• Set a unique IP address using the adapter switches, a BOOTP server, or by disabling BOOTP and using adapter parameters. • Configure the adapter to use a unique IP address and cycle power. • 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.

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. The diagnostic parameters for the embedded EtherNet/IP adapter can be viewed using the PowerFlex 22-HIM-A3/-C2S HIM.

Embedded EtherNet/IP Adapter Diagnostic Parameters

No.	Name	Description
F681	Comm Sts - DSI	Displays the status of the RS485 serial (DSI) port to the drive.
F682	Comm Sts - Opt	Displays the status of the internal communication to the drive.
F683	Com Sts-Emb Enet	Displays the status of the embedded EtherNet/IP interface to the drive.
F684	EN Addr Src	Displays the network configuration source currently used by the embedded EtherNet/IP interface.
F685	EN Rate Act	Displays the network data rate currently used by the embedded EtherNet/IP interface.
F686	DSI I/O Act	Displays the drives that are active in Multi-drive mode.

Embedded EtherNet/IP Adapter Diagnostic Parameters

No.	Name	Description
F687 F688 F689 F690 F691 F692	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]
F693 F694 F695 F696	EN IP Addr Act 1 EN IP Addr Act 2 EN IP Addr Act 3 EN IP Addr Act 4	Value of each byte in the adapter's present IP address. A value of "0" appears if the adapter does not currently have an IP address. **255.255.255.255** [EN IP Addr Act 1] [EN IP Addr Act 2] [EN IP Addr Act 3] [EN IP Addr Act 4]
F697 F698 F699 F700	EN Subnet Act 1 EN Subnet Act 2 EN Subnet Act 3 EN Subnet Act 4	Value of each byte in the adapter's present subnet mask. A value of "0" appears if the adapter does not currently have a subnet mask. **255.255.255.255** [EN Subnet Act 1] [EN Subnet Act 2] [EN Subnet Act 3] [EN Subnet Act 4]
F701 F702 F703 F704	EN Gateway Act 1 EN Gateway Act 2 EN Gateway Act 3 EN Gateway Act 4	Value of each byte in the adapter's present gateway address. A value of "0" appears if the adapter does not currently have a gateway address. **255.255.255.255** [EN Gateway Act 1] [EN Gateway Act 2] [EN Gateway Act 3] [EN Gateway Act 4]
F705 F709 F713 F717 F721	Drv 0 Logic Cmd Drv 1 Logic Cmd Drv 2 Logic Cmd Drv 3 Logic Cmd Drv 4 Logic Cmd	In Multi-drive mode, this is the logic command being transmitted to drive 0. In Single-drive mode, this is the logic command being used by the drive (whether HS-DSI, EtherNet/IP, or DSI) at the time. If comms control is NOT being used, and the drive is in single-drive mode, then this parameter will show 0.
F706 F710 F714 F718 F722	Drv 0 Reference Drv 1 Reference Drv 2 Reference Drv 3 Reference Drv 4 Reference	In Multi-drive mode, this is the reference being transmitted to drive 0. In Single-drive mode, this is the reference being used by the drive (whether HS-DSI, EtherNet/IP, or DSI) at the time. If comms control is NOT being used, and the drive is in Single-drive mode, then this parameter will show 0.
F707 F711 F715 F719 F723	Drv 0 Logic Sts Drv 1 Logic Sts Drv 2 Logic Sts Drv 3 Logic Sts Drv 4 Logic Sts	In Multi-drive mode, this is the logic status being received from drive 0. In Single-drive mode, this is the logic status of the drive at the time.

Embedded EtherNet/IP Adapter Diagnostic Parameters

No.	Name	Description
F708 F712 F716 F720 F724	Drv 0 Feedback Drv 1 Feedback Drv 2 Feedback Drv 3 Feedback Drv 4 Feedback	In Multi-drive mode, this is the feedback being received from drive 0. In Single-drive mode, this is the feedback of the drive at the time.
F725	EN Rx Overruns	A count of the number of receive overrun errors reported by the embedded EtherNet/IP interface.
F726	EN Rx Packets	A count of the number of Ethernet packets that the adapter has received.
F727	EN Rx Errors	A count of the number of receive errors reported by the Ethernet hardware.
F728	EN Tx Packets	A count of the number of Ethernet packets that the adapter has sent.
F729	EN Tx Errors	A count of the number of transmit errors reported by the Ethernet hardware.
F730	EN Missed IO Pkt	A count of the number of incoming I/O connection packets that the adapter did not receive.
F731	DSI Errors	A count of the number of DSI errors.

IMPORTANT The diagnostic parameters for the dual port EtherNet/IP adapter option card cannot be viewed using the HIMs. Use Connected Components Workbench instead.

Viewing and Clearing Events

The adapter has an event queue to record significant events that occur in the operation of the adapter. When such an event occurs, an entry consisting of the event's numeric code and a timestamp is put into the event queue. You can view the event queue using the PowerFlex 22-HIM-A3/-C2S HIM or Connected Components Workbench.

The event queue can contain up to 32 entries, which are stored in RAM—making the event queue volatile, meaning a power cycle will clear the event queue. If the event queue becomes full, a new entry replaces the oldest entry. Only a power cycle, event queue clear operation, or the corruption of the RAM group containing the event queue will clear the event queue contents.

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:

Adapter Events

Code	Event	Description
Adapter Events		
0	No Event	Text displayed in an empty event queue entry.
1	Normal Startup	Power is applied to the adapter.
2	Manual Reset	The adapter was reset.
3	Watchdog T/O Flt	The software watchdog detected a failure and reset the adapter.
4	App Updated	The adapter application firmware was flash updated.
5	Boot Updated	The adapter boot firmware was flash updated.
6	EEPROM Sum Flt	The EEPROM checksum/CRC is incorrect, which limits adapter functionality. Default parameter values must be loaded to clear this condition.
DSI Events		

Adapter Events

Code	Event	Description
10	Slave Detected	The adapter detected that the slave is connected.
11	Slave Removed	The adapter detected that the slave was disconnected.
12	Slave Logon	The adapter has established communications with the slave.
13	Slave Timeout	The adapter has lost communications with the slave.
14	Slave Brand Flt	The brand of the slave is different from the adapter.
15	Host 0 Logon	The adapter has established communications with host 0.
16	Host 1 Logon	The adapter has established communications with host 1.
17	Host 2 Logon	The adapter has established communications with host 2.
18	Host 3 Logon	The adapter has established communications with host 3.
19	Host 4 Logon	The adapter has established communications with host 4.
20	Host 0 Timeout	The adapter has lost communications with host 0.
21	Host 1 Timeout	The adapter has lost communications with host 1.
22	Host 2 Timeout	The adapter has lost communications with host 2.
23	Host 3 Timeout	The adapter has lost communications with host 3.
24	Host 4 Timeout	The adapter has lost communications with host 4.
25	Host 0 Brand Flt	The brand of host 0 is different from the adapter.
26	Host 1 Brand Flt	The brand of host 1 is different from the adapter.
27	Host 2 Brand Flt	The brand of host 2 is different from the adapter.
28	Host 3 Brand Flt	The brand of host 3 is different from the adapter.
29	Host 4 Brand Flt	The brand of host 4 is different from the adapter.
Network Events		
40	EN Link Up	A network link was available for the adapter.
41	EN Link Down	The network link was removed from the adapter.
42	Dup IP Addr	The adapter uses the same IP address as another device on the network.
43	EN Open	An I/O connection from the network to the adapter has been opened.
44	EN Close	An I/O connection from the network to the adapter was closed.
45	EN Timeout	An I/O connection from the network to the adapter has timed out.
46	EN Comm Flt	An I/O connection from the network to the adapter.
47	EN Idle Flt	The adapter received “idle” packets from the network.
48	PCCC IO Open	The adapter has begun receiving PCCC Control messages (the PCCC Control Timeout was previously set to a non-zero value).
49	PCCC IO Close	The device sending PCCC Control messages to the adapter has set the PCCC Control Timeout to zero.
50	PCCC IO Time Flt	The adapter has not received a PCCC Control message for longer than the PCCC Control Timeout.
51	EN Sent Reset	The adapter received a reset from the network.
52	Msg Ctrl Open	The adapter has begun receiving Client-Server-Control messages (the Client-Server-Control Timeout was previously set to a non-zero value).
53	Msg Ctrl Close	The device sending Client-Server-Control messages to the adapter has set the Client-Server-Control Timeout to zero.
54	Msg Ctrl Timeout	The adapter has not received a Client-Server-Control message for longer than the established timeout period.
Adapter Specific Events		
60	BOOTP Response	The module received a response to its BOOTP request.
61	E-mail Failed	The module encountered an error attempting to send a requested e-mail message.

Notes:

Specifications

Appendix A presents the specifications for the adapter.

Communication

Network	
Protocol	EtherNet/IP
Data Rates	10 Mbps Full Duplex, 10 Mbps Half Duplex, 100 Mbps Full Duplex or 100 Mbps Half Duplex
Connection Limit	8 TCP connections 4 simultaneous CIP connections including 1 exclusive-owner I/O connection The following activities use a CIP connection: • Class I I/O connections (for example, from a ControlLogix or CompactLogix controller) or CIP Motion connection • Explicit messaging where "connected" is chosen (for example, in a check box in RSLogix 5000/Logix Designer) The following activities DO NOT use a CIP connection: • Explicit messaging-based control using PCCC or the Register or Assembly objects, including the MicroLogix 1100/1400 examples in Chapter 4 • Explicit messaging where "connected" is NOT chosen, which is typically the default
Requested Packet Interval (RPI)	5 ms minimum
Packet Rate	Up to 200 total I/O packets per second (100 in and 100 out)

Regulatory Compliance

See the PowerFlex 525 Adjustable Frequency AC Drive User Manual, publication 520-UM001 for regulatory compliance information.

Notes:

Adapter Parameters

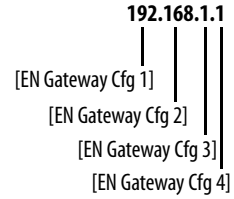
Appendix B provides information about the parameters used to configure the embedded EtherNet/IP adapter.

Parameter List

The parameters are displayed in a **Numbered List** view order.

Parameter No.	Name and Description	Details	
C121	**[Comm Write Mode]** Saves parameter values in active drive memory (RAM) or in drive non-volatile memory (EEPROM). **Important:** Parameter values set prior to setting 1 "RAM only" are saved in RAM.	Default: Values: Type: Reset Required:	0 = Save 0 = Save 1 = RAM only Read/Write
C122	**[Cmd Stat Select]** Selects velocity-specific or position/fibers-specific Command and Status Word bit definitions for use over a communication network. This parameter cannot be changed when an I/O connection is established through the communication adapter or the drive's embedded EtherNet/IP port.	Default: Values: Type: Reset Required:	0 = Velocity 0 = Velocity 1 = Position Read/Write Yes
C123	**[RS485 Data Rate]** Sets the communications baud rate (bits/second) for the RS485 port. A reset or power cycle is required after selection is made.	Default: Values: Type: Reset Required:	3 = 9600 0 = 1200 1 = 2400 2 = 4800 3 = 9600 4 = 19,200 5 = 38,400 Read/Write Yes
C124	**[RS485 Node Addr]** Sets the Modbus drive node number (address) for the RS485 port if using a network connection. A reset or power cycle is required after selection is made.	Default: Minimum: Maximum: Type: Reset Required:	100 1 247 Read/Write Yes
C125	**[Comm Loss Action]** Sets the drive's response to a loss of connection or excessive communication errors on the RS485 port.	Default: Values: Type: Reset Required:	0 = Fault 0 = Fault 1 = Coast Stop 2 = Stop 3 = Continu Last Read/Write No
C126	**[Comm Loss Time]** Sets the time that the drive remains in communication loss with the RS485 port before taking the action specified in C125 [Comm Loss Action]. **Important:** This setting is effective only if I/O that controls the drive is transmitted through the RS485 port.	Default: Minimum: Maximum: Type: Reset Required:	5.0 s 0.1 s 60.0 s Read/Write No

Parameter No.	Name and Description	Details	
C127	**[RS485 Format]** Determines the details related to the specific Modbus protocol used by the drive. A reset or power cycle is required after selection is made.	Default: Values: Type: Reset Required:	0 = RTU 8-N-1 0 = RTU 8-N-1 1 = RTU 8-E-1 2 = RTU 8-O-1 3 = RTU 8-N-2 4 = RTU 8-E-2 5 = RTU 8-0-2 Read/Write Yes
C128	**[EN Addr Sel]** Enables BOOTP to set the IP address, subnet mask and gateway address with a BOOTP server. Identifies the connections that would be attempted on a reset or power cycle. A reset or power cycle is required after selection is made.	Default: Values: Type: Reset Required:	2 = BOOTP 1 = Parameters 2 = BOOTP Read/Write Yes
C129 C130 C131 C132	**[EN IP Addr Cfg 1]** **[EN IP Addr Cfg 2]** **[EN IP Addr Cfg 3]** **[EN IP Addr Cfg 4]** Sets the bytes in the IP address. A reset or power cycle is required after selection is made. **192.168.1.62** [EN IP Addr Cfg 1] [EN IP Addr Cfg 2] [EN IP Addr Cfg 3] [EN IP Addr Cfg 4] **Important:** To set the IP address using these parameters, **C128 [EN Addr Sel]** must be set to 1 “Parameters”.	Default: Default: Default: Default: Minimum: Maximum: Type: Reset Required:	0 0 0 0 0 255 Read/Write Yes
C133 C134 C135 C136	**[EN Subnet Cfg 1]** **[EN Subnet Cfg 2]** **[EN Subnet Cfg 3]** **[EN Subnet Cfg 4]** Sets the bytes of the subnet mask. A reset or power cycle is required after selection is made. **255.255.255.0** [EN Subnet Cfg 1] [EN Subnet Cfg 2] [EN Subnet Cfg 3] [EN Subnet Cfg 4] **Important:** To set the subnet mask using these parameters, **C128 [EN Addr Sel]** must be set to 1 “Parameters”.	Default: Default: Default: Default: Minimum: Maximum: Type: Reset Required:	0 0 0 0 0 255 Read/Write Yes

Parameter No.	Name and Description	Details
C137 C138 C139 C140	**[EN Gateway Cfg 1]** **[EN Gateway Cfg 2]** **[EN Gateway Cfg 3]** **[EN Gateway Cfg 4]** Sets the bytes of the gateway address. A reset or power cycle is required after selection is made.  **Important:** To set the gateway address using these parameters, **C128 [EN Addr Sel]** must be set to 1 "Parameters".	Default: 0 Default: 0 Default: 0 Default: 0 Minimum: 0 Maximum: 255 Type: Read/Write Reset Required: Yes
C141	**[EN Rate Cfg]** Sets the network data rate at which EtherNet/IP communicates. A reset or power cycle is required after selection is made.	Default: 0 = Auto detect Values: 0 = Auto detect 1 = 10Mbps Full 2 = 10Mbps Half 3 = 100Mbps Full 4 = 100Mbps Half Type: Read/Write Reset Required: Yes
C143	**[EN Comm Flt Actn]** Sets the action that the EtherNet/IP interface and drive takes if the EtherNet/IP interface detects that Ethernet communications have been disrupted. **Important:** This setting is effective only if I/O that controls the drive is transmitted through the EtherNet/IP interface.	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 C143 [EN Comm Flt Actn] lets you determine the action of the EtherNet/IP interface and connected drive if 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 drive).

Parameter No.	Name and Description	Details	
C144	**[EN Idle Flt Actn]** Sets the action that the EtherNet/IP interface and drive takes if the EtherNet/IP interface detects that the scanner is idle because the controller was switched to program mode.	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 C144 [EN Idle Flt Actn] lets you determine the action of the EtherNet/IP interface and connected drive if the scanner 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 disconnected drive).		
C145	**[EN Flt Cfg Logic]** Sets the Logic Command data that is sent to the drive if any of the following is true: • C143 [EN Comm Flt Actn] is set to 4 "Send Flt Cfg" and communications are disrupted. • C144 [EN Idle Actn] is set to 4 "Send Flt Cfg" and the scanner is put into Program or Test mode.	Default: Minimum: Maximum: Type: Reset Required:	0000 0000 0000 0000 0000 0000 0000 0000 1111 1111 1111 1111 Read/Write No
C146	**[EN Flt Cfg Ref]** Sets the Reference data that is sent to the drive if any of the following is true: • C143 [EN Comm Flt Actn] is set to 4 "Send Flt Cfg" and communications are disrupted. • C144 [EN Idle Actn] is set to 4 "Send Flt Cfg" and the scanner is put into Program or Test mode.	Default: Minimum: Maximum: Type: Reset Required:	0 0 50000 Read/Write No
C147 C148 C149 C150	**[EN Flt Cfg DL 1]** **[EN Flt Cfg DL 2]** **[EN Flt Cfg DL 3]** **[EN Flt Cfg DL 4]** Sets the data that is sent to the Datalink in the drive if any of the following is true: • C143 [EN Comm Flt Actn] is set to 4 "Send Flt Cfg" and the I/O Communication is disrupted. • C144 [EN Idle Flt Actn] is set to 4 "Send Flt Cfg" and the controller is set into Program or Idle mode.	Default: Default: Default: Default: Minimum: Maximum: Type: Reset Required:	0 0 0 0 0 65535 Read/Write No
C153 C154 C155 C156	**[EN Data In 1]** **[EN Data In 2]** **[EN Data In 3]** **[EN Data In 4]** Datalink parameter number whose value is written from the embedded EtherNet/IP data table. This parameter cannot be changed when an I/O connection is established through the drive's embedded EtherNet/IP port.	Default: Default: Default: Default: Minimum: Maximum: Type: Reset Required:	0 0 0 0 0 800 Read/Write Yes

Parameter			
No.	**Name and Description**	**Details**	
C157	**[EN Data Out 1]**	Default:	0
C158	**[EN Data Out 2]**	Default:	0
C159	**[EN Data Out 3]**	Default:	0
C160	**[EN Data Out 4]**	Default:	0
	Datalink parameter number whose value is read from the embedded EtherNet/IP data table.	Minimum:	0
		Maximum:	800
	This parameter cannot be changed when an I/O connection is established through the drive's embedded EtherNet/IP port.	Type:	Read/Write
		Reset Required:	Yes
C161	**[Opt Data In 1]**	Default:	0
C162	**[Opt Data In 2]**	Default:	0
C163	**[Opt Data In 3]**	Default:	0
C164	**[Opt Data In 4]**	Default:	0
	Datalink parameter number whose value is written from the High Speed Drive Serial Interface (HSDSI) data table.	Minimum:	0
		Maximum:	800
	This parameter cannot be changed when an I/O connection is established through the communication adapter.	Type:	Read/Write
		Reset Required:	Yes
C165	**[Opt Data Out 1]**	Default:	0
C166	**[Opt Data Out 2]**	Default:	0
C167	**[Opt Data Out 3]**	Default:	0
C168	**[Opt Data Out 4]**	Default:	0
	Datalink parameter number whose value is read from the HSDSI data table.	Minimum:	0
		Maximum:	800
	This parameter cannot be changed when an I/O connection is established through the communication adapter.	Type:	Read/Write
		Reset Required:	Yes
C169	**[MultiDrvSel]**	Default:	0 = Disabled
	Sets the configuration of the drive that is in Multi-drive mode. A reset or power cycle is required after selection is made.	Values:	0 = Disabled
			1 = Network Opt
			2 = EtherNet/IP
		Type:	Read/Write
		Reset Required:	Yes
C171	**[Drive 1 Addr]**	Default:	2
C172	**[Drive 2 Addr]**	Default:	3
C173	**[Drive 3 Addr]**	Default:	4
C174	**[Drive 4 Addr]**	Default:	5
	Sets the corresponding node addresses of the daisy-chained drives when C169 [MultiDrv Sel] is set to 1 "Network Opt"' or 2 "EtherNet/IP". A reset or power cycle is required after selection is made.	Minimum:	1
		Maximum:	247
		Type:	Read/Write
		Reset Required:	Yes
C175	**[DSI I/O Cfg]**	Default:	0 = Drive 0
	Sets the configuration of the Drives that are active in the Multi-drive mode. Identifies the connections that would be attempted on a reset or power cycle. A reset or power cycle is required after selection is made.	Values:	0 = Drive 0
			1 = Drive 0-1
			2 = Drive 0-2
			3 = Drive 0-3
			4 = Drive 0-4
		Type:	Read/Write
		Reset Required:	Yes

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, see Chapter 6, Using Explicit Messaging.

TIP See 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_Attribute_All

Instances (Single-Drive)

Instance, Ins	DPI Inst#	Port#	Device
0x0000...0x3FFF	Ins	0	Drive
0x4000...0x43FF	Ins & 0x3FF	Interface	Interface(1)
0x4400...0x47FF	Ins & 0x3FF	1	DSI
0x4800...0x4BFF	Ins & 0x3FF	2	Option
0x4C00...0x4FFF	–	–	–
0x5000...0x53FF	–	–	–
0x5400...0x57FF	–	–	–

(1) Interface is the current interface being used to access the information.

Instances (Multi-Drive)

Instance, Ins	DPI Inst#	Port#	Device
0x0000...0x3FFF	Ins	0	Drive 0
0x4000...0x43FF	Ins & 0x3FF	0	Interface(1)
0x4400...0x47FF	Ins & 0x3FF	1	Drive 1
0x4800...0x4BFF	Ins & 0x3FF	2	Drive 2
0x4C00...0x4FFF	Ins & 0x3FF	3	Drive 3
0x5000...0x53FF	Ins & 0x3FF	4	Drive 4
0x5400...0x57FF	Ins & 0x3FF	5	Option

(1) Interface is the current interface being used to access the information.

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	150
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
9	Get	Configuration Consistency Value	UNIT	Current Parameter NVS CRS value

Assembly Object

Class Code

Hexadecimal	Decimal
0x04	4

Services

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

Instances

Instance	Description
Status Data	All I/O data being read from the DSI device (read-only)
Command Data	All I/O data written to the DSI 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	Members List	ARRAY of STRUCT: UINT UINT Packed EPATH	 Size of member data Size of member path Member path
3	Get	Conditional[1]	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

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

Instances (Single-Drive)

Instance	Description
1	Status Image – All I/O data being read from the embedded adapter (read-only)
2	Command Image – All I/O data written to the embedded adapter (read/write)
3	Logic Status and Feedback (read-only)
4	Logic Command and Reference (read/write)
5	Datalink To Net 1 (read only)
6	Datalink From Net 1 (read/write)
7	Datalink To Net 2 (read only)
8	Datalink From Net 2 (read/write)
9	Datalink To Net 3 (read only)
10	Datalink From Net 3 (read/write)
11	Datalink To Net 3 (read only)
12	Datalink From Net 4 (read/write)
13	Logic Command (Masked)(1) (read/write)
14	Logic Command (Masked)(1) (read/write)
15	Logic Command (Masked)(1) (read/write)
16	Logic Command (Masked)(1) (read/write)
17	Logic Command (Masked)(1) (read/write)
18	Logic Command (Masked)(1) (read/write)
19	Logic Status (read-only)
20	Logic Command (read/write)
21	Feedback (read-only)
22	Reference (read/write)
23	Logic Status (read-only)
24	Logic Command (read/write)
25	Feedback (read-only)
26	Reference (read/write)
27	Logic Command (read/write)
28	Logic Status (read-only)
29	Reference (read/write)
30	Feedback (read-only)
31	Logic Command (read/write)
32	Logic Status (read-only)
33	Reference (read/write)
34	Feedback (read-only)
35	Logic Command (read/write)

Instance	Description
36	Logic Status (read-only)
37	Reference (read/write)
38	Feedback (read-only)

(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.

Instances (Multi-Drive)

Instance	Description
1	Status Image – All I/O data being read from the embedded adapter (read-only)
2	Command Image – All I/O data written to the embedded adapter (read/write)
3	Logic Status and Feedback 0 (read-only)
4	Logic Command and Reference 0 (read/write)
5	Logic Status and Feedback 1 (read-only)
6	Logic Command and Reference 1 (read/write)
7	Logic Status and Feedback 2 (read-only)
8	Logic Command and Reference 2 (read/write)
9	Logic Status and Feedback 3 (read-only)
10	Logic Command and Reference 3 (read/write)
11	Logic Status and Feedback 4 (read-only)
12	Logic Command and Reference 4 (read/write)
13	Logic Command, all drives (Masked)(1) (read/write)
14	Logic Command 0 (Masked)(1) (read/write)
15	Logic Command 1 (Masked)(1) (read/write)
16	Logic Command 2 (Masked)(1) (read/write)
17	Logic Command 3 (Masked)(1) (read/write)
18	Logic Command 4 (Masked)(1) (read/write)
19	Logic Status 0 (read-only)
20	Logic Command 0 (read/write)
21	Feedback 0 (read-only)
22	Reference 0 (read/write)
23	Logic Status 1 (read-only)
24	Logic Command 1 (read/write)
25	Feedback 1 (read-only)
26	Reference 1 (read/write)
27	Logic Command 2 (read/write)
28	Logic Status2 (read-only)
29	Reference 2 (read/write)
30	Feedback 2 (read-only)
31	Logic Command 3 (read/write)
32	Logic Status 3 (read-only)
33	Reference 3 (read/write)
34	Feedback 3 (read-only)
35	Logic Command 4 (read/write)
36	Logic Status 4 (read-only)
37	Reference 4 (read/write)
38	Feedback 4 (read-only)

(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 Instance
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 = Product Register (drive to network) 1 = Consume Register (network to drive)
3	Get	Size	UINT	Size of register data in bits
4	Conditional(1)	Data	Array of Bits	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.

Parameter Object

Class Code

Hexadecimal	Decimal
0x0F	15

Services

Service Code	Implemented for:		Service Name
	Class	Instance	
0x01	Yes	Yes	Get_Attribute_All
0x05	Yes	No	Reset
0x0E	Yes	Yes	Get_Atribute_Single
0x10	No	Yes	Set_Attribute_Single
0x4B	No	Yes	Get_Enum_String

Instances (Single-Drive)

Instance	Description
0	Class
1	Drive Parameter 1
⋮	⋮
n	Drive Parameter n(1)

(1) n represents the number of parameters in the drive.

Instances (Multi-Drive)

Instance	Description
0	Class (Drive 0)
1	Drive 0 Parameter 1
⋮	⋮
n	Drive 0 Parameter n(1)
16384	Class (Interface(2))
16384 + 1	Interface Parameter 1
⋮	⋮
16384 + n	Interface Parameter n(1)
17408	Class (Drive 1)
17408 + 1	Drive 1 Parameter 1
⋮	⋮
17408 + n	Drive 1 Parameter n(1)
18432	Class (Drive 2)
18432 + 1	Drive 2 Parameter 1
⋮	⋮
18432 + n	Drive 2 Parameter n(1)
19456	Class (Drive 3)
19456 + 1	Drive 3 Parameter 1
⋮	⋮
19456 + n	Drive 3 Parameter n(1)
20480	Class (Drive 4)
20480 + 1	Drive 4 Parameter 1
⋮	⋮
20480 + n	Drive 4 Parameter n(1)

(1) n represents the number of parameters in the drive.

(2) Interface is the current interface being used to access the information. Note: If using the embedded EtherNet/IP adapter, this instance range contains the same data as Drive 0.

In addition, the parameters for the other DSI devices can be accessed using the instance-offset encoding shown in the table below:

Instances (Hex.)	Instances (Dec.)	Single-Drive Mode	Multi-Drive Mode
0x0000...0x3FFF	0...16383	Instances 0...1023 in the drive	Instances 0...1023 in Drive 0
0x4000...0x43FF	16384...17407	Instances 0...1023 in the interface(1)	Instances 0...1023 in the interface(1)
0x4400...0x47FF	17408...18431	DSI	Instances 0...1023 in Drive 1
0x4800...0x4BFF	18432...19455	Option	Instances 0...1023 in Drive 2
0x4C00...0x4FFF	19456...20479	Not supported	Instances 0...1023 in Drive 3
0x5000...0x53FF	20480...21503	Not supported	Instances 0...1023 in Drive 4
0x5400...0x57FF	21504...22527	Not supported	Instances 0...1023 in the Option

(1) Interface is the current interface being used to access the information. Note: If using the embedded EtherNet/IP adapter, this instance range contains the same data as Drive 0.

Class Attributes

Attribute ID	Access Rule	Name	Data Type	Description
1	Get	Revision	UINT	1
2	Get	Max Instance	UINT	Number of parameters
8	Get	Parameter Class Descriptor	WORD	0 = False, 1 = True Bit 0 = Supports parameter instances Bit 1 = Supports full attributes Bit 2 = Must do NVS save command Bit 3 = Parameters are stored in NVS
9	Get	Configuration Assembly Instance	UINT	0
10	Get	Native Language	USINT	1 = English 2 = French 3 = Spanish 4 = Italian 5 = German 6 = Japanese 7 = Portuguese 8 = Chinese Simplified 9 = Reserved 10 = Reserved 11 = Korean 12 = Polish 13 = Reserved 14 = Turkish 15 = Czech

Instance Attributes

Attribute ID	Access Rule	Name	Data Type	Description
1	(1)	Parameter Value	(2)	(3)
2	Get	Link Path Size	USINT	0 = No link specified n = The size of Attribute 3 in bytes
3	Get	Link Path		(4)
4	Get	Descriptor	WORD	0 = False, 1 = True Bit 1 = Supports ENUMs Bit 2 = Supports scaling Bit 3 = Supports scaling links Bit 4 = Read only Bit 5 = Monitor Bit 6 = Extended precision scaling

Attribute ID	Access Rule	Name	Data Type	Description
5	Get	Data Type	USINT	0xC2 = SINT (8-bits) 0xC3 = INT (16-bits) 0xC4 = DINT (32-bits) 0xC6 = USINT (8-bits) 0xC7 = UINT (16-bits) 0xCA = REAL (32-bits) 0xD2 = WORD (16-bits)
6	Get	Data Size	USINT	(3)
7	Get	Parameter Name String	SHORT_STRING	(3)
8	Get	Units String	SHORT_STRING	(3)
9	Get	Help String	SHORT_STRING	(3)
10	Get	Minimum Value	(1)	(3)
11	Get	Maximum Value	(1)	(3)
12	Get	Default Value	(1)	(3)
13	Get	Scaling Multiplier	UINT	(3)
14	Get	Scaling Divisor	UINT	(3)
15	Get	Scaling Base	UINT	(3)
16	Get	Scaling Offset	UINT	(3)
17	Get	Multiplier Link	UINT	(3)
18	Get	Divisor Link	UINT	(3)
19	Get	Base Link	UINT	(3)
20	Get	Offset Link	UINT	(3)
21	Get	Decimal Precision	USINT	(3)

(1) Access rule is defined in bit 4 of instance attribute 4. 0 = Get/Set, 1 = Get.

(2) Specified in descriptor, data type, and data size.

(3) Value varies based on parameter instance.

(4) Refer to the CIP Common specification for a description of the link path.

PCCC Object

Class Code

Hexadecimal	Decimal
0x67	103

Services

Service Code	Implemented for:		Service Name
	Class	Instance	
0x4B	No	Yes	Execute_PCCC
0x4C	No	Yes	Execute_DH+

Instances

Supports Instance 1.

Class Attribute

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 request
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 request
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
0x0F	0x95	Encapsulate the other protocol
0x0F	0x00	Word range read
0x0F	0x01	Word range write

For more information regarding PCCC commands, see DFI Protocol and Command Set Manual (Allen-Bradley publication 1770-6.5.16).

N-File	Description	
N41	**For Single-Drive Mode Only** 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 value of N42:3 is set to a non-zero value.	
	Write	*Read*
N41:0	Logic Command Word	Logic Status Word
N41:1	Unused	Unused
N42:2	Reference	Feedback
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 N41 or N44 file. If the adapter does not receive a message in the specified time, it performs the fault action configured in parameter C143 [EN Comm Flt Actn].	
N42:7	Adapter Port Number (read only): DPI port on the drive to which the adapter is connected.	
42:8	Peer Adapters (read only): Bit field of devices having DPI Peer capabilities.	

N-File	Description	
N44	**For Multi-drive mode Only** 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 value of N42:3 is set to a non-zero value.	
	Write	Read
N44:0	Drive 0 Logic Command	Drive 0 Logic Status
N44:1	Unused	Unused
N44:2	Drive 0 Reference	Drive 0 Feedback
N44:3	Drive 1 Logic Command	Drive 1 Logic Status
N44:4	Drive 0 Reference	Drive 1 Feedback
N44:5	Drive 2 Logic Command	Drive 2 Logic Status
N44:6	Drive 2 Reference	Drive 2 Feedback
N44:7	Drive 3 Logic Command	Drive 3 Logic Status
N44:8	Drive 3 Reference	Drive 3 Feedback
N44:9	Drive 4 Logic Command	Drive 4 Logic Status
N44:10	Drive 4 Reference	Drive 4 Feedback

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.)	Instances (Dec.)	Single-Drive Mode	Multi-Drive Mode
0x0000...0x3FFF	0...16383	Instances 0...1023 in the drive	Instances 0...1023 in Drive 0
0x4000...0x43FF	16384...17407	Instances 0...1023 in the interface(1)	Instances 0...1023 in the interface(1)
0x4400...0x47FF	17408...18431	DSI	Instances 0...1023 in Drive 1
0x4800...0x4BFF	18432...19455	Option	Instances 0...1023 in Drive 2
0x4C00...0x4FFF	19456...20479	Not supported	Instances 0...1023 in Drive 3
0x5000...0x53FF	20480...21503	Not supported	Instances 0...1023 in Drive 4
0x5400...0x57FF	21504...22527	Not supported	Instances 0...1023 in the Option

(1) Interface is the current interface being used to access the information. Note: If using the embedded EtherNet/IP adapter, this instance range contains the same data as Drive 0.

Class Attributes

Attribute ID	Access Rule	Name	Data Type	Description
0	Get	Family Code	USINT	0x00 = DSI Peripheral 0x09 = Single Mode 0x0B = Multi-Drive Mode 0x1E = 25-COMM-X Option Module 0xFF = HIM
1	Get	Family Text	STRING[16]	Text identifying the device.
2	Set	Language Code	USINT	1 = English 2 = French 3 = Spanish 4 = Italian 5 = German 6 = Japanese 7 = Portuguese 8 = Chinese Simplified 9 = Reserved 10 = Reserved 11 = Korean 12 = Polish 13 = Reserved 14 = Turkish 15 = Czech
3	Get	Product Series	USINT	1 = A 2 = B …
4	Get	Number of Components	USINT	Number of components (for example, 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 = DSI
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

Attribute ID	Access Rule	Name	Data Type	Description
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[16]	 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.
38	Set	ADC Configuration Signature	USINT[16]	Value stored by the device and zeroed if its configuration changes.
39	Get	SI Driver 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 Attribute

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	STRING	Name of the component with support for Unicode.

DPI Parameter Object

Class Code

Hexadecimal	Decimal
0x93	147

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, Attribute 0.

Instances (Hex.)	Instances (Dec.)	Single-Drive Mode	Multi-Drive Mode
0x0000...0x3FFF	0...16383	Instances 0...1023 in the drive	Instances 0...1023 in Drive 0
0x4000...0x43FF	16384...17407	Instances 0...1023 in the interface(1)	Instances 0...1023 in the interface(1)
0x4400...0x47FF	17408...18431	DSI	Instances 0...1023 in Drive 1
0x4800...0x4BFF	18432...19455	Option	Instances 0...1023 in Drive 2
0x4C00...0x4FFF	19456...20479	Not supported	Instances 0...1023 in Drive 3
0x5000...0x53FF	20480...21503	Not supported	Instances 0...1023 in Drive 4
0x5400...0x57FF	21504...22527	Not supported	Instances 0...1023 in the Option

(1) Interface is the current interface being used to access the information. Note: If using the embedded EtherNet/IP adapter, this instance range contains the same data as Drive 0.

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 = DSI
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 Attribute

Attribute ID	Access Rule	Name	Data Type	Description
6	Get	DPI Offline Read Full	STRUCT of:	
			BOOL[32]	Descriptor
			CONTAINER	Offline Minimum value
			CONTAINER	Offline Maximum value
			CONTAINER	Offline Default value
			STRING[16]	Parameter name
			STRING[4]	Offline parameter units
			UINT	Online minimum parameter instance
			UINT	Online maximum parameter instance
			UINT	Online default parameter instance
			UINT	Multiplier parameter instance
			UINT	Divisor parameter instance
			UINT	Base parameter instance
			UINT	Offset parameter instance
			USINT	Formula number
			USINT	Pad byte (always zero)
			UINT	Help instance
			UINT	Pad word (always a value of zero)
			CONTAINER	Parameter value
			UINT	Multiplier
			UNIT	Divisor
			UNIT	Base
			INT	Offset
7	Get	DPI Online Read Full	STRUCT of:	
			BOOL[32]	Descriptor (see page 144)
			CONTAINER[(1)]	Parameter value
			CONTAINER	Minimum value
			CONTAINER	Maximum value
			CONTAINER	Default value
			UINT	Next parameter
			UINT	Previous parameter
			STRING[4]	Units (for example, Amps, Hz)
			UINT	Multiplier[(2)]
			UINT	Divisor[(2)]
			UINT	Base[(2)]
			INT	Offset[(2)]
			USINT[3]	Link (source of the value) (0 = no link)
			USINT	Always zero (0)
			STRING[16]	Parameter name
8	Get	DPI Descriptor	BOOL[32]	Descriptor (see page 144)
9	Get/Set	DPI Parameter Value	Various	Name of the component with support for Unicode.
10	Get/Set	DPI RAM Parameter Value	Various	Parameter value in NVS. [(3)]
11	Get/Set	DPI Link	USINT[3]	Parameter value in temporary memory. Valid only for DSI drives.
12	Get	Help Object Instance	UINT	Link (parameter or function block that is the source of the value) (0 = no link)
13	Get	DPI Read Basic	STRUCT of:	
			BOOL[32]	Descriptor (see page 144)
			CONTAINER	Parameter value
			CONTAINER	Minimum value
			CONTAINER	Maximum value
			CONTAINER	Default value
			STRING[16]	Parameter name
			STRING[4]	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.

Attribute ID	Access Rule	Name	Data Type	Description
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
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 unit
21	Get	DPI Extended Descriptor	UDINT	Extended Descriptor (see page 145)
22	Get	International DPI Offline Read Full	Struct of: BOOL CONTAINER CONTAINER CONTAINER UINT 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 DSI descriptor International DSI parameter name International DSI 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. See Formulas for Converting on page 146.

(3) Do NOT continually write parameter data to NVS. See the attention on page 69.

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 = Read only 1 = Read/write
10	Instance	0 = Writable when enabled (e.g., drive running) 1 = Not writable when enabled
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 DSI Limits Object	Parameter uses the DSI 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 DSI 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 3	
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.
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

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

Object Specific Services

	Implemented for:			Allocation Size (in bytes)	
Service Code	**Class**	**Instance**	**Service Name**	**Par. Number**	**Par. Value**
0x32	Yes	No	Get_Attributes_Scattered	4	4
0x34	Yes	Yes	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 write (zero when reading)

The response data appears in the following format:

Name	Data Type	Description
Parameter Number	UDINT	Parameter read or write(1)
Parameter Value	UDINT	Parameter value read (zero when writing)(2)

(1) If an error occurred, bit 15 will be turned on in the response.
(2) If an error occurred, the error code will appear instead of the value.

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

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 faults or events supported in the queue. The maximum number of faults/events can be read in Instance 0, Attribute 2.

Instances (Hex.)	Instances (Dec.)	Single-Drive Mode	Multi-Drive Mode
0x0000...0x3FFF	0...16383	Instances 0...1023 in the drive	Instances 0...1023 in Drive 0
0x4000...0x43FF	16384...17407	Instances 0...1023 in the interface(1)	Instances 0...1023 in the interface(1)
0x4400...0x47FF	17408...18431	DSI	Instances 0...1023 in Drive 1
0x4800...0x4BFF	18432...19455	Option	Instances 0...1023 in Drive 2
0x4C00...0x4FFF	19456...20479	Not supported	Instances 0...1023 in Drive 3
0x5000...0x53FF	20480...21503	Not supported	Instances 0...1023 in Drive 4
0x5400...0x57FF	21504...22527	Not supported	Instances 0...1023 in the Option

(1) Interface is the current interface being used to access the information. Note: If using the embedded EtherNet/IP adapter, this instance range contains the same data as Drive 0.

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.
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 Attribute

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 DSI port DSI 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 DSI port DSI 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.

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 Attribute
1	Object Attribute

Class Attributes

Attribute ID	Access Rule	Name	Data Type	Description
1	Get	Class Revision	UINT	The revision of this object

Instance Attribute

Attribute ID	Access Rule	Name	Data Type	Description
1	Get	Status of TCP/IP Network Interface	UDINT	0 = Not configured 1 = Valid configuration 2…15 = Reserved
2	Get	Configuration Capability	UDINT	Bit | 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 using TCP/IP object) 5…31 = Reserved
3	Set	Configuration Control	UDINT	Bit | Value 1…3 = Startup configuration 0 = Use configuration saved in NVS 1 = Obtain configuration using BOOTP 2 = Obtain configuration using DHCP 3…15 = Reserved 4 = DNS Enabled (resolves host names by query to DNS server) 5…31 = Reserved

Attribute ID	Access Rule	Name	Data Type	Description
4	Get	Physical Link Object	STRUCT of:	
			UINT	Path size
			Padded EPATH	Path
5	Get	Interface Configuration	STRUCT of:	
			UDINT	Adapter IP address
			UDINT	Adapter subnet mask
			UDINT	Adapter gateway address
			UDINT	Primary name server
			UDINT	Secondary name server
			STRING	Default domain name
6	Get	Host Name	STRING	Host name when using DHCP

Ethernet Link Object

Class Code

Hexadecimal	Decimal
0xF6	246

Services

	Implemented for:		
Service Code	Class	Instance	Service Name
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 Attribute
1	Object Attribute

Class Attributes

Attribute ID	Access Rule	Name	Data Type	Description
1	Get	Class Revision	UINT	The revision of this object

Instance Attribute

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 | Value 0 = Link status (0 = inactive, 1 = active) 1 = Duplex (0 = half duplex, 1 = full duplex) 2…31 = Reserved
3	Set	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

Logic Command/Status Words: PowerFlex 525 Drives

Appendix D presents the definitions of the Logic Command and Logic Status words that are used for PowerFlex 525 drives.

Logic Command Word

Velocity Bit Definitions

Comm Logic Command – C122 = 0 "Velocity"

Logic Bits																	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Command	Description
															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 = Jog
												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 = No Command
									x							Keypad	0 = Not Force Keypad Control 1 = Force Keypad Control
								x								MOP Increment	0 = Not MOP Increment 1 = MOP Increment
						x	x									Accel Time	00 = No Command 01 = Use Accel Rate 1 (P041 [Accel Time 1]) 10 = Use Accel Rate 2 (A442 [Accel Time 2]) 11 = Hold Accel Rate Selected
				x	x											Decel Time	00 = No Command 01 = Use Decel Rate 1 (P042 [Decel Time 1]) 10 = Use Decel Rate 2 (A443 [Decel Time 2]) 11 = Hold Decel Rate Selected
			x													Ref Select 1	000 = No Command 001 = Freq. Source = P047 [Speed Reference1] 010 = Freq. Source = P049 [Speed Reference2] 011 = Freq. Source = P051 [Speed Reference3] 100 = A410 [Preset Freq 0] 101 = A411 [Preset Freq 1] 110 = A412 [Preset Freq 2] 111 = A413 [Preset Freq 3]
		x														Ref Select 2	
	x															Ref Select 3	
x																MOP Decrement	0 = Not MOP Decrement 1 = MOP Decrement

(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 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."

Position Bit Definitions

Comm Logic Command – C122 = 1 "Position"

Logic Bits																	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Command	Description
															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 = Jog
												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 = No Command
									x							Logic Input 1	1 = Logic In 1
								x								Logic Input 2	1 = Logic In 2
					x	x	x									Frequency and Position Steps	000 = Frequency and Position Step 0 001 = Frequency and Position Step 1 010 = Frequency and Position Step 2 011 = Frequency and Position Step 3 100 = Frequency and Position Step 4 101 = Frequency and Position Step 5 110 = Frequency and Position Step 6 111 = Frequency and Position Step 7
				x												Find Home	1 = Find Home
			x													Hold Step	1 = Hold Step
		x														Redefine Position	1 = Pos Redefine
	x															Enable Sync	1 = Sync Enable
x																Disable Travel	1 = Travel Disable

(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 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."

Logic Status Word

Velocity Bit Definitions

Comm Logic Status – C122 = 0 "Velocity"

Logic Bits																	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Command	Description
															x	Run Ready	0 = Not Ready to Run 1 = Ready to Run
														x		Active	0 = Not Active 1 = Active (Running)
													x			Command Direction	0 = Reverse 1 = Forward
												x				Actual Direction	0 = Rotating Reverse 1 = Rotating Forward
											x					Accel	0 = Not Accelerating 1 = Accelerating
										x						Decel	0 = Not Decelerating 1 = Decelerating
									x							Reserved	–
								x								Fault	0 = Not Faulted 1 = Faulted
							x									At Speed	0 = Not at Reference 1 = At Reference
						x										Main Frequency	0 = Not Controlled by Active Com 1 = Controlled by Active Com
					x											Operation Command	0 = Not Controlled by Active Com 1 = Controlled by Active Com
				x												Parameters	0 = Not Locked 1 = Locked
			x													Digital Input 1 Status	–
		x														Digital Input 2 Status	–
	x															Digital Input 3 Status	–
x																Digital Input 4 Status	–

Position Bit Definitions

Comm Logic Status – C122 = 1 "Position"																	
Logic Bits																	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Command	Description
															x	Run Ready	0 = Not Ready to Run 1 = Ready to Run
														x		Active	0 = Not Active 1 = Active (Running)
													x			Command Direction	0 = Reverse 1 = Forward
												x				Actual Direction	0 = Rotating Reverse 1 = Rotating Forward
											x					Accel	0 = Not Accelerating 1 = Accelerating
										x						Decel	0 = Not Decelerating 1 = Decelerating
									x							Travel Position	0 = Reverse Travel Position 1 = Forward Travel Position
								x								Fault	0 = Not Faulted 1 = Faulted
							x									At Speed	0 = Not at Reference 1 = At Reference
						x										At Position	0 = Not at Position 1 = At Position
					x											Drive Home	0 = Not at Home 1 = At Home
				x												Commanded Home	0 = Not Drive Homed 1 = Drive Homed
			x													Sync Hold	0 = Not Sync Hold 1 = Sync Hold
		x														Sync Ramp	0 = Not Sync Ramp 1 = Ramp Sync
	x															Traverse	0 = Traverse Off 1 = Traverse On
x																Traverse Decel	0 = Not Traverse Decel 1 = Traverse Decel

Notes:

The following terms and abbreviations are used throughout this manual.For definitions of terms not listed here, see the Allen-Bradley Industrial Automation Glossary, publication AG-7.1.

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 its PowerFlex 525 drive to an EtherNet/IP network. Adapters are sometimes also called “cards,” “embedded communication options,” “gateways,” “modules,” and “peripherals.”

ADC (Automatic Device Configuration) An RSLogix 5000 (version 20) and Logix Designer (version 21 or greater) feature that supports the automatic download of configuration data upon the Logix controller establishing an EtherNet/IP network connection to a PowerFlex525 drive and its associated peripherals.

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 or CompactLogix controller that connects the controller to a network. See also Scanner.

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.

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.

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 525 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 16-bit Datalink in a PowerFlex 525 drive consumes 4 bytes in the input image table and/or 4 bytes in the output image table of the controller.

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.

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.

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 its PowerFlex 525 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.

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.

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. The PowerFlex 22-HIM-A3 or 22-HIM-C2S HIM can be used to configure PowerFlex 525 drives and their connected peripherals.

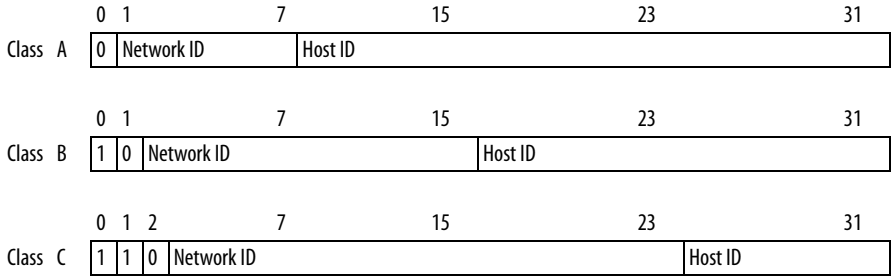
Hold Last When communication is disrupted (for example, a cable is disconnected), the adapter and its PowerFlex 525 drive can respond by holding last. Hold last results in the drive receiving the last data received through the network connection before the disruption. If the drive was running and using the Reference from the adapter, it will continue to run at the same Reference.

Idle Action An idle action determines how the adapter and its PowerFlex 525 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” (To Net) and “output” (From Net) 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.

Logic Command/Logic Status

The Logic Command is used to control the PowerFlex 525 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 525 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.

Logix Designer

The Logix Designer application is the rebranding of RSLogix 5000 software and will continue to be the product to program Logix 5000 controllers for discrete, process, batch, motion, safety, and drive-based solutions. It is a 32-bit application that runs on various Windows operating systems. Information about Logix Designer software can be found at http://www.software.rockwell.com/rslogix.

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 multi-master hierarchy), each slave device must have a scanner specified as a master.

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."

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.

Ping A message that is sent on the network to determine if a node exists.

PowerFlex 525 Drives The Allen-Bradley PowerFlex 525 drives are part of the PowerFlex 520-series of drives.

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.

RSLogix 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.

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.

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.

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).

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.

Rockwell Automation Support

Rockwell Automation provides technical information on the Web to assist you in using its products. At http://www.rockwellautomation.com/support/, you can find technical manuals, a knowledge base of FAQs, technical and application notes, sample code and links to software service packs, and a MySupport feature that you can customize to make the best use of these tools.

For an additional level of technical phone support for installation, configuration, and troubleshooting, we offer TechConnect support programs. For more information, contact your local distributor or Rockwell Automation representative, or visit http://www.rockwellautomation.com/support/.

Installation Assistance

If you experience a problem within the first 24 hours of installation, review the information that is contained in this manual. You can contact Customer Support for initial help in getting your product up and running.

United States or Canada	1.440.646.3434
Outside United States or Canada	Use the Worldwide Locator at http://www.rockwellautomation.com/support/americas/phone_en.html, or contact your local Rockwell Automation representative.

New Product Satisfaction Return

Rockwell Automation tests all of its products to ensure that they are fully operational when shipped from the manufacturing facility. However, if your product is not functioning and needs to be returned, follow these procedures.

United States	Contact your distributor. You must provide a Customer Support case number (call the phone number above to obtain one) to your distributor to complete the return process.
Outside United States	Please contact your local Rockwell Automation representative for the return procedure.

Documentation Feedback

Your comments will help us serve your documentation needs better. If you have any suggestions on how to improve this document, complete this form, publication RA-DU002, available at http://www.rockwellautomation.com/literature/.

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 NV, Pegasus Park, De Kleetlaan 12a, 1831 Diegem, 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 520COM-UM001B-EN-E - March 2013
Supersedes Publication 520COM-UM001A-EN-E - February 2013