Allen-Bradley

1768 CompactLogix Controllers

1768-L43, 1768-L45

Firmware Revision 16

User Manual

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://literature.rockwellautomation.com) describes some important differences between solid state equipment and hard-wired electromechanical devices. Because of this difference, and also because of the wide variety of uses for solid state equipment, all persons responsible for applying this equipment must satisfy themselves that each intended application of this equipment is acceptable.

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

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

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

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

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

WARNING	Identifies information about practices or circumstances that can cause an explosion in a hazardous environment, which may lead to personal injury or death, property damage, or economic loss.
IMPORTANT	Identifies information that is critical for successful application and understanding of the product.
ATTENTION	Identifies information about practices or circumstances that can lead to: personal injury or death, property damage, or economic loss. Attentions help you identify a hazard, avoid a hazard, and recognize the consequences.
SHOCK HAZARD	Labels may be on or inside the equipment, such as a drive or motor, to alert people that dangerous voltage may be present.
BURN HAZARD	Labels may be on or inside the equipment, such as a drive or motor, to alert people that surfaces may reach dangerous temperatures.

Introduction

The release of this document contains new and updated information. To find new and updated information, look for change bars, as shown next to this paragraph.

Updated Information

The document contains these changes This table represents major topics. Make sure to look for the change bars throughout this document.

Topic	Page
Who Should Use This Publication	9
1768-L45 CompactLogix Controller	Throughout manual
About the 1768 CompactLogix Controllers	11
Complex CompactLogix System	12
Additional Resources	9, 13, 44, 50, 68, 83, 104, 112
Connect a Controller Via the Serial Port	16
Set the IP Address via a Serial Port	20
Set the IP Address Using the BOOTP/DHCP Utility	23
Communication I/O Modules for DeviceNet	30
Modbus RTU	35
Modbus Support	40
Station Address	43
Optimize network performance	43
Token hold factor	44
Producing and Consuming (interlock) Data	46
1768 Module Placement	51
1769 Module Placement	53
Reconfigure an I/O Module	65
Reconfigure a Module via a MSG Instruction	66
Managing Tasks	69
Defining Tasks	71
Defining Programs	72
System Overhead Time Slice	81
Set Up Each Axis	95
PhaseManager programs, Minimum System Requirements	110
Preventing a Major Fault During a Load	112
1768 CompactLogix Controller Specifications	113
CompactLogix Controller Status Indicators	117

Notes:

Table of Contents

Chapter 8

Develop Motion Applications

Chapter 9

Configure PhaseManager

Chapter 10

Maintain Nonvolatile Memory

Purpose of This Manual

This manual describes the necessary tasks to install, configure, program, and operate a CompactLogix system.

Who Should Use This Publication

This manual is for automation engineers and control system developers who design, program, and commission 1768 CompactLogix control systems.

Additional Resources

These documents address Logix5000 controllers.

Resource	Description
1768-L43 and 1768-L45 CompactLogix Controller Installation Instructions, publication <u>1768-IN004</u>	Details how to assemble and install the controllers and provides technical specifications.
CompactLogix L43 and L45 Controller Version 17 Firmware Release Note, publication <u>1768-RN016</u>	Provides revision, restriction, and anomaly information for this controller.
Logix5000 Controllers Execution Time and Memory Use Reference Manual, publication <u>1756-RM087</u>	Details how to estimate the execution time and memory use of the controller's logic.
PhaseManager User Manual, publication <u>LOGIX-UM001</u>	Details how to use equipment phases, set up state models for equipment, and program in a way that resembles S88 and PackML models.
Logix5000 Controllers Quick Start, publication <u>1756-QS001</u>	Details how to begin programming and maintaining Logix5000 controllers.
Logix5000 Controllers Common Procedures Programming Manual, publication <u>1756-PM001</u>	Provides guidance on the development of controller projects.
Logix5000 Controllers System Quick Reference, publication <u>1756-QR107</u>	Provides general information on Logix5000 controllers.
Logix5000 Controllers General Instruction Set Reference Manual, publication <u>1756-RM003</u>	Details how to program the controller for sequential applications.
Logix5000 Controllers Process Control/Drives Instruction Set Reference Manual, publication <u>1756-RM006</u>	Details how to program the controller for process or drives applications.
Logix5000 Controllers Motion Instructions Reference Manual, publication <u>1756-RM007</u>	Details how to program the controllers for motion applications.
EtherNet/IP Communication Modules in Logix5000 Control Systems User Manual, publication <u>ENET-UM001</u>	Details how to configure and use the modules.
ControlNet Communication Modules in Logix5000 Control Systems User Manual, publication <u>CNET-UM001</u>	Details how to configure and use the modules and provides technical specifications.

You can view or download publications at <u>http://literature.rockwellautomation.com</u>. To obtain paper copies of technical documentation, contact your Rockwell Automation distributor or sales representative.

Notes:

1768 CompactLogix Overview

Introduction

This chapter explains the design and installation requirements of the 1768 CompactLogix controllers.

Topic	Page
About the 1768 CompactLogix Controllers	11
Designing a System	13
Additional Resources	13

About the 1768 CompactLogix Controllers

The CompactLogix system is designed to provide a Logix solution for machine-level control applications with I/O, motion, and network requirements.

The 1768-L43 and 1768-L45 controllers offer one built-in serial port and have a key on the front panel so you can change controller modes.

CompactLogix Chassis

CompactLogix Controller	Available 1768 Slots	Number of 1768 Communication Modules, Max	Type of Communication Modules	Number of 1768-M04SE Modules, Max	Number of Connections Supported, Max
1768-L43	2	2	1768-ENBT 1768-EWEB 1768-CNB 1768-CNBR	2	250
1768-L45	4	2		4	

CompactLogix Communication Modules

Communication Module	Function
1768-ENBT	Module for Ethernet/IP communication
1768-M04SE	Module for motion control of SERCOS drives
1768-EWEB	Web server module for the remote monitoring and modification of data via an XML web page raw socket
1768-CNB and 1768-CNBR	Modules for ControlNet communication

Example - Standalone CompactLogix Controller with I/O and DeviceNet Communication

1768 Backplane
– 1768 Controller Plus Two 1768 Modules

1769 Backplane
– 1769-SDN Scanner for the DeviceNet Network
– As many as eight 1769 I/O modules

Complex CompactLogix System

Computers or Other Controllers

EtherNet/IP Network via 1768-ENBT Communication Module or ControlNet network via 1768-CNB or 1768-CNBR Communication Modules

Other Networks

Built-in Serial Port

1769 Local I/O

1768 backplane contains the 1768-L43 and as many as two 1768 modules or the 1768-L45 controller and as many as four 1768 modules.

1768-L43 - Maximum 3 I/O Banks for a Total of 16 I/O Modules
1768-L45 - Maximum 3 I/O Banks for a Total of 30 I/O Modules

SERCOS Motion via 1768-M04SE Adapter Module

SERCOS Drives

Distributed I/O

DeviceNet Network via 1769-SDN Scanner Module

IMPORTANT	You can attach up to eight of the local I/O modules to your 1768 controller.
	Install the remaining modules in one or two additional I/O banks attached to the 1768/1769 system.
	See 1769 Module Placement on page 53 for more information.
	The additional banks are powered by any standard 1769 power supply and connect to the main rack by using standard 1769-CRL*x* extension cables.

Designing a System

When designing a CompactLogix system, determine the network configuration and the placement of components in each location. You need to:

- select I/O devices for your DIN-rail or panel-mounted system.
- establish motion control and drives requirements.
- select communication modules with a built-in RS-232 serial port (DF1 or ASCII).
- select controllers.
- select power supplies.
- mount the system.
- select ViewAnyWare products.
- select RSLogix 5000 programming software.

Additional Resources

These documents contain additional information on selecting components and designing your system.

Resource	Description
CompactLogix Selection Guide, publication <u>1769-SG001</u>	Provides the information necessary for designing a CompactLogix system.
Logix5000 Controllers Design Considerations Reference Manual, publication <u>1756-RM094</u>	Provides information pertinent to the design of Logix5000 systems.
1768-L43 and 1768-L45 CompactLogix Controller Installation Instructions, publication <u>1768-IN004</u>	Provides details about how to assemble and mount the controllers, and controller technical specifications.
CompactLogix SERCOS Interface Module Installation Instructions, publication <u>1768-IN005</u>	Provides information about installing the SERCOS module and technical specifications.
1768 CompactLogix Quick Start Guide, publication <u>IASIMP-QS001</u>	Provides examples and procedures for the use of a 1768-L43 CompactLogix system.

Notes:

Directly Connect to a Controller Via the Serial Port

Introduction

This chapter explains how to establish a serial connection to a controller via the serial port so you can configure a controller and upload and download a project to it.

Topic	Page
Connect a Controller Via the Serial Port	16
Configure the Serial Driver	17
Select the Controller Path	19
Set the IP Address via a Serial Port	20

For a CompactLogix controller to operate on a serial network, you need:

- a workstation with a serial port.
- RSLinx software to configure the serial communication driver.
- RSLogix 5000 programming software to configure the serial port of the controller.

Connect a Controller Via the Serial Port
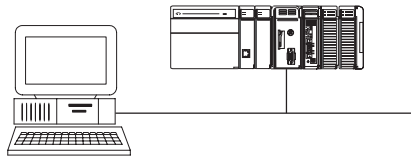
Channel 0 on a CompactLogix controller is fully isolated and does not need a separate isolation device.

To connect a serial cable, perform this procedure.

1. Obtain a serial cable.

If you	Then
Make your own cable	a. Limit the length to 15.2 m (50 ft). b. Wire the connectors. c. Attach the shield to both connectors.
Do not make your own cable	Obtain one of these serial cables: • 1747-CP3 • 1756-CP3

2. Connect the cable to your controller and workstation.
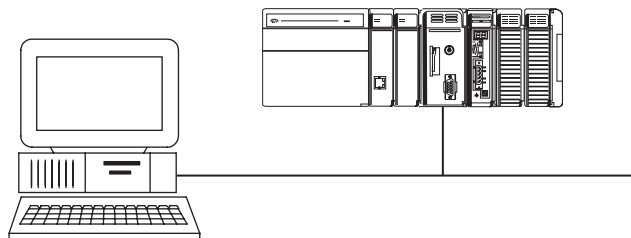
Configure the Serial Driver

Use RSLinx software to configure the RS-232 DF1 device driver for serial communication. To configure the driver, perform this procedure.

1. In RSLinx software, from the Communications menu, choose Configure Drivers.
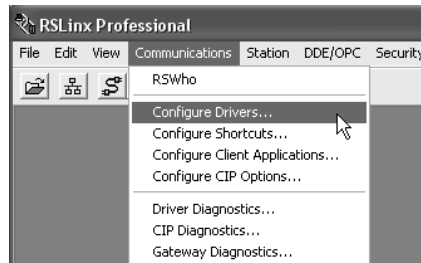
The Configure Drivers dialog appears.
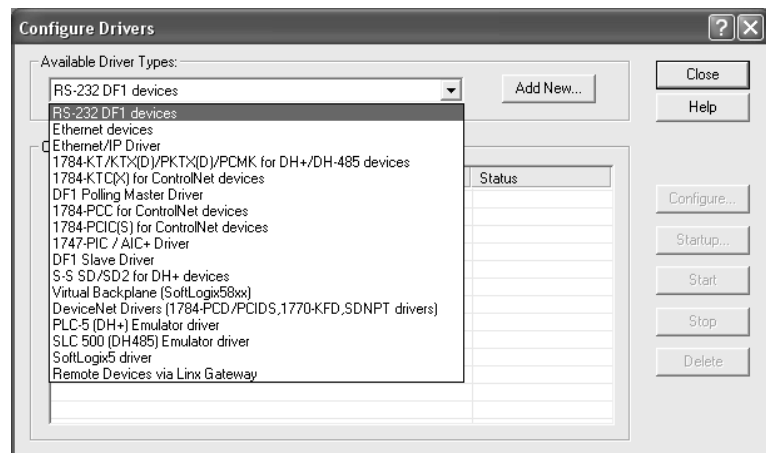
2. From the Available Driver Types pull-down menu, choose RS-232 DF1 devices.

3. Click Add New to add the driver.

The Add New RSLinx Driver dialog appears.
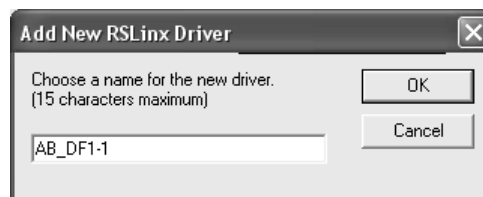
4. Specify the driver name and click OK.

The Configure RS-232 DF1 Devices dialog appears.

5. From the Comm Port pull-down menu, choose the serial port on the workstation to which the cable is connected.

6. From the Device pull-down menu, choose Logix 5550/CompactLogix.

7. Click Auto-Configure.

8. Verify that the auto configuration was successful.

If	Then
Yes	Click OK.
No	Return to step 5 and verify that you selected the correct communication port.

9. In the Configure Drivers dialog, click Close.

Select the Controller Path

IMPORTANT	Remember to update your controller's firmware prior to establishing communication and downloading a program.

To select the controller path, perform this procedure.

1. In RSLogix 5000 programming software, open a project for the controller.
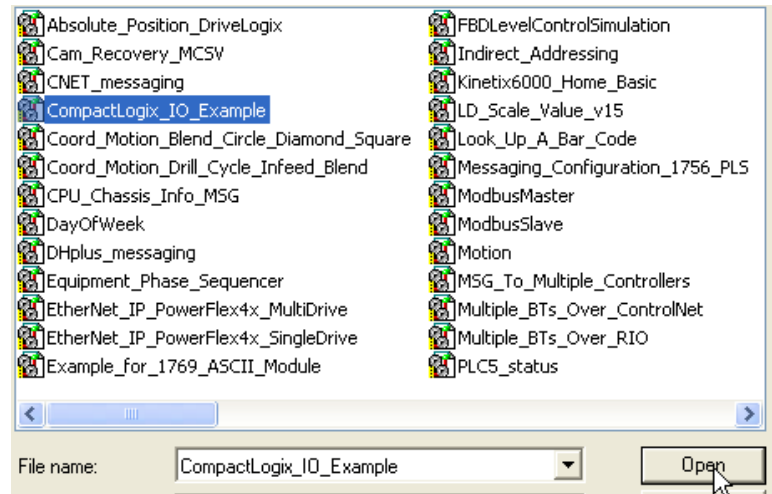
2. From the Communications menu, choose Who Active.
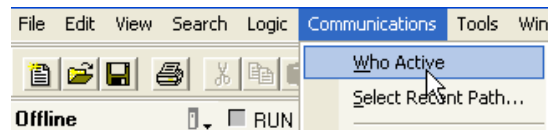
3. Expand the communication driver to the level of the controller.
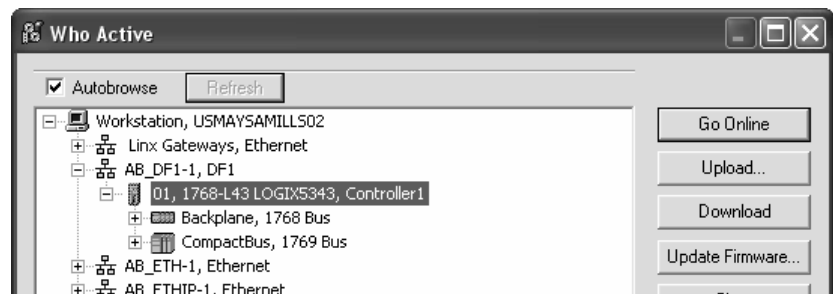
4. Select the controller.

To	Click
Monitor the project in the controller	Go Online
Transfer a copy of the project from the controller to RSLogix 5000 programming software	Upload
Transfer the open project to the controller	Download

Set the IP Address via a Serial Port

To set the IP address via a serial port, perform this procedure.

1. Make sure the module is installed, started, and connected to the controller via a serial connection.

2. In the Controller Organizer of RSLogix 5000 programming software, right-click the module and choose Properties.
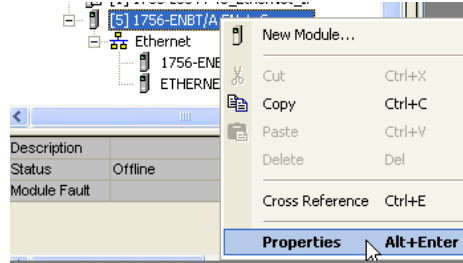
The Module Properties dialog appears.
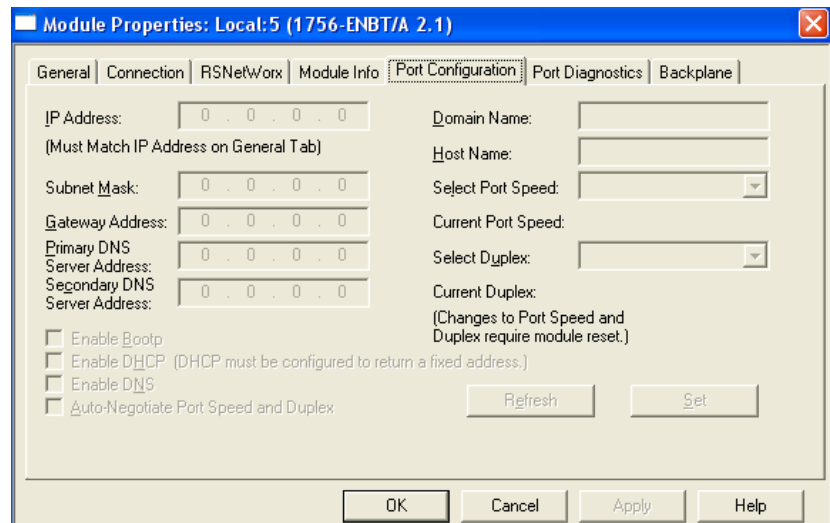
3. Click the Port Configuration tab.

4. In the IP Address box, type the IP address.

5. In the other boxes, type the other network parameters, if needed.

> **IMPORTANT**   The boxes that appear vary from one module to another.
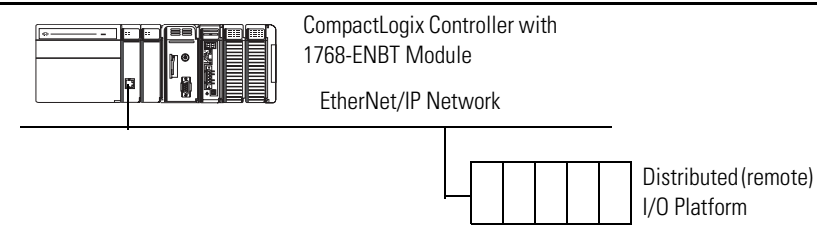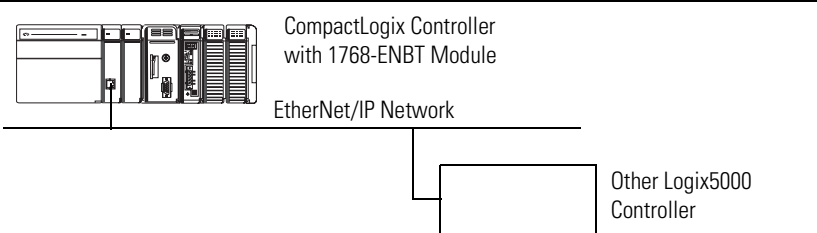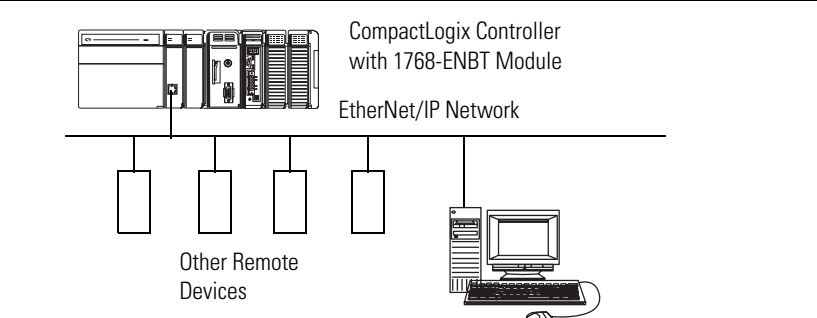
6. Click Apply.

7. Click OK.

Communicate over Networks

Introduction

This chapter explains how CompactLogix controllers support multiple networks.

Network Support	Example
Control distributed (remote) I/O via: • the EtherNet/IP network. • the ControlNet network. • the DeviceNet network.	CompactLogix Controller with 1768-ENBT Module EtherNet/IP Network Distributed (remote) I/O Platform
Produce and consume data between controllers via: • the EtherNet/IP network. • the ControlNet network.	CompactLogix Controller with 1768-ENBT Module EtherNet/IP Network Other Logix5000 Controller
Send and receive messages to and from other devices, including access to the controller via RSLogix 5000 programming software, via: • the EtherNet/IP network. • the DeviceNet network (to devices only). • the ControlNet network. • serial networks. • DH-485 networks.	CompactLogix Controller with 1768-ENBT Module EtherNet/IP Network Other Remote Devices

EtherNet/IP Network Communication

The EtherNet/IP network offers a full suite of control, configuration, and data collection services by layering the Common Industrial Protocol (CIP) over the standard Internet protocols, such as TCP/IP and UDP. With this combination of well-accepted standards, the EtherNet/IP network supports information data exchange and control applications.

The EtherNet/IP network also uses commercial, off-the-shelf Ethernet components and physical media, providing you with a cost-effective plant-floor solution.

For EtherNet/IP communication, the controller needs a 1768-ENBT or 1768-EWEB module. You can install up to two of these modules in the 1768 backplane for each controller.

Use these software products for EtherNet/IP communication.

Required Software for EtherNet/IP Communication

Software	Functions	Requirement
RSLogix 5000	• Configure CompactLogix projects. • Define EtherNet/IP communication.	Yes
RSLogix 5000 BOOTP/DHCP Utility	Assign IP addresses to devices on an EtherNet/IP network.	No
RSNetWorx for EtherNet/IP	• Configure EtherNet/IP devices by IP addresses and/or host names. • Provide bandwidth status.	
RSLinx	• Configure communication devices. • Provide diagnostics. • Establish communication between devices.	Yes

Set the IP Address Using the BOOTP/DHCP Utility

To set the IP address using the BOOTP/DHCP utility, perform this procedure.

1. From the Tools menu of RSLogix 5000 programming software, choose BootP-DHCP Server.
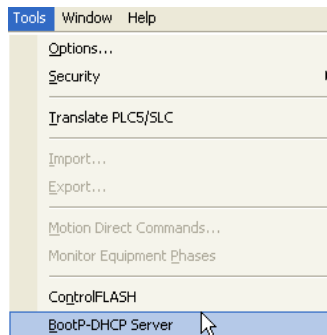
The Network Settings dialog appears.
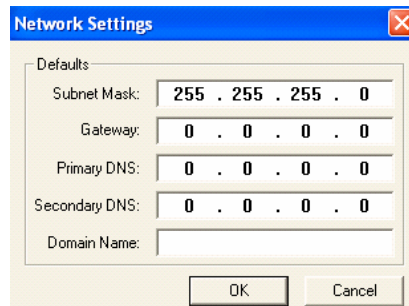
2. Enter at least one subnet mask setting and click OK.

The BOOTP/DHCP Server dialog appears. You will see BOOTP requests fill the upper portion of the dialog.
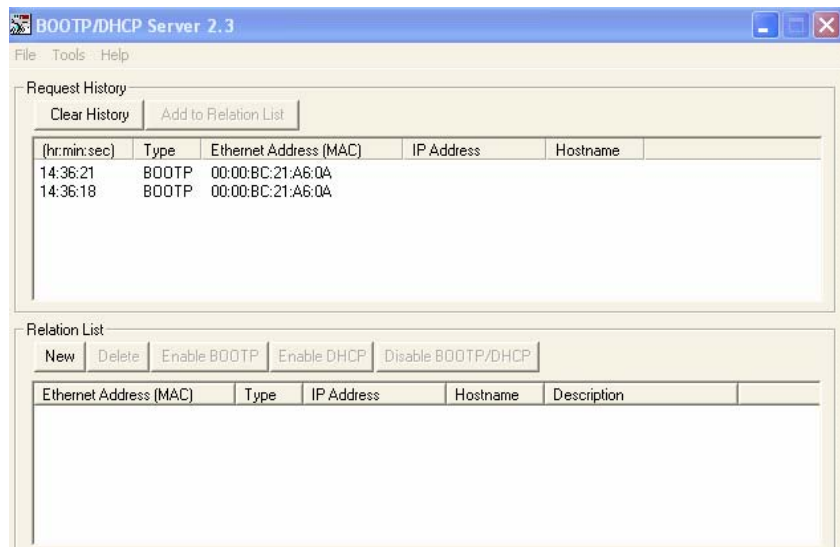
3. Select one of the requests and click Add to Relation List.
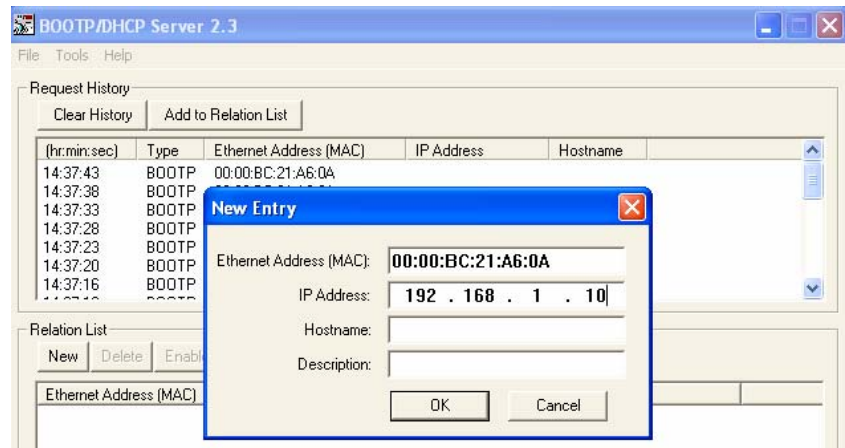
The New Entry dialog appears, showing the Ethernet address (MAC).

4. In the IP Address box, type the IP address and click OK.

1768-ENBT EtherNet/IP Communication Module

The 1768-ENBT EtherNet/IP communication module:

- supports messaging, produced/consumed tags, HMI, and distributed I/O.
- encapsulates messages within standard TCP/UDP/IP protocol.
- shares a common application layer with ControlNet and DeviceNet networks.
- connects via RJ45 connector.
- supports half-/full-duplex 10 MB or 100 MB operation.
- supports standard switches.

In this example:

- the controllers produce and consume tags.
- the controllers initiate MSG instructions that send and receive data or configure devices.
- the personal computer uploads or downloads projects to the controllers.
- the personal computer configures devices on an EtherNet/IP network.
- the controllers establish I/O and drive control over an EtherNet/IP network.

CompactLogix EtherNet/IP Network Overview

1768-L43 or 1768-L45 CompactLogix
Controller with 1768-ENBT Module

Computer with
Standard
Ethernet Port

ControlLogix
Controller

Ethernet
Industrial
Switch

PowerFlex 70 Drive
with 20-COMM-E
Ethernet Adapter

Distributed 1734
POINT I/O with
1734-AENT Ethernet
Adapter

PowerFlex 40 Drive
with 22-COMM-E
Ethernet Adapter

PanelView Plus Terminal
with Built-in EtherNet/IP Port

1768-EWEB Web Server Module

The 1768-EWEB EtherNet/IP web server module supports:

- the bridging and routing of messages, but no I/O control.
- data access (read and write) to controllers via a standard web browser.
- custom web pages.
- email.
- raw, open-socket interface.

In this example:

- you can route messages, upload/download programs, and flash upgrade modules by using the web server module as part of the communication path to access the target device.
- you can view and modify data that resides in a 1768 CompactLogix controller via a standard web browser.
- you can create custom web pages that are tailored to your application.

  Use ASP functions to populate your web pages with live controller data.

- you can send an email initiated by a Logix controller via a MSG instruction.
- you can open TCP or UDP communication links to other standard Ethernet devices via open sockets.

CompactLogix EtherNet/IP Web Network
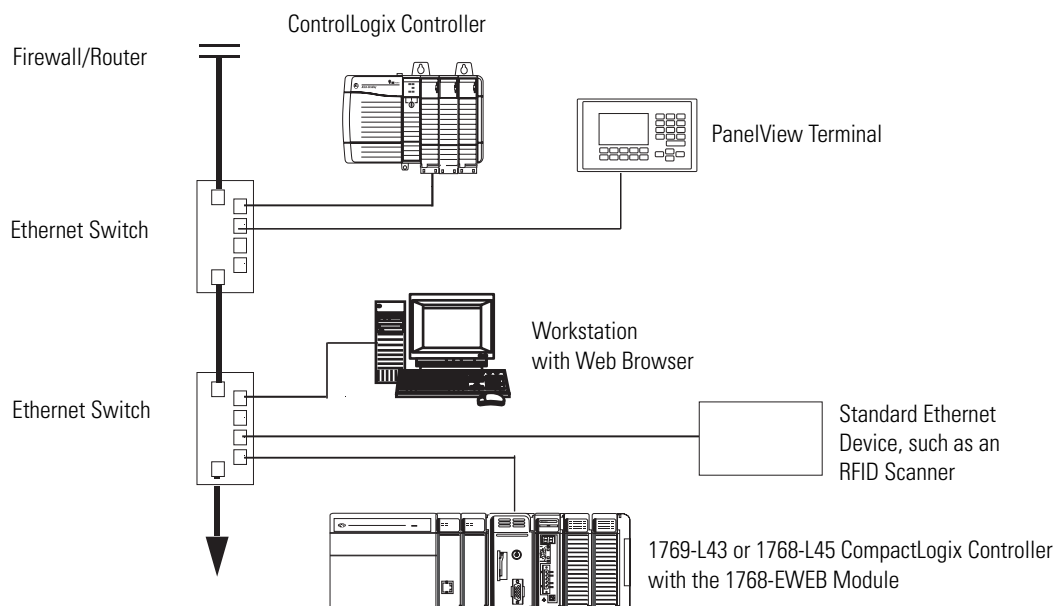
Connections Over an EtherNet/IP Network

Each 1768-ENBT or 1768-EWEB module in an EtherNet/IP network can provide messaging support for 64 CIP connections and 32 TCP/IP connections.

For more information, see the EtherNet/IP Modules in Logix5000 Control Systems User Manual, publication ENET-UM001.

ControlNet Network Communication

The ControlNet network is a real-time control network that provides high-speed transport of both time-critical I/O and interlocking data and messaging data, including uploading and downloading of programming and configuration data on a single physical-media link. The ControlNet network's efficient data transfer capability significantly enhances I/O performance and peer-to-peer communication in any system or application.

The ControlNet network is deterministic and repeatable and remains unaffected as devices are connected or disconnected from the network. This robust quality results in dependable, synchronized, and coordinated real-time performance.

The ControlNet network often functions as:

- the default network for the CompactLogix platform.
- a substitute/replacement for the remote I/O (RIO) network because the ControlNet network adeptly handles large numbers of I/O points.
- a backbone to multiple distributed DeviceNet networks.
- a peer interlocking network.

Required Software for ControlNet Communication

Software	Functions	Requirement
RSLogix 5000	• Configure CompactLogix projects. • Define ControlNet communication.	Yes
RSNetWorx for ControlNet	• Configure ControlNet devices by IP addresses and/or host names. • Schedule a network.	
RSLinx	• Configure communication devices. • Provide diagnostics. • Establish communication between devices.	

1768-CNB and 1768-CNBR ControlNet Modules

CompactLogix ControlNet communication modules bridge ControlNet links to route messages to devices on other networks. The modules also monitor and control I/O modules located remotely from CompactLogix controllers.

The 1768-CNB and 1768-CNBR ControlNet modules:

- provide messaging data for configuration and programming.
- support operator interfaces and uploading and downloading.
- support I/O bridging.
- support the transfer of scheduled data via produced/consumed tags.
- support unscheduled MSG instruction communication with other ControlNet nodes.
- support local communication network access through the network access port (NAP).
- support redundant media (1768-CNBR module only).

In this example:

- the controllers produce and consume tags.
- the controllers initiate MSG instructions that send and receive data or configure devices.
- the personal computer uploads or downloads projects to the controllers.
- the personal computer configures devices on the ControlNet network and the network itself.

CompactLogix ControlNet Network Overview

ControlLogix
Controller

1768-L43 or 1768-L45 CompactLogix Controller
with 1768-CNB or 1768-CNBR Module

Workstation with
1784-PCIC or
1784-PCICS
ControlNet
Communication Cards

ControlNet
Network
with Taps

Distributed 1734
POINT I/O Modules with
1734-ACNR ControlNet
Adapter

PowerFlex 70 Drive
with 20-COMM-C
ControlNet Adapter

PowerFlex 40 Drive
with 22-COMM-C
ControlNet Adapter

PanelView Plus Terminal with
ControlNet Communication Adapter

Connections Over a ControlNet Network

Each 1768-CNB or 1768-CNBR module in a ControlNet network can provide a certain amount of connected messaging support.

ControlNet Connected Messaging Support

Each	Supports
1768-CNB module	Supports 64 connections.
1768-CNBR module	• Five controllers can have a rack-optimized connection to the module. • Five controllers can have a rack-optimized, listen-only connection to the module.

DeviceNet Network Communication

The DeviceNet network uses the Common Industrial Protocol (CIP) to provide the control, configuration, and data collection capabilities for industrial devices. The DeviceNet network uses the proven Controller Area Network (CAN) technology, which lowers installation costs and decreases installation time and costly downtime.

A DeviceNet network provides access to the intelligence present in your devices by letting you connect devices directly to plant-floor controllers without having to hard wire each device into an I/O module.

DeviceNet Interfaces

Application	Required Interface
• Communicates with other DeviceNet devices • Uses the controller as a master on a DeviceNet network	1769-SDN DeviceNet scanner
• Accesses remote Compact I/O modules over a DeviceNet network • Sends remote I/O data for as many as 30 modules back to a scanner or controller	1769-ADN DeviceNet adapter[1]

[1] This table specifically describes using the 1769-ADN adapter to access remote Compact I/O modules over the DeviceNet network. However, CompactLogix controllers can access other Allen-Bradley remote I/O modules over the DeviceNet network. In those cases, you must select the appropriate interface. For example, if accessing remote POINT I/O modules, you must select the 1734-ADN adapter.

Communication I/O Modules for DeviceNet

The following table lists the communication I/O modules available for use with the DeviceNet network.

Catalog Number	Adapter	Specifications
Distributed Block I/O		
1791D CompactBlock I/O	1791D CompactBlock I/O contains a built-in adapter in the base block	• removable terminal block • 50% smaller than FLEX I/O • 24V DC and selectable combination analog block • 4...16 points • DeviceLogix-enabled • expandable up to 32 digital points
1790 CompactBlock LDX I/O	CompactBlock LDX I/O contains a built-in adapter in the base block	• lowest price-per-point I/O • digital: 24V DC, 120V AC, • analog: current, voltage, RTD and thermocouple • 4...16 points • expandable up to 64 points

Catalog Number	Adapter	Specifications
1732 ArmorBlock I/O	ArmorBlock I/O contains a built-in adapter in the base block	• 8 points • 8 in, 8 out, or 8 self-configuring • front or side mount • M12 or M8 I/O connector
1792 ArmorBlock MaXum I/O	ArmorBlock MaXum I/O contains a built-in adapter in the base block	• maximum value IP67 I/O • lowest overall installed cost on KwikLink • point-level diagnostics • 4...16 points • DeviceLogix-enabled • input-only, output-only, or combos
Distributed Modular I/O		
1734 POINT I/O	1734D POINTBlock I/O 1734-ADN 1734-ADNX (with subnet connectivity) 1734-PDN (DeviceNet power supply)	• highly granular (2, 4, or 8 points) • digital, analog, relay outputs, isolated temperature, RTD thermocouple, counter, and ASCII modules • channel-level Logix diagnostics: wire off, short circuit, OR, UR, CALIB, fault, and more • removal and insertion under power (RIUP) • expansion power supply and a POINT Power Bus Isolation Module available • removable terminal blocks • a total of 63 POINT I/O modules can be assembled on a single DeviceNet node • 1734-ADNX increases the reach of DeviceNet from 500 to 1500 meters • 1734-ADNX allows an expansion power supply for additional Pinatubo backplane current • up to 504 points maximum using 8-point digital I/O modules
1738 ArmorPoint I/O	1738-ADN12 1738-ADN18 1738-ADN18P 1738-ADNX	• DIN rail mounting • 1P67 and NEMA 4 rated • highly granular (mix and match to your exact needs) • removal and insertion under power (RIUP) • full range of digital, analog, specialty, and temperature modules • up to 252 points per adapter

Catalog Number	Adapter	Specifications
1794 FLEX I/O	1794-ADN	• I/O can be removed and inserted under power (RIUP) • digital: 24/48V DC, 120/230V AC, relay contact, protected, diagnostic, isolated • analog: current/voltage selectable, temperature, isolated • specialty: counters and frequency modules • 4...32 points per module, 8 modules per adapter • conformally coating versions on selected modules
1797 FLEX Ex I/O	1794-ADN To be used with 1797-BIC and 1797-CEC for connecting to hazardous areas	• intrinsically-Safe FLEX I/O for hazardous areas • I/O can be removed and inserted under power (RIUP) • digital (NAMUR), 0...20 mA analog, temperature, frequency • all modules conformally coated • 4...32 points per module, 8 modules per adapter
1798 FLEX Armor I/O	1798-ADN Also order either of the following: • 1798-DFTP1 (Terminator for 12mm DeviceNet cable plugs) • 1798-DFTP2 (Terminator for 18mm DeviceNet cable plugs)	• on-machine mounting • IP67 and NEMA 4X rated • outdoor applications • 24V DC analog and digital • 4 and 8 points per module; up to 64 per node

In addition to communication hardware for DeviceNet networks, these software products are available.

Required Software for DeviceNet Communication

Software	Functions	Requirement
RSLogix 5000	• Configure CompactLogix projects. • Define EtherNet/IP communication.	
RSNetWorx for DeviceNet	• Configure DeviceNet devices. • Define the scan list for those devices.	Yes
RSLinx	• Configure communication devices. • Provide diagnostics. • Establish communication between devices.	

The DeviceNet communication modules:

- support messaging to a device, not controller to controller.
- share a common application layer with ControlNet and EtherNet/IP networks.
- offer diagnostics for improved data collection and fault detection.
- require less wiring than traditional, hardwired systems.

CompactLogix DeviceNet Network Overview

1768-L43 or 1768-L45
CompactLogix Controller
with 1769-SDN Scanner

Distributed
1734 POINT I/O
Modules
with 1734-ADN
Adapter

PowerFlex 70
Drive with
20-COMM-D
Adapter

PowerFlex 40
Drive with
22-COMM-D
Adapter

Workstation with
1784-PCID, 1784-PCIDS,
and 1770-KFD DeviceNet
Cards

DeviceNet Network
with KwikLink
Flat Cable and
Micro Connectors

1606-XLDNET8
DeviceNet
Power Supply

PanelView Plus
Terminal

Serial Network Communication

The 1768 CompactLogix controllers have a built-in RS-232 serial port that you can configure.

IMPORTANT	Limit the length of serial (RS-232) cables to 15.2 m (50 ft).

DF1 Modes for Logix5000 Controllers

Mode	Functions
DF1 Point-to-Point	Communication between a controller and one other DF1-protocol-compatible device. This is the Default System mode. Default parameters are: • Communication Rate: 19,200 bps. • Data Bits: 8. • Parity: None. • Stop Bits: 1. • Control Line: No Handshake. • RTS send Delay: 0. • RTS Off Delay: 0. This mode is typically used to program a controller through its serial port.
DF1 Master	• Control of polling and message transmission between the master and slave nodes. • The master/slave network includes one controller configured as the master node and up to 254 slave nodes. Link slave nodes using modems or line drivers. • A master/slave network can have node numbers from 0...254. Each node must have a unique node address. Also, for your link to be a network, it must consist of one master and one slave station.
DF1 Slave	• A controller to operate as a slave station in a master/slave serial communication network. • When there are multiple slave stations on the network, link slave stations by using modems or line drivers to the master. When you have a single slave station on the network, you do not need a modem to connect the slave station to the master. You can configure the control parameters for no handshaking. You can connect 2...255 nodes to a single link. In DF1 Slave mode, a controller uses DF1 half-duplex protocol. • One node is designated as the master and controls who has access to the link. All of the other nodes are slave stations and must wait for permission from the master before transmitting.
DF1 Radio Modem	• Compatible with SLC 500 and MicroLogix 1500 controllers. • This mode supports Master and Slave, and Store and Forward modes.
User (channel 0 only)	• Communication with ASCII devices. • This requires your program to use ASCII instructions to read and write data from and to an ASCII device.
DH-485	• Communication with other DH-485 devices. • This multi-master, token-passing network permits programming and peer-to-peer messaging.

Master and Slave Communication

You can establish Modbus RTU Master and Slave communication via Application 129 provided in the samples directory of RSLogix 5000 programming software.

Communicate with DF1 Devices

You can configure a controller as a master or slave on a serial network. Use serial communication to get information to and from remote controllers (stations) when:

- the system contains three or more stations.
- communication occurs on a regular basis and requires leased-line, radio, or power-line modems.
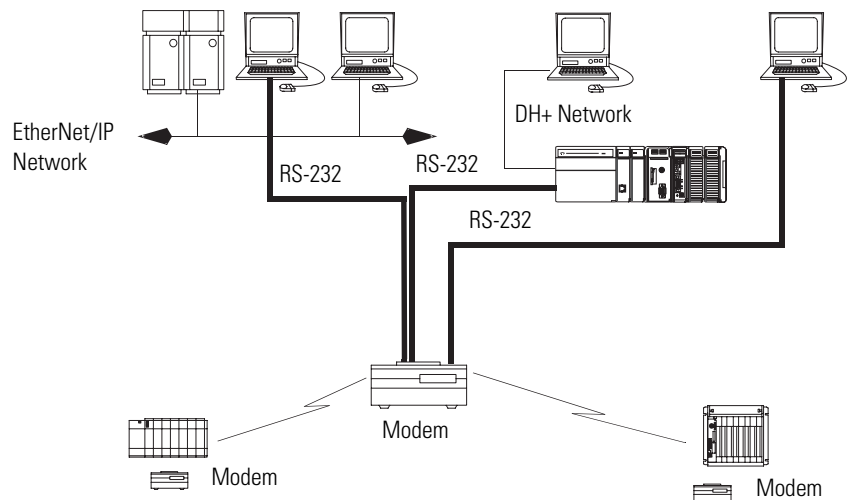
To configure your controller for DF1 communication, perform this procedure.

1. In the Controller Organizer of RSLogix 5000 programming software, right-click your controller and choose Properties.
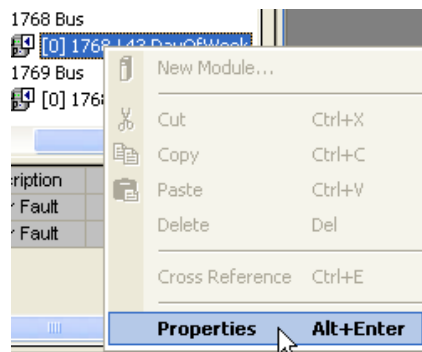
The Controller Properties dialog appears.
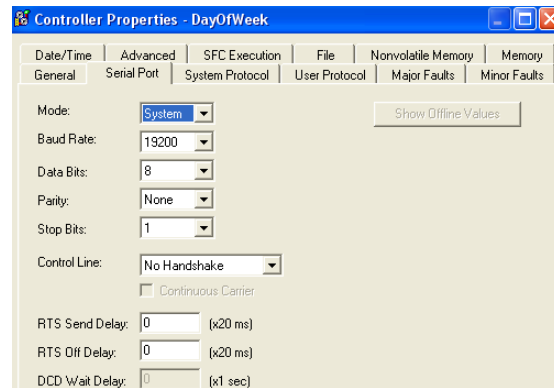
2. Click the Serial Port tab.

3. From the Mode pull-down menu, choose System.

4. Specify DF1 communication settings.

5. Click the System Protocol tab.

6. From the Protocol pull-down menu, choose DF1 Point-to-Point.

7. Specify DF1 system protocol settings.

8. Click OK.

Communicate with ASCII Devices

When you configure the serial port for User mode, you can use it to:

- read ASCII characters from a weigh scale module or bar code reader.
- send and receive messages from an ASCII triggered device, such as a MessageView terminal.

Connection from the Controller's Serial Port to the ASCII Device
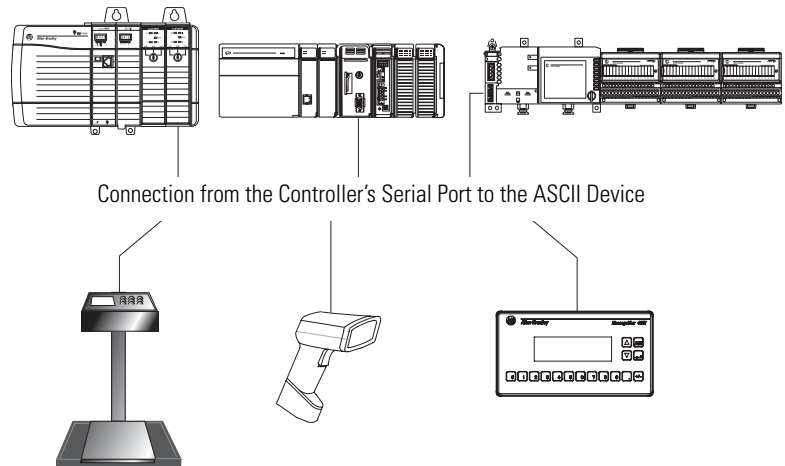
To communicate with ASCII devices, perform this procedure.

1. In the Controller Organizer of RSLogix 5000 programming software, right-click your controller and choose Properties.
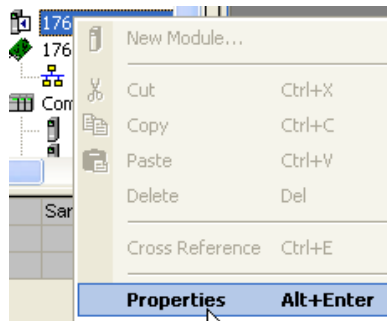
The Controller Properties dialog appears.
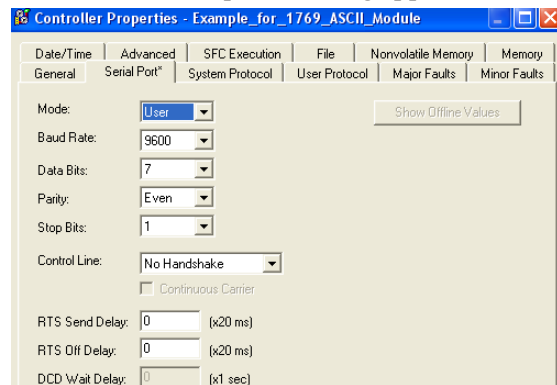
2. Click the Serial Port tab.

3. From the Mode pull-down menu, choose User.

4. Specify ASCII communication settings.

5. Click the User Protocol tab.
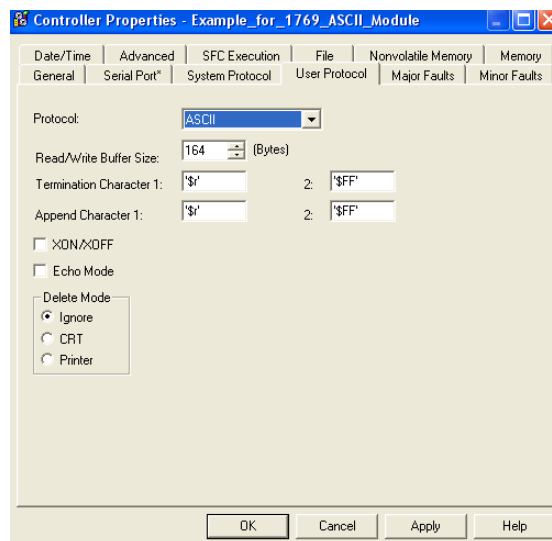
6. Specify ASCII user protocol settings.

7. Click OK.

Each controller supports several ladder diagram (LD) and structured text (ST) instructions to manipulate ASCII characters.

Read and Write ASCII Characters

Instruction	Command
ABL	Determine when the buffer contains termination characters.
ACB	Count the characters in the buffer.
ACL	Clear the buffer.
	Clear out ASCII serial port instructions that are currently executing or are in the queue.
AHL	Obtain the status of the serial port control lines.
	Turn the DTR signal on or off.
	Turn the RTS signal on or off.
ARD	Read a fixed number of characters.
ARL	Read a varying number of characters, up to and including the first set of termination characters.
AWA	Send characters and automatically append one or two additional characters to mark the end of the data.
AWT	Send characters.

Create and Modify Strings of ASCII Characters

Instruction	Command
CONCAT	Add characters to the end of a string.
DELETE	Delete characters from a string.
FIND	Determine the starting character of a sub-string.
INSERT	Insert characters into a string.
MID	Extract characters from a string.

Convert Data To or From ASCII Characters

Instruction	Command
STOD	Convert the ASCII representation of an integer value to a SINT, INT, DINT, or REAL value.
STOR	Convert the ASCII representation of a floating-point value to a REAL value.
DTOS	Convert a SINT, INT, DINT, or REAL value to a string of ASCII characters.
RTOS	Convert a REAL value to a string of ASCII characters.
UPPER	Convert the letters in a string of ASCII characters to upper case.
LOWER	Convert the letters in a string of ASCII characters to lower case.

Modbus Support

To use Logix5000 controllers on the Modbus protocol, you will need to access two example programs in the samples directory of RSLogix 5000 programming software that emulate the Modbus protocol. These programs are entitled ModbusMaster and ModbusSlave. To execute these programs, you will need to use the CH0 serial port.

For more information on these applications, see the Using Logix5000 Controllers as Masters and Slaves on Modbus Application Solution, publication CIG-AP129.

DH-485 Network Communication

For DH-485 communication, use the serial port of the controller. 1768 CompactLogix controllers can send and receive messages to and from other controllers on a DH-485 network. The DH-485 connection supports remote programming and monitoring via RSLogix 5000 programming software. However, excessive traffic over a DH-485 connection can adversely affect overall performance and lead to timeouts and decreased RSLogix 5000 configuration performance.

IMPORTANT   Use Logix5000 controllers on DH-485 networks only when you want to add controllers to an existing DH-485 network. For new applications with Logix5000 controllers, we recommend you use networks in the NetLinx open architecture.

The DH-485 protocol uses RS-485 half-duplex as its physical interface. RS-485 is a definition of electrical characteristics, not a protocol. You can configure the RS-232 port of a CompactLogix controller to act as a DH-485 interface. By using a 1761-NET-AIC converter and the appropriate RS-232 cable (catalog number 1756-CP3 or 1747-CP3), a CompactLogix controller can send and receive data on a DH-485 network.

CompactLogix DH-485 Network Communication Overview

CompactLogix Controller

Connection from CompactLogix
Controller to Port 1 or Port 2

1761-NET-AIC+
Converter

1747-AIC Programmable
Controller Isolated Link
Coupler

DH-485 Network

SLC 5/03 Controller
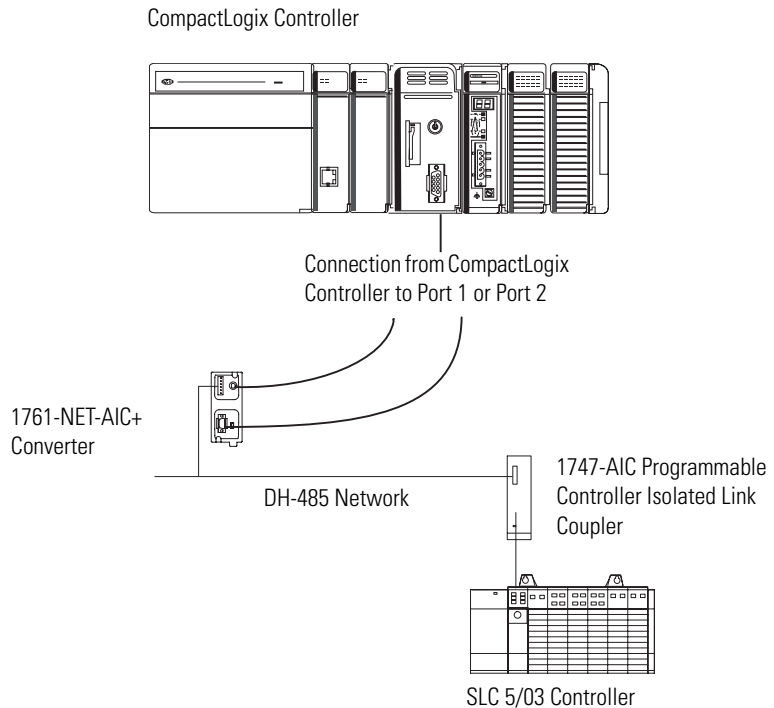
IMPORTANT	A DH-485 network consists of multiple cable segments. Limit the total length of all the segments to 1219 m (4000 ft).

You can have two controllers for each 1761-NET-AIC converter, but you need a separate cable for each controller. Connect the serial port of the controller to either port 1 or port 2 of the 1761-NET-AIC converter. Use the RS-485 port to connect the converter to the DH-485 network.

Cable Selection

Connection	**Required Cable**
Port 1 DB-9 RS-232, DTE connection	1747-CP3 or 1761-CBL-AC00
Port 2 Mini-DIN 8 RS-232 connection	1761-CBL-AP00 or 1761-CBL-PM02

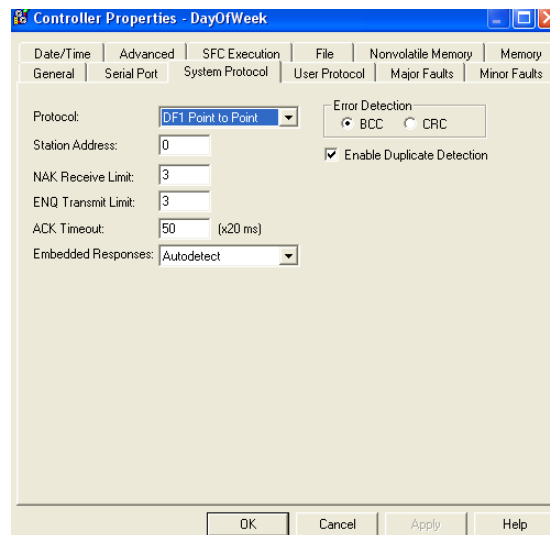To communicate with DH-485 devices, perform this procedure.

1. In the Controller Organizer of RSLogix 5000 programming software, right-click your controller and select Properties.
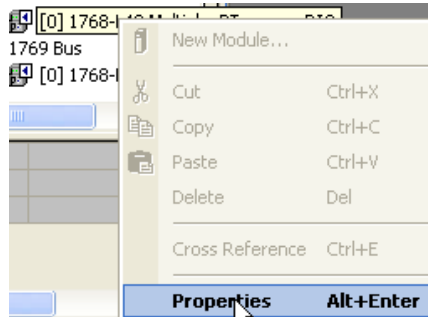
The Controller Properties dialog appears.
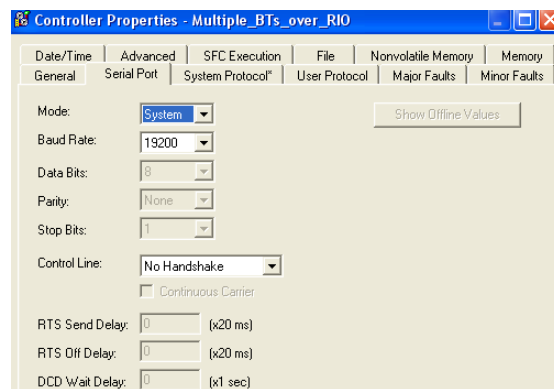
2. Click the Serial Port tab.

   a. From the Mode pull-down menu, choose System.

   b. From the Baud Rate pull-down menu, choose 9600 or 19200 bps.

IMPORTANT	The baud rate specifies the communication rate for the DH-485 port. All devices on the same DH-485 network must be configured for the same baud rate.

3. Click the System Protocol tab.
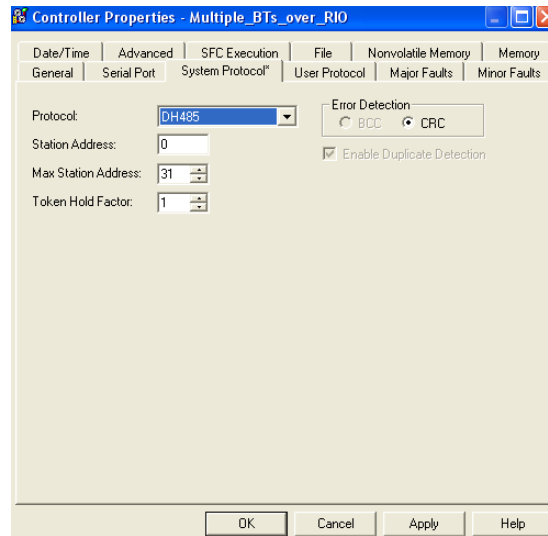
c. From the Station Address pull-down menu, choose a station address number from 1...31, decimal.

IMPORTANT	The station address specifies the node address of the controller on the DH-485 network.
	To optimize network performance, assign station addresses in sequential order.
	Assign initiators, such as workstations, the lowest station address numbers to minimize the time required to initialize the network.
	The maximum station address specifies the maximum node address of all the devices on the DH-485 network.

d. From the Max Station Address pull-down menu, choose a maximum station address number from 1...31, decimal.

IMPORTANT	To optimize network performance, make sure:
	• the maximum station address is the highest node number being used on the network.
	• that all the devices on the same DH-485 network have the same selection for the maximum station address.

e. In the Token Hold Factor box, type a token hold factor value from 1...4.

| IMPORTANT | The token hold factor is the number of transmissions (plus retries) that an address holding a token can send onto the data link each time that it receives the token. The default is 1. |

4. Click OK.

Additional Resources

These documents contain additional information pertinent to communicating over networks.

Resource	Description
EtherNet/IP Web Server Module User Manual, publication ENET-UM527	Explains how to use and troubleshoot the 1768-EWEB Web Server Module.
EtherNet/IP Modules in Logix5000 Control Systems User Manual, publication ENET-UM001	Explains how to use EtherNet/IP modules with Logix5000 controllers.
EtherNet/IP Performance Application Solution, publication ENET-AP001	Explains how to plan an EtherNet/IP network and improve overall network performance.
Logix5000 Controllers Design Considerations Reference Manual, publication 1756-RM094	Provides information pertinent to the design of Logix5000 systems.
ControlNet Modules in the Logix5000 Control Systems User Manual, publication CNET-UM001	Explains how to use ControlNet modules with Logix5000 controllers and provides technical specifications.
DeviceNet Modules in Logix5000 Control Systems User Manual, publication DNET-UM004	Explains how to use EtherNet/IP modules with Logix5000 controllers.
SCADA System Application Guide, publication AG-UM008	Explains how to configure a SCADA system.
Logix5000 Controllers General Instructions Reference Manual, publication 1756-RM003	Explains how to program Logix5000 controllers for sequential applications.
Logix5000 Controllers Common Procedures Programming Manual, publication 1756-PM001	Provides guidelines for the development of programs for Logix5000 controllers.
Using Logix5000 Controllers as Masters or Slaves on Modbus Application Solution, publication CIG-AP129	Details how to use Logix5000 controllers as Modbus RTU masters or slaves.
Data Highway/Data Highway Plus/Data Highway II/Data Highway-485 Cable Installation Manual, publication 1770-6.2.2	Explains how to plan and construct a Data Highway cable system.

Manage Controller Communication

Introduction

This chapter explains how to manage controller communication.

Topic	Page
Connection Overview	45
Producing and Consuming (interlock) Data	46
Sending and Receiving Messages	47
Calculating Connection Use	48
Connections Example	49
Additional Resources	50

Connection Overview

A Logix5000 system uses a connection to establish a communication link between two devices. There are several types of connections.

- Controller to local I/O modules or local communication modules
- Controller to remote I/O or remote communication modules
- Controller to remote I/O (rack-optimized) modules
- Produced and consumed tags
- Messages
- Controller access by RSLogix 5000 programming software
- Controller access by RSLinx software for HMI or other applications

Producing and Consuming (interlock) Data

The controller supports the ability to produce (broadcast) and consume (receive) system-shared tags over EtherNet/IP and ControlNet networks. Produced and consumed tags each require connections.

Produced and Consumed Tags

Tag Type	Description
Produced	A produced tag allows other controllers to consume the tag, which means that a controller can receive the tag data from another controller. The producing controller uses one connection for the produced tag and one connection for each consumer. The controller's communication device uses one connection for each consumer.
	As you increase the number of controllers that can consume a produced tag, you also reduce the number of connections the controller and communication device have available for other operations, such as communication and I/O.
Consumed	Each consumed tag requires one connection for the controller that is consuming the tag. The controller's communication device uses one connection for each consumer.

For two controllers to share produced or consumed tags, both controllers must be attached to the same Ethernet/IP or ControlNet network. You cannot bridge produced and consumed tags over two networks.

The number of available connections limits the number of tags that can be produced or consumed. If the controller uses all of its connections for I/O and communication devices, no connections are left for produced and consumed tags.

Sending and Receiving Messages

Messages transfer data to other devices, such as other controllers or operator interfaces. Some messages use connections to send or receive data. These connected messages can leave the connection open (cache) or close the connection when the message is done transmitting. Each message uses one connection, regardless of how many devices are in the message path.

Message Types

Message Type	Communication Method	Connected Message	Message Can Be Cached
CIP data table read or write	N/A	Yes[1]	Yes
PLC-2, PLC-3, PLC-5, or SLC (all types)	CIP	No	No
	CIP with Source ID	No	No
	DH+	Yes	Yes
CIP generic	N/A	Optional [2]	Yes[3]
Block-transfer read or write	N/A	Yes	Yes

[1] Starting with version 16 of RSLogix 5000 programming software, you can initiate these messages as unconnected.

[2] You can connect CIP generic messages. However, for most applications we recommend you leave CIP generic messages unconnected.

[3] Consider caching only if the target module requires a connection.

To conserve connections, configure one message to read from or write to multiple devices. You can programmatically change the target of a MSG instruction to optimize program size.

Determining Whether to Cache Message Connections

When you configure a MSG instruction, you can choose whether or not to cache the connection.

Caching Message Connections

Message Execution	Function
Repeatedly	Cache the connection. This keeps the connection open and optimizes execution time. Opening and closing a connection each time the message executes increases execution time.
Infrequently	Do not cache the connection. This closes the connection upon completion of the message, which frees up that connection for other uses.

Calculating Connection Use

The total connection requirements for a 1768 CompactLogix system include both local and remote (distributed) connections. You do not have to tally local controller connections because the controllers support all of the connections required for the maximum number of I/O modules and 1769-SDN modules in one system.

Tallying Remote Connection Use

Remote Connection Type	Device Quantity	Connections per Device	Total Connections
Remote EtherNet/IP communication module			
I/O configured as direct connection (none)		0 **or**	
I/O configured as rack-optimized connection		1	
Remote I/O module over an EtherNet/IP network (direct connection)		1	
Produced tag		1	
Each consumer		1	
Consumed tag		1	
Message (depending on type)		1	
Block-transfer message		1	
		Total	

Connections Example

In this example system, the 1768-L43 or 1768-L45 CompactLogix controller:

- monitor input or status data via the 1769-L35E CompactLogix controller.
- sends and receives messages to and from a ControlLogix controller on an EtherNet/IP network.
- produces a tag for the 1769-L35E CompactLogix controller, which the controller consumes.
- is programmed via RSLogix 5000 programming software.

Example CompactLogix System

RediSTATION Operator Interface

Series 9000 Photoeye

1769-ADN Adapter with Compact I/O Modules

DeviceNet Network

1756 ControlLogix Controller with 1756-ENBT Module

1768-L43 CompactLogix Controller with 1768-ENBT Modules

EtherNet/IP Network

Workstation

1769-L35E CompactLogix Controller
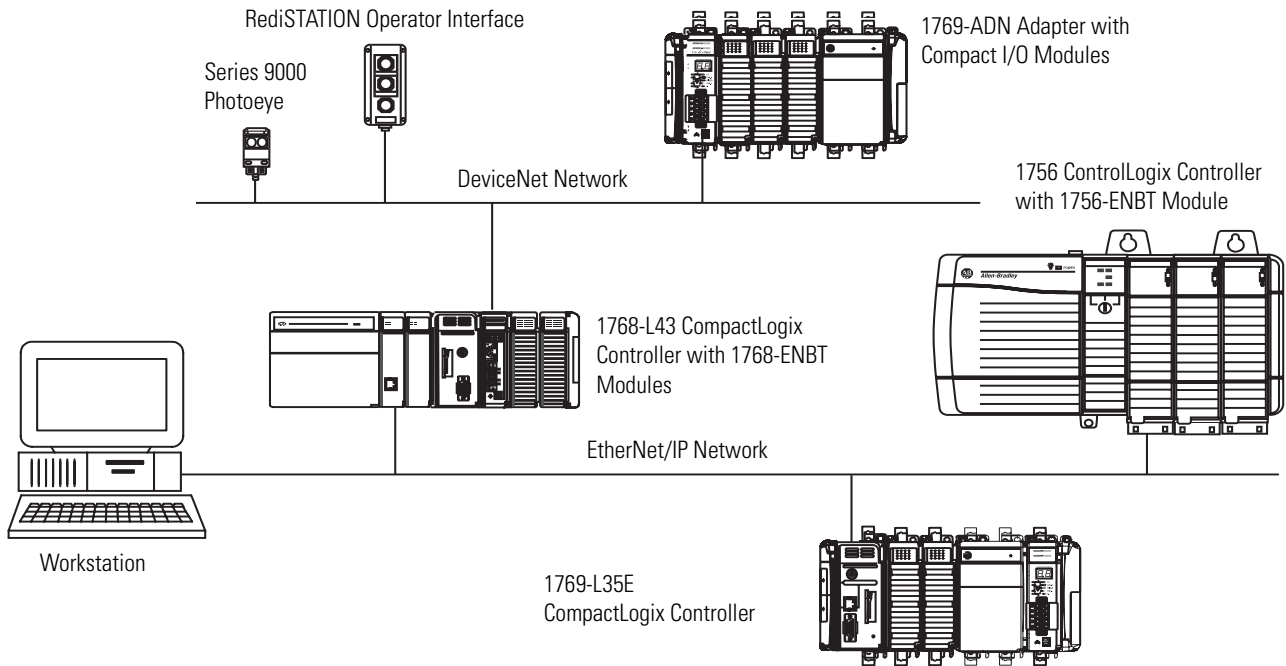
The 1756-ENBT and 1768-ENBT modules in this system use these connections.

Example Connection Types

Connection Type	Device Quantity	Connections per Device	Total Connections
Controller to RSLogix 5000 programming software	1	1	1
Message to 1756 ControlLogix controller	1	1	1
Message to 1769-L35E controller	1	1	1
Tag produced for the 1769-L35E CompactLogix controller	1	1	1
Tag consumed by the 1769-L35E CompactLogix controller	1	1	1
		Total	5

Additional Resources

These documents contain additional information pertinent to managing controller communication.

Resource	Description
Logix5000 Controllers Common Procedures Programming Manual, publication 1756-PM001	Provides guidelines for the development of projects for Logix5000 controllers.
Logix5000 Controllers Design Considerations Reference Manual, publication 1756-RM094	Provides information pertinent to the design of Logix5000 systems.
Logix5000 Controllers General Instructions Reference Manual, publication 1756-RM003	Explains how to program Logix5000 controllers for sequential applications.

Place 1768 and 1769 Modules

Introduction

This chapter explains the placement of 1768 and 1769 modules.

The 1768 CompactLogix controllers combine a 1768 backplane with a 1769 backplane. This combination includes the advantages of the 1768 architecture while retaining the advantages of 1769 I/O support.

Topic	Page
1768 Module Placement	51
1769 Module Placement	53

1768 Module Placement

Follow these guidelines as you place modules in the 1768 backplane.

1768 CompactLogix Controller	Guidelines
1768-L43 and 1768-L45	The 1768 power supply must be the leftmost module in the 1768 backplane.
	The controller must be the rightmost module in the 1768 backplane.
	Up to two 1768 communication modules can reside between the controller and power supply in any one of these combinations: • 1768-ENBT or 1768-EWEB for EtherNet/IP communication (maximum of two) • 1768-CNB or 1768-CNBR for ControlNet communication (maximum of two)
1768-L43	Two chassis slots are available.
	The controller supports a maximum of 3 banks for a total of 16 modules.
1768-L45	Four chassis slots are available.
	The controller supports a maximum of 3 banks for a maximum of 30 modules.
	Up to four 1768-M04SE for SERCOS motion control modules can be used.

1768 Module Placement Overview

Place 1768 modules in the 1768 backplane.   1768-L43 controller is shown.

1768 Power Supply   1768 Modules

L1
L2
PE

Slot 2   Slot 1   Slot 0
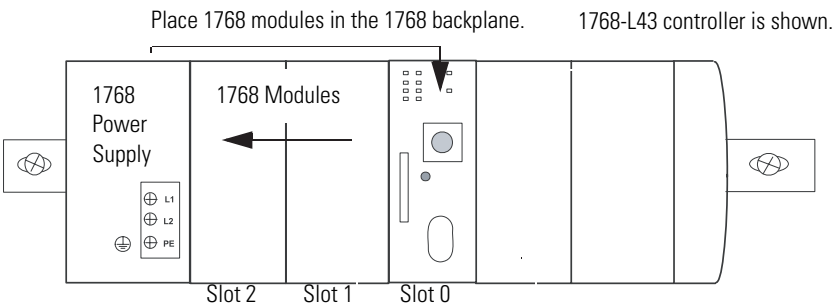
- The 1768 slots are numbered right to left, starting with the controller in slot 0.
- The 1768 backplane requires one 1768 power supply.
- The 1768-PA3 power supply is a dual input supply that operates in these ranges:
  - 86...265V ac
  - 108...132V dc

The 1768-PB3 power supply is a single-input power supply whose power supply range is 16.8...31.2V dc.

1768 Power Supply

Place 1768 modules in the 1768 backplane.

1768 Power Supply

1768 Modules

1769 I/O

L1
L2
PE

Controller sends:
– 5V dc to 1768 modules.
– 5V/24V dc to 1769 I/O modules.
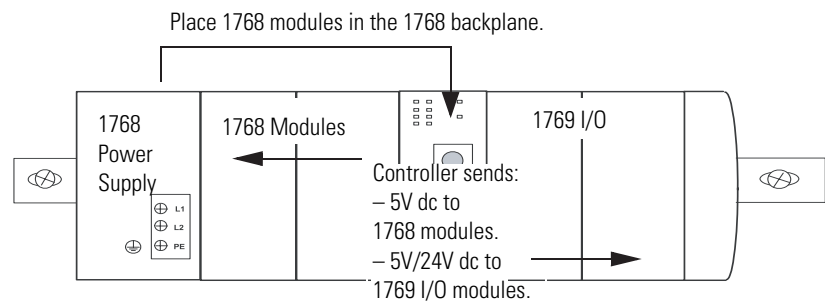
The 1768-PA3 and 1768-PB3 power supplies also offer a 24V dc external power source. These power supplies require that a 1768 CompactLogix controller be installed.

- The power supply sends 24V dc to the controller in slot 0.
- The controller converts the 24V dc to 5V dc and 24V dc, and distributes the power as needed.
  - 5V/24V power to 1769 I/O modules on the right side of the controller
  - 5V power to communication or motion modules on the left side of the controller

The 1768 modules do not have a distance rating to the 1768 power supply.

1769 Module Placement

The CompactLogix controllers support the following:

- 1768-L43, a maximum of 16 local 1769 I/O modules
- 1768-L45, a maximum of 30 local 1769 I/O modules

Follow these guidelines as you place 1769 modules to the right of the 1768 controller:

- Up to eight 1769 modules can be attached to the right of the 1768 system.
- The 1769 I/O modules connected directly to the 1768 controller do not need a 1769 power supply.

IMPORTANT	Never put a 1769 power supply in the 1768 backplane. A 1769 power supply in the 1768 backplane causes the controller to generate a major fault that cannot be cleared until you remove the 1769 power supply.

- Additional 1769 modules must be in additional I/O banks.
- Each additional I/O bank must have its own power supply. Use any 1769 power supply.
- Each 1769 module also has a power supply distance rating, which is the number of modules from the power supply.

IMPORTANT	Each module must be within its distance rating. See the specifications for the module to determine its distance rating.

- Place up to eight 1769 I/O modules to the left or the right of the 1769 power supply.
- Each additional I/O bank must connect to the main rack by using standard 1769-CRL*x* extension cables.

1769 Module Placement Overview

Place 1769 I/O modules to the right of the controller.

1769 I/O

Slot 0   Slot 1   Slot 2
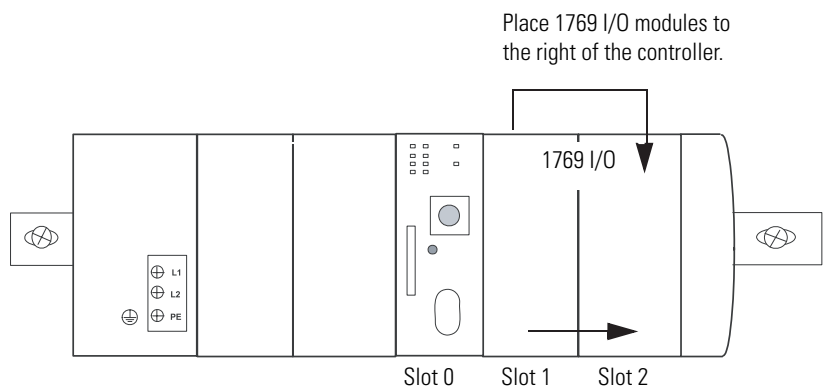
The 1769 slots are numbered left to right, starting with the controller as slot 0.

Notes:

Configure and Monitor I/O

Introduction

This chapter explains how to configure and monitor I/O.

Selecting I/O Modules

When selecting 1769 I/O modules, choose:

- specialty I/O modules, when appropriate.

  Some modules have field-side diagnostics, electronic fusing, or individually isolated inputs/outputs.

- a 1492 wiring system for each I/O module as an alternative to the terminal block that comes with the module.
- 1492 PanelConnect modules and cables if you are connecting input modules to sensors.

Each 1769 I/O module includes a built-in removable terminal block with finger-safe cover for connections to I/O sensors and actuators. The terminal block is behind a door at the front of the module. I/O wiring can be routed from beneath the module to the I/O terminals. When planning I/O communication, consider:

- which CompactLogix I/O modules to use.
- where to place CompactLogix I/O modules.
- how CompactLogix I/O modules operate.

Local I/O Performance

For the best local I/O performance in a 1768 CompactLogix system:

- set an individual requested packet interval (RPI) for each local 1769 I/O module.
- use a faster RPI for time critical I/O without impacting overall 1769 I/O performance.

I/O update times do not affect overall 1768 bus performance, such as motion performance or controller performance.

Configuring I/O

To communicate with an I/O module in your system, add the module to the I/O Configuration folder of the controller.

```
Controller example_1768
    Controller Tags
    Controller Fault Handler
    Power-Up Handler
Tasks
    MainTask
        MainProgram
    Unscheduled Programs / Phases
Motion Groups
    Ungrouped Axes
Trends
Data Types
    User-Defined
    Strings
    Predefined
    Module-Defined
I/O Configuration
    1768 Bus
        [0] 1768-L43 example_1768
    1769 Bus
        [0] 1768-L43 example_1768
```

Add I/O modules to the 1769 backplane.
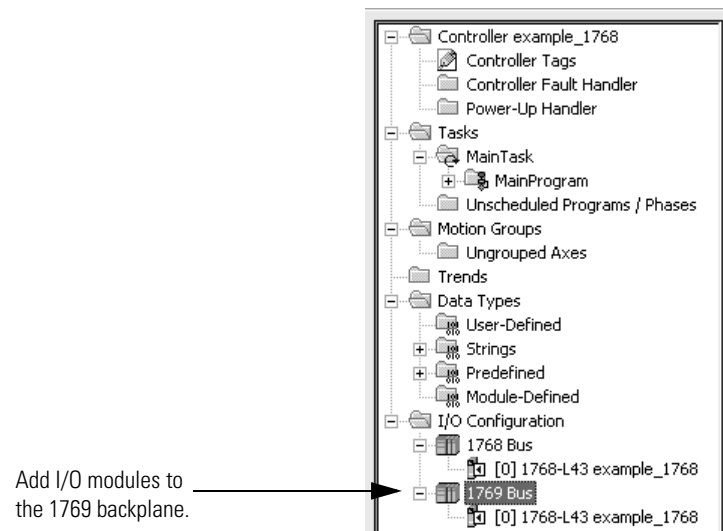
When you add a module, you also define a specific configuration for the module. While the configuration options vary from module to module, there are some common options that you typically configure.

Configuration Options for I/O Modules

Configuration Option	Description
Requested Packet Interval (RPI) (applies to local 1769 I/O and distributed I/O)	The RPI specifies the period at which data updates over a connection. For example, an input module sends data to a controller at the RPI that you assign to the module. • Typically, you configure an RPI in milliseconds (ms). The minimum RPI for 1769 I/O is 1 millisecond. • If a ControlNet network connects the devices, the RPI reserves a slot in the stream of data flowing across the ControlNet network. The timing of this slot may not coincide with the exact value of the RPI, but the control system guarantees that the data transfers at least as often as the RPI.
Change of State (COS) (only applies to distributed I/O)	Digital I/O modules use COS to determine when to send data to the controller. If a COS does not occur within the RPI time frame, the module multicasts data at the specified RPI. Because the RPI and COS functions are asynchronous to the logic scan, it is possible for an input to change state during program scan execution. If this is a concern, buffer input data so your logic has a stable copy of data during its scan. Use the Synchronous Copy (CPS) instruction to copy the input data from your input tags to another structure and use the data from that structure.
Communication Format (only applies to distributed I/O)	Many I/O modules support different formats. The chosen communication format also determines: • data structure of tags. • connections. • network use. • ownership. • whether the module returns diagnostic information.
Electronic Keying (applies to local 1769 I/O and distributed I/O)	When you configure a module, you specify the slot number for the module. However, it is possible to place a different module in that slot. Electronic keying lets you protect your system against the accidental placement of the wrong module in a slot. The chosen keying option determines how closely any module in a slot must match the configuration for that slot before the controller will open a connection to the module. Keying options differ depending on your application needs.

IMPORTANT   The RSLogix 5000 configuration dialogs for 1769 I/O modules offer a Hold Last State option for how to react when the controller faults.

Although the 1768-L43 an 1768-L45 controllers do not support the Hold Last State option for 1769 I/O modules when they are configured locally, this feature is available when they are connected on DeviceNet using a 1769-ADN adapter.

I/O Connections

A Logix5000 system uses connections to transmit I/O data.

Logix5000 Connection Types

Connection	Description
Direct (applies to all 1769 I/O)	A direct connection is a real-time, data-transfer link between the controller and an I/O module. The controller maintains and monitors the connection between the controller and the I/O module. Any break in the connection, such as a module fault or the removal of a module while under power, causes the controller to set fault status bits in the data area associated with the module. Typically, analog I/O modules, diagnostic I/O modules, and specialty modules require direct connections.
Rack-optimized (applies to distributed I/O)	For digital I/O modules, you can select rack-optimized communication. A rack-optimized connection consolidates connection usage between the controller and all of the digital I/O modules on a rack (or DIN rail). Rather than having individual, direct connections for each I/O module, there is one connection for the entire rack (or DIN rail).

Configure Distributed I/O on an EtherNet/IP Network

To communicate with distributed I/O modules over an EtherNet/IP network, add an EtherNet/IP adapter and I/O modules to the controller's I/O Configuration folder.

Within the I/O Configuration folder, organize the modules into a hierarchy of tree/branch and parent/child.

Configuring I/O on EtherNet/IP Network

For a typical distributed I/O network…
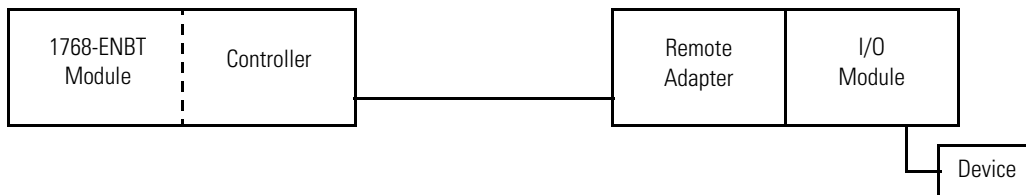
…you build the I/O configuration in this order.

1. Add the remote adapter for the distributed I/O chassis or DIN rail.

2. Add the distributed I/O modules.
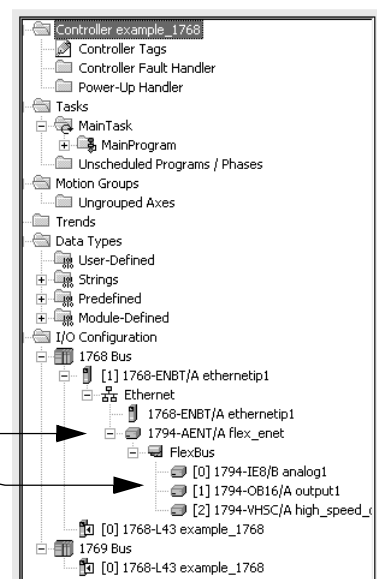
Configuring Distributed I/O on a DeviceNet Network

To communicate with the I/O modules over a DeviceNet network, add the DeviceNet scanner (catalog number 1769-SDN) to the I/O Configuration folder of the controller. You define a scanlist within the DeviceNet scanner to communicate data between devices and the controller.

For a typical distributed I/O network…

Single Network
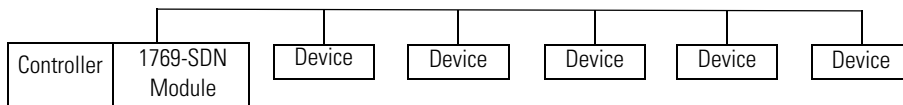
Several Smaller Distributed Networks (subnets)
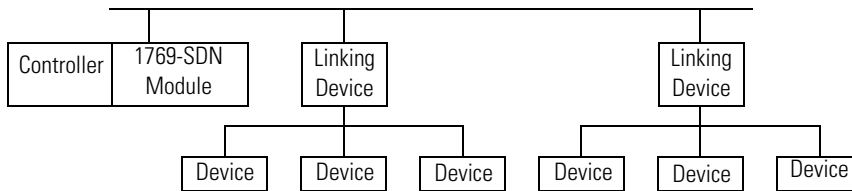
…you build the I/O configuration in this order.

Add the local scanner module.
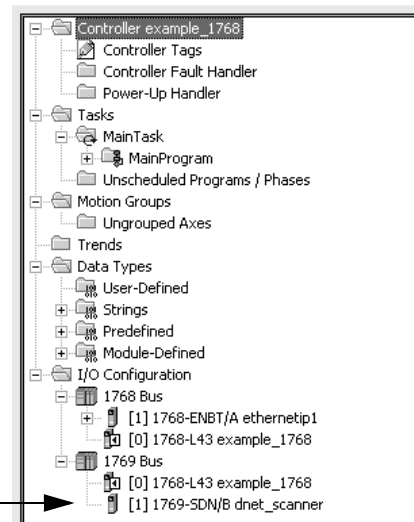
Address I/O Data

I/O information is presented as a set of tags.

- Each tag uses a structure of data. The structure depends on the specific features of the I/O module.
- The name of the tags is based on the location of the I/O module in the system.

An I/O address follows this format.

| `Location` | `:Slot` | `:Type` | `.Member` | `.SubMember` | `.Bit` |

☐ = Optional

I/O Address Components

Where	Is
Location	Network location
	LOCAL = same chassis or DIN rail as the controller
	ADAPTER_NAME = identifies remote communication adapter or bridge module
Slot	Slot number of I/O module in its chassis or DIN rail
Type	Type of data
	I = input
	O = output
	C = configuration
	S = status
Member	Specific data from the I/O module, depending on what type of data the module can store
	• For a digital module, a data member usually stores the input or output bit values
	• For an analog module, a channel member (CH#) usually stores the data for a channel
Submember	Specific data related to a member
Bit	Specific point on a digital I/O module; depends on the size of the I/O module (0...31 for a 32-point module)

Determine When Data is Updated

CompactLogix controllers update data asynchronously with the execution of logic. Use the flowchart to determine when a producer, such as a controller, input module, or bridge module, will send data.

Overview - Updating Data

```
Input or output data? ──Output──▶ Local or distributed? ──────Local──────┐
        │                              │                                  │
      Input                        Distributed                            │
        ▼                              │                                  │
Local or distributed? ──Distributed──▶│                                   │
        │                             │                                   │
      Local                           ▼                                   ▼
        ▼                    ┌──────────────────┐           ┌──────────────────────────┐
┌──────────────────┐        │ Over an EtherNet/ │           │ Data is written to the   │
│ Data is read from│        │ IP network,       │           │ module at the RPI        │
│ the module at the│        │ distributed data  │           │ and at the end of every  │
│ configured RPI.  │        │ is usually sent   │           │ task.                    │
└──────────────────┘        │ close to the RPI. │           └──────────────────────────┘
                            └──────────────────┘
```
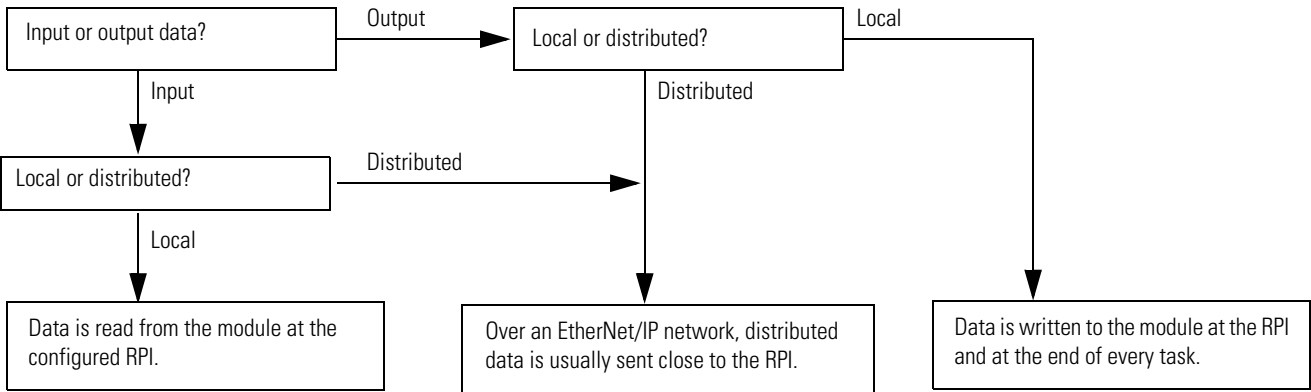
IMPORTANT	If you need I/O values used during logic execution to be from one moment in time, such as at the beginning of a ladder program, use the Synchronous Copy (CPS) instruction to buffer I/O data.

Monitor I/O Modules

To monitor I/O modules, you can:

- use RSLogix 5000 programming software to display fault data.
- program logic to monitor fault data so you can take appropriate action.

Display Fault Data

Fault data for certain types of module faults can be viewed through the programming software.

To display fault data, perform this procedure.

1. In RSLogix 5000 programming software, right-click Controller Tags in the Controller Organizer and choose Monitor Tags.

The Monitor Tags display appears.

The default display setting for the fault data is decimal.

2. Change the display setting for the fault data to Hex to read the fault code.

If the module faults but maintains an open connection to the controller, the controller tags database displays the fault value 16#0E01_0001.

| 31 | | 27 | | 23 | | 19 | | 15 | | 11 | | 7 | | 3 | | 0 | |

| Fault_Code_Value | | | Reserved | | | |
| Reserved | FaultCode | FaultInfo | | | | |

0 = Connection Open

1 = Connection Closed

}

Connection_Closed
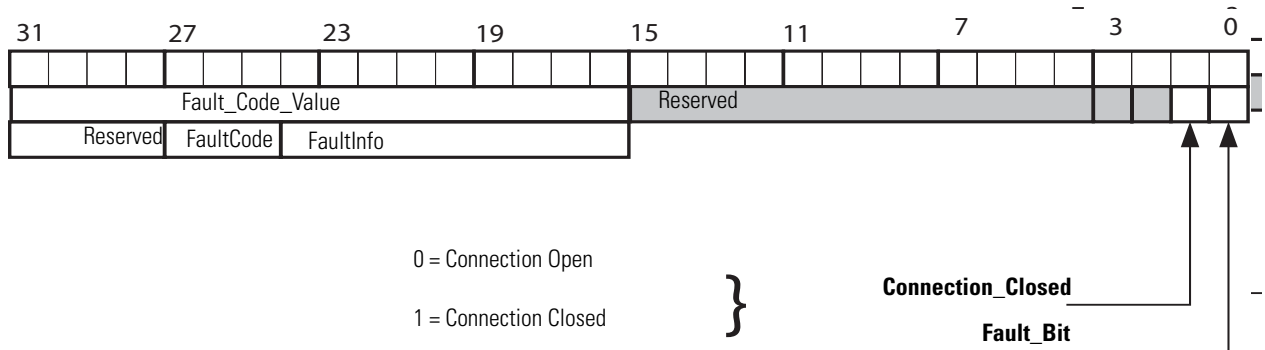
Fault_Bit

Fault Word Bits

Bit	Description
Fault_Bit	This bit indicates that at least one bit in the fault word is set (1). If all of the bits in the fault word are cleared (0), this bit is cleared (0).
Connection_Closed	This bit indicates whether the connection to the module is open (0) or closed (1). If the connection is closed (1), the Fault_Bit it set (1).

Display Fault Data via Module Properties Dialog
To display fault data via another another option in RSLogix 5000 programming software, perform this procedure.

1. Right-click your 1769 I/O module and choose Module Properties.
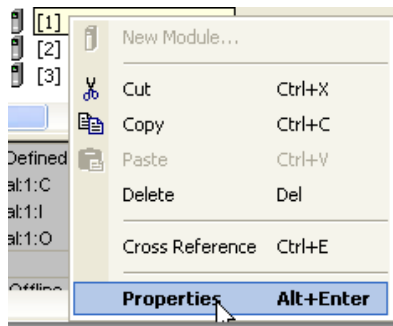
The Module Properties dialog appears.
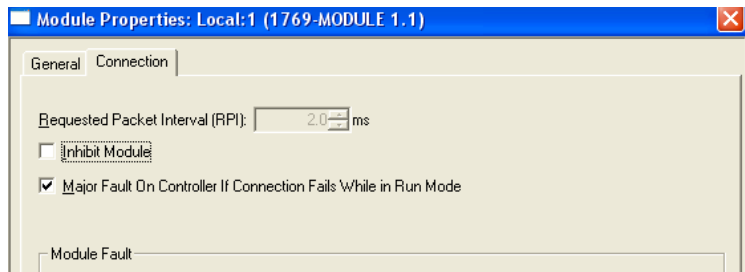
2. Click the Connection tab.

3. From the Module Fault pull-down menu, view any faults affecting your 1769 I/O module.

End-cap Detection and Module Faults

If the module adjacent to the end cap faults, or any other fault that the controller interprets as bus integrity lost (such as power loss in an expansion I/O rack) occurs, communication with all of the local 1769 I/O ceases. If any of these 1769 I/O modules are configured as required, the controller faults.

Reconfigure an I/O Module

If an I/O module supports reconfiguration, you can reconfigure the module via:

- the Module Properties dialog in RSLogix 5000 programming software.
- a MSG instruction in program logic.

Reconfigure a Module via RSLogix 5000 Programming Software

To change the configuration of an I/O module via RSLogix 5000 programming software, perform this procedure.

1. In the Controller Organizer of RSLogix 5000 programming software, right-click the module in the I/O Configuration tree and choose Properties.
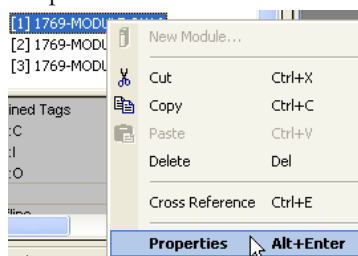
The Module Properties dialog appears. The appearance of the Module Properties dialog differs from one I/O module to the next.
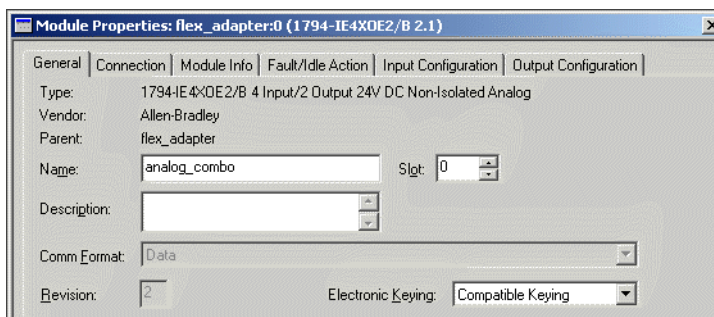
2. Configure the I/O module.

Reconfigure a Module via a MSG Instruction

Use a MSG instruction of type Module Reconfigure to send new configuration information to an I/O module. During the reconfiguration:

- input modules continue to send input data to the controller.
- output modules continue to control their output devices.

1. Click ☐ in the MSG box.

   The Module Configuration dialog appears.
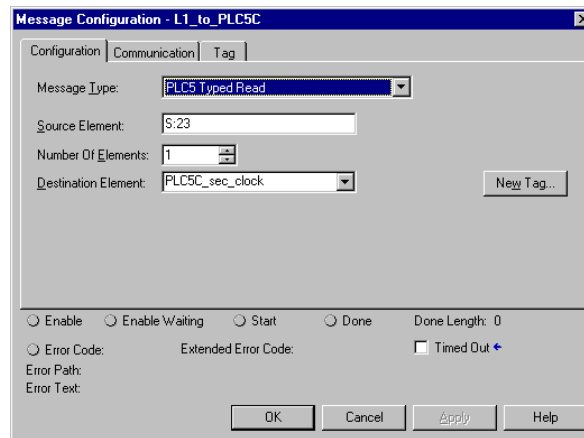
   a. From the Message Type pull-down menu, choose Module Reconfigure.
   b. In the Source Element box, type the appropriate information.
   c. In the Number of Elements box, type the number of elements.
   d. From the Destination Element pull-down menu, choose the instruction's destination element.

   The message instruction's destination determines how the message is configured.

2. Click the Communication tab.
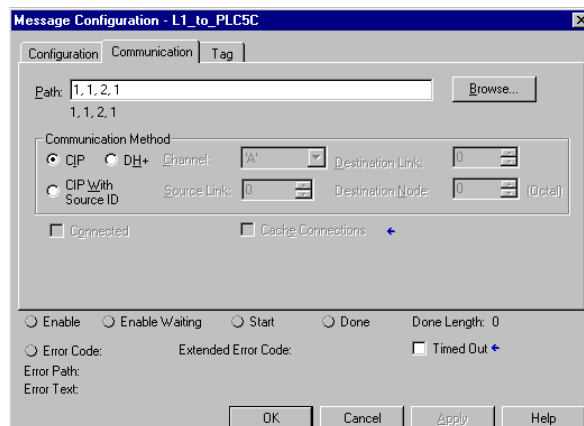
a. Specify the path of the module for which you sent the message instruction to the I/O configuration tree.

If the module	Then
Has been added	Click Browse to choose the path.
Has not been added	Type the path in the Path box.

b. Click OK.

Additional Resources

These documents contain additional information pertinent to configuring and monitoring I/O.

Resource	Description
Compact I/O Selection Guide, publication 1769-SG002	Details Compact I/O products for every application.
Compact I/O Analog Modules User Manual, publication 1769-UM002	Explains how to design, program, and troubleshoot Compact I/O analog modules.
Compact I/O 1769-IR6 RTD/Resistance Input Module User Manual, publication 1769-UM005	Explains how to design, program, and troubleshoot Compact I/O, CompactLogix, or MicroLogix 1500 controllers.
Compact I/O 1769-IT6 Thermocouple/mV Input Module User Manual, publication 1769-UM004	Explains how to design, program, and troubleshoot a CompactLogix system that use this Compact I/O 1769-IT6 module.
Logix5000 Controllers Common Procedures Programming Manual, publication 1756-PM001	Provides guidelines for the development of programs for Logix5000 controllers.
Logix5000 Controllers Design Considerations Reference Manual, publication 1756-RM094	Provides information pertinent to the design of Logix5000 systems.
EtherNet/IP Communication Modules in Logix5000 Control Systems User Manual, publication ENET-UM001	Explains how to configure, use, and troubleshoot EtherNet/IP communication modules and also provides technical specifications.
DeviceNet Communication Modules in Logix5000 Control Systems User Manual, publication DNET-UM004	Explains how to configure and use DeviceNet communication modules.
ControlNet Communication Modules in Logix5000 Control Systems User Manual, publication CNET-UM001	Explains how to configure and use ControlNet communication modules.
Logix5000 Controllers General Instruction Set Reference Manual, publication 1756-RM003	Details how to program the controller for sequential applications.

Develop Applications

Introduction

This chapter explains how to develop applications.

Topic	Page
Managing Tasks	69
Developing Programs	70
Organizing Tags	74
Selecting a Programming Language	75
Monitoring Controller Status	77
Monitoring Connections	77
Select a System Overhead Percentage	80
Additional Resources	83

Managing Tasks

A Logix5000 controller lets you use multiple tasks to schedule and prioritize the execution of your programs based on specific criteria. This multitasking allocates the controller's processing time among the different operations in your application.

IMPORTANT	• The controller executes only one task at a time.
	• One task can interrupt another executing task and take control.
	• In any given task, only one program executes at a time.

Developing Programs

The controller operating system is a preemptive multitasking system that is IEC 1131-3 compliant. This environment provides:

- tasks to configure controller execution.
- programs to group data and logic.
- routines to encapsulate executable code written in a single programming language.

Program Development Overview

Control Application

```
┌─────────────────────────────────────────────────────────────────────┐
│              ┌──────────────────────────────┐                        │
│              │   Controller Fault Handler    │                        │
│              └──────────────────────────────┘                        │
│    ┌──────────────────────────────────────────────────────────┐      │
│    │ Task 8                                                    │      │
│   ┌──────────────────────────────────────────────────────────┐│      │
│   │ Task 1                                                    ││      │
│   │                                    ┌──────────────────┐   ││      │
│   │                                    │  Configuration   │   ││      │
│   │                                    └──────────────────┘   ││      │
│   │                                    ┌──────────────────┐   ││      │
│   │                                    │     Status       │   ││      │
│   │                                    └──────────────────┘   ││      │
│   │                                    ┌──────────────────┐   ││      │
│   │        ┌─────────────────────────┐ │    Watchdog      │   ││      │
│   │        │ Program 32              │ └──────────────────┘   ││      │
│   │       ┌─────────────────────────┐                        ││      │
│   │       │ Program 1               │                        ││      │
│   │       │ ┌─────────────┐ ┌──────────────┐                 ││      │
│   │       │ │Main Routine │ │Program (local)│                ││      │
│   │       │ └─────────────┘ │     Tags      │                ││      │
│   │       │ ┌─────────────┐ └──────────────┘                 ││      │
│   │       │ │Fault Routine│                                  ││      │
│   │       │ └─────────────┘ ┌──────────────┐                 ││      │
│   │       │          │ Other Routines │                      ││      │
│   │       │          └──────────────┘                        ││      │
│   │       └─────────────────────────┘                        ││      │
│   └──────────────────────────────────────────────────────────┘│      │
│    └──────────────────────────────────────────────────────────┘      │
│  ┌───────────────────┐ ┌─────────────┐ ┌────────────────────┐        │
│  │Controller (global)│ │  I/O Data   │ │ System-shared Data │        │
│  │       Tags        │ │             │ │                    │        │
│  └───────────────────┘ └─────────────┘ └────────────────────┘        │
└─────────────────────────────────────────────────────────────────────┘
```
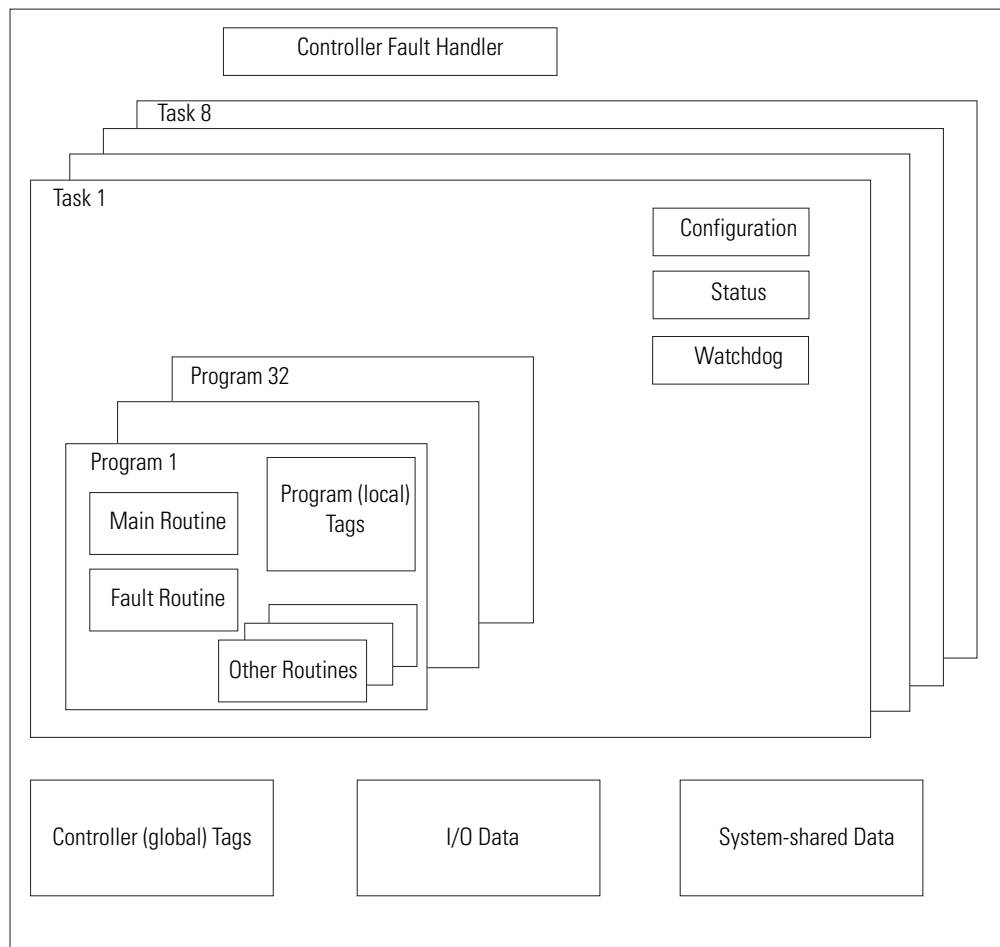
Defining Tasks

A task provides scheduling and priority information for a set of one or more programs. You can configure tasks as continuous, periodic, or event.

Types of Logix5000 Controller Tasks

Task Execution	Task Type	Description
All of the time	Continuous	The continuous task runs in the background. Any CPU time not allocated to other operations (such as motion, communication, and other tasks) is used to execute the programs in the continuous task. • The continuous task runs constantly. When the continuous task completes a full scan, it restarts immediately. • A project does not require a continuous task. If used, there can be only one continuous task.
• At a set interval, such as every 100 ms • Multiple times within the scan of your other logic	Periodic	A periodic task performs a function at a specific interval. • Whenever the time for the periodic task expires, the task interrupts any lower priority tasks, executes once, and then returns control to where the previous task left off. • You can configure the time period from 0.1...2000 μs. The default is 10 ms. It is also controller and configuration dependent. • The performance of a periodic task depends on the type of Logix controller and on the logic in the task.
Immediately when an event occurs	Event	An event task performs a function only when a specific event (trigger) occurs. In a 1768 CompactLogix controller, the trigger for the event task can be: • a consumed tag trigger. • an EVENT instruction. • an axis trigger. • a motion event trigger.

• The 1768-L43 controller supports 16 tasks, only 1 of which can be continuous.
• The 1768-L45 controller supports 30 tasks, only 1 of which can be continuous.

A task can have as many as 32 separate programs, each with its own executable routines and program-scoped tags. Once a task is triggered (activated), all of the programs assigned to the task execute in the order in which they are grouped. A program can appear only once in the Controller Organizer and cannot be shared by multiple tasks.

Specifying Task Priorities

Each task in the controller has a priority level. The operating system uses the priority level to determine which task to execute when multiple tasks are triggered. You can configure periodic tasks to execute from the lowest priority of 15 up to the highest priority of 1. Higher-priority tasks interrupt any lower-priority tasks. Periodic tasks take priority and always interrupt continuous tasks, which have the lowest priority.

Defining Programs

Each program contains program tags, a main executable routine, other routines, and an optional fault routine. Each task can schedule up to 32 programs.

Scheduled programs within a task execute to completion from first to last. Programs not attached to any task show up as unscheduled programs. You must specify (schedule) a program within a task before the controller can scan the program.

Unscheduled programs within a task are downloaded to the controller with the entire project. The controller verifies unscheduled programs, but does not execute them.

Defining Routines

A routine is a set of logic instructions in a single programming language, such as ladder logic. Routines provide the executable code for the project in a controller. A routine resembles a program file or subroutine in a PLC or SLC controller.

Each program has a main routine. This is the first routine to execute when the controller triggers the associated task and calls the associated program. Use logic, such as the Jump to Subroutine (JSR) instruction, to call other routines.

You can also specify an optional program fault routine. The controller executes this routine if it encounters an instruction-execution fault within any of the routines in the associated program.

Sample Controller Projects

RSLogix 5000 programming software includes sample projects that you can copy and then modify to fit your application.

To obtain a list of sample projects, perform this procedure.

1. In RSlogix 5000 programming software, from the Help menu, choose Vendor Sample Projects.
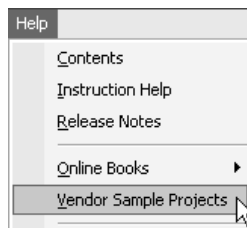
2. Scroll down to select the appropriate set of sample projects.

Organizing Tags
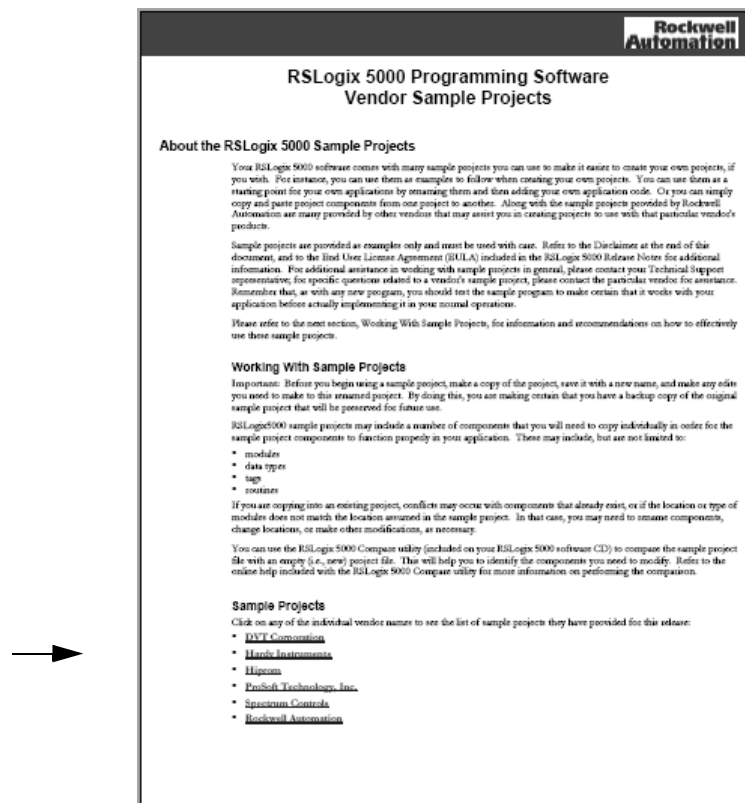
With a Logix5000 controller, you use a tag (alphanumeric name) to address data (variables). In Logix5000 controllers, there is no fixed, numeric format. The tag name itself identifies the data. This naming convention lets you:

- organize your data to mirror your machinery.
- document your application as you develop it.

Tag Organization Overview

	Tag Name	Alias For	Base Tag	Type
	north_tank_mix			BOOL
	north_tank_pressure			REAL
	north_tank_temp			REAL
	⊞-one_shots			DINT
	⊞-recipe			TANK[3]
	⊞-recipe_number			DINT
	replace_bit			BOOL
	⊞-running_hours			COUNTER
	⊞-running_seconds			TIMER
	start			BOOL
	stop			BOOL

Program Tags - MainProgram

Scope: MainProgram   Show: Show All   Sort: Tag Nam

Analog I/O Device →
Integer Value →
Storage Bit →
Counter →
Timer →
Digital I/O Device →

Monitor Tags **Edit Tags**

When you create a tag, assign these properties to the tag:

- Tag type
- Data type
- Scope

Selecting a Programming Language

The CompactLogix controller supports these programming languages, both online and offline.

Logix5000 Programming Languages

Required Language	Programs
Ladder diagram (LD)	Continuous or parallel execution of out-of-sequence operations
	Boolean or bit-based operations
	Complex logical operations
	Message and communication processing
	Machine interlocking
	Operations that service or maintenance personnel may have to interpret in order to troubleshoot the machine or process
Function block diagram (FBD)	Continuous process and drive control
	Loop control
	Calculations in circuit flow
Sequential function chart (SFC)	High-level management of multiple operations
	Repetitive sequence of operations
	Batch process
	Motion control using structured text
	State machine operations
Structured text (ST)	Complex mathematical operations
	Specialized array or table loop processing
	ASCII string handling or protocol processing

Add-On Instructions

With RSLogix 5000 programming software, version 16, you can design and configure sets of commonly used instructions to increase project consistency. Similar to the built-in instructions contained in Logix5000 controllers, these instructions you create are called Add-On Instructions. Add-On Instructions reuse common control algorithms. With them, you can:

- ease maintenance by animating logic for a single instance.
- protect intellectual property with locking instructions.
- reduce documentation development time.

You can use Add-On Instructions across multiple projects. You can define your instructions, obtain them from somebody else, or copy them from another project.

Once defined in a project, Add-On Instructions behave similarly to the built-in instructions in Logix5000 controllers. They appear on the instruction tool bar for easy access, as do internal RSLogix 5000 software instructions.

Saving Time

With Add-On Instructions, you can combine your most commonly used logic into sets of reusable instructions. You save time when you create instructions for your projects and then share them with others. Add-On Instructions increase project consistency since commonly used algorithms all work in the same manner, regardless of who implements the project.

Using Standard Editors

You create Add-On Instructions by using one of three RSLogix 5000 software programming editors.

- Standard Ladder
- Function Block Diagram
- Structured Text

Once you have created instructions, you can use them in any RSLogix 5000 editor.

Exporting Add-On Instructions

You can export Add-On Instructions to other projects as well as copy and paste them from one project to another. Give each instruction a unique name so that you don't accidentally overwrite another instruction of the same name.

Using Context Views

Context views let you visualize an instruction's logic for a specific instant, simplifying online troubleshooting of your Add-On Instructions. Each instruction contains a revision, a change history, and an auto-generated help page.
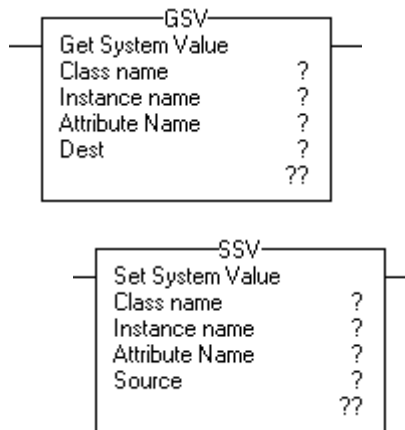
Creating Custom Help

When you create an instruction, you enter information for the description boxes in software dialogs, information that becomes what is known as Custom Help. Custom Help makes it easier for users to get the help they need when implementing the instructions.

Applying Source Protection

As the creator of Add-On Instructions, you can limit users of your instructions to read-only access, or you can bar access to the internal logic or local parameters used by the instructions. This source protection lets you prevent unwanted changes to your instructions and protects your intellectual property.

Monitoring Controller Status

```
┌────────GSV────────┐
│ Get System Value  │
│ Class name      ? │
│ Instance name   ? │
│ Attribute Name  ? │
│ Dest            ? │
│                ?? │
└───────────────────┘

   ┌────────SSV────────┐
   │ Set System Value  │
   │ Class name      ? │
   │ Instance name   ? │
   │ Attribute Name  ? │
   │ Source          ? │
   │                ?? │
   └───────────────────┘
```

The CompactLogix controller uses Get System Value (GSV) and Set System Value (SSV) instructions to get and set (change) controller data. The controller stores system data in objects. There is no status file, as in the PLC-5 processor.

The GSV instruction retrieves the specified information and places it in the destination. The SSV instruction sets the specified attribute with data from the source.

When you enter a GSV/SSV instruction, the programming software displays the valid object classes, object names, and attribute names for each instruction. For the GSV instruction, you can get values for all available attributes. For the SSV instruction, the software only displays attributes you are permitted to set.

Some object types appear repeatedly, so you may have to specify the object name. For example, there can be several tasks in your application. Each task has its own TASK object that you access by the task name.

You can access these object types:

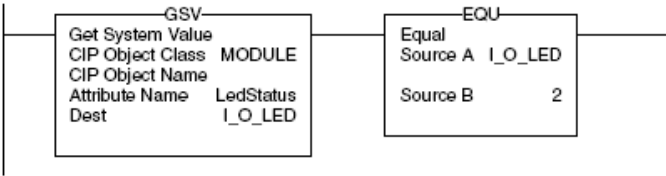|	
• AXIS	• MODULE
• CONTROLLER	• MOTIONGROUP
• CONTROLLERDEVICE	• PROGRAM
• CST	• ROUTINE
• DF1	• SERIALPORT
• FAULTLOG	• TASK
• MESSAGE	• WALLCLOCKTIME

Monitoring Connections

If communication with a device in the I/O configuration of the controller does not occur for 100 ms or 4 times the RPI, whichever is less, the communication times out and the controller produces these warnings.

- The I/O status indicator on the front of the controller flashes green.

- A ⚠ shows over the I/O configuration folder and the timed-out device.

- A module fault code is produced, which you can access via:
  - the Module Properties dialog.
  - a GSV instruction.

Determining if Communication has Timed Out with Any Device

If communication times out with at least one device (module) in the I/O configuration of the controller, the I/O status indicator on the front of the controller flashes green.

- The GSV instruction gets the status of the I/O status indicator and stores it in the I_O_LED tag
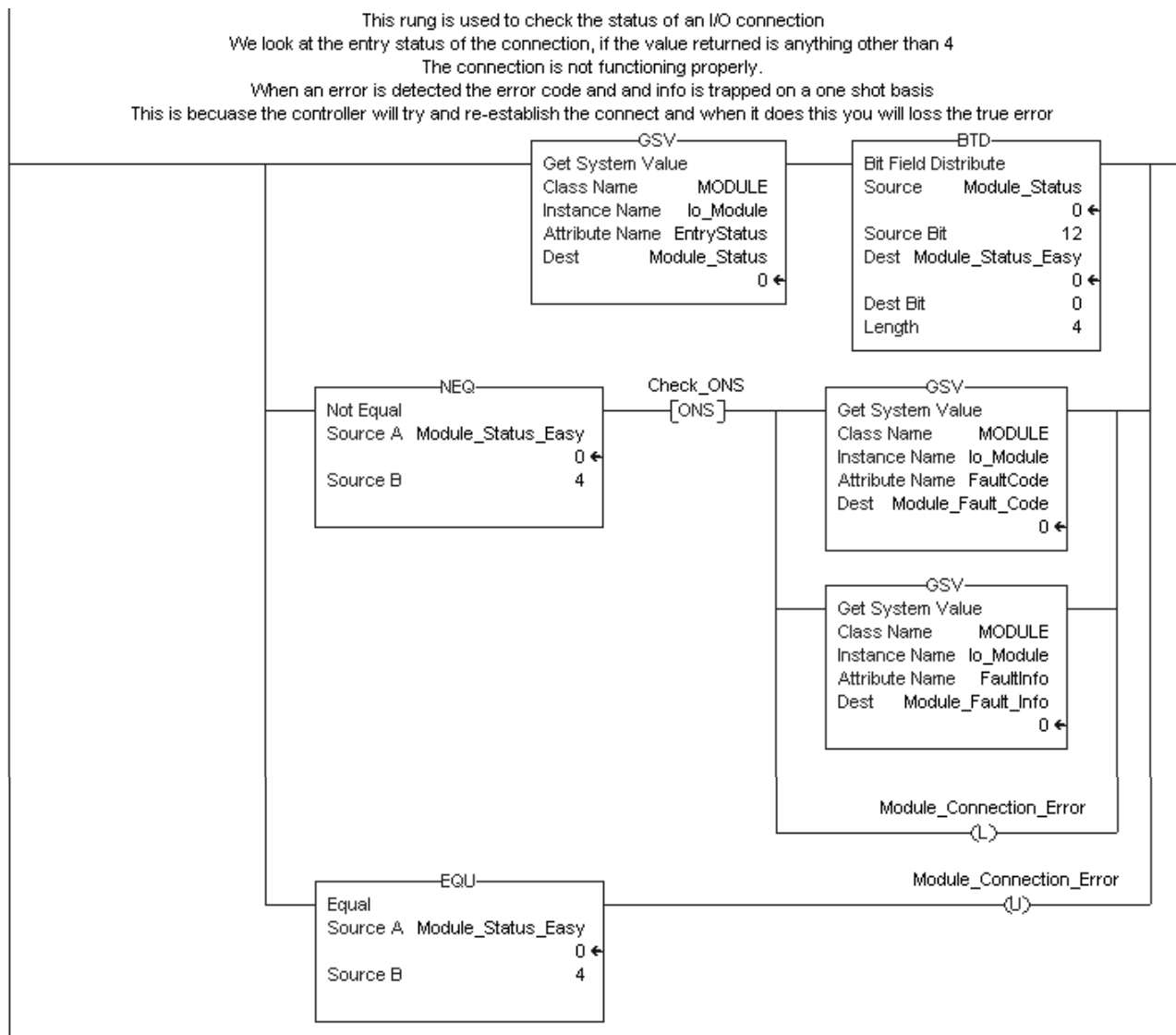- If the I_O_LED tag equals 2, the controller has lost communication with at least one device

```
          ┌──────GSV──────────────┐      ┌────EQU────────────┐
          │ Get System Value       │      │ Equal              │
──────────┤ CIP Object Class  MODULE│──────┤ Source A  I_O_LED  ├──────────
          │ CIP Object Name        │      │                    │
          │ Attribute Name  LedStatus│     │ Source B        2  │
          │ Dest            I_O_LED │      └────────────────────┘
          └────────────────────────┘
```

where:

I_O_LED is a DINT tag that stores the status of the I/O status indicator on the front of the controller.

Determining if Communication has Timed Out with a Specific I/O Module

If communication times out with a device (module) in the I/O configuration of the controller, the controller produces a fault code for the module.

- The GSV instruction gets the fault code for Io_Module and stores it in the Module_Status tag.
- If Module_Status is any value other than 4, the controller is not communicating with the module.

This rung is used to check the status of an I/O connection
We look at the entry status of the connection, if the value returned is anything other than 4
The connection is not functioning properly.
When an error is detected the error code and and info is trapped on a one shot basis
This is becuase the controller will try and re-establish the connect and when it does this you will loss the true error

```
                                    ─────GSV─────                          ─────BTD─────
                                    Get System Value                       Bit Field Distribute
                                    Class Name      MODULE                 Source      Module_Status
                                    Instance Name   Io_Module                                      0 ←
                                    Attribute Name  EntryStatus            Source Bit            12
                                    Dest          Module_Status            Dest  Module_Status_Easy
                                                              0 ←                                 0 ←
                                                                           Dest Bit               0
                                                                           Length                 4

              ─────NEQ─────          Check_ONS                  ─────GSV─────
              Not Equal              ─[ONS]─                    Get System Value
              Source A  Module_Status_Easy                     Class Name      MODULE
                                0 ←                             Instance Name  Io_Module
              Source B            4                             Attribute Name  FaultCode
                                                                Dest   Module_Fault_Code
                                                                                       0 ←

                                                                ─────GSV─────
                                                                Get System Value
                                                                Class Name      MODULE
                                                                Instance Name  Io_Module
                                                                Attribute Name   FaultInfo
                                                                Dest    Module_Fault_Info
                                                                                       0 ←

                                                                Module_Connection_Error
                                                                         ─(L)─

                                                                Module_Connection_Error
              ─────EQU─────                                              ─(U)─
              Equal
              Source A  Module_Status_Easy
                                0 ←
              Source B            4
```

## Select a System Overhead Percentage

With RSLogix 5000 programming software, you can specify a percentage for the system overhead, or time slice. The system overhead time slice specifies the percentage of controller time, excluding the time for periodic tasks, devoted to communication and background functions.

To select a system overhead percentage slice, perform this procedure.

1. In the Controller Organizer of RSLogix 5000 programming software, right-click the controller and choose Properties.

The Controller Properties dialog appears.
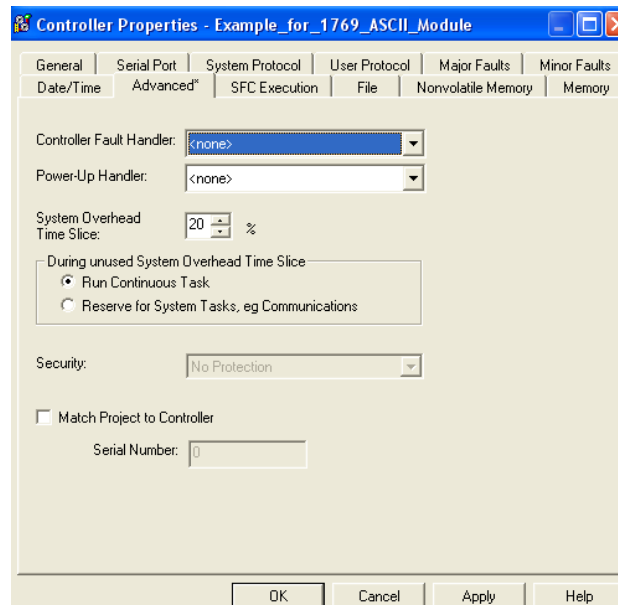
2. Click the Advanced tab.

   a. From the Controller Fault Handler pull-down menu, choose the program that will run as the result of a system fault.

   b. From the Power-Up Handler pull-down menu, choose the program the processor executes when it starts in Run mode after having been powered down in Run mode.

c. From the System Overhead Time Slice pull-down menu, choose the percentage of time your controller spends running its system task, relative to running its user tasks.

IMPORTANT   System overhead tasks include:

- communication with programming and HMI devices (such as RSLogix 5000 programming software).

- responding to messages.

- transmission of messages.

---

3. Click OK.

The controller performs system overhead functions for up to 1 ms at a time. If the controller completes the overhead functions in less than 1 ms, it resumes the continuous task.

As the system overhead percentage increases, time allocated to executing the continuous task decreases. If there is no communication for the controller to manage, the controller uses the communication time to execute the continuous task. While increasing the system overhead percentage does increase communication performance, it also increases the amount of time it takes to execute a continuous task, increasing overall scan time.

Ratio between the Continuous Task and System Overhead Functions

Time Slice	Continuous Task Length	Overhead Function Time, Max
10%	9 ms	1 ms
20%	4 ms	1 ms
33%	2 ms	1 ms
50%	1 ms	1 ms

At a time slice of 10%, system overhead interrupts the continuous task every 9 ms (of continuous task time).

Legend:

████████ Task executes.

▒▒▒▒▒▒▒▒ Task is interrupted (suspended).

Periodic

System Overhead   1 ms   1 ms

Continuous Task   9 ms   9 ms

Elapsed Time (ms)   0   5   10   15   20   25
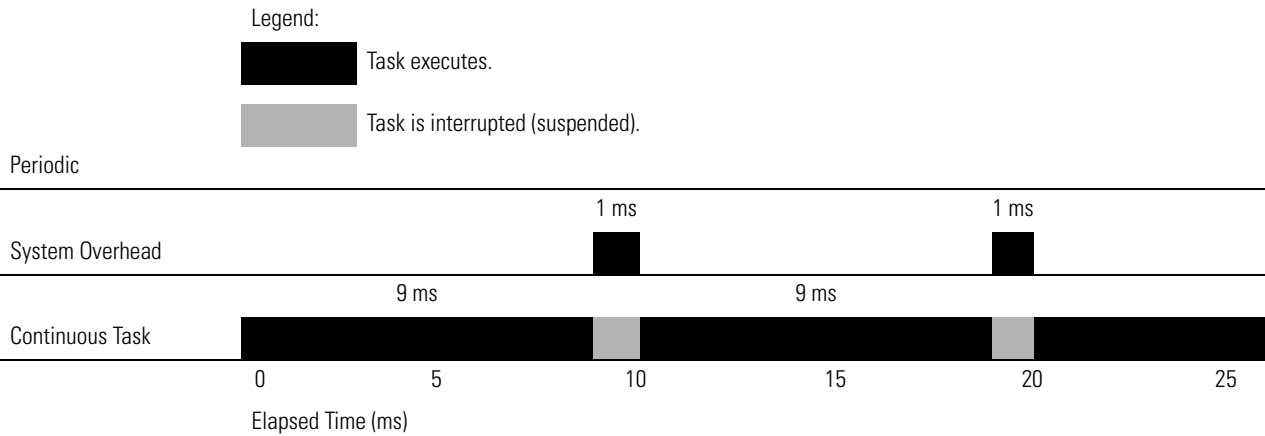
The interruption of a periodic task increases the elapsed time (clock time) between the execution of system overhead functions.

Periodic Task   1 ms   1 ms   1 ms   1 ms   1 ms

System Overhead   1 ms   1 ms

Continuous Task   9 ms of Continuous Task Time   9 ms of Continuous Task Time

Elapsed Time (ms)   0   5   10   15   20   25
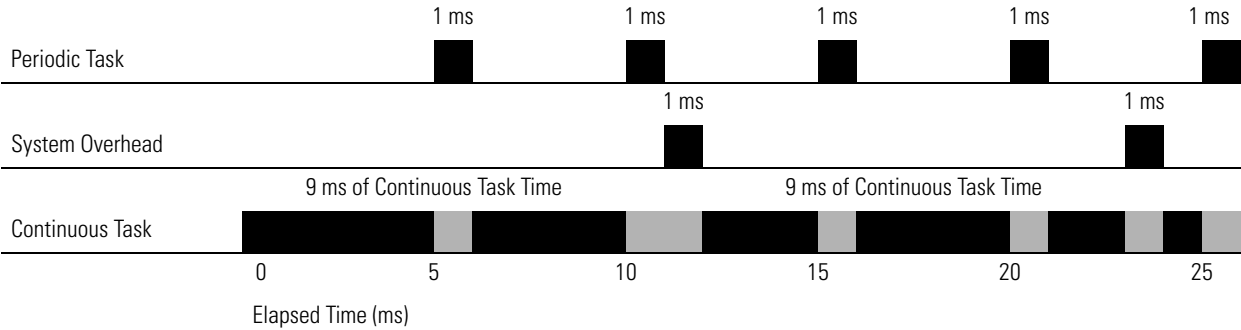
If you use the default time slice of 20%, the system overhead interrupts the continuous task every 4 ms.

System Overhead   1 ms   1 ms   1 ms   1 ms   1 ms

Continuous Task   4 ms   4 ms   4 ms   4 ms   4 ms

Elapsed Time (ms)   5   10   15   20   25
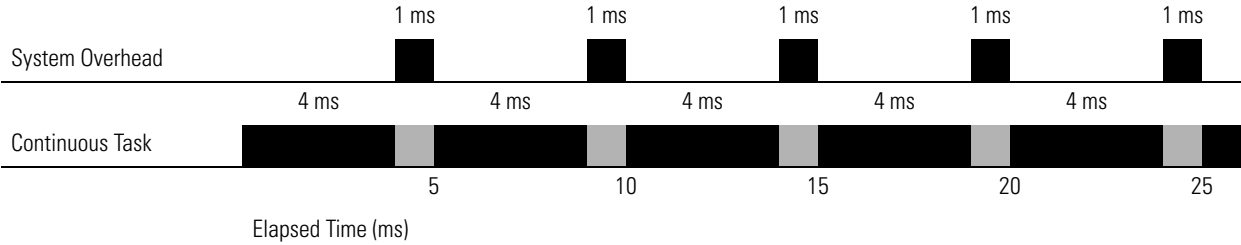
If you increase the time slice to 50%, the system overhead interrupts the continuous task every millisecond.

1 ms

System Overhead

1 ms

Continuous Task

5   10   15   20   25

Elapsed Time (ms)
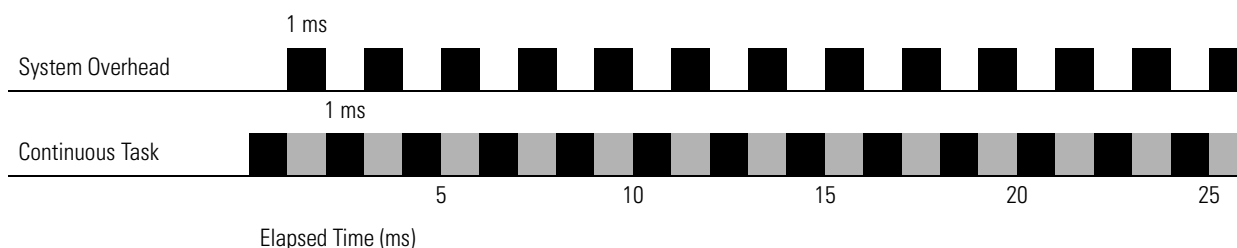
If the controller only contains periodic tasks, the system overhead time slice value has no effect. System overhead runs whenever a periodic task is not running.
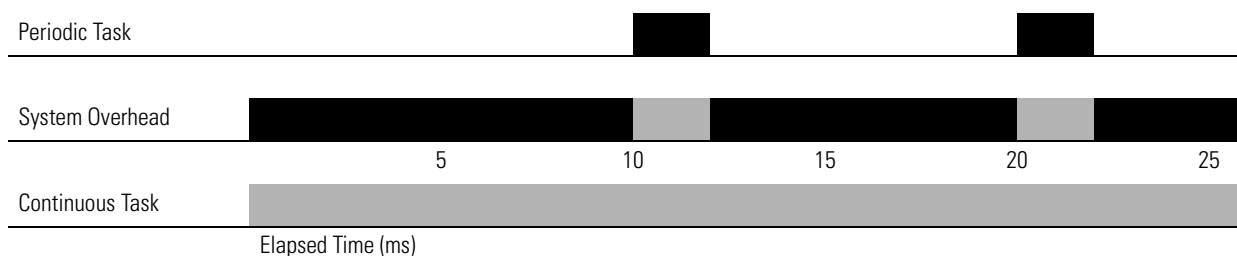
Periodic Task

System Overhead

5   10   15   20   25

Continuous Task

Elapsed Time (ms)

Additional Resources

These documents contain additional information pertinent to the development of applications.

Resource	Description
Logix5000 Controllers Common Procedures Programming Manual, publication 1756-PM001	Provides guidance on the development of controller projects.
Logix5000 Controllers Design Considerations Reference Manual, publication 1756-RM094	Provides information pertinent to the design of Logix5000 systems.
Logix5000 Controllers Execution Time and Memory Use Reference Manual, publication 1756-RM087	Details how to estimate the execution time and memory use of the controller's logic.
Logix5000 Controllers General Instructions Reference Manual, publication 1756-RM003	Details how to program the controller for sequential applications.

Notes:

Develop Motion Applications

Introduction

This chapter explains how to develop motion control applications.

Motion Performance

The CompactLogix L4x controllers utilize the 1768-M04SE SERCOS modules for motion control. Each 1768-M04SE modules support up to 4 axis. The following table describes motion capability.

1768 CompactLogix Controller Motion Support

Controller	SERCOS modules	Axes	Kinetix Drives	Motors Axes	Feedback Axes	Virtual Axes
1768-L43	2	4	4	4	2	6
1768-L45	4	8	8	8	4	6

If your solution requires more than eight Kinetix drives, consider the ControlLogix platform. For more information, refer to Additional Resources on page 104.

Make the Controller the Master Clock

You must make one module in the chassis the master clock for motion control. This module is called the coordinated system time (CST) master. The motion modules set their clocks to the CST master.

In most cases, make the controller the CST master.

To make the controller the master clock for a motion control application, perform this procedure.

1. In the Controller Organizer of RSLogix 5000 programming software, right-click your controller and choose Properties.
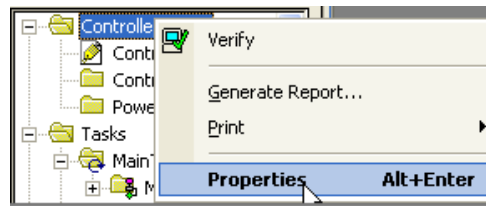
The Controller Properties dialog appears.
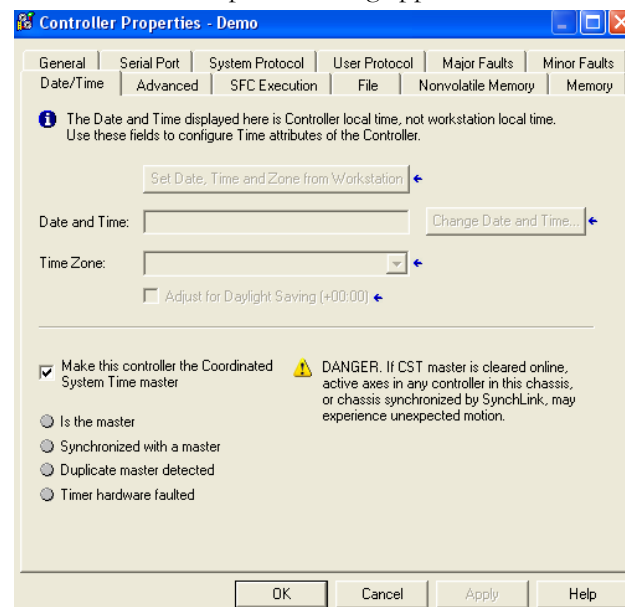
2. Click the Date/Time tab.

3. Click Make This Controller the Coordinated System Time Master.

4. Click OK.

Add the Motion Modules

For your motion modules, use the firmware revision that matches the firmware revision of your controller. See the release notes for your controller's firmware revision.

To add motion modules, perform this procedure.

1. In the I/O configuration tree in RSLogix 5000 programming software, right-click your module and choose New Module.
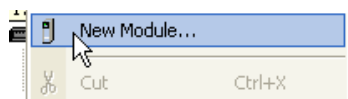
The Select Module dialog appears.
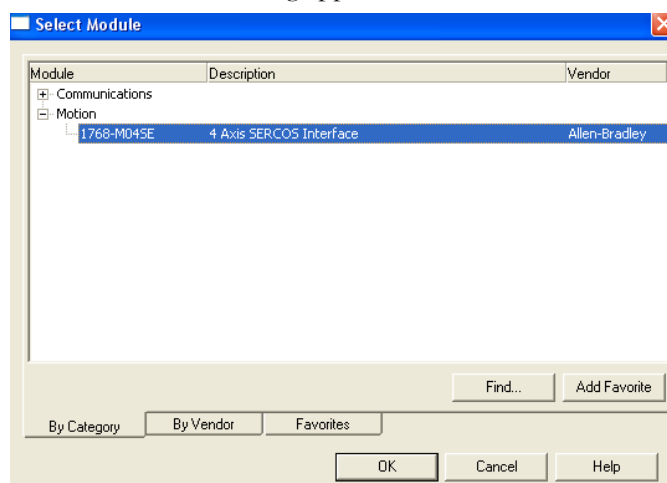
2. Double-click Motion.

3. Select the 1768-M04SE interface module.

4. Click OK.

The New Module dialog appears.
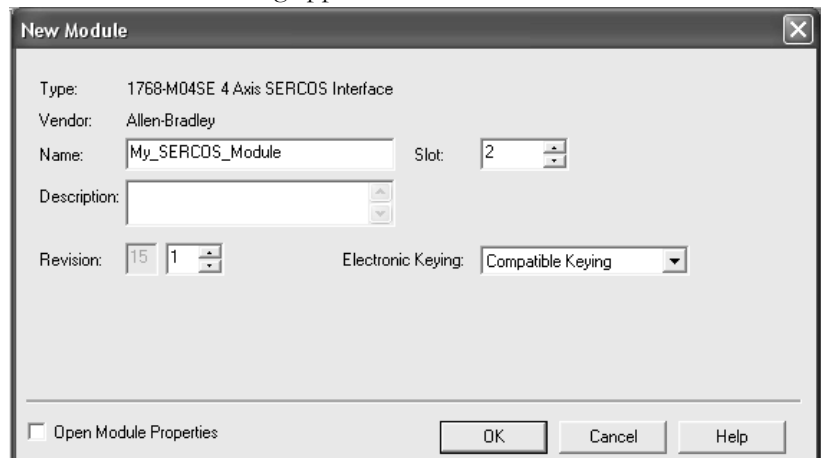
5. In the Name box, type the module name.

6. In the Slot box, type the slot the new module will occupy.

7. Leave Open Module Properties unchecked.

8. Click OK.

Add SERCOS Interface Modules

With the addition of SERCOS interface modules to a controller's I/O configuration, you can use RSLogix 5000 programming software to set up the modules.

To add SERCOS interface modules, perform this procedure.

1. In the I/O configuration tree in RSLogix 5000 programming software, right-click your module and choose New Module.
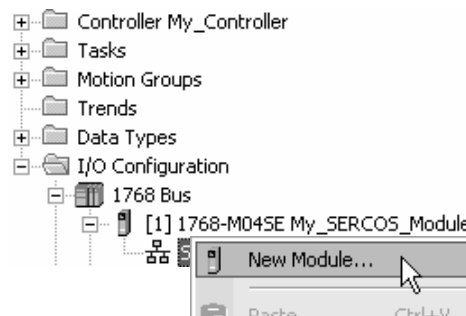
The Select Module dialog appears.
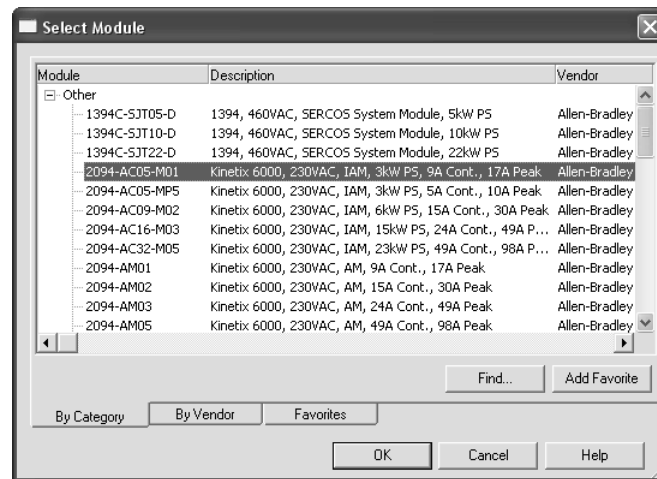
2. Double-click By Category.

3. Double-click Other.

4. Select a module.

5. Click OK.

The New Module dialog appears.
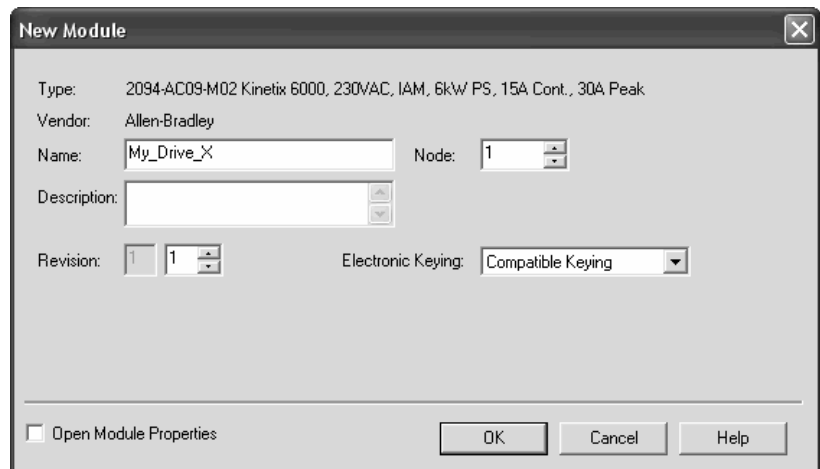
6. In the Name box, type a module name.

7. In the Node box, type the node number of the drive on the SERCOS ring.

8. Leave Open Module Properties unchecked.

9. Click OK.

Set Up Each SERCOS Interface Module

To set the data rate and cycle time for each SERCOS interface module in your project, perform this procedure.

1. In the I/O configuration tree in RSLogix 5000 programming software, right-click your module and choose Properties.
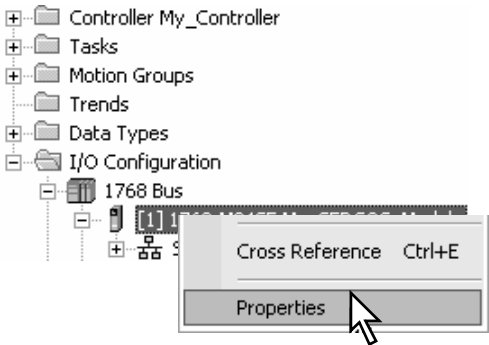
The Module Properties dialog appears.
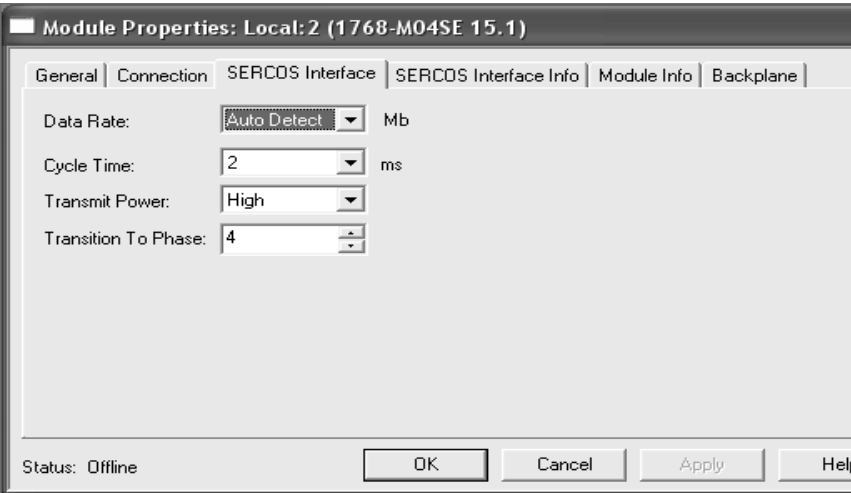
2. Click the SERCOS Interface tab.

3. From the Data Rate pull-down menu, choose Auto Detect.

4. From the Cycle Time pull-down menu, choose the cycle time.

Baud Rate of Drives	Number of Drives on Ring	Type of Drive	Cycle Time
4 MB	1...2	Kinetix 2000 or 6000	0.5 ms
	3...4	⟶	1 ms
8 MB	1…8	Kinetix 2000 or 6000	0.5 ms

5. Click OK.

Add the Motion Group

You can add a motion group to set up the motion planner.

Motion Group Overview

Motion Planner	Part of the controller that takes care of position and velocity information for your axes.
Coarse Update Period	How often the motion planner runs. When the motion planner runs, it interrupts all other tasks regardless of their priority.

Motion Planner

Scans of your code, system overhead, and so on.

0 ms   10 ms   20 ms   30 ms   40 ms

In this example, the coarse update period = 10 ms. Every 10 ms the controller stops scanning your code and whatever else it is doing and runs the motion planner.

IMPORTANT   Add only one motion group for the project. RSLogix 5000 programming software does not let you add more than one motion group.

To add a motion group to set up the motion planner, perform this procedure.

1. Choose your coarse update period.

IMPORTANT   The coarse update period is the interval between updating your axes' positions and scanning your code.
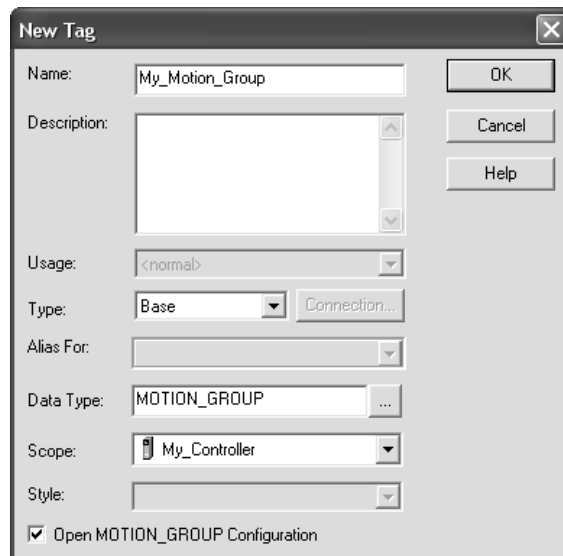
a. Set the coarse update period to 10 ms.

b. Leave at least half the controller's time for the scanning of all your code.

c. Set the coarse update period to a multiple of the motion module's cycle time.

Example: If the cycle time is 2 ms, set the coarse update period to 8 ms, 10 ms, 12 ms, and so on.

---

2. In RSLogix 5000 programming software, right-click Motion Groups and choose New Motion Group.
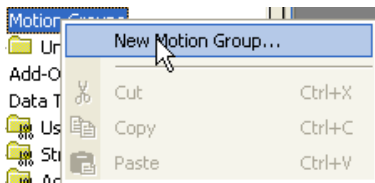
The New Tag dialog appears.
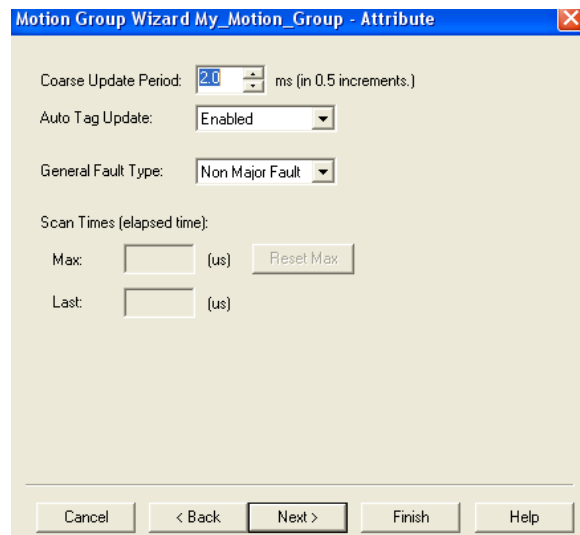
3. In the Name box, type the tag name.

4. From the Data Type pull-down menu, choose MOTION_GROUP.

5. Click OK.

The Motion Group Wizard dialog appears.

6. Click Next.

The Motion Group Wizard Attributes dialog appears.

7. In the Coarse Update Period box, type the interval at which you want to run your motion planner.
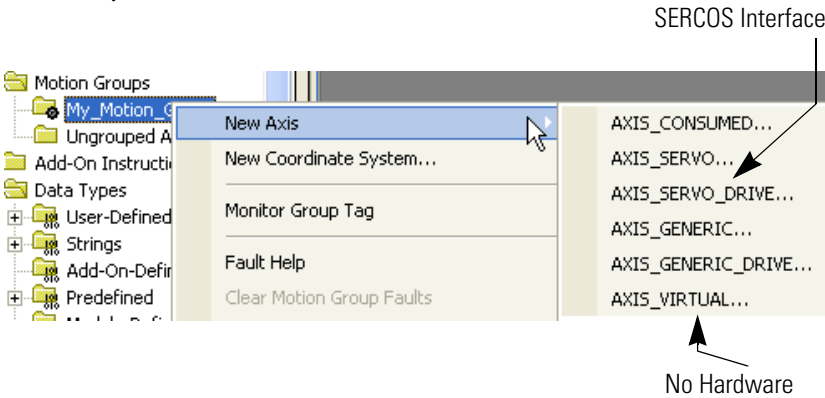
8. Click Finish.

Add Your Axes
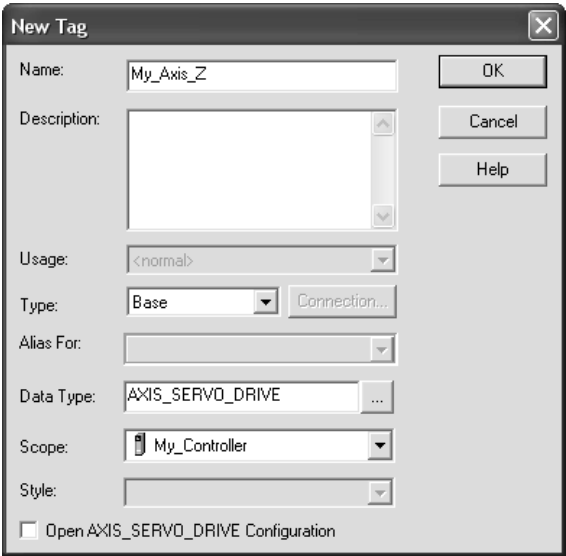
To add an axis for each of your drives, perform this procedure.

1. Decide which data type to use.

Motion Module	Data Type
1768-M04SE	AXIS_SERVO_DRIVE
No hardware	AXIS_VIRTUAL

2. In Motion Groups within RSLogix 5000 programming software, right-click My Motion Group and choose New Axis and then the type of axis you want to add.

The New Tag dialog appears.

3. In the Name box, type the name of your new axis.

4. Leave Open AXIS_SERVO_DRIVE Configuration unchecked.

5. Click OK.

Set Up Each Axis

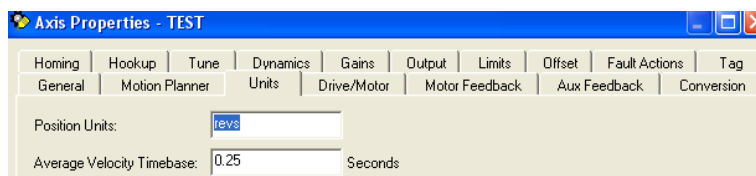To set up the axis of a SERCOS interface drive, perform this procedure.

1. In Motion Groups within RSLogix 5000 programming software, click Motion_Group, and then right-click an axis and choose Properties.
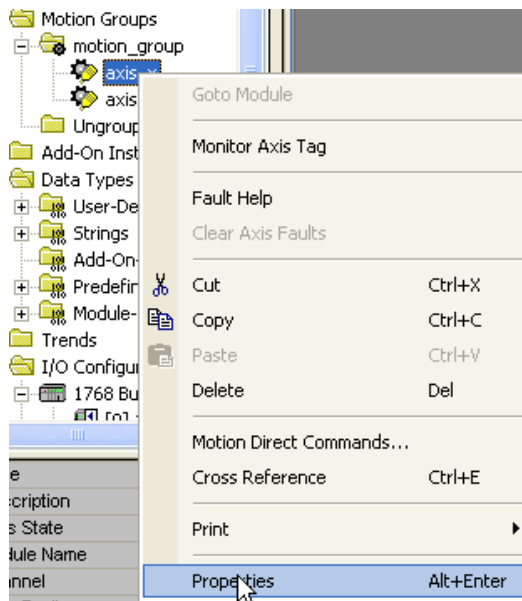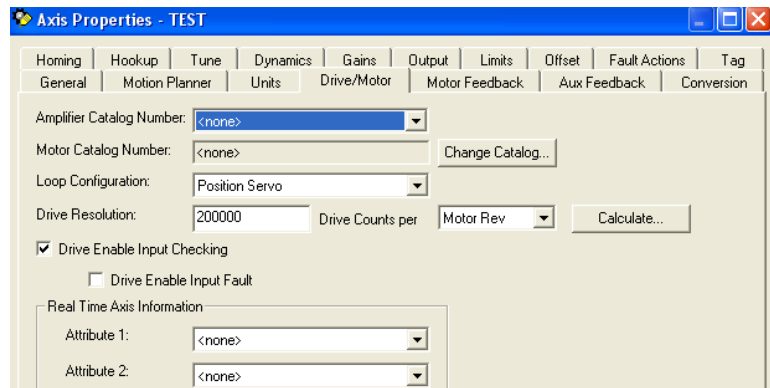
The Axis Properties Dialog appears.
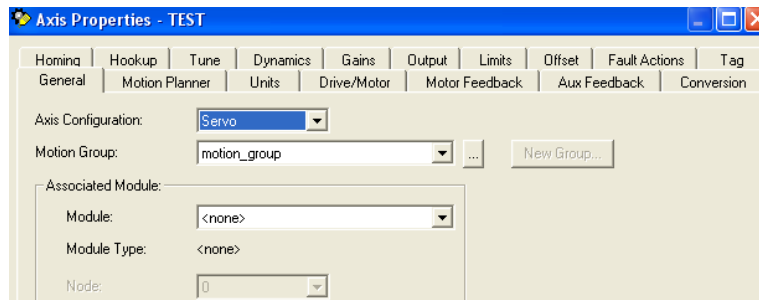
2. From the Module pull-down menu, choose the name that you gave to the drive for this axis.

3. Click the Units tab.

4. In the Position Units box, type the units of measure, such as revolutions (revs), degrees, inches, or millimeters.

5. Click the Drive/Motor tab.

a. From the Amplifier Catalog Number pull-down menu, choose the catalog number of your drive.

b. Click Change Catalog to select your motor's catalog number.

> **IMPORTANT** The default, and only, setting for Loop Configuration is Position Servo.

c. In the Drive Resolution box, type the number of feedback counts.

You can collect data as the motor turns:

- a revolution.
- a millimeter.
- an inch.

d. From the Drive Counts per pull-down menu, choose one of these feedback count options:
- Motor Rev - the default setting
- Motor Inch
- Motor Millimeter

6. Click the Conversion tab.
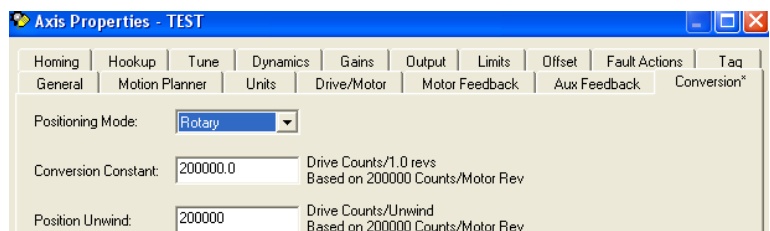
a. From the Positioning Mode pull-down menu, choose between the axis as either rotary or linear.

IMPORTANT	For an axis of the data type AXIS_CONSUMED, you cannot edit this parameter because it is set in and taken from a producing axis in a Logix processor connected to a network.

- Linear - provides a maximum total linear travel of 1 billion feedback counts. This mode disables the unwind feature. You can limit the axis' linear travel distance by specifying its positive and negative travel limits.

- Rotary - enables the axis' rotary unwind feature. This feature provides infinite position range by unwinding the axis position whenever the axis moves through a complete unwind distance. The number of encoder counts per unwind of the axis is specified by the Position Unwind parameter.

---

b. In the Conversion Constant box, specify the number of drive feedback counts per rotary or linear revolution.

c. For a rotary axis, in the Position Unwind box, type the number of feedback counts needed to automatically unwind the axis.

7. Click the Homing tab.

a. From the Sequence pull-down menu, choose one of these types of homing sequences:

- Immediate

- Switch

- Marker

- Switch-Marker

b. Within Active Home Sequence Group, in the Speed and Return Speed boxes, type the homing speeds and click OK.

Check the Wiring of Each Drive

Use these tests to check a drive's wiring.

Wiring Tests

| Test	Function	Requirement
Test marker	Verify that the encoder A, B, and Z channels are connected and phased properly for marker detection.	You must manually move the axis for this test.
Test feedback	Verify the polarity of the feedback.	You must manually move the axis for this test.
Test command and feedback	Verify the polarity of the drive.	N/A

ATTENTION

These tests move the axis even with the controller in Remote Program mode.

- Before you do the tests, make sure no one is in the path of the axis.

- Do not change the polarity after you do the tests, or you may cause an axis-runaway condition.

To check the wiring of each drive, perform this procedure.

1. In Motion Groups within RSLogix 5000 programming software, click Motion_Group, and then right-click an axis and choose Properties.
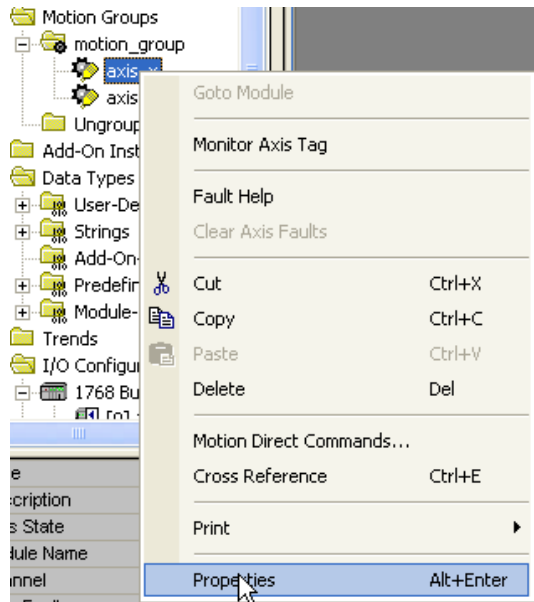
The Axis Properties Dialog appears.
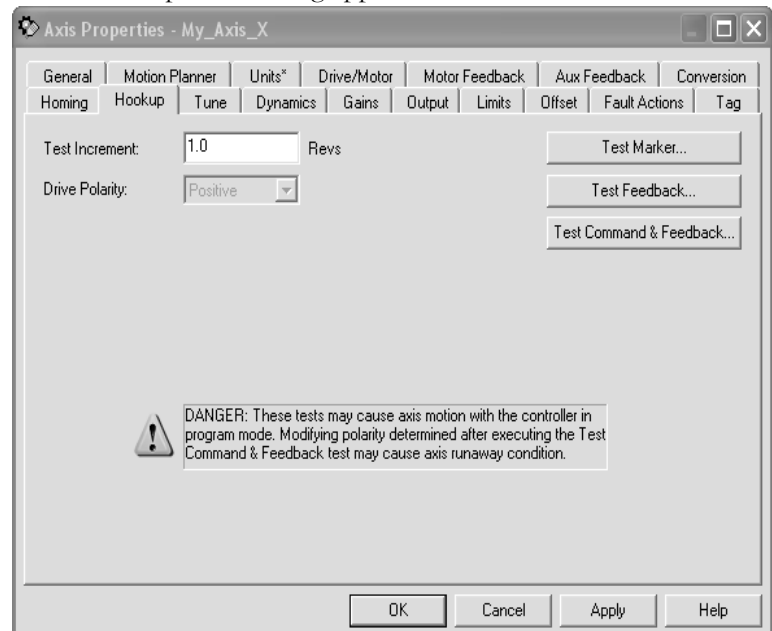
2. Click the Hookup tab.

   a. In the Test Increment box, specify how many revolutions you want the axis to turn during each test.

   b. Click Test Marker to check the channels for proper connection and phasing.

   c. Click Test Feedback to test the polarity of the feedback.

   d. Click Test Command & Feedback to test the drive's polarity.

3. Click OK.

Tune Each Axis

You need to tune each axis.

ATTENTION	When you tune an axis, it moves even with the controller in Remote Program mode. In that mode, your code is not in control of the axis.
⚠	Before you tune an axis, make sure no one is in its path.

To tune each axis, perform this procedure.

1. In Motion Groups within RSLogix 5000 programming software, click Motion_Group, and then right-click an axis and choose Properties.
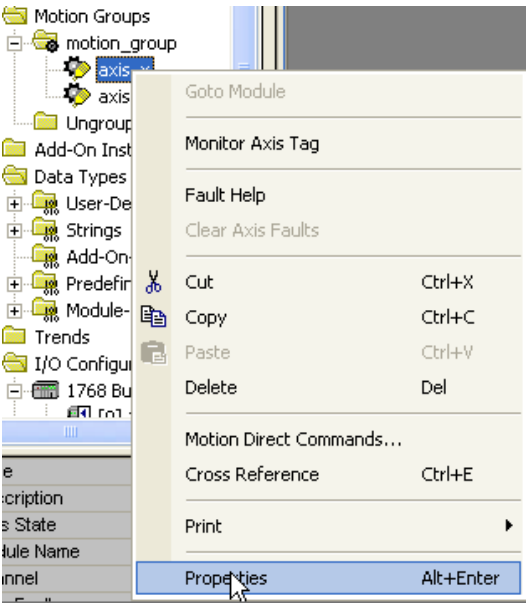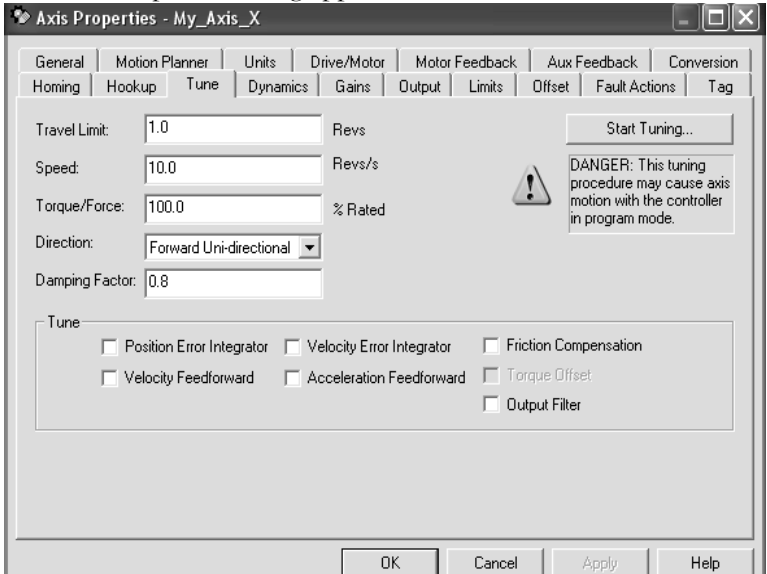

The Axis Properties dialog appears.

2. Click the Tune tab.

    a. In the Travel Limit box, type the number of revolutions to which you want to limit your axis during its tuning.

    b. In the Speed box, type the number of revolutions per second to which you want to limit your axis during its tuning.

    c. Click Start Tuning.

3. Click OK.

Obtain Axis Information

You can obtain axis information through any one of several methods.

To obtain axis information, take any one, or all, of these steps.

Obtaining Axis Information

Use the Axis Properties dialog to configure the axis.

Monitor the state and faults of an axis.

Use a Get System Value (GSV) instruction or Set System Value (SSV) instruction to read or change the configuration at run-time.

Monitor the axis tag for status and faults.
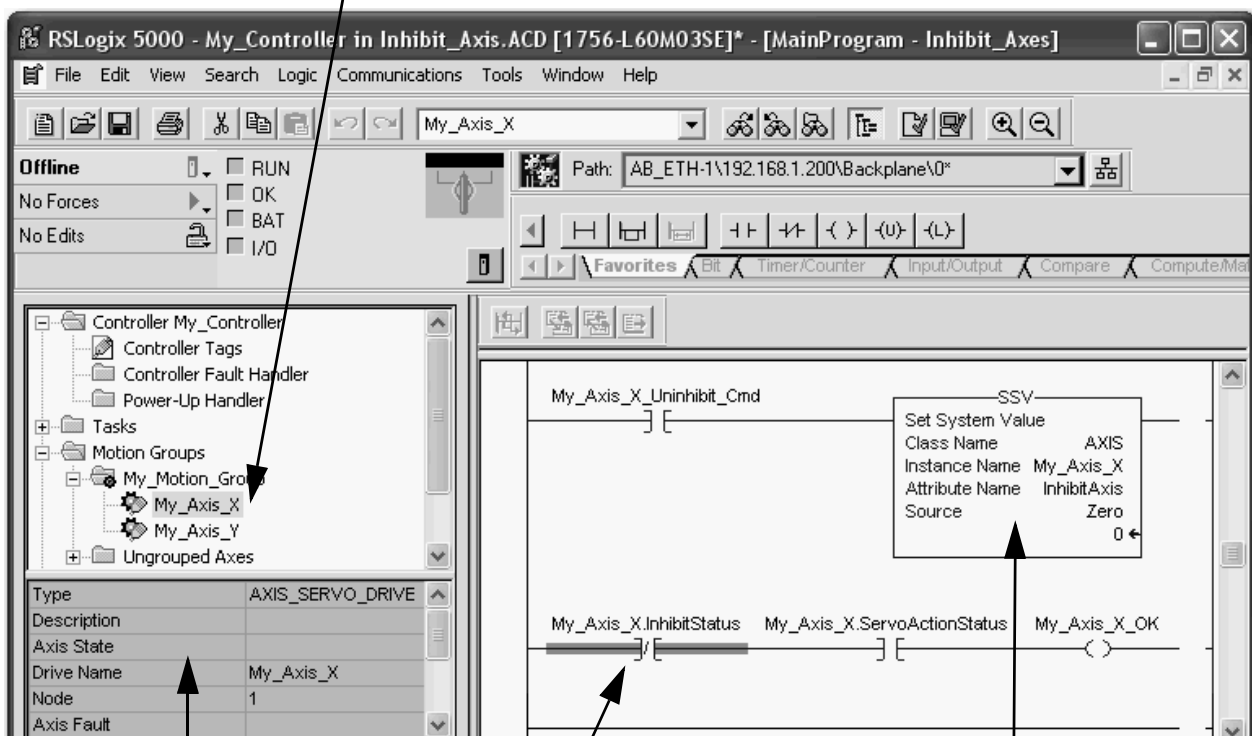
Programming Motion Control

The controller gives you a set of motion control instructions for your axes and:

- uses these instructions just like the rest of the Logix5000 instructions.

  You can program motion control in these programming languages:

  – Ladder diagram (LD)
  – Structured text (ST)
  – Sequential function chart (SFC)

- each motion instruction works on one or more axes.
- each motion instruction needs a motion control tag. The tag uses a MOTION_INSTRUCTION data type and stores the instruction's information status.

Motion Control Instruction

Motion Control Tag

	MSO	
	Motion Servo On	(EN)
	Axis ? ... ◄	(DN)
	Motion control ?	(ER)
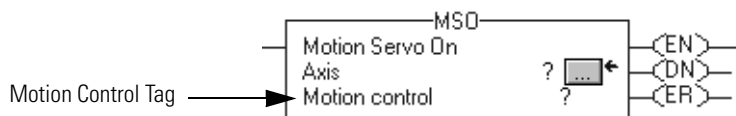
ATTENTION	Use the tag for the motion control operand of motion instruction only once. Reuse of the same motion control tag in other instructions may cause unintended operation of the control variables.

Example

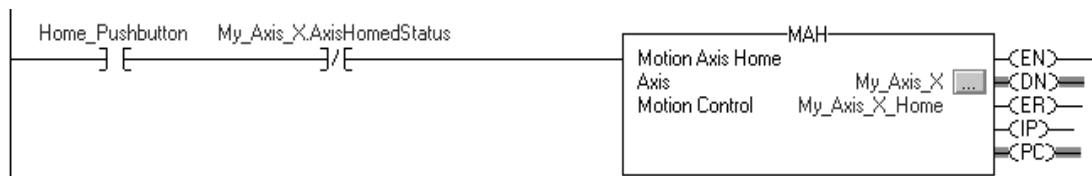Here is an example of a simple ladder diagram that homes, jogs, and moves an axis.

If Initialize_Pushbutton = on and the axis = off (My_Axis_X.ServoActionStatus = off) then
    The MSO instruction turns on the axis.

Initialize_Pushbutton	My_Axis_X.ServoActionStatus	MSO	
] [	]/[	Motion Servo On	(EN)
		Axis My_Axis_X ...	(DN)
		Motion Control My_Axis_X_On	(ER)

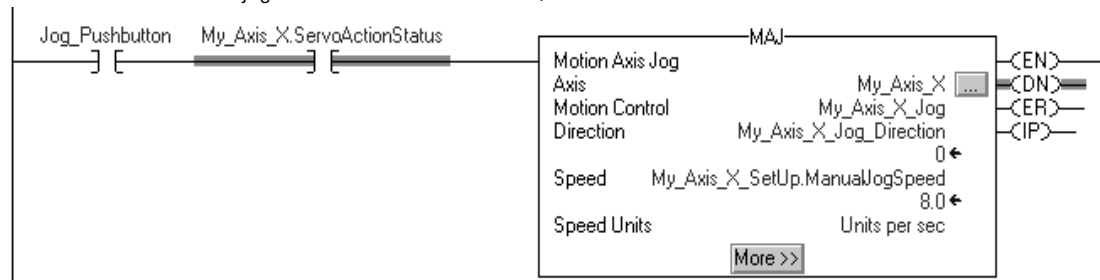If Home_Pushbutton = on and the axis hasn't been homed (My_Axis_X.AxisHomedStatus = off) then
    The MAH instruction homes the axis.

Home_Pushbutton	My_Axis_X.AxisHomedStatus	MAH	
] [	]/[	Motion Axis Home	(EN)
		Axis My_Axis_X ...	(DN)
		Motion Control My_Axis_X_Home	(ER)
			(IP)
			(PC)

If Jog_Pushbutton = on and the axis = on (My_Axis_X.ServoActionStatus = on) then

    The MAJ instruction jogs the axis forward at 8 units/second.

```
Jog_Pushbutton  My_Axis_X.ServoActionStatus                 ┌─────────MAJ─────────┐
     ] [ ─────────────] [──────────────────────────────────│Motion Axis Jog         │──(EN)
                                                            │Axis          My_Axis_X │  (DN)
                                                            │Motion Control My_Axis_X_Jog│(ER)
                                                            │Direction  My_Axis_X_Jog_Direction│(IP)
                                                            │                    0 ← │
                                                            │Speed  My_Axis_X_SetUp.ManualJogSpeed│
                                                            │                  8.0 ← │
                                                            │Speed Units    Units per sec│
                                                            │      [ More >> ]       │
                                                            └────────────────────────┘
```

If Jog_Pushbutton = off then

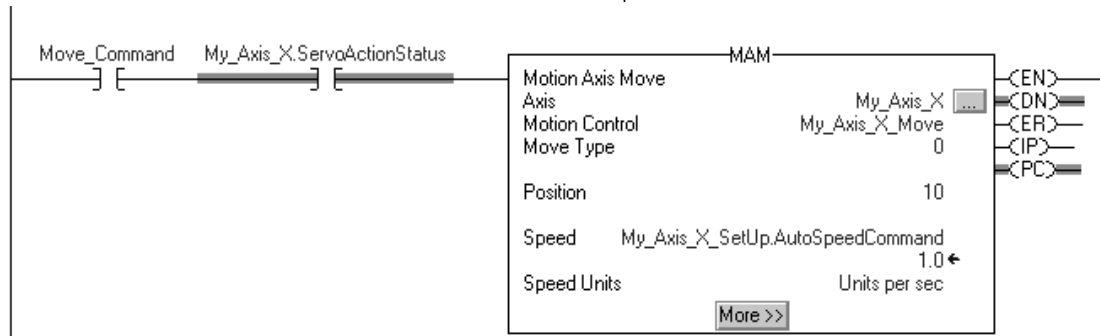   The MAS instruction stops the axis at 100 units/second^2
    Make sure that Change Decel is Yes. Otherwise, the axis decelerates at its maximum speed.

```
Jog_Pushbutton                                              ┌─────────MAS─────────┐
     ]/[ ──────────────────────────────────────────────────│Motion Axis Stop        │──(EN)
                                                            │Axis          My_Axis_X │  (DN)
                                                            │Motion Control My_Axis_X_MAS│(ER)
                                                            │Stop Type          Jog  │  (IP)
                                                            │Change Decel       Yes  │  (PC)
                                                            │Decel Rate My_Axis_X_SetUp.ManualJogDecel│
                                                            │                100.0 ← │
                                                            │Decel Units   Units per sec2│
                                                            │      [ << Less ]       │
                                                            └────────────────────────┘
```
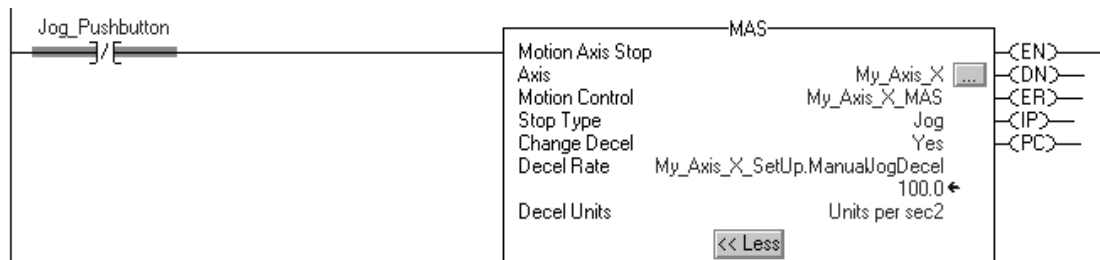
If Move_Command = on and the axis = on (My_Axis_X.ServoActionStatus = on) then

    The MAM instruction moves the axis. The axis moves to the position of 10 units at 1 unit/second.

```
Move_Command  My_Axis_X.ServoActionStatus                   ┌─────────MAM─────────┐
     ] [ ────────────] [────────────────────────────────────│Motion Axis Move       │──(EN)
                                                            │Axis          My_Axis_X │  (DN)
                                                            │Motion Control My_Axis_X_Move│(ER)
                                                            │Move Type          0    │  (IP)
                                                            │                        │  (PC)
                                                            │Position          10    │
                                                            │                        │
                                                            │Speed  My_Axis_X_SetUp.AutoSpeedCommand│
                                                            │                  1.0 ← │
                                                            │Speed Units    Units per sec│
                                                            │      [ More >> ]       │
                                                            └────────────────────────┘
```

Additional Resources

These documents contain information pertinent to the development of motion applications.

Resource	Description
Motion Modules in Logix5000 Control Systems User Manual, publication LOGIX-UM002	Explains how to set up and program motion control modules in Logix5000 control systems.
Logix5000 Controllers Common Procedures Programming Manual, publication 1756-PM001	Provides guidance on the development of controller projects.
Logix5000 Controllers Motion Instructions Reference Manual, publication 1756-RM007	Details how to program Logix5000 controllers for motion applications.
Logix5000 Controllers General Instructions Reference Manual, publication 1756-RM003	Details how to program Logix5000 controllers for sequential applications.

Configure PhaseManager

Introduction

This chapter explains how to configure PhaseManager.

Within RSLogix 5000 programming software, PhaseManager provides a state model for your equipment.

Topic	Page
PhaseManager Overview	105
State Model Overview	107
Comparing PhaseManager to Other State Models	109
Minimum System Requirements	110
Equipment Phase Instructions	110
Additional Resources	110

PhaseManager Overview

PhaseManager lets you add equipment phases to your controller. An equipment phase helps you lay out your code in sections that are easier to write, find, follow, and change.

PhaseManager Terms

Term	Description
Equipment phase	• As with a program, an equipment phase is run in a task and is given a set of routines and tags. • Unlike a program, an equipment phase runs by a state model and lets you do one activity.
State model	• A state model divides the operating cycle of your equipment into a series of states. Each state is an instant in the operation of the equipment, the actions, or conditions of the equipment at a given time. • The state model of an equipment phase resembles that of the S88 and Pack*ML* state models.
State machine	An equipment phase includes an embedded state machine that: • calls the main routine (state routine) for an acting state. • manages the transitions between states with minimal coding. • makes sure that the equipment goes from state to state along an allowable path.
PHASE tag	When you add an equipment phase, RSLogix 5000 programming software makes a tag for the equipment phase. The tag uses the PHASE data type.

PhaseManager Overview

A PHASE tag gives you the status of an equipment phase.

Name	△	Data Type
− Add_Water		PHASE
+ Add_Water.State		DINT
Add_Water.Running		BOOL
Add_Water.Holding		BOOL
Add_Water.Restarting		BOOL

Controller
└ Controller Tags

Tasks
└ MainTask

An equipment phase directs one activity of your equipment.

A state model divides the activity into a series of states.

Running State Routine

How to add water

Add Water Phase

Mix Phase

Drain Phase

Space Parts Phase

MainProgram

Equipment phase instructions control the transitions between states and handle faults.

PSC	POVR	PCLF	PRNP	PATT
PCMD	PFL	PXRQ	PPD	PDET

My Equipment Program

Other code controls the specific actions of your equipment.

Water Feed   Conveyor   Enable Axes

State Model Overview

A state model defines what your equipment does under different conditions, such as run, hold, and stop.

PhaseManager States

State	Description
Acting	Does something or several things for a certain time or until certain conditions are met. An acting state runs once or repeatedly.
Waiting	Shows that certain conditions are met and the equipment is waiting for the signal to go to the next state.

PhaseManager State Transitions

Your equipment can go from any state in the box to the stopping or aborting state.

Acting states represent the things your equipment does at a given time.

Waiting states represent the condition of your equipment when it is between acting states.
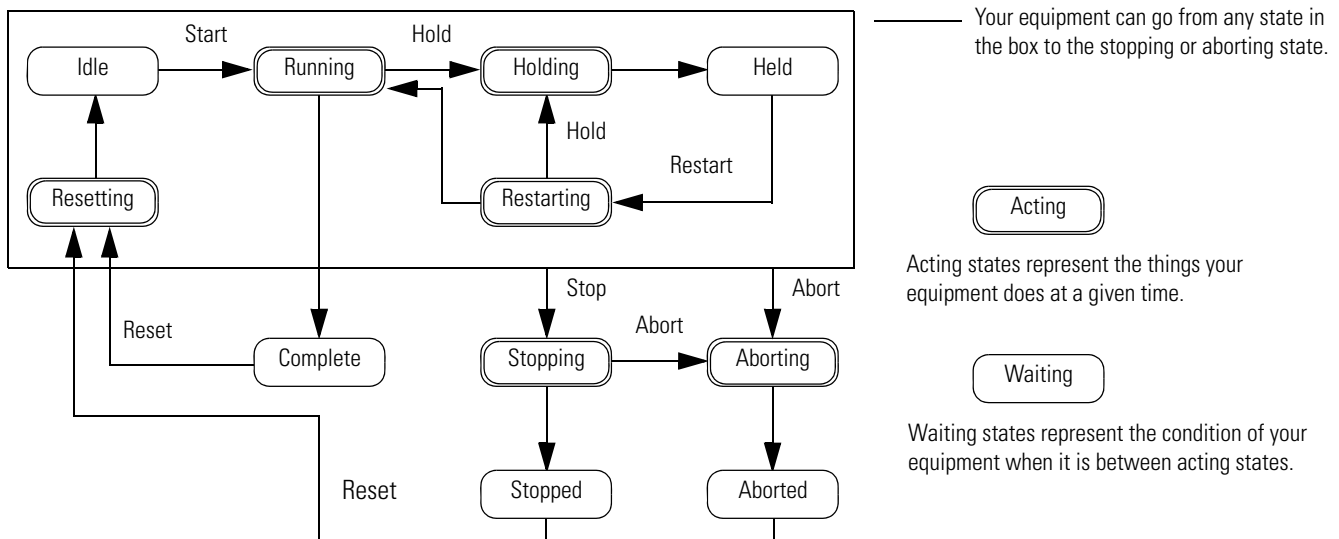
With a state model, you define the behavior of your equipment.

PhaseManager State Models

State	Question to be asked
Stopped	What happens when you turn on power?
Resetting	How does the equipment get ready to run?
Idle	How do you tell that the equipment is ready to run?
Running	What does the equipment do to make product?
Holding	How does the equipment temporarily stop making product without making scrap?
Held	How do you tell if the equipment is safely holding?
Restarting	How does the equipment resume production after holding?
Complete	How do you tell when the equipment is done with what it had to do?
Stopping	What happens during a normal shutdown?
Aborting	How does the equipment shut down if a fault or failure occurs?
Aborted	How do you tell if the equipment is safely shut down?

How Equipment Changes States

The state model's arrows show the states through which your equipment progresses.

- Each arrow is called a transition.
- A state model lets the equipment make only certain transitions. This restriction standardizes the equipment's behavior so that other equipment using the same model will behave the same way.

PhaseManager State Model
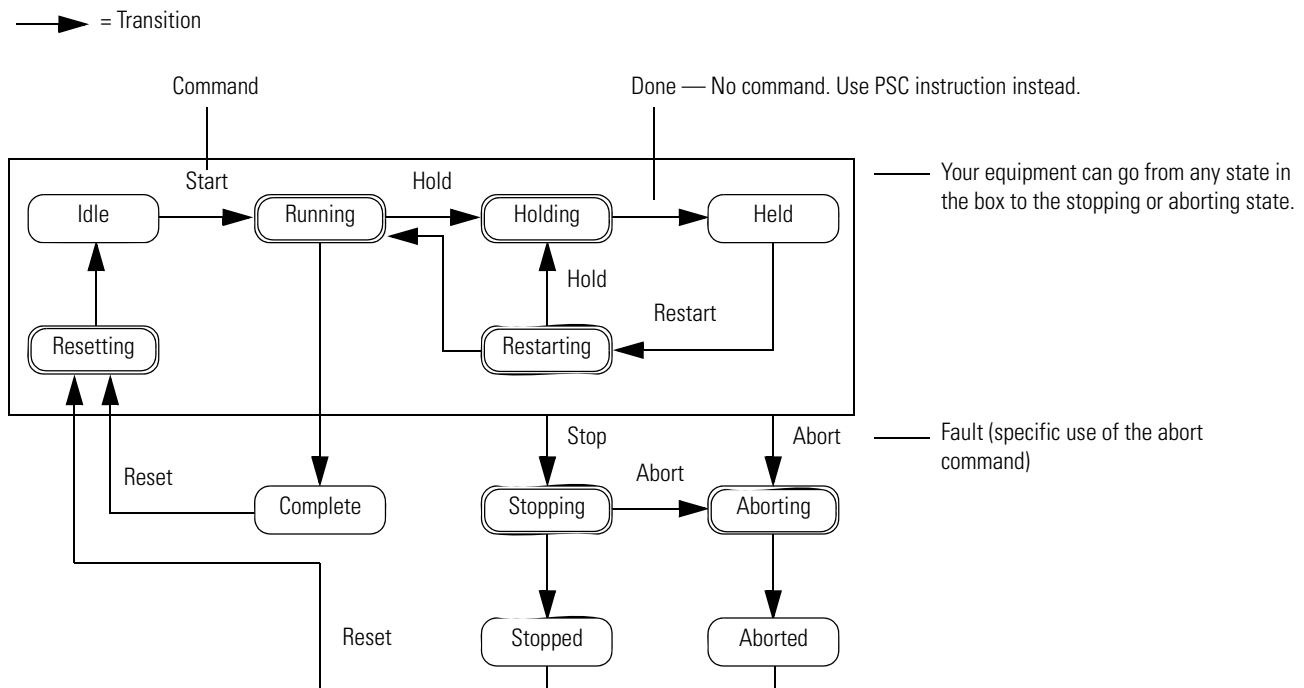
PhaseManager Transition Commands

Type of Transition	Description
Command	A command tells the equipment to do something. For example, the operator pushes the start button to start production and the stop button to halt production. PhaseManager uses these commands:
	Reset Stop Restart
	Start Hold Abort
Done	Equipment goes to a waiting state when it is finished with what it is doing. You do not give the equipment a command. Instead, you set up your code to signal when the equipment is finished.
Fault	A fault tells you that something out of the ordinary has happened. You set up your code to look for faults and take action if it finds any. If you want to shut down your equipment as quickly as possible when it detects a fault, set up your code to look for that fault and give the abort command if it finds it.

Manually Change States

With RSLogix 5000 programming software, you can manually change an equipment phase.

To manually change a PhaseManager state, perform this procedure.

Current Equipment Phase

1. Take ownership of the equipment phase.

2. Give a command.
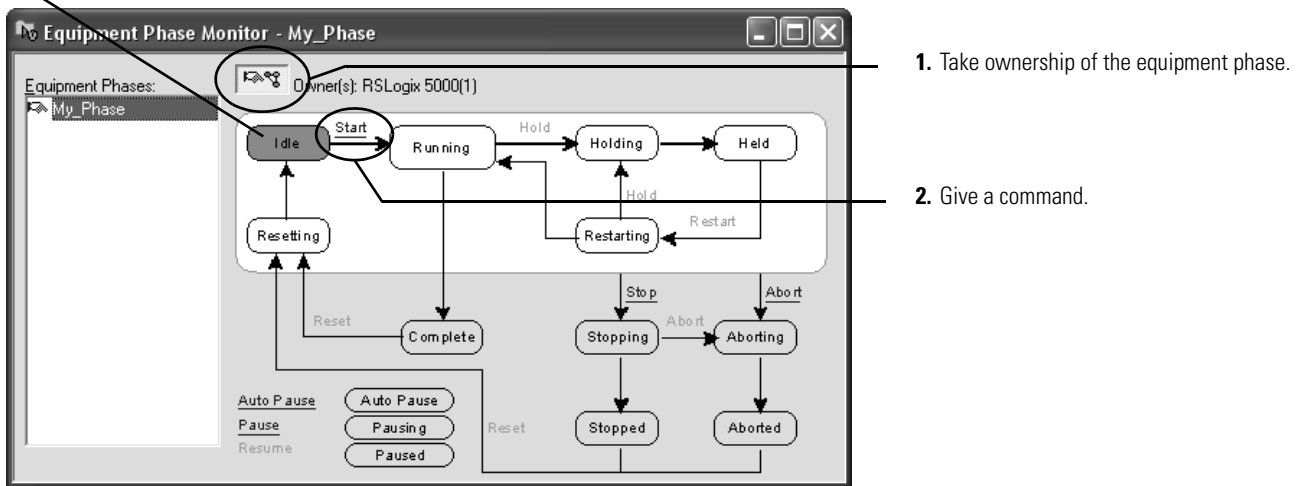
Comparing PhaseManager to Other State Models

You can compare PhaseManager state models to other state models.

State Model Comparisons

S88	Pack*ML*	PhaseManager
Idle	Starting $\Rightarrow$ Ready	Resetting $\Rightarrow$ Idle
Running $\Rightarrow$ Complete	Producing	Running $\Rightarrow$ Complete
Pausing $\Rightarrow$ Paused	Standby	Subroutines or breakpoints
Holding $\Rightarrow$ Held	Holding $\Rightarrow$ Held	Holding $\Rightarrow$ Held
Restarting	None	Restarting
Stopping $\Rightarrow$ Stopped	Stopping $\Rightarrow$ Stopped	Stopping $\Rightarrow$ Stopped
Aborting $\Rightarrow$ Aborted	Aborting $\Rightarrow$ Aborted	Aborting $\Rightarrow$ Aborted

Minimum System Requirements

To develop PhaseManager programs, you need:

- a CompactLogix controller, firmware revision 15.0 or later.
- a communication path to the controller.
- RSLogix 5000 software, version 15.0 or later.

    The major revision of the controller and software must match.

To enable PhaseManager support, you need the full or professional editions of RSLogix 5000 programming software or RSLogix 5000 with PhaseManager software (9324-RLDPMENE).

Equipment Phase Instructions

The controller supports several equipment-phase ladder diagram (LD) and structured text (ST) instructions.

PhaseManager Instructions

Instruction	Instruction Function
PSC	Signal a phase that the state routine is complete and to proceed to the next state.
PCMD	Change the state or substate of a phase.
PFL	Signal a failure for a phase.
PCLF	Clear the failure code of a phase.
PXRQ	Initiate communication with RSBizWare Batch software.
PRNP	Clear the NewInputParameters bit of a phase.
PPD	Set up breakpoints within the logic of a phase.
PATT	Take ownership of a phase to either: • prevent another program or RSBizWare Batch software from commanding a phase. • make sure another program or RSBizWare Batch software does not already own a phase.
PDET	Relinquish ownership of a phase.
POVR	Override a command.

Additional Resources

For more information about how to configure PhaseManager, refer to The PhaseManager User Manual, publication LOGIX-UM001.

Maintain Nonvolatile Memory

Introduction

This chapter explains how to maintain nonvolatile memory.

CompactLogix controllers support the 1784-CF64 CompactFlash card for nonvolatile memory. Nonvolatile memory lets you keep a copy of your project on the controller. The controller does not need power to keep this copy.

Topic	Page
No Battery is Required	111
Preventing a Major Fault During a Load	112
Using a CompactFlash Reader	112
Additional Resources	112

You can load the copy from nonvolatile memory to the controller's user memory:

- each time you apply power.
- whenever there is no project in the controller and you apply power.
- at any time via RSLogix 5000 programming software.

IMPORTANT	Nonvolatile memory stores the contents of the user memory when you store the project.
	• Changes made after storing a project are not reflected in nonvolatile memory.
	• If you change the project but do not store those changes, you overwrite them when you load the project from nonvolatile memory. If this occurs, you have to upload or download the project to go online.
	If you want to store changes such as online edits, tag values, or a ControlNet network schedule, store the project again after you make the changes.

No Battery is Required

The 1768 controller does not require a battery. When it is being shut down, the controller uses internal flash memory to store its program. Energy stored in the 1768 power supply maintains controller power long enough to store the program to internal flash memory, not the external CompactFlash card.

Preventing a Major Fault During a Load

If the major and minor revisions of the project in nonvolatile memory do not match the major and minor revisions of the controller, a major fault may occur during a load.

Project Load and Controller

Controller	Consequences
Does not use a CompactFlash card	Make sure that the major and minor revisions of the project in nonvolatile memory match the major and minor revisions of the controller. The nonvolatile memory of the controller stores only the project. It does not store the firmware for the controller.
Uses a CompactFlash card	The CompactFlash card stores the firmware for projects ≥ 12.0. Depending on the current revision of the controller, you may be able to use the CompactFlash card to update the firmware of the controller and load the project.

> **ATTENTION**
>
> ⚠️
>
> Do not remove the CompactFlash card while the controller is reading from or writing to the card, as indicated by a flashing green CF status indicator. This could corrupt the data on the card or in the controller, as well as corrupt the latest firmware in the controller.

Using a CompactFlash Reader

All CompactLogix controllers support the FAT16 file system used with the CompactFlash card.

Typically, you do not have to manage the files on a CompactFlash card. The card automatically loads the project that you most recently stored. For additional flexibility, the file system also lets you:

- manually change which project loads from the CompactFlash card.
- manually change a project's load parameters.

Additional Resources

For more information about the maintenance of nonvolatile memory, refer to The Logix5000 Controllers Common Procedures Programming Manual, publication 1756-PM001.

1768 CompactLogix Controller Specifications

Introduction

This appendix contains the specifications for the 1768-L43 and 1768-L45 CompactLogix controllers.

1768-L43 and L45 CompactLogix Controllers General Specifications

Controller	User Memory	Nonvolatile Memory	Communication Options	Number of Tasks Supported	Number of Local I/O Modules Supported
1768-L43	2 MB	CompactFlash	• Maximum of two 1768 modules in the 1768 backplane • EtherNet/IP network (1768-ENBT and 1768-EWEB modules) • ControlNet network (1768-CNB and 1768-CNBR modules) • DeviceNet network (1769-SDN scanner) • Serial (built-in) • Modbus via ladder routine • DH-485	• 16 tasks (only 1 continuous) • Event tasks: supports consumed tag trigger, EVENT instruction, axis, and motion event triggers	• Max of 16 1769 I/O modules (in 3 banks) in the 1769 backplane
1768-L45	3 MB	CompactFlash	• Maximum of four 1768 modules in the 1768 backplane • EtherNet/IP network (1768-ENBT and 1768-EWEB modules) • ControlNet network (1768-CNB and 1768-CNBR modules) • DeviceNet network (1769-SDN scanner) • Serial (built-in) • Modbus via ladder routine • DH-485	• 30 tasks (only 1 continuous) • Event tasks: supports consumed tag trigger, EVENT instruction, axis, and motion event triggers	Max of 30 1769 I/O modules (in 3 banks) in the 1769 backplane

1768-L43 and 1768-L45 CompactLogix Controller General Specifications

Attribute	Value
Backplane current at 24 V	1.3 A
Backplane current output	1769 Backplane: 2.0 A @ 5.2V 1.0 A @ 24V (sourced by 1768 power supply) 1768 Backplane: 2.8 A @ 5.2V
Power consumption	31.3 W
Power dissipation	6.3 W

Power Dissipated (Watts) vs 1768 and 1769 Bus 5.2V Load (Watts)
2.9 W, 6.3 W
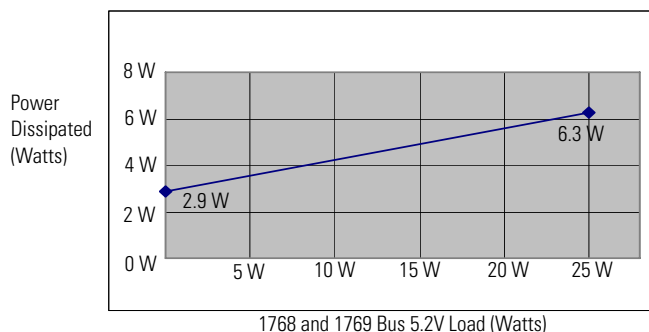
Isolation (continuous-voltage rating)	30V, functional insulation type, tested @ 710V dc for 60 s, RS-232 to system
Wiring category[1]	2 - on communication ports
North American temperature code	T4
Mounting screw torque	1.16 Nm (10 lb-in), using M4 or #8 screws
Dimensions (HxWxD), approx.	131.6 x 67.4 x 121.8 mm (5.18 x 2.65 x 4.80 in.)
Weight, approx.	0.34 kg (11.9 oz)

[1] Use this Conductor Category information for planning conductor routing. Refer to Industrial Automation Wiring and Grounding Guidelines, publication 1770-4.1.

1768-L43 and 1768-L45 CompactLogix Controller Environmental Specifications

Attribute	Value	
Temperature, operating	IEC 60068-2-1: IEC 60068-2-2: IEC 60068-2-14:	(Test Ad, Operating Cold), (Test Bd, Operating Dry Heat), (Test Nb, Operating Thermal Shock): 0...60 °C (32...140 °F)
Temperature, nonoperating	IEC 60068-2-1: IEC 60068-2-2: IEC 60068-2-14:	(Test Ab, Unpackaged Nonoperating Cold), (Test Bb, Unpackaged Nonoperating Dry Heat), (Test Na, Unpackaged Nonoperating Thermal Shock): -40...85 °C (-40...185 °F)
Relative humidity	IEC 60068-2-30:	(Test Db, Unpackaged Nonoperating Damp Heat): 5...95% noncondensing
Vibration	IEC 60068-2-6:	(Test Fc, Operating): 5 g @ 10...500 Hz
Shock, operating	IEC 60068-2-27:	(Test Ea, Unpackaged Shock): 30 g
Shock, nonoperating	IEC 60068-2-27:	(Test Ea, Unpackaged Shock): 50 g
Emissions	CISPR 11: Group 1, Class A	
ESD immunity	IEC 61000-4-2:	6 kV contact discharges 8 kV air discharges

1768-L43 and 1768-L45 CompactLogix Controller Environmental Specifications

Attribute	Value	
Radiated RF immunity	IEC 61000-4-3:	10V/M with 1 kHz sine-wave 80%AM from 80…2000 MHz 10V/M with 200 Hz 50% Pulse 100%AM @ 900 MHz 10V/M with 200 Hz 50% Pulse 100%AM @ 1890 MHz 1V/M with 1 kHz sine-wave 80%AM from 2000…2700 MHz
EFT/B immunity	IEC61000-4-4:	±4 kV @ 5 kHz on communication ports
Surge transient immunity	IEC 61000-4-5:	±2 kV line-earth (CM) on communication ports
Conducted RF immunity	IEC 61000-4-6:	10V rms with 1 kHz sine-wave 80%AM from 150 kHz…80 MHz
Enclosure type rating	None (open-style)	

Notes:

Status Indicators

Introduction
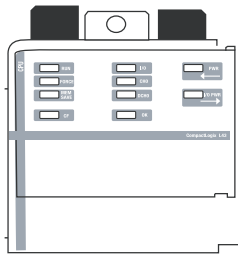
This appendix explains how to interpret the status indicators on your CompactLogix controllers.

Topic	Page
CompactLogix Controller Status Indicators	117
RS-232 Serial Port Status Indicators	122
Faceplate Pushbutton	122

CompactLogix Controller Status Indicators

The following table describes the CompactLogix controller status indicators The table also provides controller behavior interpretations and offers you recommended actions.

CompactLogix Controller Status Indicators

Indicator	Condition	Interpretation	Recommended Actions
RUN	Off	The controller is in Program or Test mode.	Normal Operation - No action is required.
	Green	The controller is in Run mode.	
FORCE	Off	• No tags contain I/O force values. • I/O forces are inactive (disabled).	
	Steady amber	• I/O forces are active (enabled). • I/O force values may or may not exist.	
	Flashing amber	One or more input or output addresses have been forced to an On or Off state, but the forces have not been enabled.	Enable forces, or remove the individual I/O from being forced.

CompactLogix Controller Status Indicators

Indicator	Condition	Interpretation	Recommended Actions
MEM SAVE	Off	The user program and configuration data are not actively being saved to flash memory.	Normal Operation - No action is required.
	Green	The user program and configuration data are being saved to flash memory.	
I/O	Off	• There are no devices in the I/O configuration of the controller. • The controller does not contain a project (controller memory is empty).	
	Steady green	The controller is communicating with all of the devices in its I/O configuration.	
I/O	Flashing green	One or more devices in the controller's I/O configuration are not responding.	To determine why a device may not be responding, perform this procedure. 1. Verify that all I/O modules in your project are installed in the same order. 2. Verify that all devices have been updated to the latest major and minor firmware revisions. 3. Use RSLogix 5000 programming software Online help to determine which I/O module is not responding.
	Flashing red	The controller is not communicating with any of the devices in its I/O configuration.	
OK	Off	• No power is applied. • If MEM SAVE is green, the user program and configuration data are being saved to flash memory.	Normal Operation - No action is required.
	Flashing red	The controller requires a firmware update.	1. Download the latest firmware from http://support.rockwellautomation.com. 2. To install the latest firmware, either • use the ControlFlash utility. • use AutoFlash. • use your CompactFlash card. For additional information, see the CompactLogix Controllers Installation Guide, publication 1768-IN004.
		A major recoverable fault occurred on the controller.	To clear the fault, perform this procedure. 1. Turn the controller keyswitch from PROG to RUN to PROG. 2. Go online with RSLogix 5000 programming software.

CompactLogix Controller Status Indicators

Indicator	Condition	Interpretation	Recommended Actions
OK		• A nonrecoverable major fault occurred on the controller and produces either Fault Code 60 or Fault Code 61. • Fault Code 60 indicates that the CompactFlash card is not installed. • Fault Code 61 means that the CompactFlash card is installed. To recover from this fault, perform this procedure.	To recover from Fault Code 60 or 61, perform this procedure. 1. Clear the fault. 2. Download the project. 3. Change to Remote Run/Run mode. 4. If the problem persists: a. before you cycle power to the controller, record the state of the OK and RS232 status indicators. b. contact Rockwell Automation. See the back cover of this publication.
	Red	The controller detected a nonrecoverable major fault, so it cleared the project from memory.	To recover from this fault, perform this procedure. 1. Cycle power to the chassis. 2. Download the project. 3. Change to Run mode. If the OK Status indicator remains red, contact Rockwell Automation.
	Green	The controller is operating normally.	Normal Operation - No action is required.
	Flashing green	The controller is storing or loading a project to or from nonvolatile memory.	
PWR	Off	The power supply is turned off or lacks adequate input power.	Verify that the power supply is turned on and that adequate input power is properly connected.
		The power supply has failed.	Replace the power supply.
	Steady green	The power supply is operating properly.	Normal Operation - No action is required.

CompactLogix Controller Status Indicators

Indicator	Condition	Interpretation	Recommended Actions
PWR	Steady red	The power supply cannot produce valid 24V power to the 1768 modules.	To supply 24V power, perform this procedure. 1. Disconnect all modules from the system. 2. Reapply power. 3. Check the PWR status indicator. a. If the status indicator remains red, replace the power supply. b. If the status indicator is green, one of the other modules in the system is causing the red indicator. Move to the next step. 4. Reinstall any 1768 motion or communication modules. 5. Reapply power. a. If the status indicator is green, either the 1768 controller or one of the 1769 I/O modules is causing the red indicator. b. If the status indicator remains red, one of the 1768 communication or motion modules is causing the red indicator. Move to the next step. 6. Disconnect the 1768 communication or motion modules from the system one at a time. 7. After each module is removed from the system, reapply power to the power supply and check the PWR status indicator. a. If the status indicator is green, the most recently removed 1768 module caused the red indicator and should be replaced. b. If the status indicator remains red, continue disconnecting 1768 modules one at a time until the PWR status indicator turns green.

CompactLogix Controller Status Indicators

Indicator	Condition	Interpretation	Recommended Actions
I/O PWR	Off	Either the controller or the power supply is not operating properly.	To recover from this fault, perform this procedure. 1. Make sure all modules in the system are installed properly and fully engaged with each other. a. If the controller PWR status indicator remains off, proceed to the next step. 2. Disconnect any 1768 communication or motion modules from the system. 3. Reinstall the controller directly next to the power supply and reapply power. 4. If the controller PWR status indicator remains off, replace the controller. 5. If the controller PWR status indicator on the new controller remains off, replace the power supply.
	Steady green	The controller is sending power to the 1768 modules as it should.	Normal Operation - No action is required.

	Steady red	• The controller needs to be replaced.
		• If 1768 communication and motion modules are installed in the system, one of the 1768 modules needs to be replaced.
			Perform this procedure.
			
			1. Disconnect all of the 1768 communication and motion modules from the system.
			
			2. Reapply power.
			
			3. Check the controller PWR status indicator.
			
			   a. If the status indicator is green, the controller is operating properly and one of the other 1768 modules needs to be replaced.
			
			   b. To troubleshoot the 1768 modules, see their respective installation instructions.
			
			   c. If the status indicator remains red, replace the controller.

CompactFlash Card Status Indicator

ATTENTION	Do not remove the CompactFlash card while the controller is reading from or writing to the card, as indicated by a flashing green CF status. This could corrupt the data on the card or in the controller, as well as corrupt the latest firmware in the controller.

CompactFlash Status Indicators

Indicator	Condition	Interpretation
CF	Off	No activity.
	Flashing green	The controller is reading from or writing to the CompactFlash card.
	Flashing red	CompactFlash card does not have a valid file system.

RS-232 Serial Port Status Indicators

The RS-232 serial port has two status indicators.

RS-232 Status Indicators

Indicator	Condition	Interpretation
DCH0	Off	Channel 0 is configured differently than the default serial configuration.
	Steady green	Channel 0 has the default serial configuration.
CH0	Off	There is no RS-232 activity.
	Flashing green	There is RS-232 activity. No action is required.

Faceplate Pushbutton
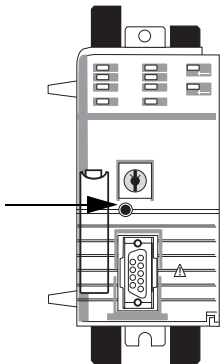
On the faceplate of the controller, there is a recessed pushbutton.

Pushbutton Actions

If you access the pushbutton	The action
After power is applied to the controller	Resets the RS-232 configuration setting to the defaults
While the controller is powering up	Clears the user program from controller memory

Symbols

69

Numerics

1768 CompactLogix controllers
overview 11
1768 module placement 51
1768 power supply 52
1768-CNB and 1768-CNBR ControlNet modules 28
1768-ENBT EtherNet/IP communication module 22
1768-EWEB Web server module 26
1768-L43 CompactLogix controller
1769 I/O module placement and support 53
environmental specifications 113
general specifications 113
overview 11
status indicators 117
1768-L45 CompactLogix controller
1769 I/O module placement and support 53
overview 11
status indicators 117
1769 module placement 53

A

add
axes 94
motion group 91
motion modules 87
SERCOS interface modules 88
additional resources 9
Add-On Instructions 75
address I/O data 61
applications
develop 69
ASCII characters
create and modify strings 39
data conversion 39
read and write 39
ASCII device 37
ASCII devices
communication 37
axes
add 94
set up 95
tune 100
axis information
obtain 101

C

cable
serial 16
cache
message connections 47
calculate
connection use 48
change
equipment phase 109
change of state 57
check
drive wiring 98
communicate
over networks 21
with DF1 devices 35
communication
communication 30
controllers 45
ControlNet network 27
determine timeout with any device 78
determine timeout with I/O module 79
DeviceNet network 30
DH-485 network 40
EtherNet/IP network 22
format 57
serial network 34
communication modules 11
CompactFlash
card Status indicators 122
reader 112
CompactLogix
56, 69
1768 controller specifications 113
1768 power supply 52
1768-L43 controller environmental specifications 113
1768-L43 controller general specifications 113
chassis 11
communication modules 11
CompactFlash card Status indicators 122
complex system 12
configure and monitor I/O 55
connection overview 45
connections example 49
controller communication 45
controller faceplate pushbutton 122
controller RS-232 serial port status indicators 122
controller Status indicators 117
ControlNet network 27, 29
design a system 13
develop motion applications 85

T

tag organization 74
tag types 46
tags
    organize 74
task execution 71
tasks
    define 71
    manage 69

tune
    axes 100

U

update 62
    data 62

How Are We Doing?

Your comments on our technical publications will help us serve you better in the future. Thank you for taking the time to provide us feedback.

You can complete this form and mail (or fax) it back to us or email us at RADocumentComments@ra.rockwell.com.

Pub. Title/Type   1768 CompactLogix Controllers

| Cat. No. | *1768-L43, 1768-L45* | Pub. No. | 1768-UM001C-EN-P | Pub. Date | November 2007 | Part No. | NA |

Please complete the sections below. Where applicable, rank the feature (1=needs improvement, 2=satisfactory, and 3=outstanding).

Overall Usefulness	1	2	3	How can we make this publication more useful for you?

Completeness (all necessary information is provided)	1	2	3	Can we add more information to help you?

procedure/step   illustration   feature

example   guideline   other

explanation   definition

Technical Accuracy (all provided information is correct)	1	2	3	Can we be more accurate?

text   illustration

Clarity (all provided information is easy to understand)	1	2	3	How can we make things clearer?

|	
Other Comments	You can add additional comments on the back of this form.

Your Name

Your Title/Function

Location/Phone

Would you like us to contact you regarding your comments?

___No, there is no need to contact me

___Yes, please call me

___Yes, please email me at _____

___Yes, please contact me via _____

Return this form to: Rockwell Automation Technical Communications, 1 Allen-Bradley Dr., Mayfield Hts., OH 44124-9705

Fax: 440-646-3525   Email: RADocumentComments@ra.rockwell.com

Other Comments

_____

_____

_____

_____

_____

_____

_____

_____

_____
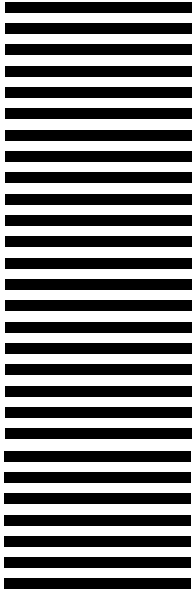
_____

_____

_____

PLEASE FOLD HERE
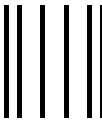
PLEASE REMOVE

BUSINESS REPLY MAIL
FIRST-CLASS MAIL PERMIT NO. 18235 CLEVELAND OH

POSTAGE WILL BE PAID BY THE ADDRESSEE

Rockwell
Automation

1 ALLEN-BRADLEY DR
MAYFIELD HEIGHTS OH 44124-9705

Rockwell Automation Support

Rockwell Automation provides technical information on the Web to assist you in using its products. At http://literature.rockwellautomation.com, 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://support.rockwellautomation.com.

Installation Assistance

If you experience a problem with a hardware module within the first 24 hours of installation, please review the information that's contained in this manual. You can also contact a special Customer Support number for initial help in getting your module up and running.

United States	1.440.646.3434 Monday – Friday, 8am – 5pm EST
Outside United States	Please contact your local Rockwell Automation representative for any technical support issues.

New Product Satisfaction Return

Rockwell 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, it may need to be returned.

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

www.rockwellautomation.com