Allen-Bradley

ControlNet™ Communications Module

User Manual

Cat. No. 1203-CN1
Firmware 2.xxx

Important User Information

Because of the variety of uses for the products described in this publication, those responsible for the application and use of this control equipment must satisfy themselves that all necessary steps have been taken to assure that each application and use meets all performance and safety requirements, including any applicable laws, regulations, codes and standards.

The illustrations, charts, sample programs and layout examples shown in this guide are intended solely for purposes of example. Since there are many variables and requirements associated with any particular installation, Rockwell Automation does not assume responsibility or liability (to include intellectual property liability) for actual use based upon the examples shown in this publication.

Rockwell Automation publication SGI-1.1, *Safety Guidelines for the Application, Installation, and Maintenance of Solid-State Control* (available from your local Rockwell Automation office), describes some important differences between solid-state equipment and electromechanical devices that should be taken into consideration when applying products such as those described in this publication.

Throughout this manual we use notes to make you aware of safety considerations:

ATTENTION: Identifies information about practices or circumstances that can lead to personal injury or death, property damage or economic loss.

Attention statements help you to:

- Identify a hazard.
- Avoid the hazard.
- Recognize the consequences.

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

Summary of Changes

Updated Information

The following changes to this manual have occurred since Publication 1203-5.13 – July 1998, P/N189939 (02).

Getting Started

Section: Establishing a Serial Connection with the 1203-CN1 Module
Paragraph added:

DriveExplorer (v1.01 or higher) software can now also be used on 1203-CN1s that are v2.001 or higher. Do not use DriveExplorer software with v1.xxx CN1s.

Appendix A

Power Consumption specification updated to 250mA at 24V DC (-20% / +30%).

Appendix B

Parameter 21 Serial Port Rate

Valid Value/Settings updated.
0 = 2400
1 = 4800
2 = 9600
3 = 19.2K
4 = 38.4K

Default setting updated.
2 = 9600

Important statement added to description.
Important: These settings are valid for v1.004 or higher CN1s. v1.003 or lower CN1s only have two settings: 0 = 9600 and 1 = 19.2K.

Appendix D

Section: Class Code 0x99 - SCANport Pass-Through Link Object Common Services:

Service Code updated.
0x0E
0x10

Notes:

Table of Contents

Troubleshooting **Chapter 7**

Specifications **Appendix A**

1203-CN1 Module Parameters **Appendix B**

N-File Structure **Appendix C**

ControlNet Objects **Appendix D**

Preface

Objectives

Read this preface to become familiar with the rest of the manual. This preface covers the following topics:

- Who should use this manual.
- An overview of the 1203-CN1 ControlNet™ communications module.
- The purpose of this manual.
- Terms and abbreviations.
- Conventions used in this manual.
- Rockwell Automation support.

Who Should Use This Manual?

Use this manual if you are responsible for installing, wiring, programming, or troubleshooting control systems that use the 1203-CN1 ControlNet communications module.

This manual is intended for qualified service personnel responsible for setting up and servicing the 1203-CN1 module. You must have previous experience with and a basic understanding of electrical terminology, programming procedures, required equipment, required software, networking, and safety precautions.

What Is the 1203-CN1 ControlNet Communications Module?

The 1203-CN1 ControlNet communications module provides an interface between a ControlNet network and a single SCANport™ product.

Purpose of this Manual

This manual is a learning and reference guide for the 1203-CN1 ControlNet communications module. It describes the procedures needed to install, configure, and troubleshoot the module. Before you initialize, operate, or service the module, you should read this manual in its entirety.

Contents of this Manual

This manual contains the following information:

Chapter	Title	Contents
	Preface	Describes the purpose, background, and scope of this manual. Also provides information on safety precautions and technical support.
1	Overview	Provides an overview of the 1203-CN1 module, ControlNet, and SCANport.
2	Installation	Provides procedures for installing the 1203-CN1 module.
3	Getting Started	Provides procedures for configuring the 1203-CN1 module, including how to set up a serial connection to the module, navigate in the module's software, edit its parameters, view its serial number, perform a flash upgrade to its firmware, and view its event queue.
4	Configuring a Controller to Communicate with the 1203-CN1 Module	Provides procedures for using RSNetWorx to set up a ControlNet network and configure controllers to communicate with devices, such as the 1203-CN1 module.
5	PLC Ladder Logic Programming	Provides information on and an example of a ladder logic program used to control the SCANport product.
6	Using Messages	Provides information on and examples of messages used to set and monitor data in the SCANport product.
7	Troubleshooting	Explains how to troubleshoot the 1203-CN1 module using its LEDs.
A	Specifications	Provides specifications for the 1203-CN1 module.
B	1203-CN1 Module Parameters	Provides information on datalinks, information on fault configurable inputs, and a list of the 1203-CN1 module's parameters.
C	N-File Structure	Lists the N-file structure for the 1203-CN1 module and attached SCANport products.
D	ControlNet Objects	Provides a reference list of ControlNet objects.
E	Supported PCCC Messages	Provides a reference list of PCCC messages supported by the 1203-CN1 module.
F	Supported Emulated Block Transfer Commands	Provides a reference list of emulated block transfer commands.

Safety Precautions

Please read the following safety precautions carefully.

ATTENTION: Only personnel familiar with SCANport devices and the associated machinery should plan or implement the installation, start-up, configuration, and subsequent maintenance of the 1203-CN1 module. Failure to comply may result in personal injury and/or equipment damage.

ATTENTION: The 1203-CN1 ControlNet communications module contains ESD (Electrostatic Discharge) sensitive parts. Static control precautions are required when installing, testing, or servicing this module. Device malfunction may occur if you do not follow ESD control procedures. If you are not familiar with static control procedures, refer to Allen-Bradley publication 8000-4.5.2, Guarding Against Electrostatic Damage, or other applicable ESD protection handbook.

Terms and Abbreviations

The following terms and abbreviations are specific to this product. For a complete listing of Allen-Bradley terminology, refer to the Allen-Bradley Industrial Automation Glossary, Publication AG-7.1.

Term:	Definition
ControlNet	An open network that provides deterministic I/O control and unscheduled messaging through a time division multiplexing scheme. ControlNet offers a redundant media option.
SCANport	A standard peripheral communications interface for various Allen-Bradley drives and power products.
SCANport Peripheral	A device that provides an interface between SCANport and a network. It is often referred to as an adapter. For example, the 1203-CN1 module is a SCANport peripheral.
SCANport Product	A device that uses the SCANport communications interface to communicate with one or more peripheral devices. For example, a motor drive such as a 1336 PLUS is a SCANport product.
1203-CN1 ControlNet Communications Module	In this manual, it is also called “1203-CN1 module” or “module.”

Conventions Used in this Manual

The following conventions are used throughout this manual:

- Bulleted lists provide information, not procedural steps.
- Numbered lists provide sequential steps or hierarchical information.
- *Italic* type is used for chapter names and for parameter names.
- **Bold** type is used for names of menus, menu options, screens, and dialog boxes.

Important: This type of paragraph contains tips or notes that have been added to call attention to useful information.

Rockwell Automation Support

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

Local Product Support

Contact your local Rockwell Automation representative for:

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

Technical Product Assistance

If you need to contact Rockwell Automation for technical assistance, please review the information in the Troubleshooting chapter first. If you are still having problems, then call your local Rockwell Automation representative.

Refer to http://www.ab.com for updates and supporting documentation.

Overview

Chapter Objectives

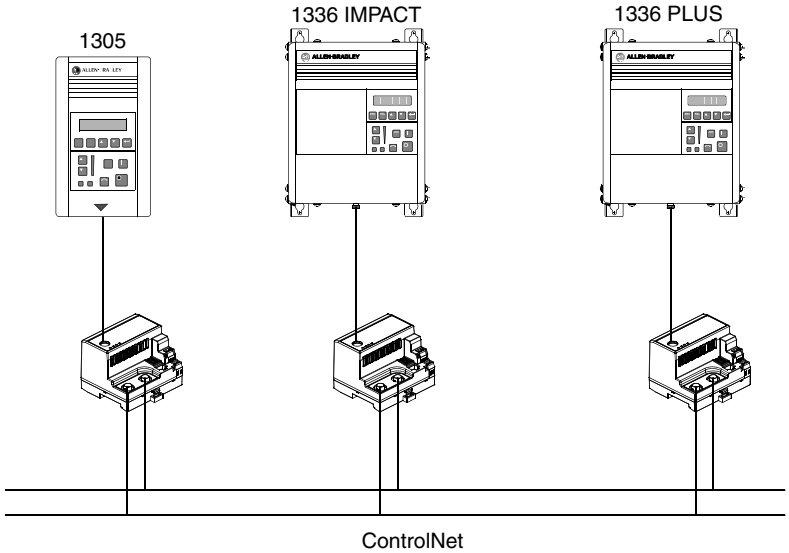
Chapter 1 provides an overview of your 1203-CN1 ControlNet communications module. It provides the following information:

- Description of how the 1203-CN1 module works.
- Overview of ControlNet.
- Overview of SCANport products.
- Parts of the 1203-CN1 module.
- Overview of setting up the module.
- Required equipment and tools.

Overview of the 1203-CN1 ControlNet Communications Module

The 1203-CN1 ControlNet communications module provides an electronic communications interface between a ControlNet network and any single SCANport product.

Figure 1.1
Example of 1203-CN1 Modules Connecting SCANport Products to ControlNet



A SCANport cable connects the module to a SCANport product through a SCANport interface port on the SCANport product. One or two ControlNet cable taps connect the module to the ControlNet bus, depending on whether you are using non-redundant or redundant connections.

The module translates the ControlNet messages into SCANport messages that can be understood by the SCANport product. Both scheduled I/O data and unscheduled messages can be transferred through the module. ControlNet capability enhances the functionality and usefulness of the connected product and lets you communicate with the SCANport product from any node on the ControlNet network.

What Is ControlNet?

ControlNet is a real-time, control-layer network providing high-speed transport of both scheduled time-critical I/O data and unscheduled messaging data, including upload/download of programming and configuration data and peer-to-peer messaging, on a single physical media link. Deterministic and repeatable, ControlNet's high-speed (5 Mbps) control and data capabilities significantly enhance the size and speed of I/O data and messaging data transfers.

Specifically, ControlNet provides:

- Bandwidth for I/O, real-time interlocking, peer-to-peer messaging and programming—all on the same link, without impacting time-critical I/O.
- Deterministic, repeatable performance for both discrete and process applications.
- Multicast of both inputs and peer-to-peer data.
- Fiber media, media redundancy and intrinsically safe options.
- Simple and flexible installation requiring no special tools to install or tune the network.
- Network access for any node.
- Support for up to 99 nodes per subnet to help flatten architectures and support more distributed systems.
- User-configured real-time remote analog updates for more flexibility and process capabilities.

SCANport Products

Some SCANport products support one peripheral; others support up to six peripherals. The table below lists SCANport products, the number of peripherals each supports, and the minimum and maximum I/O words allowed between the product and module.

Product	Number of Peripherals Supported	I/O Words	
		Minimum	Maximum
1305 AC MICRO Drive	5①	0	10
1336 IMPACT™ Drive	6②	0	10
1336 PLUS AC Drive	6②	0	10
1336 PLUS II Drive	6②	0	10
1336 FORCE™ Drive	6②	0	10
1336 Line Regeneration Package	2	0	2
1394 AC Mult-Axis Motion Control System	6	0	10
SMC Dialog Plus	1	0	2
SMP-3 Smart Motor Protector	2	0	2
1397 Digital DC Drive	5	0	10
1557 Medium Voltage Drive	5	0	10
2364 RGU DC Bus Regeneration Front End	6	0	10

① Early versions of the 1305 AC MICRO Drive firmware may not support some types of communications.

② Lower horsepower products may not support a sixth peripheral. Refer to your user manual to verify that your product supports a sixth peripheral.

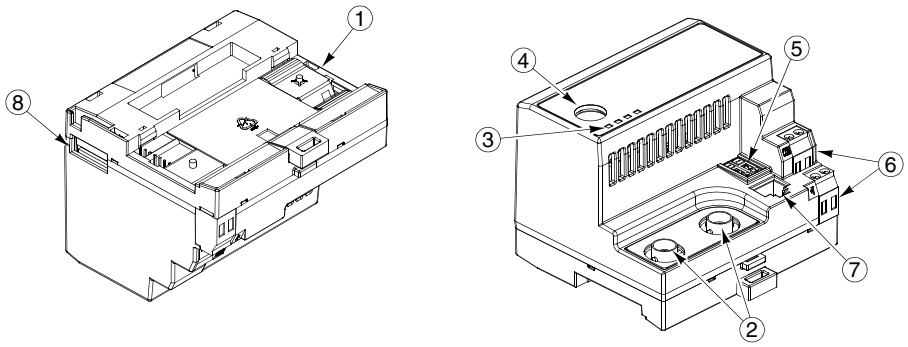
Important: If you intend to use datalinks to communicate with and control your SCANport product, verify that your SCANport product supports datalinks before enabling them in the module.

Important: To connect multiple peripherals to a SCANport product, a port expander may be required. Refer to your product's documentation for more information.

What Hardware Is Included?

Figure 1.2 and the table below illustrate and list the main parts of the 1203-CN1 ControlNet communications module:

Figure 1.2
Parts of the Communications Module



Number	Part	Description
1	DIN Rail Mount	Securely attaches and electronically grounds the module to the DIN rail.
2	ControlNet Coax Connections	Provide connections for ControlNet cable taps to allow either redundant or non-redundant communications over the ControlNet network.
3	Bi-Color LEDs	Indicate the status of each ControlNet media channel, of the SCANport connection, and of the module. For more information, refer to Chapter 7, Troubleshooting.
4	SCANport Connection	Provides a standard SCANport 8-pin circular mini-DIN connector for the SCANport cable.
5	ControlNet Node Address Indicator	Displays the ControlNet node address for the module. Use the push buttons to set the address before applying power.
6	+24V DC Power Connections	Provide for a +24V DC power supply connection. Multiple connections let you daisy chain power through a group of modules placed close to each other.
7	ControlNet Network Access Port	Provides an RJ-45 connection for devices capable of communicating over ControlNet. The module as well as other operational network devices can be accessed. A ControlNet network access cable (1786-CP) and 1784-KTCX communication card, 1784-PCC communication card, or 1770-KFC communication interface are required to use this port.
8	RS-232 Serial Port	Provides a connection for terminals capable of RS-232 serial communications. This port can be used to edit the module's parameters, download a file needed to perform a flash to the module's operating code, and support devices that monitor and test the module. A 1203-SFC serial cable and a PC running a terminal emulation program or a VT100-compatible terminal are required to use this port.

Overview of Setting Up the 1203-CN1 Module

To set up the 1203-CN1 module, you must perform the following tasks:

1. Install the module. Refer to Chapter 2, *Installation*.
2. If desired, configure the module's parameters. Refer to Chapter 3, *Getting Started*.
3. Configure the module on the ControlNet network. Refer to Chapter 4, *Configuring a Controller to Communicate with the 1203-CN1 Module*.

Required Tools and Equipment

To install and configure a 1203-CN1 module, you need the following equipment:

- Grounding wrist strap (shipped with the module).
- 1203-CN1 ControlNet communications module.
- 35 x 7.5 mm DIN rail A (Allen-Bradley part number (199-DR1; 46277-3; EN 50022).
- 1/8 in. (3.2 mm) flathead screwdriver.
- Blunt, pointed instrument (not pen or pencil) for setting the node address.
- Ohm meter.
- Appropriate cables for SCANport and ControlNet connections. Refer to Chapter 2, *Installation*, for more information.
- 1203-SFC serial cable.
- Either a PC running a Windows™ terminal emulation program (e.g., HyperTerminal) or a VT100-compatible terminal.
- PC that is:
  - Running RSNetWorx™, RSLogix5™, and RSLinx™.
  - Connected to the ControlNet network using a 1784-KTCX card, 1784-PCC card, or 1770-KFC adapter.

Notes:

Installation

Chapter Objectives

Chapter 2 provides the information that you need to install the 1203-CN1 ControlNet communications module. This information includes:

- A list of tools and equipment needed for the installation.
- A discussion of available cables for SCANport and ControlNet connections.
- Instructions for installing the module.
- Instructions for removing the module.

After installing the module, refer to Chapter 3, *Getting Started*, for procedures on how to configuring the module's parameters and refer to Chapter 4, *Configuring a Controller to Communicate with the 1203-CN1 Module*, for information on configuring the module on the ControlNet network.

Required Tools and Equipment

To install your 1203-CN1 module, you will need the following tools and equipment:

- Grounding wrist strap.
- 1203-CN1 ControlNet communications module.
- 35 x 7.5 mm DIN rail A (Allen-Bradley part number 199-DR1; 46277-3; EN 50022).
- 1/8 in. (3.2 mm) flathead screwdriver.
- Blunt, pointed instrument (not pen or pencil) for setting the node address.
- Appropriate cables for SCANport and ControlNet connections. Refer to the Selecting Cables section in this chapter.
- Ohm meter.

Selecting Cables

To connect the 1203-CN1 module to the SCANport product and the ControlNet network, you must select an appropriate SCANport cable and one or two ControlNet cable tap(s). Use the following information to select appropriate cables for each connection.

SCANport Cables

When selecting the SCANport cable to connect the module to the SCANport product, you need to:

- Use an Allen-Bradley SCANport cable. Refer to the table below.

Male to Male Connection[1]		Male to Female Connection	
Length	Catalog Number	Length	Catalog Number
1/3 m	1202-C03	1/3 m	1202-H03
1 m	1202-C10	1 m	1202-H10
3 m	1202-C30	3 m	1202-H30
9 m	1202-C90	9 m	1202-H90

[1] For most installations, a male-to-male connection on the cable is required.

- Do not exceed 10 meters (33 feet) of cable between the SCANport product and module.
- Keep SCANport cables away from high power cables to guard against introducing noise into your system.

ControlNet Cable Taps

A tap connects a node on the ControlNet network, such as a module, to the cable system via an integral 1 m (39.6 in.) drop cable. When selecting a tap to connect the module to the ControlNet network, you need to:

- Determine if your network uses a redundant media system. If so, you will need **two** taps.
- Use one or two Allen-Bradley tap(s). Refer to the table below.

Type	Catalog Number
Straight T-Tap	1786-TPS
Straight Y-Tap	1786-TPYS
Right-Angle T-Tap	1786-TPR
Right-Angle Y-Tap	1786-TPYR

For more information on ControlNet taps and ControlNet networks, refer to Publication 1786-6.2.1, ControlNet Cable System Planning and Installation Manual.

Electrostatic Discharge Precautions

Please read the following safety precautions carefully before installing the 1203-CN1 module

ATTENTION: The 1203-CN1 ControlNet communications module contains ESD (Electrostatic Discharge) sensitive parts. Static control precautions are required when installing, testing, or servicing this module. Device malfunction may occur if you do not follow ESD control procedures. If you are not familiar with static control procedures, refer to Allen-Bradley publication 8000-4.5.2, Guarding Against Electrostatic Damage, or other applicable ESD protection handbook.

Important: You must wear a grounding wrist strap that is properly grounded when you handle the 1203-CN1 module.

Installing Your 1203-CN1 Module

The following instructions explain how to physically install your 1203-CN1 module.

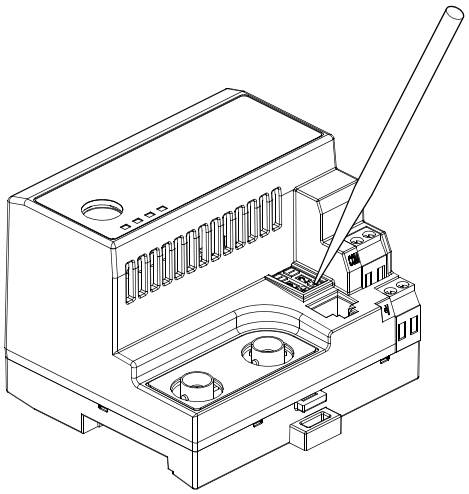
Important: To guard against device malfunction, you must wear a grounding wrist strap when installing the 1203-CN1 module.

1. Set the module's ControlNet node address by clicking the + or - button to the desired value for each digit.

Important: Each node on the ControlNet network must have a unique address.

Important: The node address must be set before power is applied because the module uses the node address it detects when it first receives power. To change a node address, you must set the new value and then remove and reapply power to or reset the module.

Figure 2.1
Setting the Module's Node Address



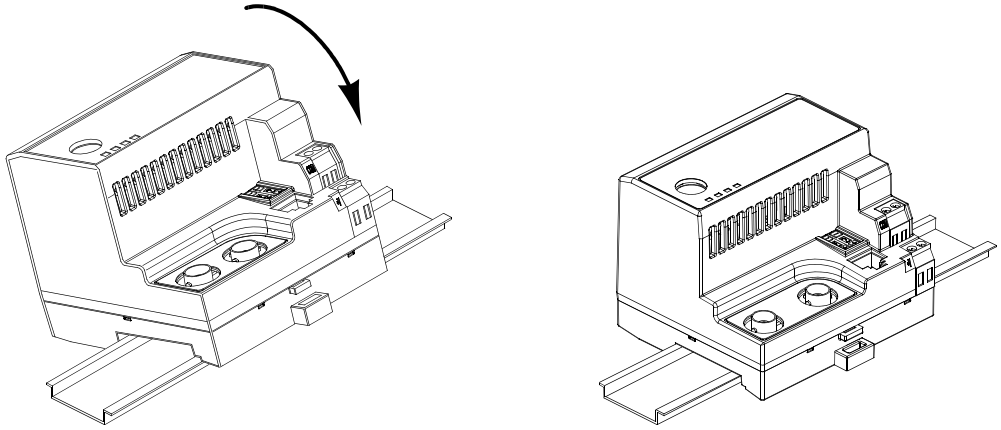
ATTENTION: When setting the node address, use a blunt, pointed instrument. Do not use a pencil or pen because lead (graphite) or ink may damage the switch assembly.

2. Ensure the DIN rail to which the module will be attached is connected to an earth ground.

Important: If EMC compliance is required, the DIN rail should be properly grounded inside a full metal enclosure. The enclosure should also be properly connected to an earth ground.

3. Hook the top lip of the module's DIN rail mount onto the top of the DIN rail and then rotate the module onto the DIN rail. You will hear the module snap into a locked position.

Figure 2.2
Connecting the Module to the DIN Rail



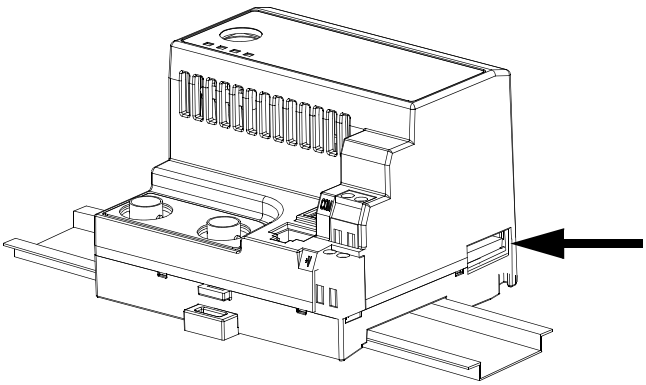
4. Verify the module is correctly grounded to the DIN rail by using an Ohm meter to measure between:

   • DIN rail's earth ground.

   • Metal shell in the module's RS-232 serial port.

   If the reading is greater than 2 Ohms, you must reconnect the module to the DIN rail, making sure it attaches securely.

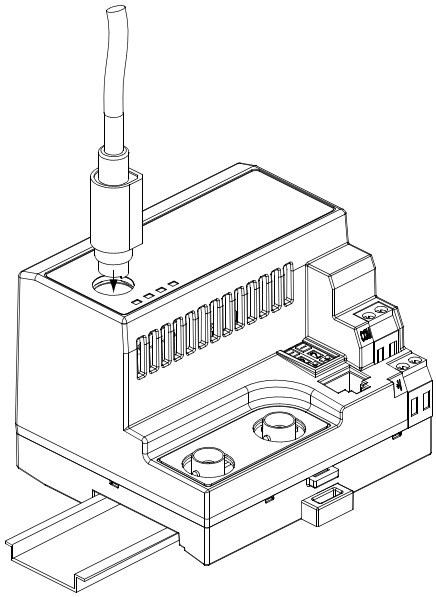
Figure 2.3
Metal Shell of the Module's RS-232 Serial Port



5. Connect the SCANport cable to the SCANport product and the module.

   To connect the cable to the module, align the pins on the cable with the holes in the SCANport connection and then insert the SCANport cable. The cable will click into a locked position.

Figure 2.4
Connecting the SCANport Cable to the Module

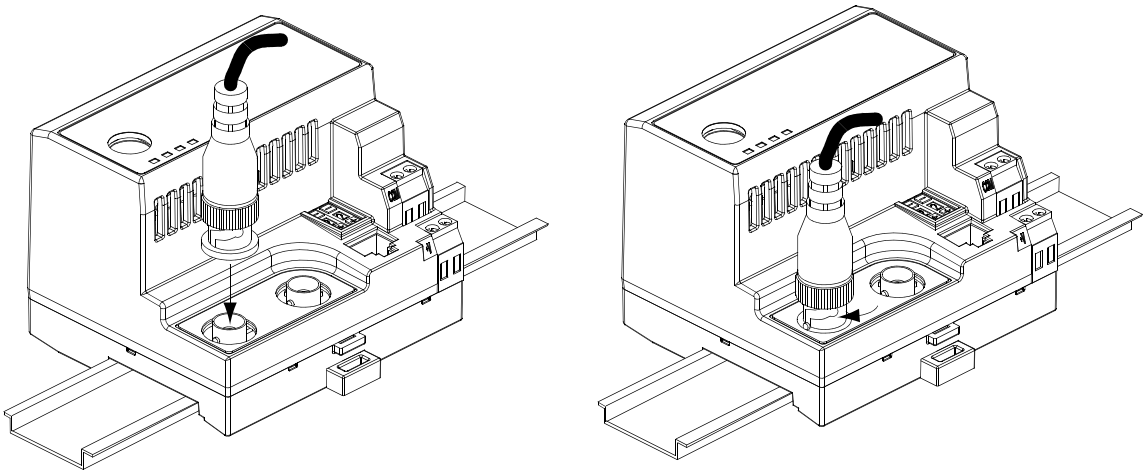


6. Connect the ControlNet cable tap(s) to the ControlNet media and the module.

   To connect the cable tap(s) to the module, twist each onto the ControlNet Coax connection(s). You will hear the tap(s) click into a locked position.

Important: Make sure you connect the Channel A cable to the Channel A connection and the Channel B cable to the Channel B connection.

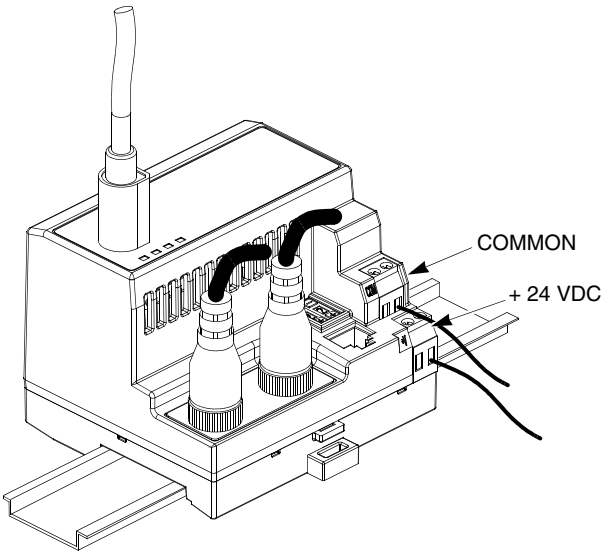
Figure 2.5
Connecting a Straight ControlNet Tap to the Module



7. Connect a +24V power supply. If necessary, loosen the screw to insert the power supply connection and then re-tighten the screw.

Important: You can use the two sets of holes to daisy chain the power supply between multiple 1203-CN1 modules placed close together.

Figure 2.6
Connecting the Power Supply to the Module



The module is now physically installed. Its SCANport and ControlNet LEDs are solid green, and its Module LED is flashing green. (If your module's LEDs differ, refer to Chapter 7, *Troubleshooting*, for troubleshooting information.)

The flashing green Module LED means that the module is not yet able to provide an interface between the ControlNet network and the SCANport product. For it to do so, you must:

1. Configure, if necessary, the module by editing its parameters. Refer to Chapter 3, *Getting Started*.

2. Configure the controller on the ControlNet network to recognize and communicate with the module. Refer to Chapter 4, *Configuring a Controller to Communicate with the 1203-CN1 Module*.

3. If desired, write a PLC Ladder Logic program to control the SCANport product. Refer to Chapter 5, *PLC Ladder Logic Programming*.

Important: The 1203-CN1 ControlNet communications module will not communicate over ControlNet without being connected to a SCANport product which is powered and operational.

Removing the 1203-CN1 Module

If you want to remove the 1203-CN1 module, you need to:

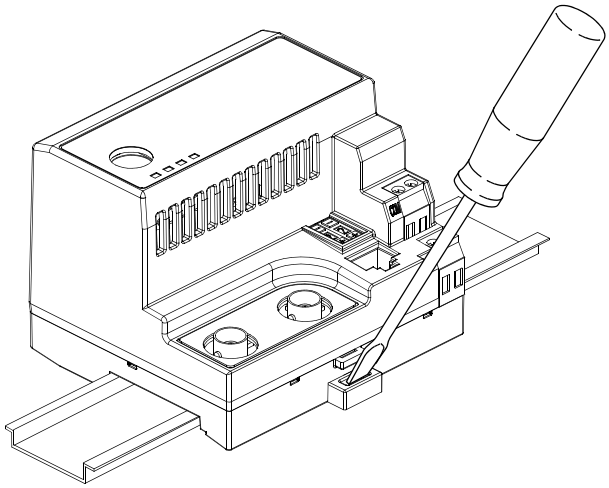
Important: To guard against device malfunction, you must wear a grounding wrist strap when removing the 1203-CN1 module.

1. Turn off the power supply to the module.
2. Disconnect all cables and the power supply from the module.

Important: To disconnect the SCANport cable, gently push in the cable and then pull it out.

3. With a screw driver in the tab release, gently push its handle towards the module to release the connection tab, and then pull the module off of the DIN Rail.

Figure 2.7
Removing the Module from the DIN Rail



Getting Started

Chapter Objectives

Chapter 3 provides information that you need to configure the 1203-CN1 ControlNet Communications module. This includes:

- Information on the 1203-CN1 module's default setting.
- Equipment needed to make a serial connection to the module.
- Instructions on how to connect a PC running terminal emulation software or a VT100-compatible terminal to the module.
- Navigation techniques to use in the module's software.
- Instructions for editing the module's parameters.
- Instructions for displaying and clearing the module's event queue.
- Instructions for displaying the modules I/O data.
- Instructions for displaying DF1 Protocol statistics.
- Instructions for viewing the module's serial number.
- Instructions for performing a flash upgrade to the module.

Factory-Default Settings for the 1203-CN1 Module's Parameters

The factory-default settings of the 1203-CN1 module enable the following functions:

- 16-bit Logic Command/Status.
- 16-bit Reference/Feedback.
- If the PLC is put into program mode or the network fails, the SCANport product will be faulted by the module.
- All datalinks are disabled.
- Baud rate is 9600.

If you wish to change any of these functions (e.g., Fault Configurable inputs) or add more functions (e.g., datalinks), you must edit the module's parameters. To do so, refer to:

- Appendix B, *1203-CN1 Module Parameters*, for detailed information about each of the module's parameters.
- Instructions in this chapter on establishing a serial connection.
- Instructions in this chapter on how to edit the parameters.

Required Tools and Equipment

To make a serial connection to the module, you need the following:

- Grounding wrist strap.
- 1203-SFC serial cable.
- Either a PC running a Windows terminal emulation program (e.g., HyperTerminal) or a VT100-compatible terminal.

Electrostatic Discharge Precautions

Please read the following safety precautions carefully before making a serial connection to the 1203-CN1 module.

ATTENTION: The 1203-CN1 ControlNet communications module contains ESD (Electrostatic Discharge) sensitive parts. Static control precautions are required when installing, testing, or servicing this module. Device malfunction may occur if you do not follow ESD control procedures. If you are not familiar with static control procedures, refer to Allen-Bradley publication 8000-4.5.2, Guarding Against Electrostatic Damage, or other applicable ESD protection handbook.

You must wear a grounding wrist strap that is properly grounded when you handle the 1203-CN1 module.

Establishing a Serial Connection with the 1203-CN1 Module

The module's software lets you do the following:

- Edit the module's parameters.
- View its serial number.
- View its event queue.
- View its current I/O data.
- View DF1 statistics.
- Perform a flash upgrade.

To access its software, you must make a serial connection between the module and either a PC running terminal emulation software or a terminal. Refer to the following table:

If Using:	Refer to Page
PC running terminal emulation software	3-3
VT100-compatible terminal	3-7

DriveExplorer (v.101 or higher) software can now also be used on 1203-CN1s that are v2.001 or higher. Do not use DriveExplorer software with v1.xxx CN1s.

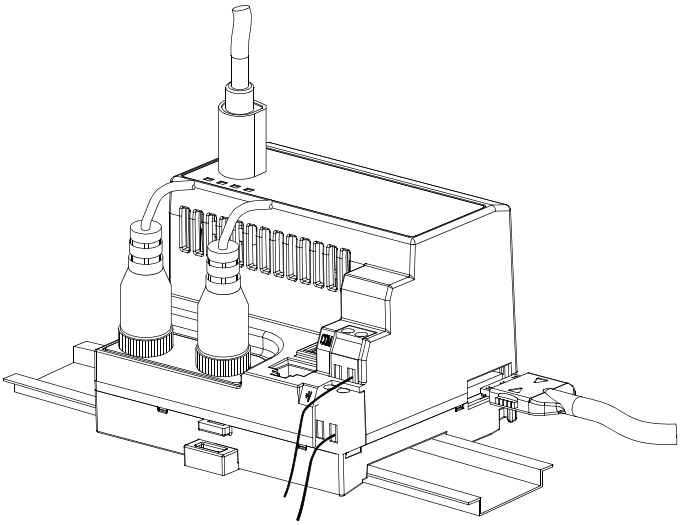
Using a PC Running Terminal Emulation Software

A variety of terminal emulation programs can be used to establish a serial connection to the module. The following instructions describe how to establish the initial serial connection to the module using a PC running Windows 95 HyperTerminal software. Future connections to the module can use this same configuration by clicking the icon added to the initial screen when the configuration is saved.

Important: If you are not using Windows 95 HyperTerminal, steps to establish a serial connection may vary from the following steps.

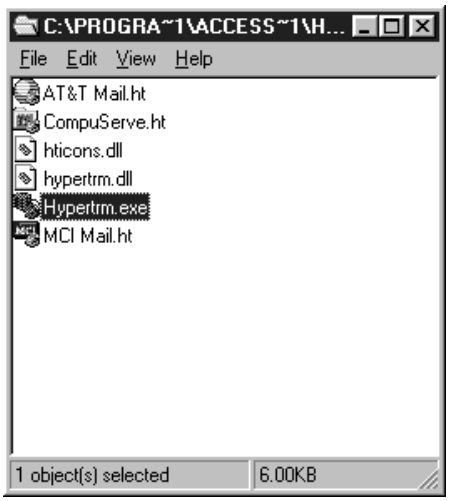
1. Connect a 1203-SFC serial cable to your PC's serial port and then to the RS-232 serial port on the module.

Figure 3.1
Connecting the 1203-SFC Serial Cable to the Module's RS-232 Serial Port



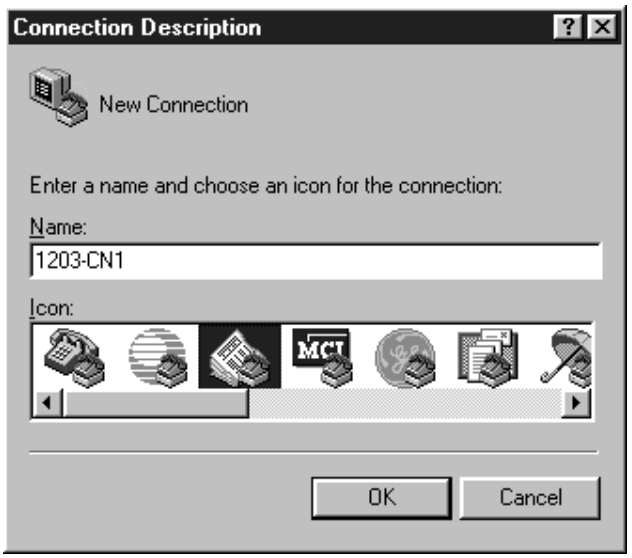
2. In the **Start** menu, select **Programs**, **Accessories**, and then **HyperTerminal**. A **HyperTerminal** dialog box appears.

Figure 3.2
Example HyperTerminal Dialog Box



3. Double-click **HyperTrm.exe**. The **Connection Description** dialog box appears.

Figure 3.3
Example Connection Dialog Box



4. Enter a name in the **Name** field and select any icon in the **Icon** field. In this example, we enter "1203-CN1" in the **Name** field.

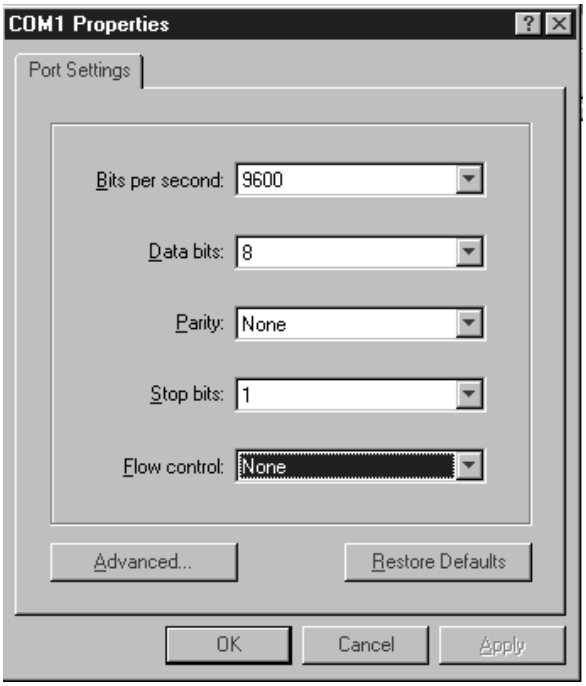
5. Click **OK**. The **Phone Number** dialog box appears.

Figure 3.4
Example Phone Number Dialog Box

6. In the **Connect Using** field, select the appropriate communications port (usually COM1 or COM2).

7. Click **OK**. The **Comm Properties** dialog box appears.

Figure 3.5
Example Comm Properties Dialog Box



8. Select the following settings:

   - **9600** in the **Bits per second** field.
     If you have previously set the module's *Serial Port Rate (Parameter 21)* to enable 19200 bps, set the bps to 19200 in this field.
   - **8** in the **Data bits** field.
   - **None** in the **Parity** field.
   - **1** in the **Stop bits** field.
   - **None** in the **Flow Control** field.

9. Click **OK**. A blank **HyperTerminal** screen appears.

10. In the **File** menu, select **Properties**. The **Properties** dialog box appears.

Figure 3.6
Properties Dialog Box

11. Click the **Settings** tab.

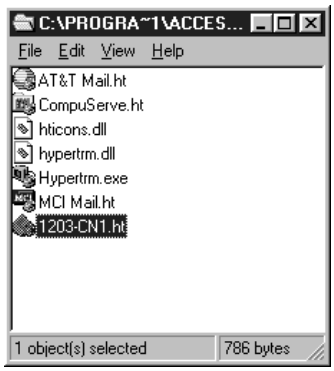
12. In the **Function, arrow, and ctrl keys act as** box, verify **Terminal keys** is selected.

13. In the **Emulation** field, verify **VT100** is selected.

14. Click **OK**.

15. In the **File** menu, select **Save**. The configuration is saved and the icon you selected will appear in the initial **HyperTerminal** window next time you start HyperTerminal.

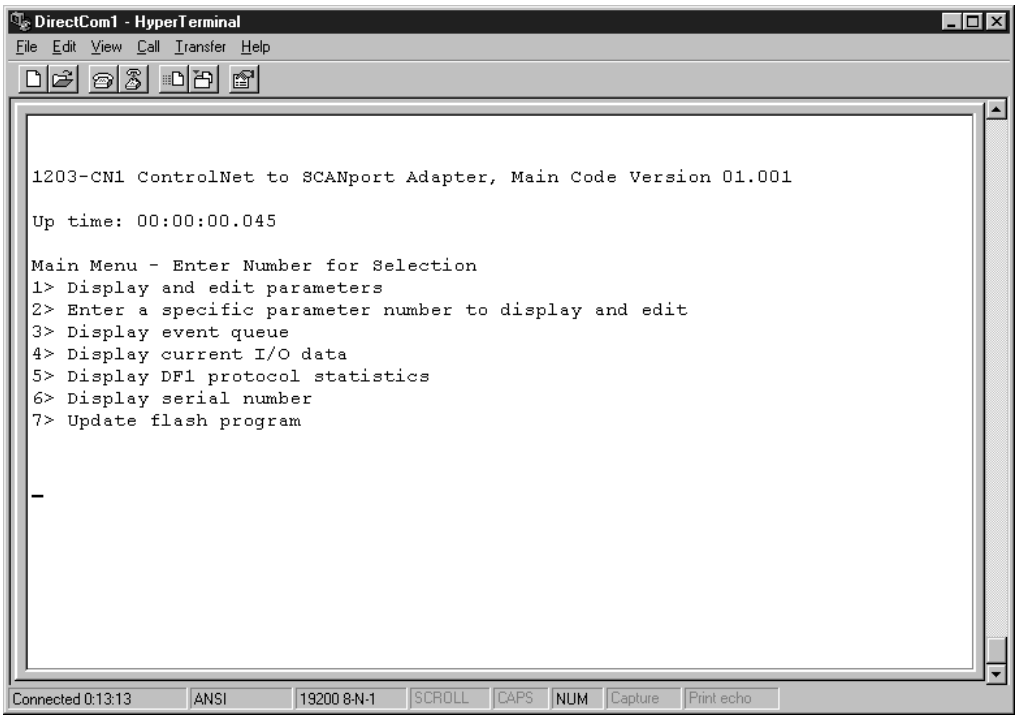
Figure 3.7
Example Initial HyperTerminal Window



Important: Next time you need to make a connection to the module, double-click the icon just created. In Figure 3.7, double-clicking the 1203-CN1.ht icon would re-establish the connection to the module.

16. Press the **Enter** key. The main menu of the1203-CN1 ControlNet to SCANport Adapter application appears.

Figure 3.8
Main Menu



You now have access to the module's software. Go to the Navigation Techniques section on page 3-8 for more information on using it.

Using a VT100-Compatible Terminal

Important: If you are using a PC, skip this section.

The following instructions describe how to establish a serial connection to the module using a VT100-compatible terminal.

1. Connect a 1203-SFC serial cable to your terminal and then to the RS-232 serial port on the module. Refer to Figure 3.1.

2. Start your terminal.

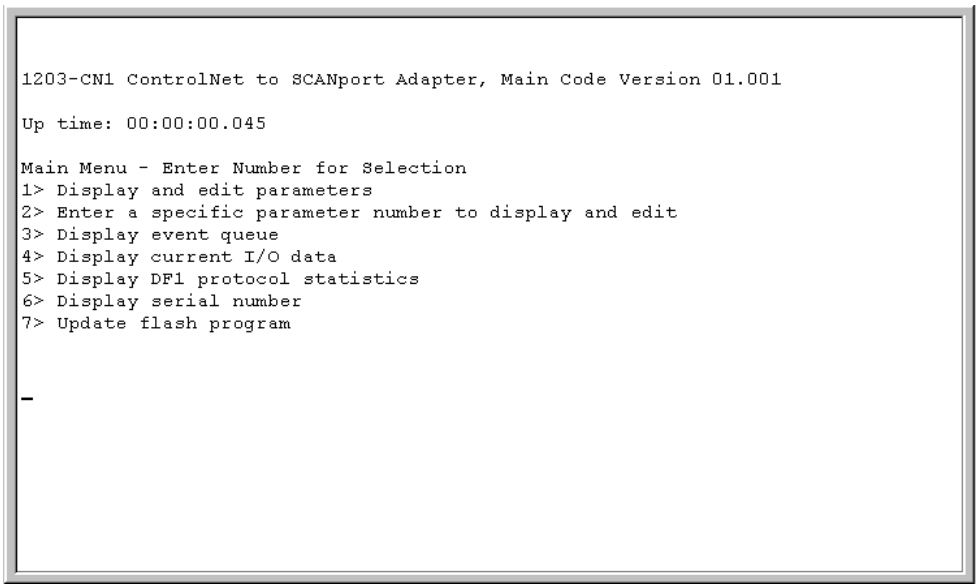
3. Select the following settings:

- **9600** in the **Bits per second** field.
  If you have previously set the module's *Serial Port Rate (Parameter 21)* to enable 19200 bps, set the bps to 19200.
- **8** in the **Data bits** field.
- **None** in the **Parity** field.
- **1** in the **Stop bits** field.
- **None** in the **Flow Control** field.

4. Press the **Enter** key. The main menu of the 1203-CN1 ControlNet to SCANport Adapter application appears.

   Refer to the Navigation Techniques section in this chapter for information on navigating in the software.

Figure 3.9
Main Menu



You now have access to the module's software. Go to the Navigation Techniques section on page 3-8 for more information on using it.

Navigation Techniques

To perform any of the functions in the software (e.g., editing parameters), you need to know the following navigation techniques:

Press:	To
1, 2, 3, 4, 5, 6, 7, 8, 9, 0	Select an option in the Main Menu (1 – 5) or enter a value for a parameter in the Parameter screen (0 – 9).
Escape	Return to Main Menu or abort changes to a parameter.
Down Arrow	View the next parameter.
Up Arrow	View the previous parameter.
Right Arrow	View the next value for a parameter.[1]
Left Arrow	View the previous value for a parameter.[1]
Enter	Save a value for a parameter.

[1] In the *Fault Config Logic Command* parameter (11), the right and left arrow keys let you navigate through the 16 bits.

Editing Parameters in the 1203-CN1 Module

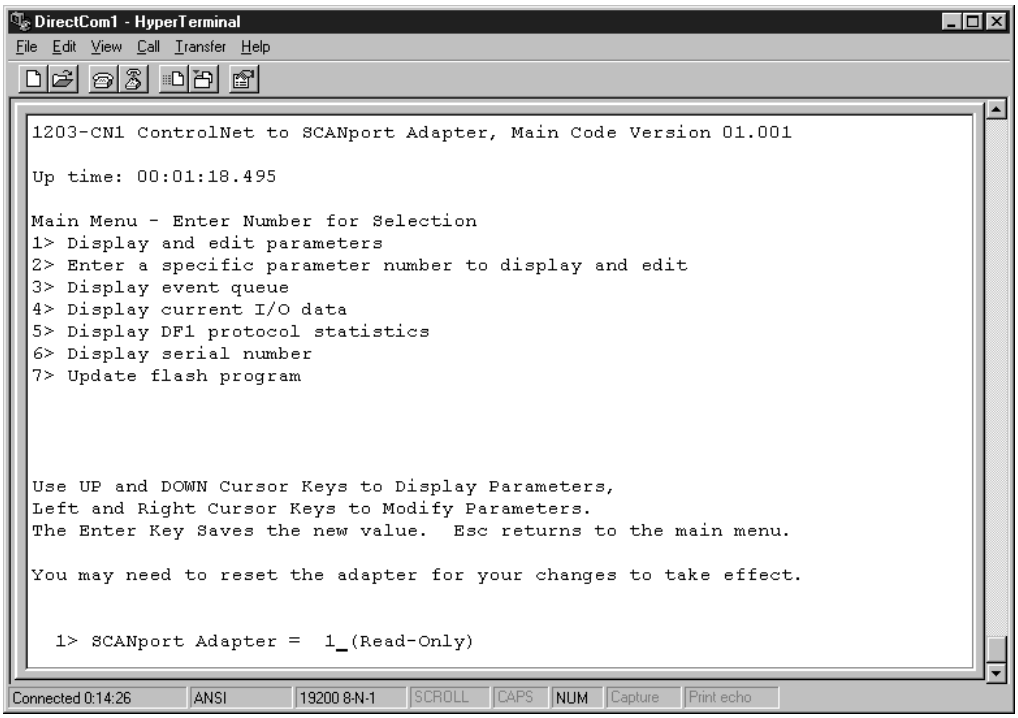
The 1203-CN1 module has many configurable parameters. Refer to Appendix B, *1203-CN1 Module Parameters*, for a detailed list. If you do not want to use the module's default settings, edit its parameters.

Important: You may also edit parameters over the ControlNet network using PCCC or emulated block transfer messages. Refer to Chapter 6, *Using Messages*, for information on and examples of messaging and refer to the appendixes for information needed to build messages.

To edit parameters using a serial connection, you need to:

1. Establish a serial connection to access the module's software. Refer to the Establishing a Serial Connection with the 1203-CN1 Module section earlier in this chapter.
2. Press **1** to select **1> Display Parameters**. The first parameter appears on the bottom of the screen.

Figure 3.10
Example Display Parameters Screen



Important: If you know the number of the parameter you intend to edit, press **2** to select **2> Enter Specific Parameter Number** and then enter the parameter number.

3. If necessary, scroll through the list of parameters by pressing the **Up Arrow** or **Down Arrow** key.

4. Edit parameters as necessary using the **Left Arrow** and **Right Arrow** keys. Refer to Navigation Techniques earlier in this chapter for information on changing values. Refer to Appendix B, *1203-CN1 Module Parameters,* for valid values.

5. If necessary, reset power to the module by enabling the *Reset Adapter* parameter (22). Refer to Appendix B, *1203-CN1 Module Parameters,* to see if the parameter you changed requires the module to be reset in order to take effect.

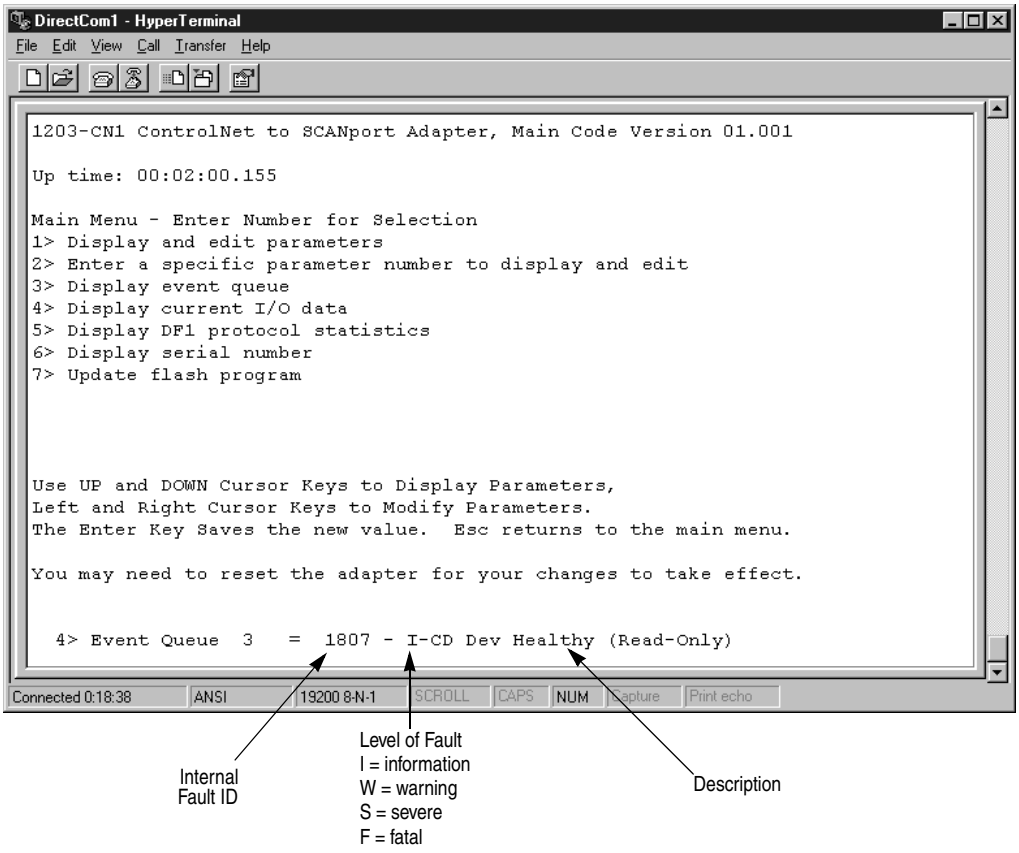
Displaying and Clearing the Event Queue in the 1203-CN1 Module

If an unexpected problem occurs with the module, you may need to check the event queue to view events that have happened in the module. Follow these instructions:

1. Establish a serial connection to access the module's software. Refer to the Establishing a Serial Connection with the 1203-CN1 Module section earlier in this chapter.

2. Press **3** to select **3> Display event queue**. The event queue appears.

Figure 3.11
Example Event Queue



3. Scroll through the list of Event Queue parameters by pressing the **Up Arrow** or **Down Arrow** key.

Number	Name	Description
1	Clr Event Queue	Enable = Clears the event queue. Ready = Leaves the event queue as is.
2 – 33	Event Queue 1 – Event Queue 32	Event in the event queue. Most recent event is listed in Event Queue 1.

4. If desired, clear the current fault in the adapter by setting *Clr Event Queue* (1) to **Enable** and pressing the **Enter** key.

Important: The Fault is cleared in the module and a "Clear Fault" event is added to the Event Queue.

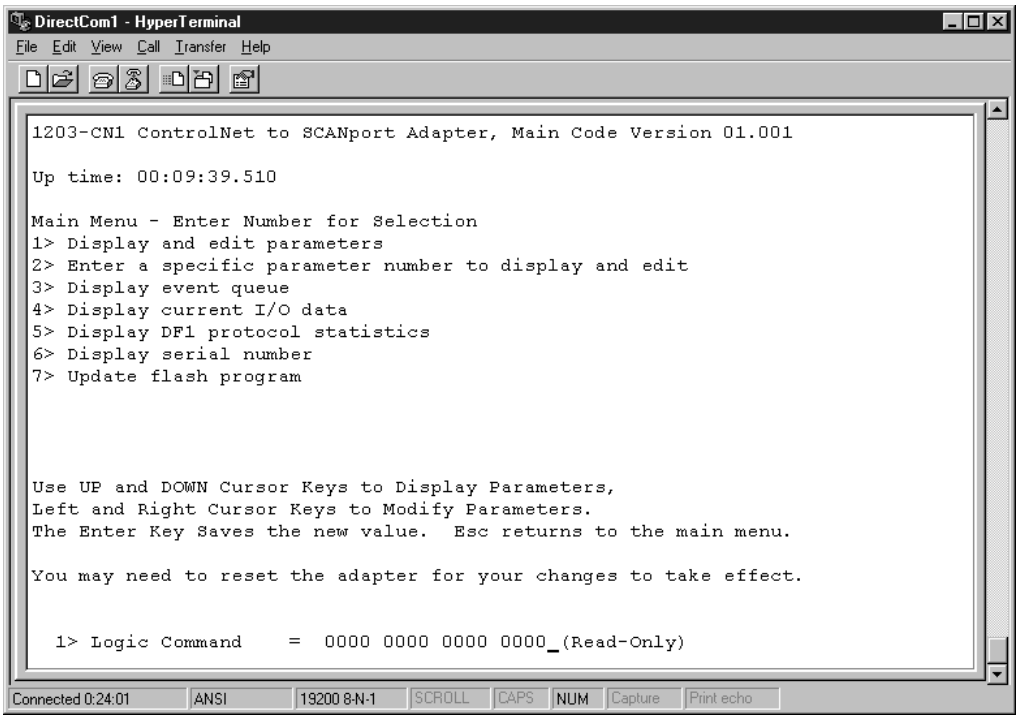
5. Press the **Escape** key to return to the main menu.

Displaying the 1203-CN1 module's Current I/O Data

You may need to do diagnostic testing to verify I/O data is passing through the module. To view the I/O data transmitted through the module, follow these instructions:

1. Establish a serial connection to access the module's software. Refer to the Establishing a Serial Connection with the 1203-CN1 Module section earlier in this chapter.
2. Press **4** to select **4> Display current I/O data**. The I/O Data screen appears.

Figure 3.12
Example I/O Data Screen



3. Scroll through the list of I/O Data parameters by pressing the **Up Arrow** or **Down Arrow** key.

Number	Name	Description
1	Logic Command	Buffer for Logic Command data
2	Logic Status	Buffer for Logic Status data
3	Reference Data	Buffer for Reference data
4	Feedback Data	Buffer for Feedback data
5 – 20	Data A1 In Val – Data D2 Out Val	Data going to (Input) or coming from (Output) the SCANport device

4. Press the **Escape** key to return to the main menu.

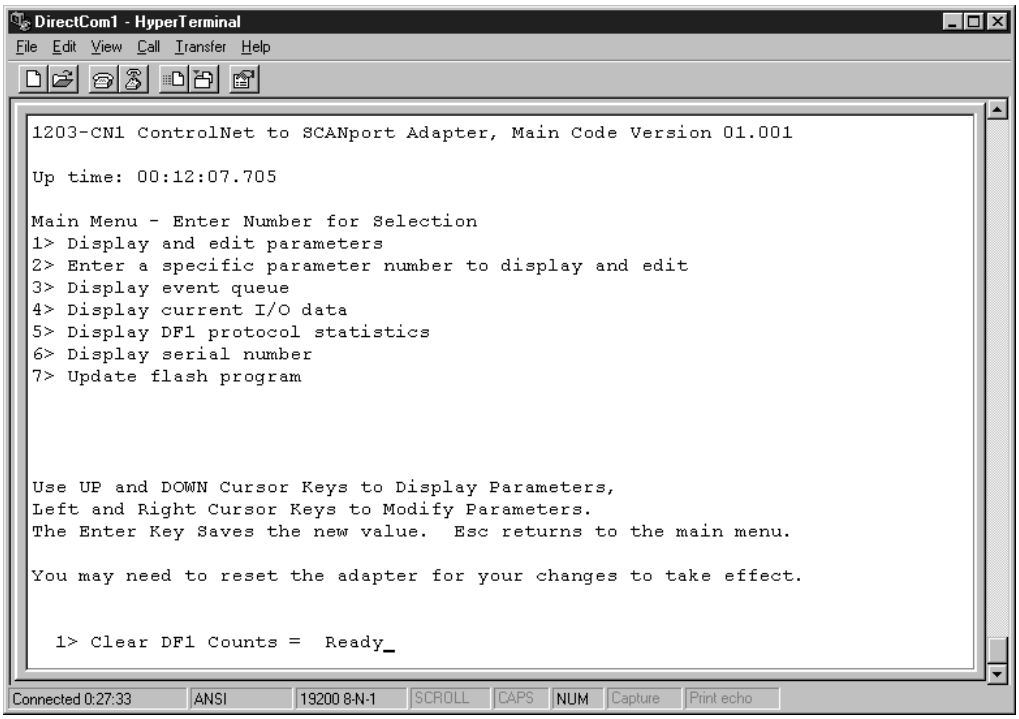
Displaying the DF1 Protocol Statistics in the 1203-CN1 Module

If you are working with DF1 communications, you may need to know how the module is using DF1 communications. To view statistics on DF1 communications, follow these directions:

1. Establish a serial connection to access the module's software. Refer to the Establishing a Serial Connection with the 1203-CN1 Module section earlier in this chapter.

2. Press **5** to select **5> Display DF1 protocol statistics**. The DF1 Protocol screen appears.

Figure 3.13
Example DF1 Protocol Statistic



3. Scroll through the list of DF1 statistic parameters by pressing the **Up Arrow** or **Down Arrow** key.

Number	Name	Description
1	Clear DF1 Counts	Ready = Accept default Enable = Reset all status parameters
2	DF1 Packets Sent	Total number of DF1 packets sent by the module
3	DF1 Packets Rcvd	Total number of DF1 packets received by the module
4	Undelivered Msgs	Total number of messages sent that were not acknowledged
5	ENQ Sent	Total number of inquiries sent by the module
6	ENQ Rcvd	Total number of inquiries received by the module
7	NAKs Received	Total number of NAKs received by the module
8	NAK Bad Packet	Total number of NAKs received by the module
9	NAK No Memory	Total number of NAKs sent by the module because the previous command did not yet complete and there was no place to save the new command
10	Duplicate Msgs	Total number of messages received by the module with the TNS number

4. If desired, reset the current DF1 protocol statistics by setting *Clear DF1 Counts* to **Enable** and pressing the **Enter** key.

5. Press the **Escape** key to return to the main menu.

Viewing Your 1203-CN1 Module's Serial Number

Each 1203-CN1 module has a unique serial number. To view the serial number, follow these instructions:

1. Establish a serial connection to access the module's software. Refer to the Establishing a Serial Connection with the 1203-CN1 Module section earlier in this chapter.

2. Press **6** to select **6> Display serial number**. The serial number for your communications module appears.

3. Press **Escape** to return to the Main Menu.

Performing a Flash Upgrade to the 1203-CN1 Module

To upgrade the 1203-CN1 module's flash firmware using the module's serial port, you need to perform a flash upgrade.

Important: To perform a flash upgrade to your module's firmware, you must use a PC running terminal emulation software.

Important: To exit the flash upgrade option before the download has started, simultaneously press the **Control** and **X** keys.

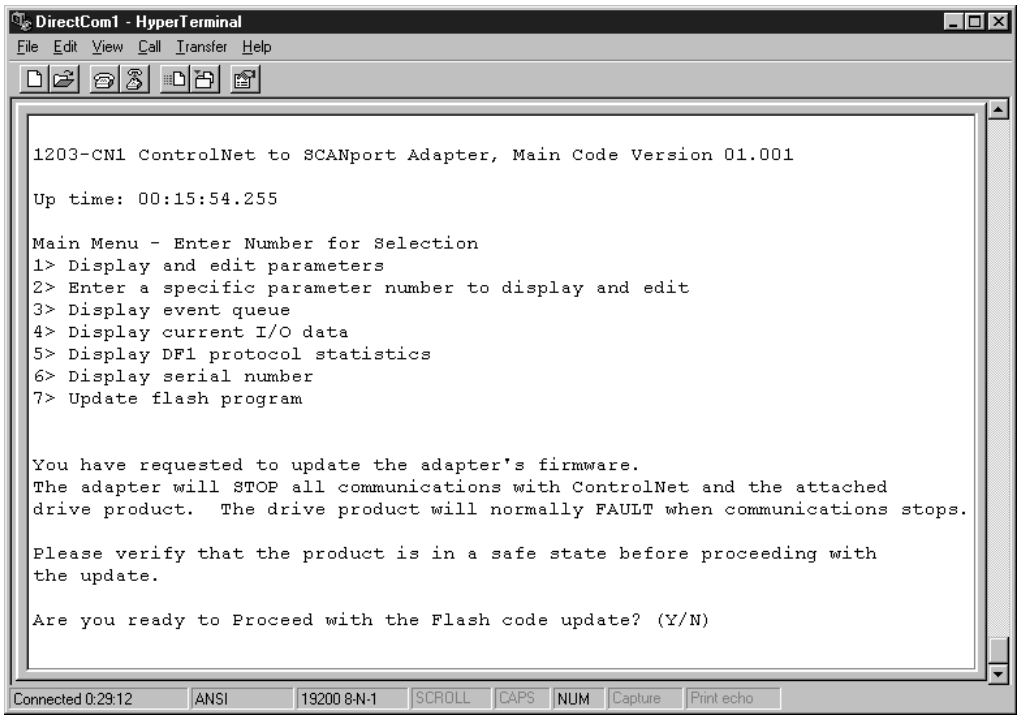
ATTENTION: Exiting the flash upgrade procedure once the download has begun can cause the module to become inoperable. If the module becomes inoperable, you must perform and complete a flash upgrade to fix the module.

To perform a flash upgrade, you need to:

1. Obtain the software file that contains the upgrade and record its location.
2. Establish a serial connection to access the module's software. Refer to the Establishing a Serial Connection with the 1203-CN1 Module section earlier in this chapter.
3. Press **7** to select **7> Update Flash Program**. The following screen appears.

ATTENTION: Hazard of personal injury or death exists when stopping a drive to perform a flash upgrade. When you perform a flash upgrade, the drive will fault and stop the motor if the drive is receiving control data from the 1203-CN1 module. Make sure the motor will stop safely or the drive will receive control data from an alternate source before beginning a flash upgrade.

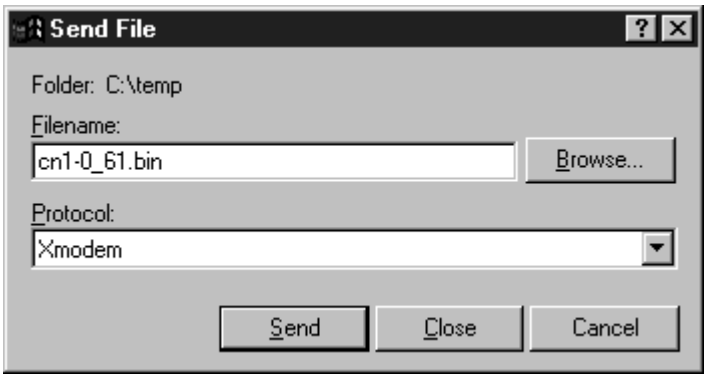
Figure 3.14
Initial Update Flash Program Screen



4. Press **Y** to verify that you want to perform a flash upgrade when prompted.

5. In the **Transfer** menu, select **Send File**. The **Send File** dialog box appears.

Figure 3.15
Send File Dialog Box



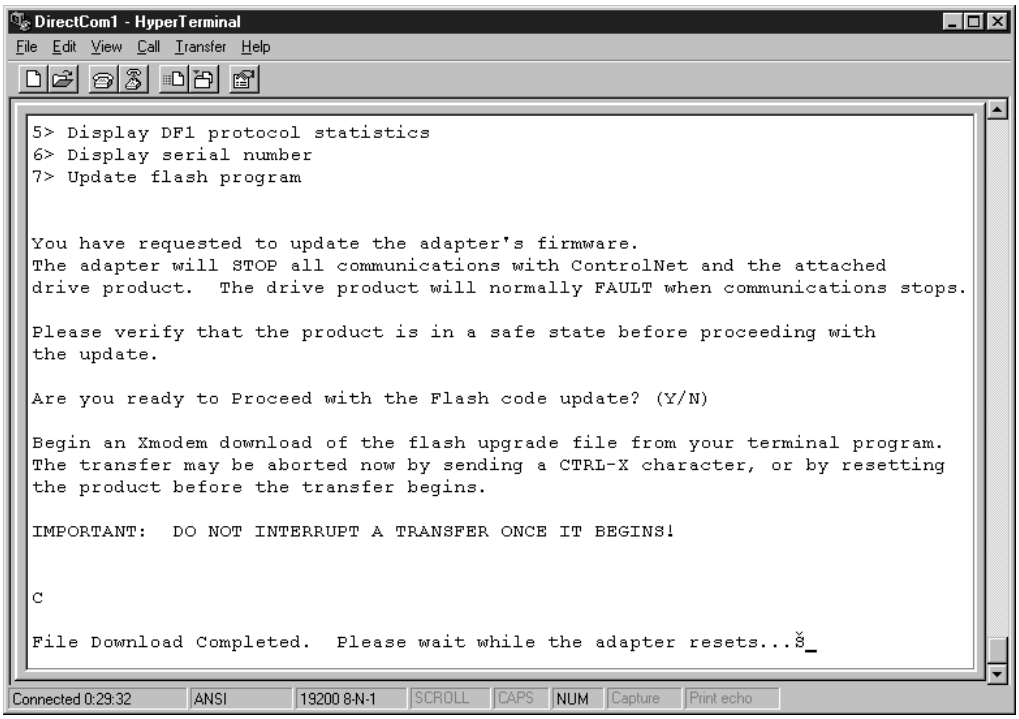
6. In the **Filename** field, select the file that contains the flash upgrade.

Important: You can click the **Browse** button to locate the file that contains the flash upgrade.

7. In the **Protocol** field, select **Xmodem**.

8. Click **Send**. A dialog box appears to report the flash is in progress. When the flash is complete, a message appears to tell you the download is complete and the module will reset itself.

Figure 3.16
Message Reporting the Flash Is Complete



The module resets itself, and then the Main menu appears. You must clear the drive's fault in order to restart the drive.

Configuring a Controller to Communicate with the 1203-CN1 Module

Chapter Objectives

Chapter 4 provides instructions for configuring your controller to communicate with the 1203-CN1 ControlNet communications module. This allows the product connected to the 1203-CN1 module to be an active node on the ControlNet network. This chapter provides information on:

- RSNetWorx.
- Equipment and software needed for the configuration.
- Configuring a controller to communicate with the 1203-CN1 module.

This chapter assumes you have experience using RSNetWorx to configure a ControlNet network.

What is RSNetWorx?

RSNetWorx for ControlNet is a 32-bit Windows application program that lets you configure ControlNet networks. Using a graphical representation of your network, you can configure network-wide parameters and the network-wide schedule.

After installing and configuring the module, you must use RSNetWorx to configure the controller to recognize and communicate with the module.

For more information on RSNetWorx, refer to:

- Getting Results with RSNetWorx for ControlNet, Doc. ID 9399-CNETGR.
- Hardware Configuration Reference Guide, Doc. ID 9399-HDWAREREF.
- RSNetWorx online help.

Required Equipment and Software

Before configuring the controller, your PC must be:

- Running RSNetWorx and RSLinx applications. Refer to http://www.software.rockwell.com for more information on these products.
- Connected to and communicating with the ControlNet network using a 1784-KTCX card, 1784-PCC card, or 1770-KFC adapter.

Configuring a Controller to Communicate with the 1203-CN1 Module

For the controller on the ControlNet network to transmit control I/O and/or messages to the 1203-CN1 module, you must configure it to recognize and communicate with the 1203-CN1 module.

These instructions describe how to use RSNetWorx to configure a new ControlNet network in online mode. The main steps in the configuration are:

- Using online mode in RSNetWorx.
- Mapping the 1203-CN1 module to the network.
- Verifying the network properties.

Important: RSNetWorx 1.6 and RSLinx 2.0.82 were used for these instructions. If you are using other versions, you may notice differences between these instructions and your screens.

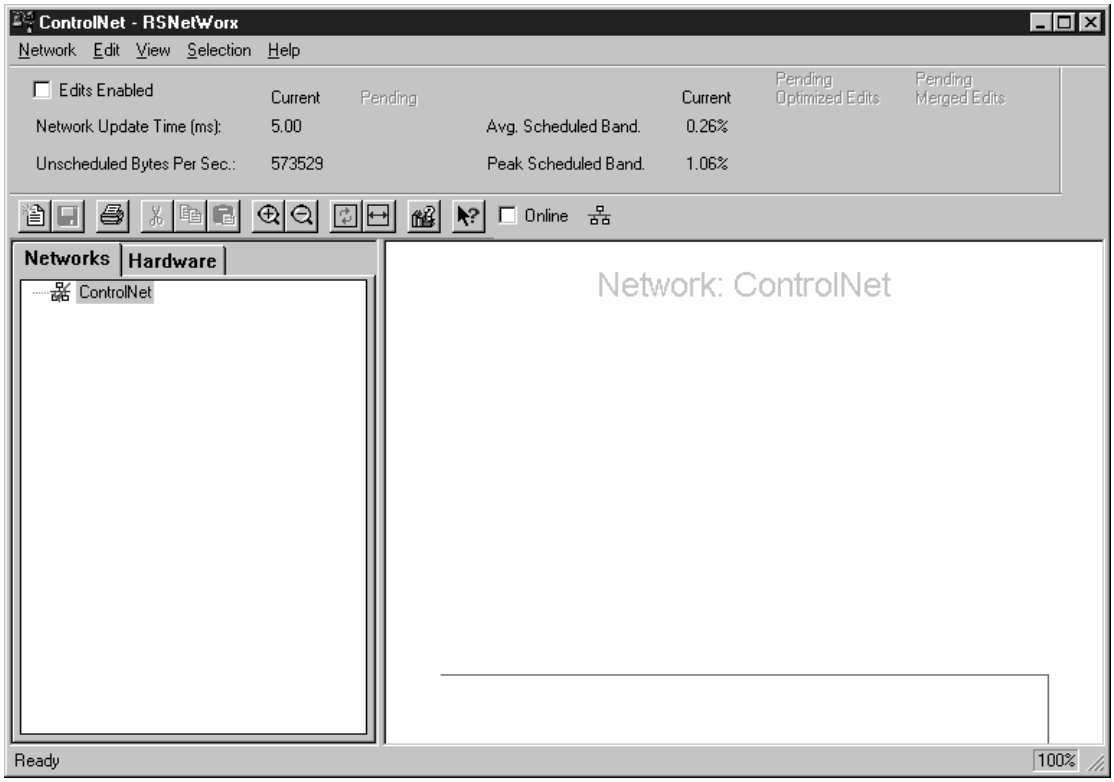
Using Online Mode in RSNetWorx

It is easiest to configure the ControlNet network online because you can view a graphical representation of your network in RSNetWorx.

Follow these instructions:

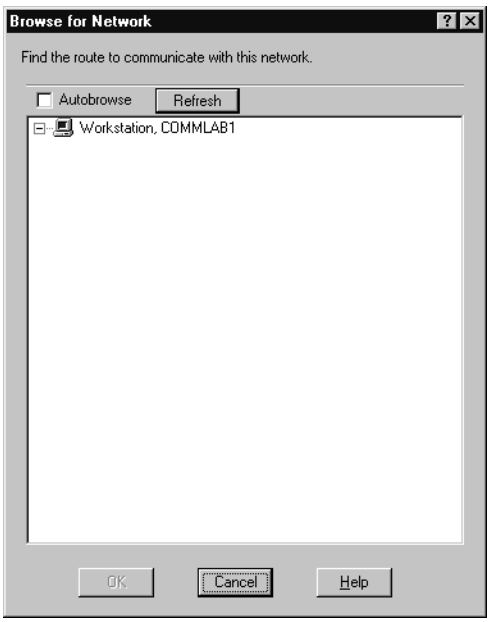
1. Start RSNetWorx. The **RSNetWorx** screen appears.

Figure 4.1
RSNetWorx Screen



2. Click the **Online** box. RSLinx starts in the background to provide a communications interface and then the **Browse for Network** dialog box appears.

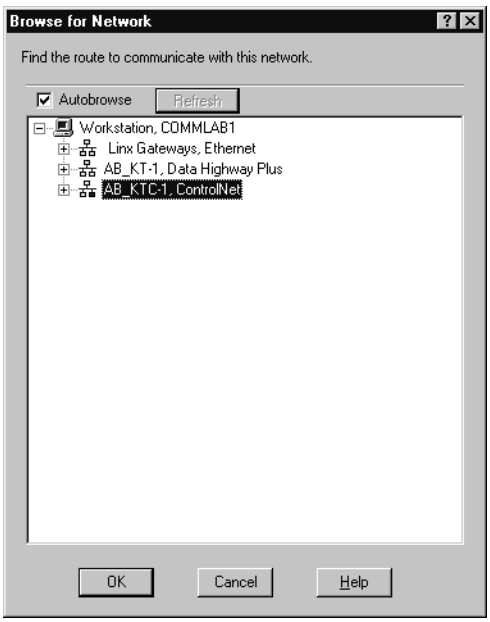
Figure 4.2
Example Browse for Network Dialog Box



3. If available, click the **Autobrowse** box.

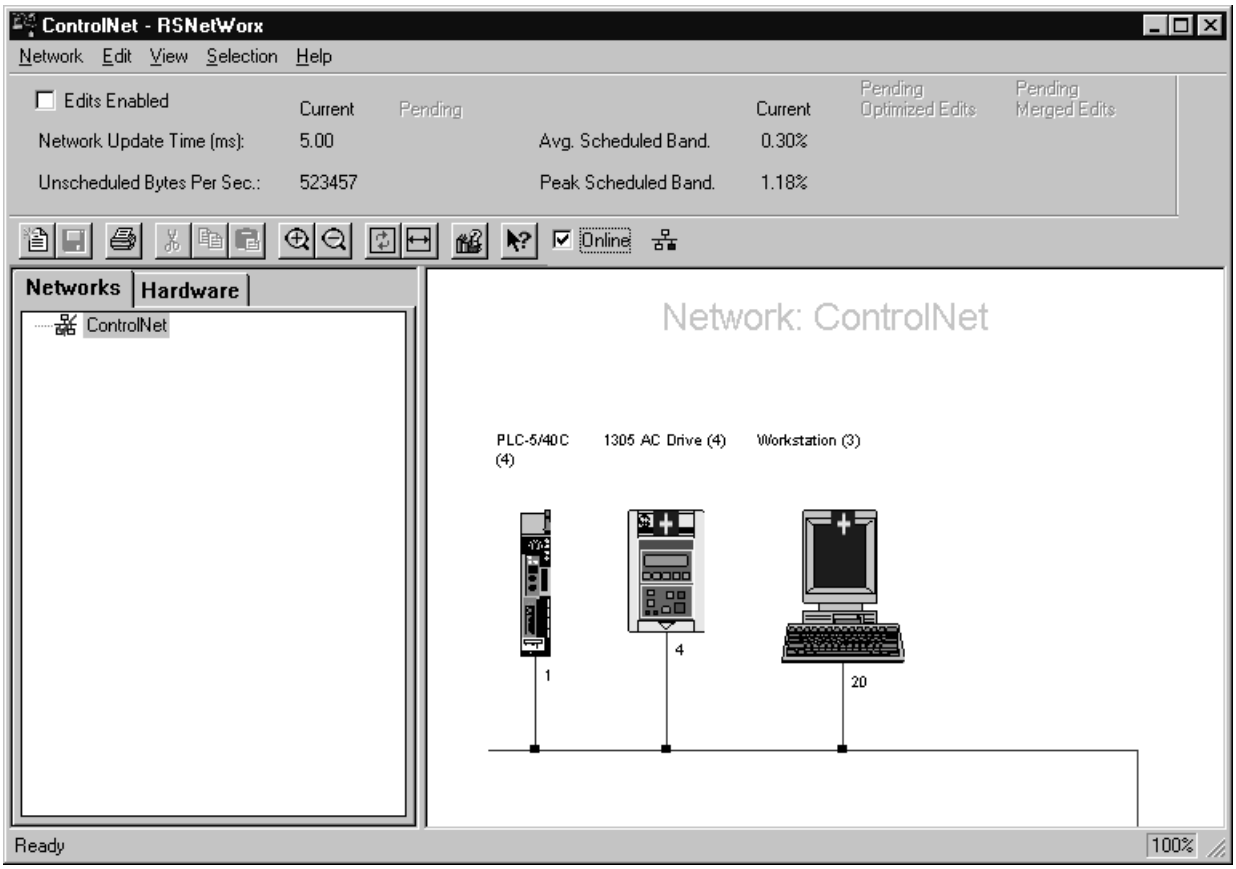
4. If necessary, click the plus sign to view the available networks.

Figure 4.3
Example Browse for Network Dialog Box with Available Networks Displayed



5. Select your ControlNet network and click **OK**. A graphical representation of your network appears.

Figure 4.4
Example Network Displayed in RSNetWorx

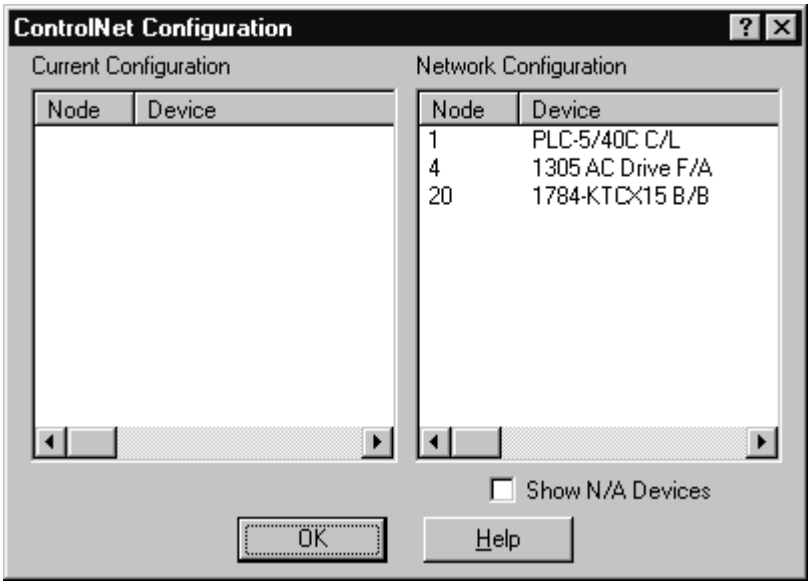


Important: Note the following about our example:

- In Figure 4.4, Node 1 is the controller (PLC-5), Node 4 is the SCANport product to which the 1203-CN1 module is connected, and Node 20 is the PC we are using to configure the network.
- The 1203-CN1 module is transparent on the ControlNet network, so an icon for the SCANport product connected to it is displayed in RSNetWorx. In our example, we used a 1305 drive and set the node address on the 1203-CN1 module to Node 4.
- Above the icons are plus flags that indicate the devices are not configured for the network.

6. In the **Network** menu, select **Actions** and then **Configurations List**. The **ControlNet Configuration** dialog box appears.

Figure 4.5
Example ControlNet Configuration Dialog Box



7. Identify the device(s) that need(s) to be mapped to the network.

   The **Current Configuration** list (blank in Figure 4.5) shows the devices configured on the ControlNet network. The **Network Configuration** list shows devices on the network. Because we are creating a new ControlNet network from scratch in our example, no devices are currently configured. Your network may already have devices configured on it.

   The only device that we will need to map to the network is the 1203-CN1 module. The other devices (PLC-5/40®) and (1784-KTCX) do not need to be configured on the network.

8. Click **OK** to close the dialog box. The RSNetWorx screen reappears.

You are now in online mode and know which device(s) need to be mapped to the network.

Mapping the 1203-CN1 Module to the ControlNet Network

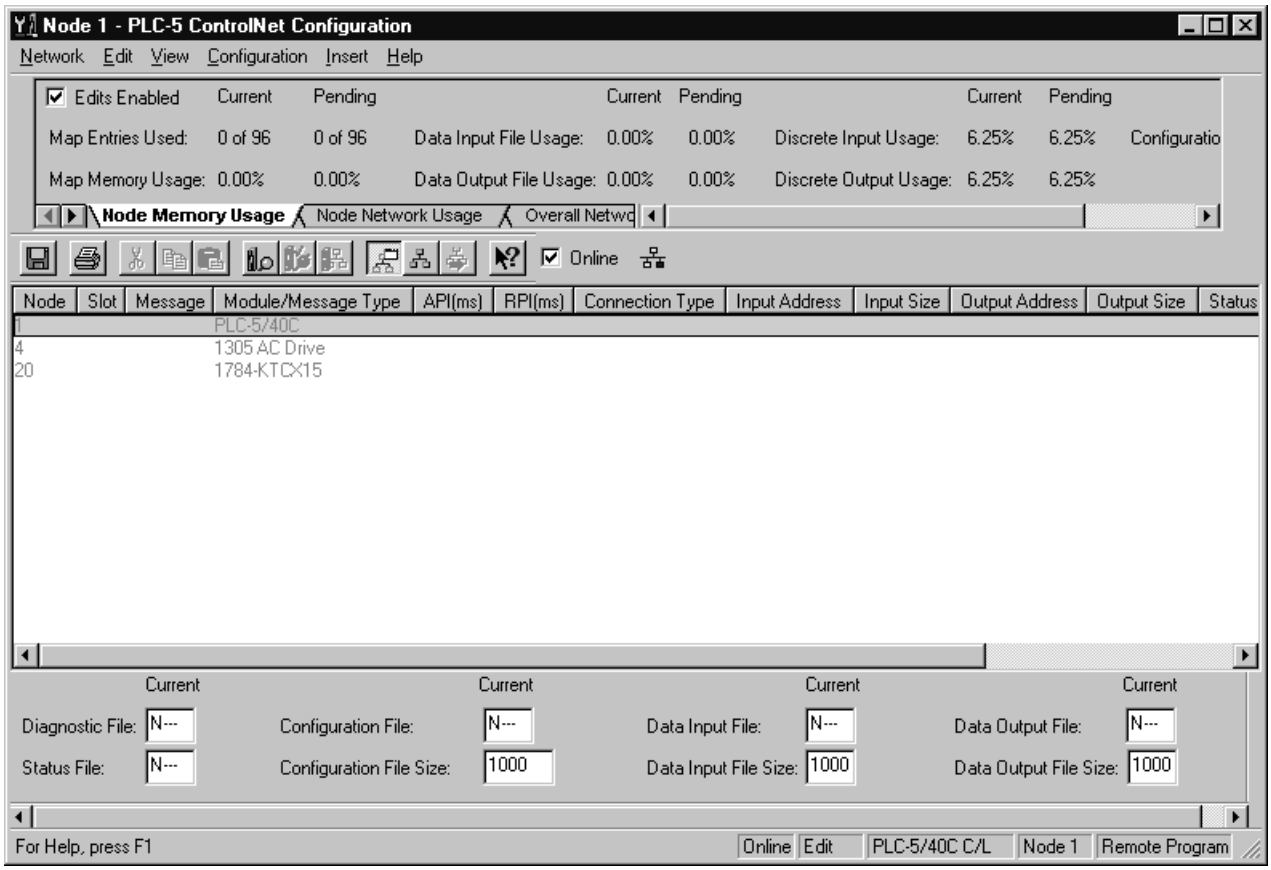
You must configure the 1203-CN1 module on the ControlNet network so that the controller can communicate with it. Follow these instructions:

1. Click the **Edits Enabled** box.

Important: If the **Online/Offline Mismatch** dialog box appears, click **OK** to use the online data. If prompted to save, save the data.

2. Right-click on the controller's icon (PLC-5) and select **ControlNet Configuration**. The **ControlNet Configuration** screen appears.

Figure 4.6
Sample ControlNet Configuration Screen

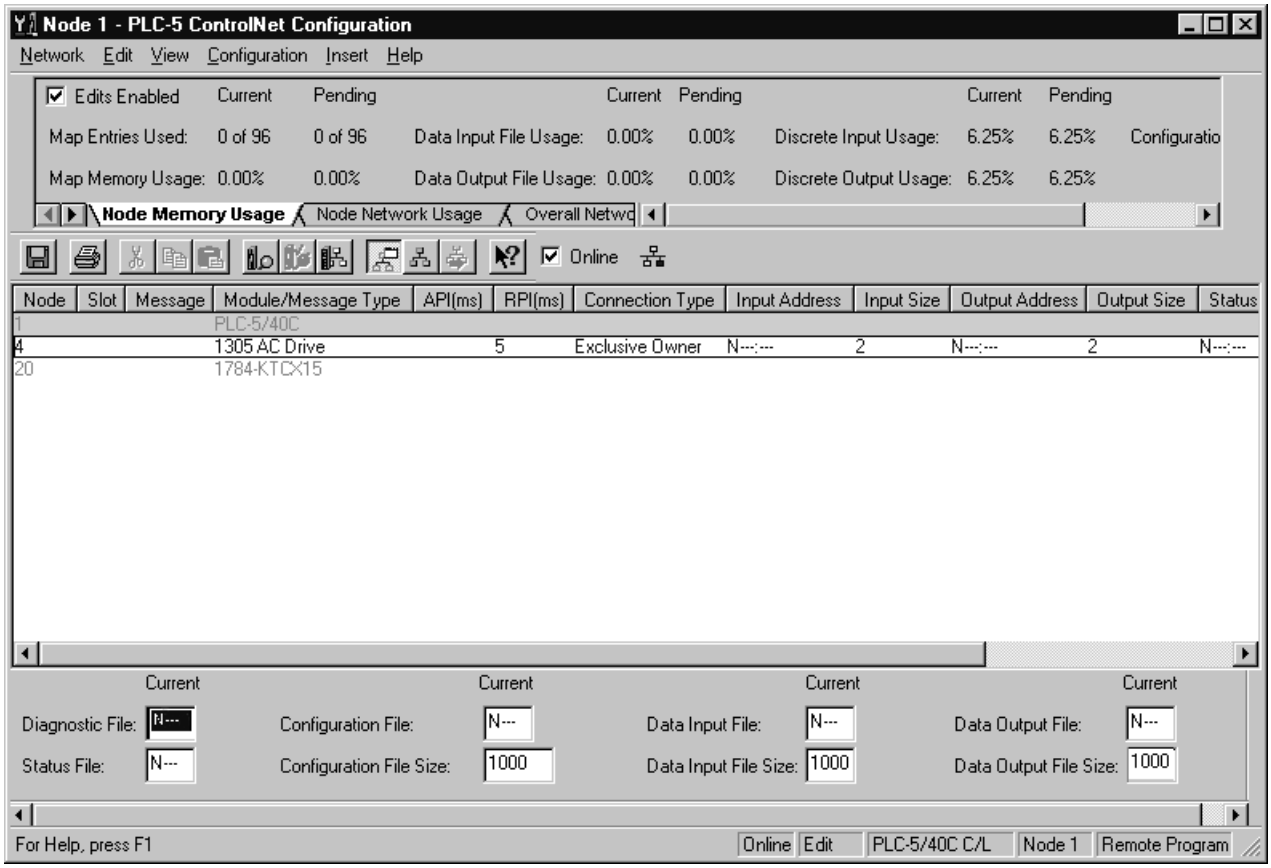


3. Select the line with the SCANport product you want to configure.

Important: You can verify that you've selected the correct line by making sure it has the same node number that you set for the module during the installation.

4. In the **Insert** menu, select **Device Connection**.

Figure 4.7
Example of the Product Line after Drive Connection Is Selected



5. Fill in each of the following fields on the bottom of the screen with unique, valid N-file numbers for the controller you are using. (Each N-file number must be unique.)

- Diagnostics
- Status
- Configuration
- Data Input
- Data Output

6. Note the N-file numbers you filled in the fields. You will need them to develop a ladder logic program.

7. Double-click the **Input Size** field in your product's row and select the appropriate size.

   Valid sizes are 2, 4, 6, 8, and 10. To determine the size, add 2 if Control I/O is enabled and add 2 for each datalink that is enabled. For example, if Control I/O data and all datalinks are enabled, the size would be 10.

8. Double-click the **Output Size** field in your product's row and select the appropriate size. This Output size should be the same value as the Input size.

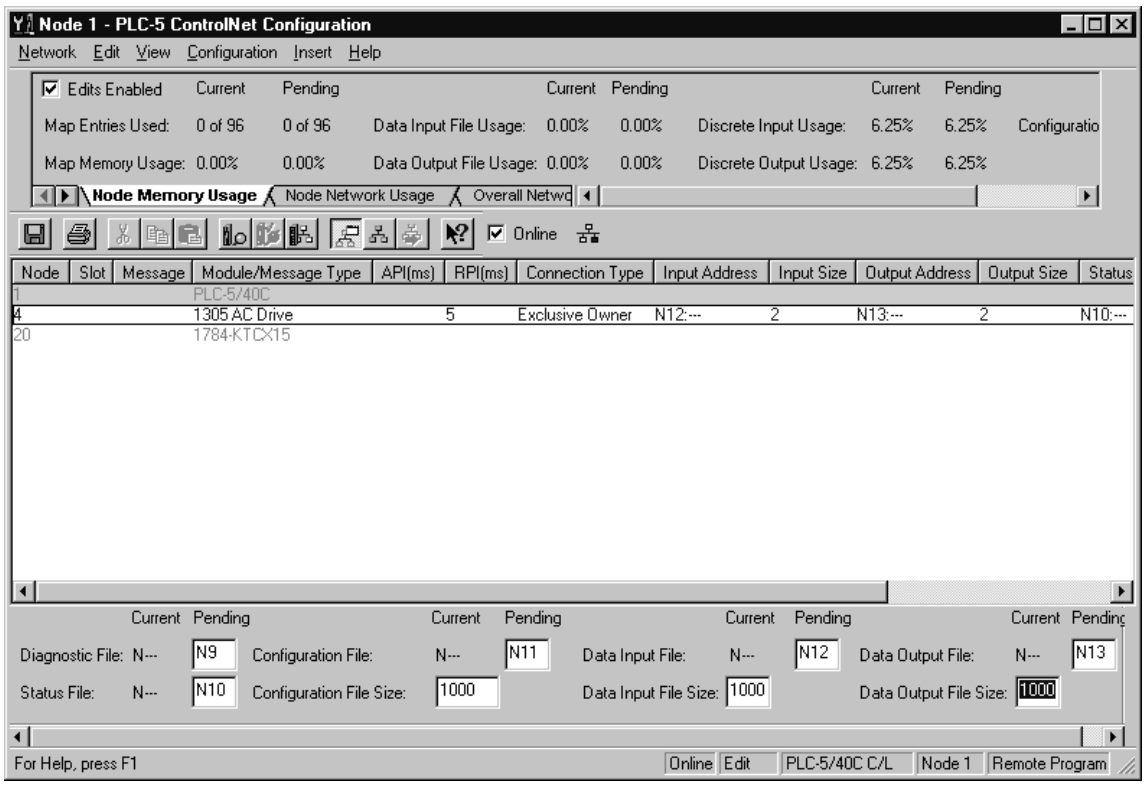
9. Click the **RPI(ms)** field in your product's row and enter the desired time for required packet interval.

Important: Note the value you enter. Ensure that this value is equal to or greater than the value you will set later in the **Network Update Time** field in the **Network Properties: ControlNet** dialog box.

Important: The actual packet interval (API) may vary from the requested packet interval (RPI).

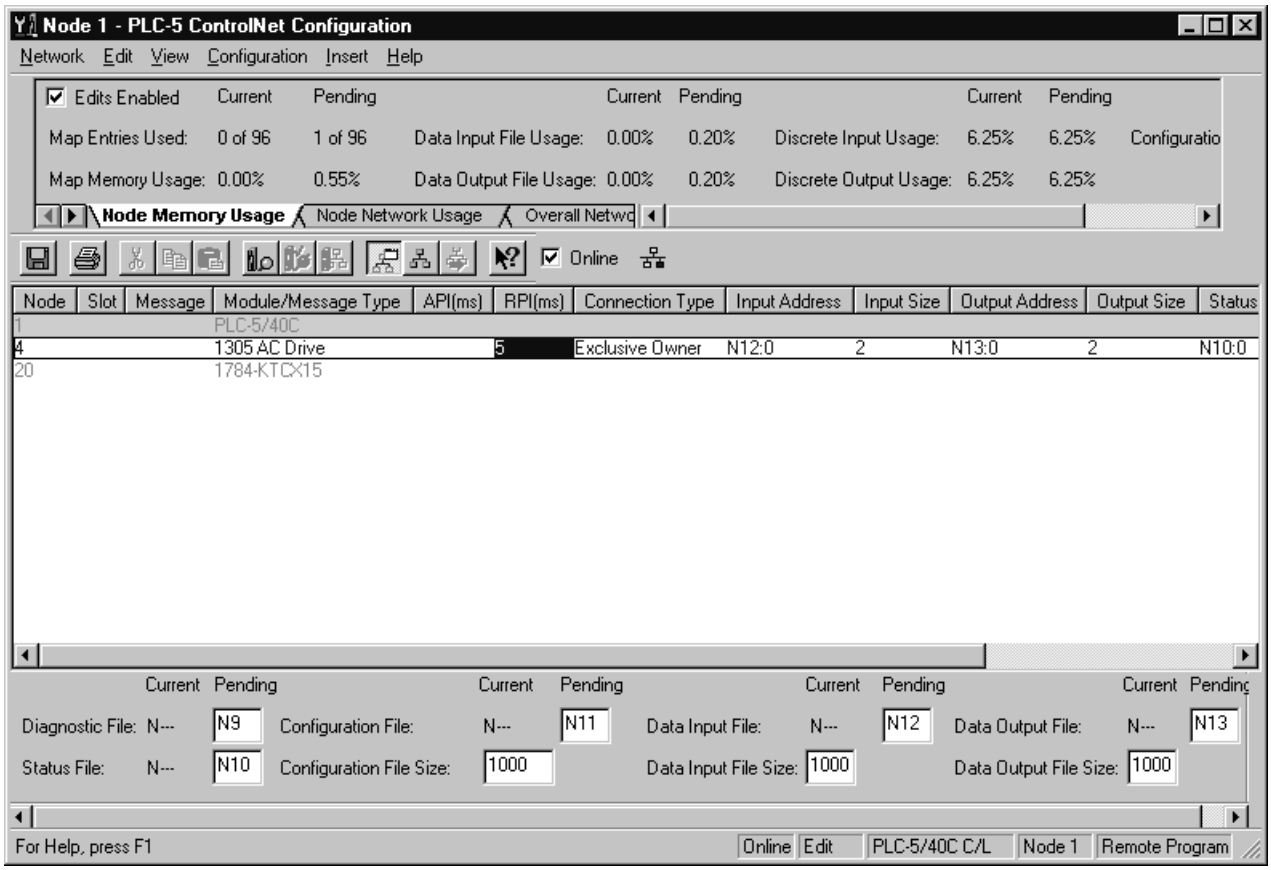
Important: The rate at which a SCANport product reports information to the module varies based on the number of datalinks enabled, the number of peripherals attached to the product, and the type of product.

Figure 4.8
Example Configuration with Size Inputs Set for Control I/O Data Only



10. In the **Configuration** menu, select **Auto Map** and then **All Entries**. Addresses are put in the **Input Address** and **Output Address** columns.

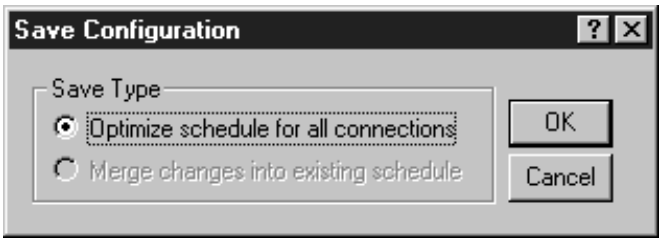
Figure 4.9
Example Configuration



Important: If you want to map the addresses manually, click in the correct field and then enter the address. Make sure you insert a valid address. Refer to the online help for information on manually mapping an address.

11. In the **Network** menu, select **Save**. The **Save Configuration** dialog box appears.

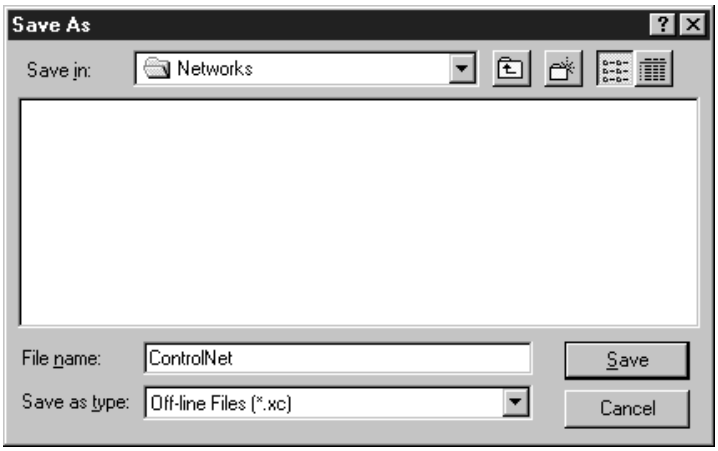
Figure 4.10
Save Configuration Dialog Box



12. Click **OK**. The **Save As** dialog box may appear.

Important: If the **ControlNet Services** dialog box appears with a warning about the MAX scheduled node being set too low, click **OK**. You will correct the problem later while verifying network properties.

Figure 4.11
Save As Dialog Box

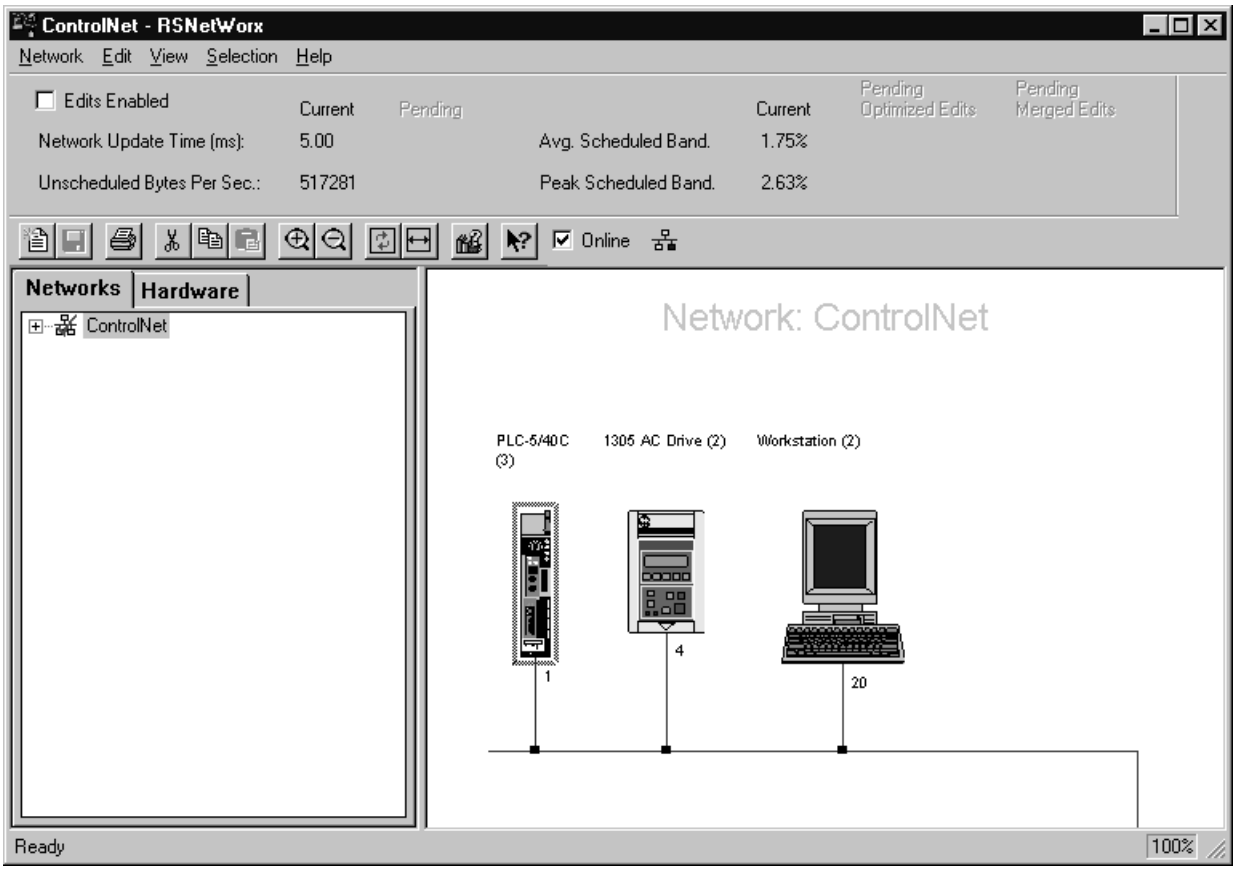


13. In the **File name** field, enter a name for the file and then click **Save**.

Clicking **Save** saves the configuration to a file on the PC and downloads the configuration to the controller on the network.

14. In the **Network** menu, select **Exit**. The **ControlNet Configuration** screen closes and the **RSNetWorx** screen reappears.

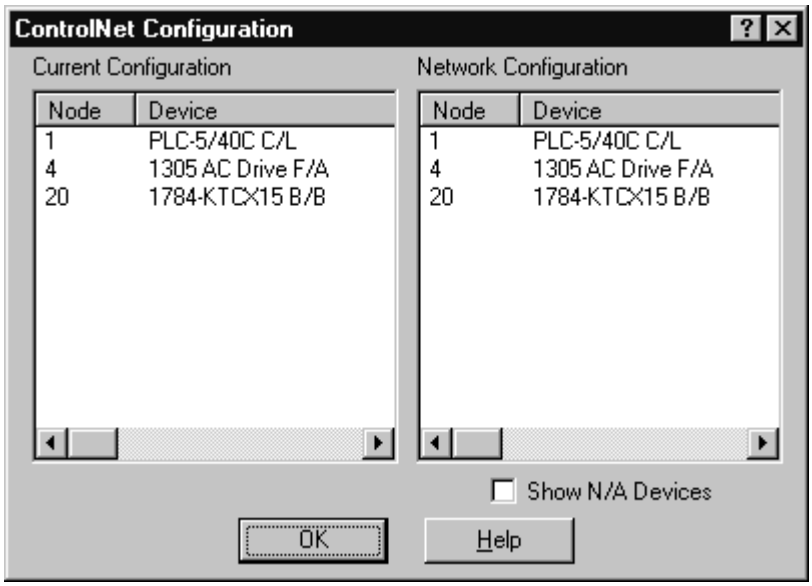
Figure 4.12
Example RSNetWorx Screen with Configured Network



Important: The icon for the SCANport product connected to the 1203-CN1 module (node 4 in Figure 4.12) should appear with a gray flag on it to indicate it is configured. The ≠ sign may appear in the flag to indicate your drive's firmware revision is not matching that in RSNetWorx. If so, proceed.

15. In the **Network** menu, select **Action** and then **Configuration List**. The **ControlNet Configuration** dialog box appears.

Figure 4.13
Example ControlNet Configuration Dialog Box



16. Verify all devices are configured on your network.

Devices configured on the network will appear in the **Current Configuration** list. Figure 4.13 shows all devices are configured on our example network.

17. Click **OK** to close the dialog box.

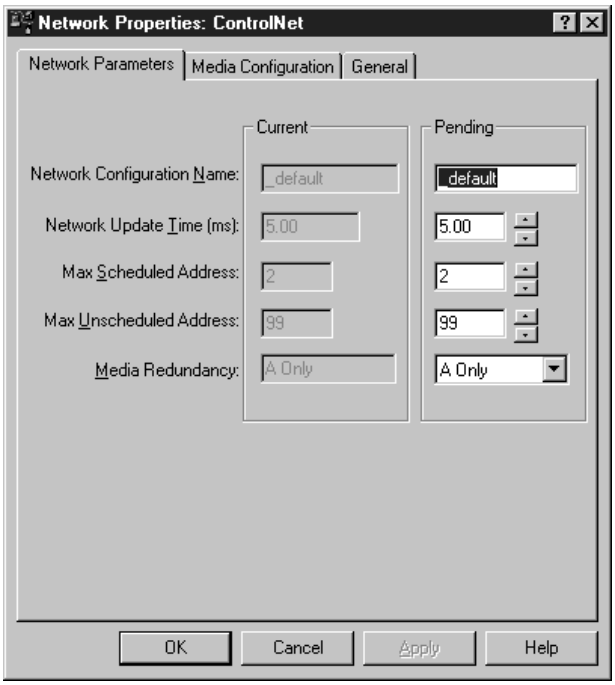
When all devices are mapped to the network, you must verify that network properties are set correctly.

Verifying Network Properties

For your network to function properly, you must set the correct properties. To verify that the correct properties are set, follow these instructions:

1. Click the **Edits Enabled** box.
2. In the **Network menu**, select **Properties**. The **Network Properties: ControlNet** dialog box appears.

Figure 4.14
Network Parameters Tab of the Network Properties: ControlNet Dialog Box



3. In the **Network Parameters** tab, verify:
   - The value in the **Network Update Time(ms)** field is 5 or greater.

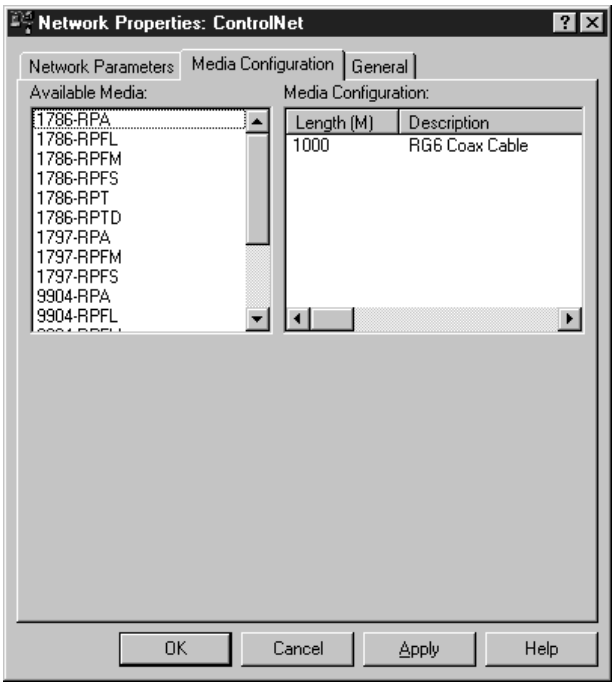
Important: Do not set the Network Update Time lower than 5 ms.

Important: Set the Network Update Time equal to or lower than the RPI times set for the devices on the network.

- The number in the **Max Scheduled Address** field is higher than or equal to the highest node number with which you'll do I/O.
- The number in the **Max Unscheduled Address** field is higher than or equal to the highest node number on the network.
- The correct type of media redundancy is selected in the **Media Redundancy** field.

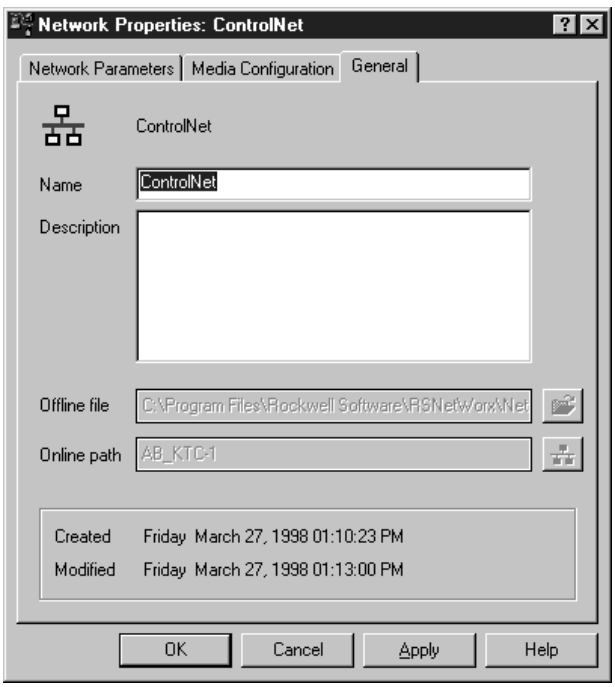
4. If a special media configuration is required (e.g., repeater), select the **Media Configuration** tab and make the appropriate changes. Refer to the online help for more information.

Figure 4.15
Media Configuration Tab



5. If desired, select the **General** tab and enter a name and description for the network.

Figure 4.16
Example General Tab



6. Click **OK**.

7. In the **Network** menu, select **Save** to save the properties and download them to the PLC.

The module is now mapped on the network and the controller will communicate with it. If all mapped devices are present and operational:

- I/O and ControlNet Channel LEDs on the controller will be solid green.
- ControlNet LED(s) on the 1203-CN1 will be solid green.

For information on how to create a PLC ladder logic program to transmit control data to the SCANport product attached to the 1203-CN1 module, refer to Chapter 5, *PLC Ladder Logic Programming*.

Notes:

PLC Ladder Logic Programming

Chapter Objectives

Chapter 5 provides information needed to create the PLC Ladder Logic program that the controller will use to transmit control I/O and messages to and from the SCANport product. This information includes:

- Discussion of PLC ladder logic programs.
- Equipment and software needed to create a PLC Ladder Logic program.
- Example PLC ladder logic program to control the drive.

This chapter assumes you already have experience creating ladder logic programs using RSLogix5.

What Is RSLogix5?

RSLogix5 software lets you create the ladder logic programs you need and download them to the PLC. It also lets you monitor the program as the PLC is using it.

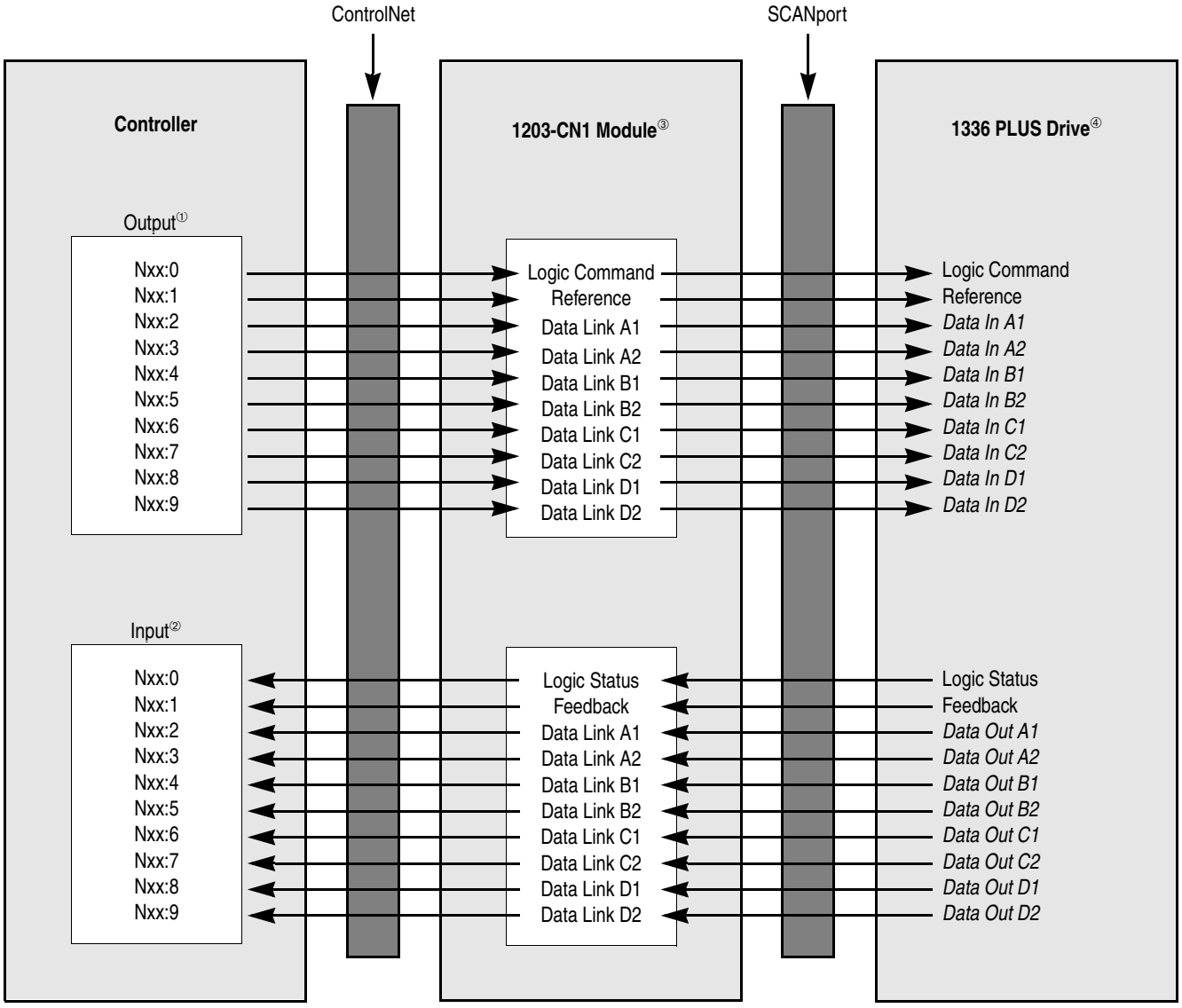
For more information on creating PLC ladder logic programs, consult the documentation for your controller.

For more information on RSLogix5, consult the documentation for RSLogix5.

What Are PLC Ladder Logic Programs?

A PLC ladder logic program lets you control the drive and the messaging from the PLC to the drive. Figure 5.1 shows how the I/O image table for the controller (e.g., PLC) relates to the 1336 PLUS drive when a 1203-CN1 module is used.

Figure 5.1
I/O Image Table



① The N-file number depends on the value entered for the Data Output File field in RSNetWorx. You can also configure this as an I/O address.
② The N-file number depends on the value entered for the Data Input File field in RSNetworx. You can also configure this as an I/O address.
③ You must enable each datalink in the module. Refer to Appendix B, *1203-CN1 Module Parameters*, and Chapter 3, *Getting Started*, for more information.
④ You must configure the datalinks in the SCANport product. Refer to your SCANport product's user manual for more information.

Required Equipment and Software

Before creating a PLC ladder logic program, your PC should be:

- Running RSLogix5 and RSLinx applications. Refer to http://www.software.rockwell.com for more information on these products.
- Connected to and communicating with the ControlNet network using a 1784-KTCX card, 1784-PCC card or 1770-KFC adapter.

Example Ladder Logic Program

The following is an example ladder logic program for a 1305 drive or a 1336 PLUS drive.

ATTENTION: The example ladder logic program shown in this manual is intended solely for purpose of example. Because there are many variables and requirements associated with any particular installation, Rockwell Automation does not assume responsibility or liability (to include intellectual property liability) for actual use based upon the example shown in this publication.

Important: Consult your SCANport product's user manual for I/O patterns. Different SCANport products have different Logic Command Data and Logic Status Data patterns.

The 1305 or 1336 PLUS drive in this example accepts the following Logic Command Data from the PLC.

Logic Status Bits																Function	Description
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
															X	Stop	1=Stop, 0=No Operation
														X		Start	1=Start, 0=No Operation
													X			Jog	1=Jog, 0=No Operation
												X				Clear Faults	1=Clear, 0=No Operation
										X	X					Direction	00=No Operation, 01=Forward, 10=Reverse
									X							Local	1=Local, 0=Multiplexed
								X								MOP Increment	1=Increment MOP, 0=No Operation
						X	X									Accel Rate Select	00=No Operation, 01=Rate 1, 10=Rate 2
				X	X											Decel Rate Select	00=No Operation, 01=Rate 1, 10=Rate 2
	X	X	X													Reference Selection	000=No Operation 001=External Reference 1 (Par 5) 010=External Reference 2 (Par 6) 011=Preset 3 100=Preset 4 101=Preset 5 110=Preset 6 111=Preset 7
X																MOP Decrement	1=Decrement MOP, 0=No Operation

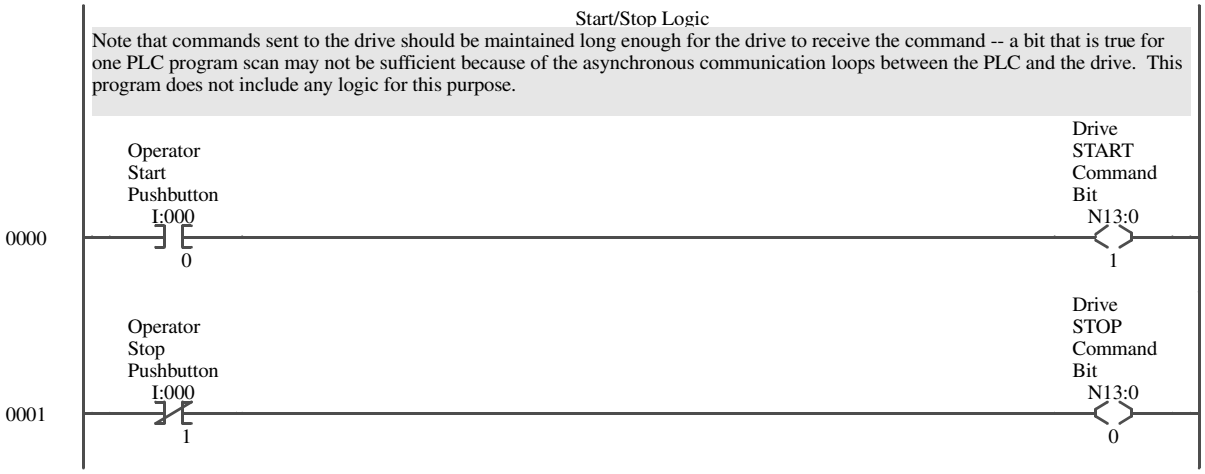
The 1305 or 1336 PLUS drive in this example sends the following Logic Status Data to the PLC.

Logic Status Bits																Function	Description
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
															X	Enabled	1=Enabled, 0=Not Enabled
														X		Running	1=Running, 0=Not Running
													X			Command Direction	1=Forward, 0=Reverse
												X				Rotating Direction	1=Forward, 0=Reverse
											X					Acceleration	1=Accelerating, 0=Not
										X						Deceleration	1=Decelerating, 0=Not
									X							Warning	1=Warning Present, 0=Not
								X								Fault	1=Faulted, 0=Not Faulted
							X									At Reference	1=At Speed, 0=Not At Speed
				X	X	X										Local	000=Terminal I/O has Local 001=Port 1 has Local 010=Port 2 has Local 011=Port 3 has Local 100=Port 4 has Local 101=Port 5 has Local 110=Port 6 has Local 111=Multiplexed Control
X	X	X	X													Reference Source	0000=External Reference 1 0001 – 0111=Presets 1 – 7 1000=External Reference 2 1001 – 1110=Port 1 – 6 Direction 1111=Jog

This example program for the 1305 or 1336 PLUS drive provides basic control functions, including starting, stopping, clearing faults, displaying drive status, controlling drive frequency reference, and displaying drive frequency feedback. Two words of control I/O have been enabled for this program by accepting the module's default configuration and by entering 2 for the Input Size and Output Size in RSNetWorx.

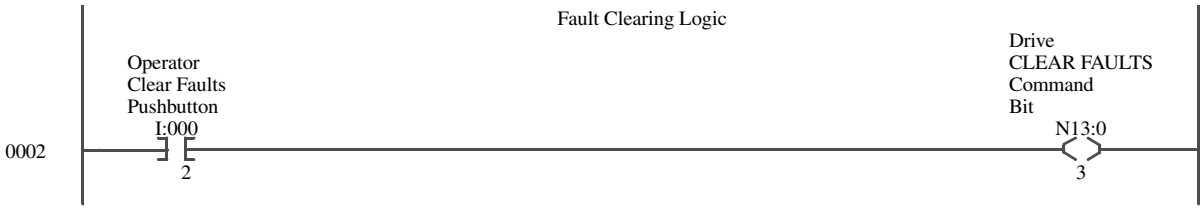
The portion of the program shown in Figure 5.2 controls the starting and stopping of the drive. When the start button is pushed, the drive will start. When the stop button is pushed, the drive will stop.

Figure 5.2
Start/Stop Logic



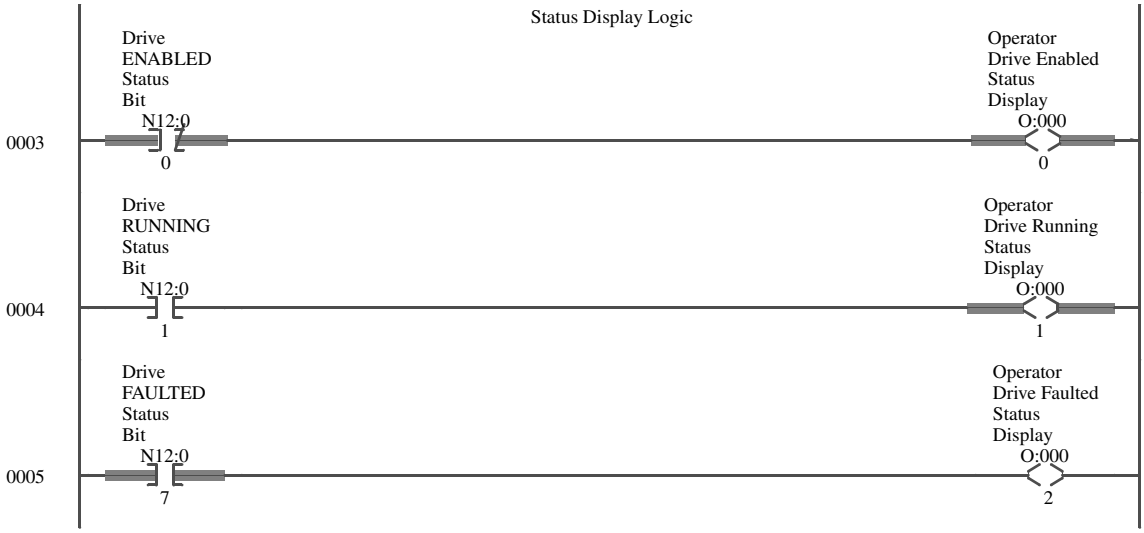
The portion of the program shown in Figure 5.3 allows the operator to clear faults in the drive by pushing a Clear Faults button.

Figure 5.3
Clear Faults Logic



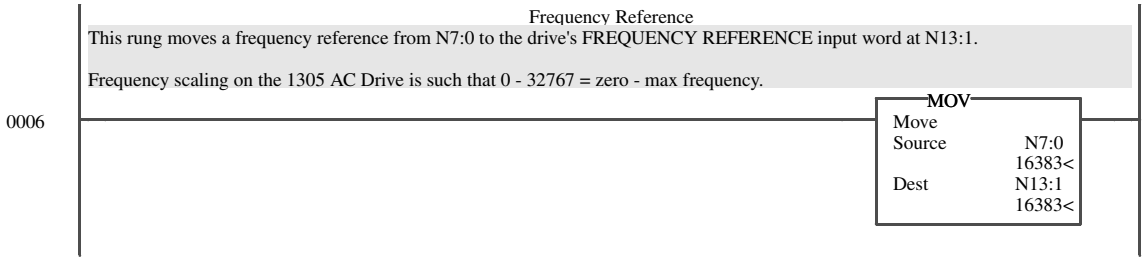
The portion of the program shown in Figure 5.4 displays the status of the drive by reading the status information in the drive's logic status word and displaying it at the operator's station.

Figure 5.4
Drive Status Display logic



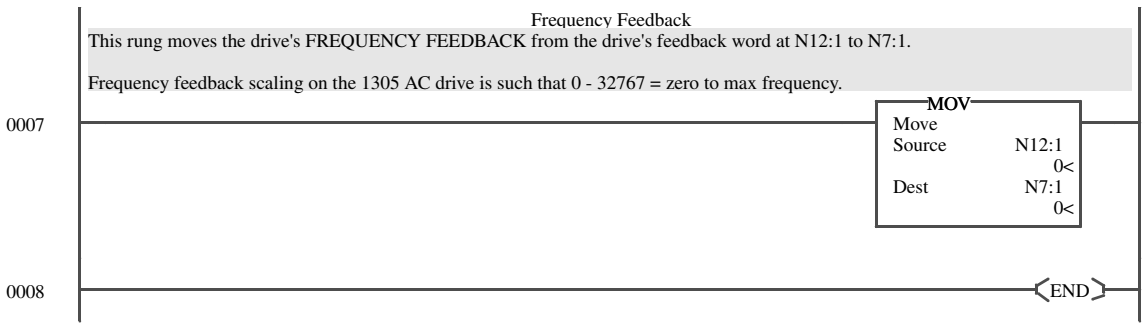
The portion of the program shown in Figure 5.5 moves a Frequency Reference to the drive. The word is scaled 0 to 32767 equals 0 to maximum frequency. Scaling may differ for other products.

Figure 5.5
Drive Frequency Reference



The portion of the program shown in Figure 5.6 moves the drive's Frequency Feedback from the drive's feedback word to the PLC. The word is scaled 0 to 32767 equals 0 to maximum frequency. This scaling may differ on other SCANport products.

Figure 5.6
Drive Frequency Feedback Display



Using Messages

Chapter Objectives

Chapter 6 provides information about using messages on the ControlNet network. It includes the following:

- Required equipment and software.
- Discussion of messaging.
- Examples of rungs that enable messaging.

This chapter assumes you already have experience creating ladder logic programs and using messages in RSLogix5.

Required Equipment and Software

Before adding messages to a PLC ladder logic program, your PC should be:

- Running RSLogix5 and RSLinx applications. Refer to http://www.software.rockwell.com for more information on these products.
- Connected to and communicating with the ControlNet network using a 1784-KTCX card, 1784-PCC card or 1770-KFC adapter.

Using Messages

Messaging lets you view and modify SCANport product parameters as well as provides access to other internal SCANport-related information and services.

To use messaging, you need to:

1. Insert a MSG rung in your ladder logic program. A unique message number must be used for each message.
2. Insert a control word in the MSG rung.
3. Configure the message using the **MSG** dialog box.

   The following must be completed in the **This PLC-5** section:

   - In the **Communication Command** field, use the drop-down list to specify whether the MSG instruction performs a read or write operation and what type of message is used.
   - In the **Data Table Address** field, enter the data table address containing the message in the controller (e.g., PLC-5).
   - In the **Size in Elements** field, enter the number of elements being transferred.
   - In the **Port Number** field, enter 2 to enable the ControlNet dialog parameters.

The following must be completed in the **Target Device** section:

- In the **Data Table Address** field, enter the starting address of the source or the destination file in the 1203-CN1 module. Refer to Appendix C, *N-File Structure*, to see what data is contained in each N-file.
- In the **ControlNet Path** field, enter the node address for the 1203-CN1 module.

Important: For more information, refer to the RSLogix5 online help.

4. Read the message by viewing the N-file set in the **Data Table Address** field in the **PLC-5** section.

Examples

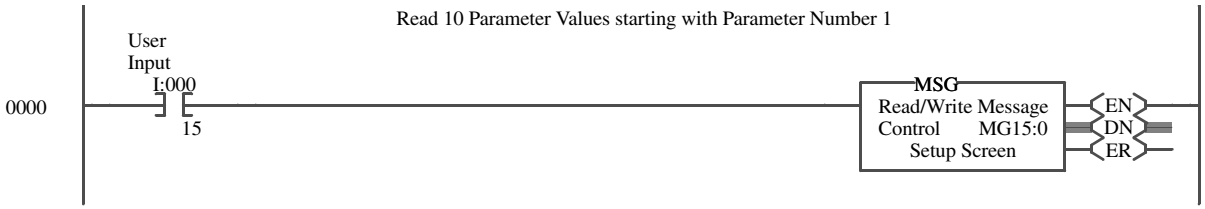
The following examples show the addition of message rungs to the example ladder logic program for the 1305 or 1336 PLUS drive shown in Chapter 5, *PLC Ladder Logic Programming*. For each type of message, the following are shown:

- Rung(s) in the ladder logic program.
- Configuration in the **MSG** dialog box.
- Example results.

Example PLC-5 Typed Read of 10 Parameter Values

Figure 6.1 contains an example of a PCCC message. This message allows the PLC to read ten parameters from the drive, beginning with parameter 1.

Figure 6.1
Read Parameters Logic



In the **MSG** dialog box (Figure 6.2), the configuration for the message is specified. Note the following:

- In the **Data Table Address** field in the **This PLC-5** section, **N20:0** is specified. This is where the results will be reported.
- In the **Size in Elements** field in the **This PLC-5** section, **10** is specified. The response will be 10 elements.
- In the **Data Table Address** field in the **Target Device** section, N10:1 is specified. This is the location where the first parameter value is found.

Important: If you refer to Appendix C, *N-File Structure*, you will find that SCANport product parameter values are contained in the N10:1 – N12:999.

Important: To view the 1203-CN1 module's parameters, you would specify N13:1 instead of N10:1.

Figure 6.2
Read Parameters Logic.

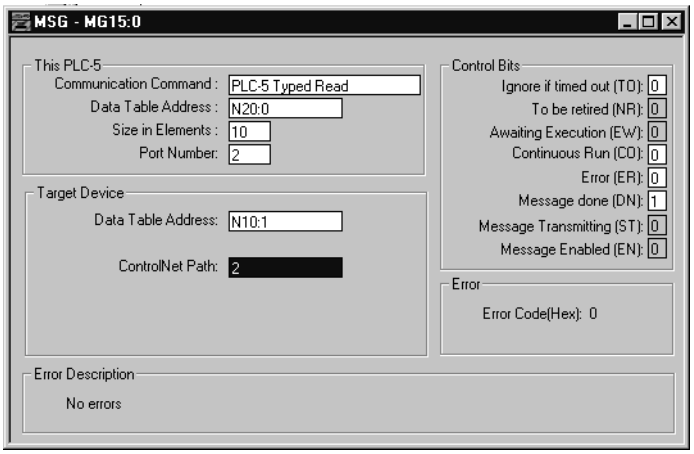
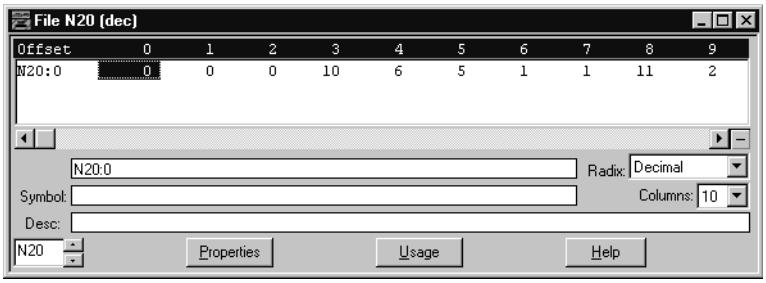


Figure 6.3 contains the values read by this example rung. The zero element contains the value of parameter 1, the one element contains the value of parameter 2, etc.

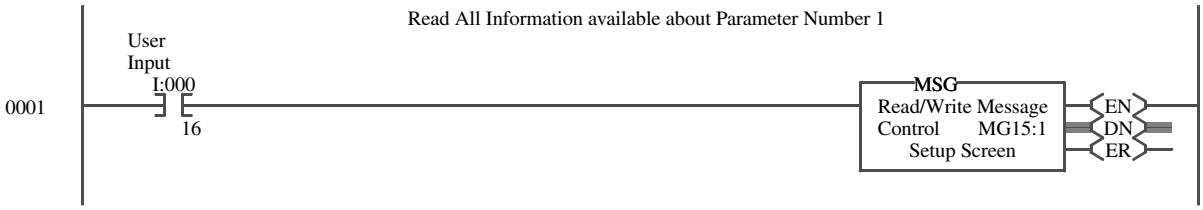
Figure 6.3
Example Results in Decimal Text



Example PLC-5 Typed Read of All Information about a Parameter

Figure 6.4 contains an example of a PCCC message. This message allows the PLC to read all information about parameter 1 in the drive.

Figure 6.4
Example Full Read Rung



In the **MSG** dialog box (Figure 6.5), the configuration for the message is specified. Note the following:

- In the **Data Table Address** field in the **This PLC-5** section, **N21:0** is specified. This is where the results will be reported.
- In the **Size in Elements** field in the **This PLC-5** section, **20** is specified. The response will be 20 elements.

Important: A read of all information about a parameter requires 20 elements. You can read all information from one to five parameters at one time. For example, you can read five parameters at one time by specifying the size as 100.

- In the **Data Table Address** field in the **Target Device** section, N30:1 is specified. This is the location where the parameter's value is found.

Important: If you refer to Appendix C, *N-File Structure*, you will find that SCANport product parameter full/all information read data is contained in N30:1 – N32:999.

Important: Parameters for the 1203-CN1 module start at N33:1.

Figure 6.5
Example MSG Dialog Box

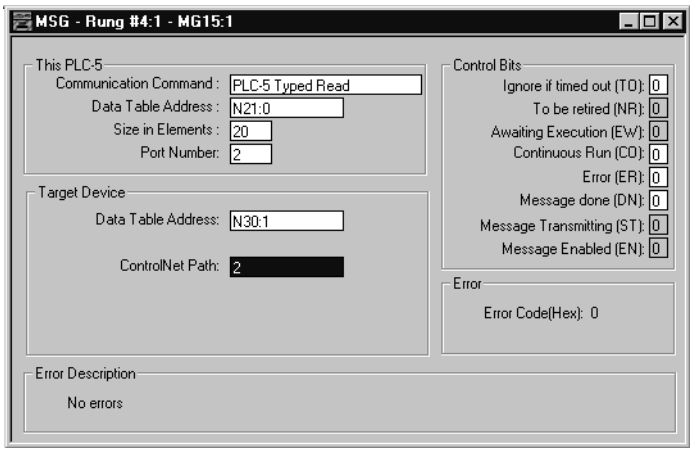
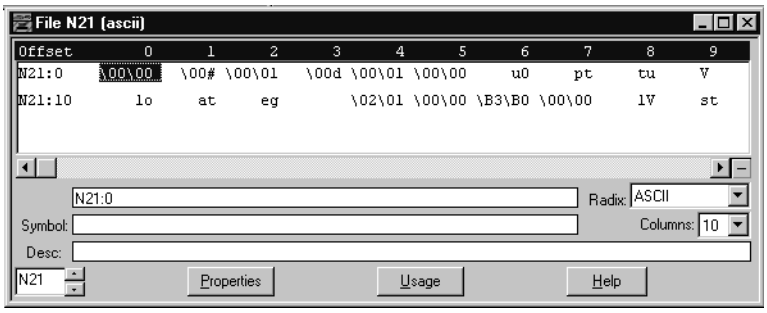


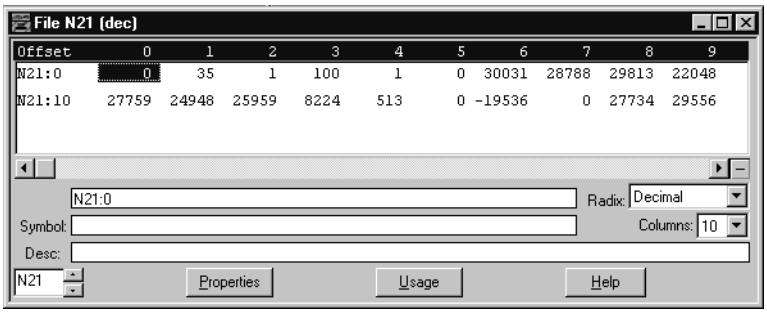
Figure 6.6 and Figure 6.7 display the same results in ASCII and decimal text, respectively.

Figure 6.6
Example Results in ASCII Text



Important: In Figure 6.6, elements 6 – 12 spell out "Output Voltage." The text in each bit is reversed (e.g., element 6 is "uO" instead of "Ou") because of the method used to display ASCII text in an N-file.

Figure 6.7
Example Results in Decimal Code

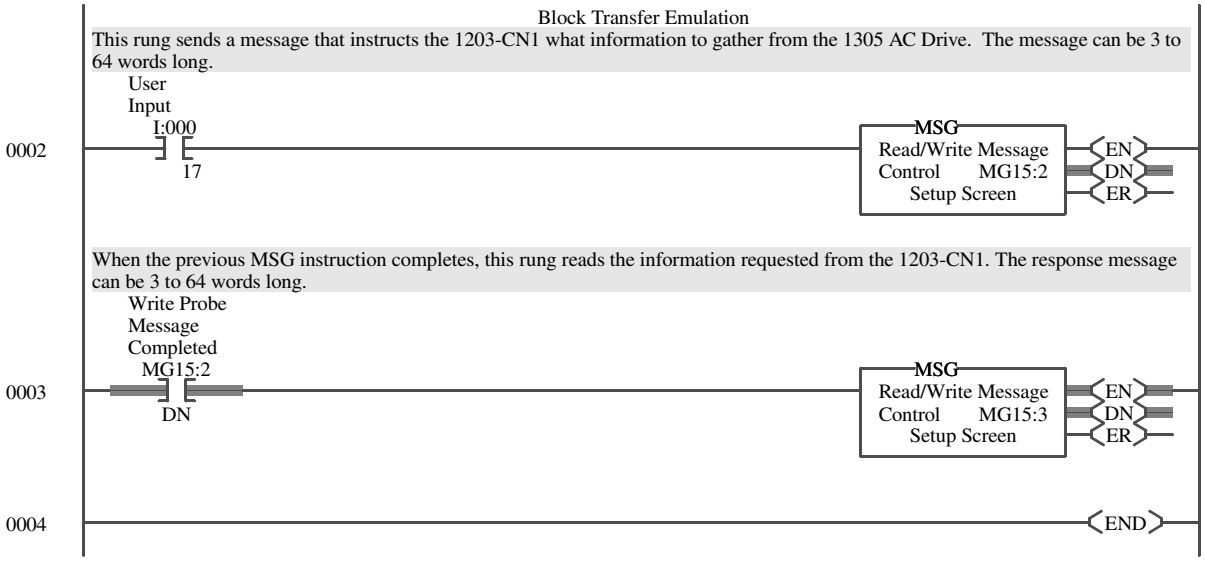


Example Fault Queue Read Emulated Block Transfer

Figure 6.8 contains an example of an emulated block transfer. In emulated block transfers, the first rung always writes the data from the drive to the N:40 file in the module; the second rung then reads the data from this file.

Important: You can activate only one emulated block transfer message at a time. If more than one message is active, data may be processed out of order and inaccurate results may be supplied.

Figure 6.8
Read Fault Queue Entry #1 Logic

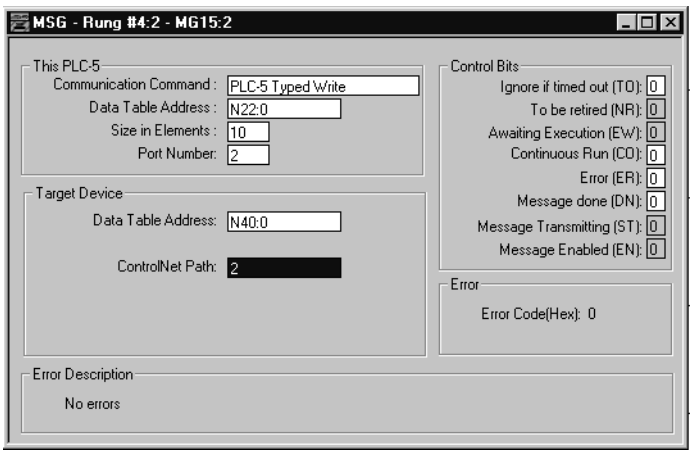


The **MSG** dialog box in Figure 6.9 specifies the configuration for the Write Probe message. Note the following:

- In the **Data Table Address** field in the **This PLC-5** section, **N22:0** is specified. This is where the results will be reported.
- In the **Size in Elements** field in the **This PLC-5** section, **10** is specified. The response will be 10 elements.
- In the **Data Table Address** field in the **Target Device** section, N40:0 is specified. This is the location where the fault queue data will be stored.

Important: If you refer to Appendix C, *N-File Structure*, you will find that block transfer emulation data is contained in N40:0 – N40:63.

Figure 6.9
MG15:1 Message Control Block Setup



The **MSG** dialog box in Figure 6.10 specifies the configuration for the Read Response message.

- In the **Data Table Address** field in the **This PLC-5** section, **N23:0** is specified. This is where the results will be reported.
- In the **Size in Elements** field in the **This PLC-5** section, **20** is specified. The response will be 20 elements.
- In the **Data Table Address** field in the **Target Device** section, N40:0 is specified. This is the location where the fault queue data is stored after the previous message block executes.

Important: If you refer to Appendix C, *N-File Structure*, you will find that block transfer emulation data is contained in N40:0 – N40:63.

Figure 6.10
MG15:2 Message Control Block Setup

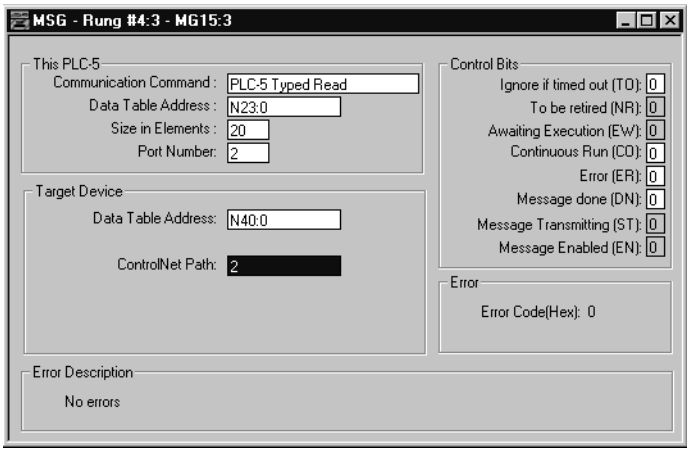


Figure 6.11 shows the results of the typed write command.

Figure 6.11
Example Results of the Typed Write

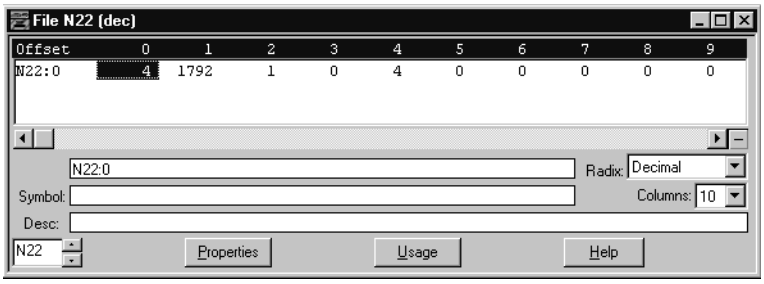
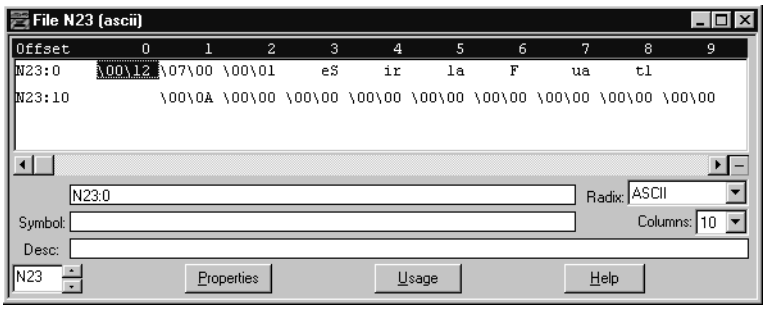


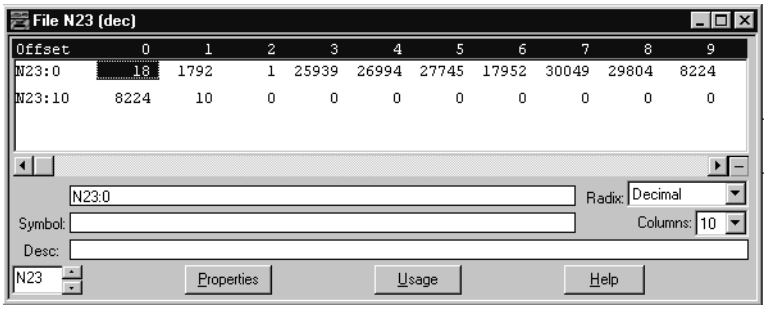
Figure 6.12 and Figure 6.13 display the same results of the typed read command in ASCII and decimal text, respectively.

Figure 6.12
Example Results in ASCII Text



Important: In Figure 6.12, elements 3 – 8 spell out "Serial Fault." The text in each element is reversed (e.g., element 3 is "eS" instead of "Se") because of the method used to display ASCII text in an N-file.

Figure 6.13
Example Results in Decimal Text



More Information on Emulated Block Transfers

The same ladder logic used to read the fault queue can also be used to obtain other information from the drive and send various commands to the drive. For more information on emulated block transfers, refer to Appendix F, *Supported Emulated Block Transfer Commands*.

Troubleshooting

Chapter Objectives

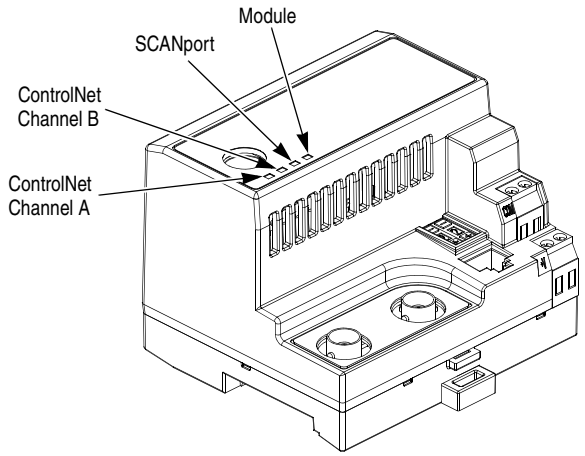
Chapter 7 provides information about the LEDs and basic troubleshooting procedures. This chapter covers the following topics:

- Locating the LEDs.
- Using the LEDs to troubleshoot.

LEDs on the 1203-CN1 Module

Your 1203-CN1 module has four LED status indicators. The LEDs provide status information about ControlNet channel(s), SCANport, and the module itself. Refer to Figure 7.1.

Figure 7.1
LED Status Indicators



Understanding the ControlNet LEDs

When viewing the ControlNet LEDs, the color and status (solid or flashing) of the LEDs are significant. In addition, the ControlNet Channel A and Channel B LEDs may need to be viewed together or independently, depending on their state. Refer to this table.

LEDs Viewed:	If:	Cause:	Action:
ControlNet A and B Together	Both LEDs are off.	A reset has occurred or there is no power.	Apply power to the module.
	Both LEDs are steady red.	A link interface failed.	1. Check media for broken cables, loose connectors, missing terminators, etc. 2. Power cycle or reset the module. If problem persists, contact Rockwell Automation support.
	LEDs are alternating red/green.	The module is in self-test mode.	None. The module will exit this mode when the self-test is completed.
	LEDs are alternating red/off.	There is a bad node configuration.	1. Verify that all node addresses are unique. 2. Check the module's ControlNet configuration parameter. 3. Check media for broken cables, loose connectors, missing terminators, etc. 4. Power cycle or reset the module.
ControlNet A or B Independently	One channel LED is steady off.	That channel is disabled or not supported.	Program the network for redundant media, if required.
	One channel LED is flashing red/green.	There is an invalid link configuration for that channel.	1. Power cycle or reset the module. 2. Reset the controller. If problem persists, contact Rockwell Automation support.
	One channel LED is flashing red/off.	There is a link fault or no frames received for that channel.	1. Check media for broken cables, loose connectors, missing terminators, etc. 2. Add other nodes to the network.
	One channel LED is flashing green/off.	A temporary channel error has occurred, or the channel is in listen-only mode.	Verify the controller is present on the network and working.
	One channel LED is steady green.	Normal operation for that channel.	None.

Understanding the SCANport LED

When viewing the SCANport LED, the color and status (solid or flashing) of the LED are significant. Refer to this table.

LEDs Viewed:	If:	Cause:	Action:
SCANport	Solid red	Either the SCANport interface is not connected to a SCANport product, or it is connected but for some reason has not detected a valid SCANport ID.	1. Verify the SCANport product is powered. 2. Check media for broken cables, loose connectors, etc.
	Flashing red	The SCANport interface has detected an error, such as no communications from the SCANport product or a datalink is configured in the module and your product does not support it or another adapter is using it.	1. Temporarily disconnect the cable to the SCANport product and then reconnect it. 2. Verify the module's configuration works with the product's configuration. 3. Power cycle or reset the module or the SCANport product.
	Solid orange	A SCANport compatibility problem exists.	Contact Rockwell Automation support.
	Solid green	The SCANport interface is fully functional for both messaging and control I/O. The adapter is communicating normally with the connected SCANport product.	None
	Flashing green	A valid SCANport ID has been detected and messages are being received from the SCANport product. The module has not been configured for control I/O messages to the SCANport product. This indicates that the SCANport interface is in a messaging-only mode and no I/O enables are present in the current configuration.	If control I/O communication is desired, configure the module's parameters to enable control I/O.

Understanding the Module LED

When viewing the Module LED, the color, status (solid or flashing), and flash pattern (number of flashes before a pause) are significant. Refer to this table.

LEDs Viewed:	If:	Cause:	Action:
Module	Off	The adapter is not receiving power.	Check that the power supply is functional and apply power.
	Solid red	The hardware or firmware is bad.	1. Ensure all connections are made properly. 2. Power cycle or reset the module. 3. Perform a flash upgrade to the module. 4. Contact Rockwell Automation customer service or replace the module.
	Flashing red (1 flash)	A flash update is in progress or the firmware has detected an error.	
	Flashing red (2 flashes)	RAM test failed.	
	Flashing red (3 flashes)	Non-Volatile Storage Cyclic Redundancy Check (CRC) failed.	1. Verify all parameters are correct. 2. Change and save at least one parameter.
	Flashing red (4 flashes)	Boot Code CRC failed.	1. Ensure all connections are made properly. 2. Power cycle or reset the module. 3. Perform a flash upgrade to the module. 4. Contact Rockwell Automation customer service or replace the module.
	Flashing red (5 flashes)	ASIC Code CRC failed.	
	Flashing red (6 flashes)	Application Code CRC failed.	
	Flashing red (7 flashes)	ControlNet chip self-test failed.	
	Flashing red (8 flashes)	SCANport Controller Test failed.	
	Flashing red (9 flashes)	Factory diagnosis mode.	
	Solid green	No failures have occurred during the diagnostics mode testing or since the module has been running.	None.
	Flashing green	The adapter is configured for I/O, but no ControlNet I/O connection has been made.	Configure the PLC to communicate with the module. Refer to Chapter 4.

Specifications

Appendix Objectives

Appendix A provides the specifications that you may need to install, repair, or use your 1203-CN1 ControlNet communications module.

Specifications

The following table gives the specifications for the 1203-CN1 ControlNet communications module.

Category	Specifications
Dimensions	3.92 x 2.69 x 3.54 in (99.5 x 68.4 x 90.0 mm)
Weight	0.52 lbs. (236.7 g)
Operating Temperature	0 to +55°C (32 to 131°F)
Storage Temperature	-40 to +85°C (-40 to 185°F)
Relative Humidity (Operating)	5 to 80% non-condensing
Relative Humidity (Non-Operating)	5 to 95% non-condensing
Shock (Operating)	30g peak acceleration, 11(+/-1)ms pulse width
Shock (Non-Operating)	50g peak acceleration, 11(+/-1)ms pulse width
Vibration (Operating)	2.5g @5Hz-2KHz
Vibration (Non-Operating)	5g @5Hz-2KHz
Power Consumption	250mA at 24V DC (-20% / +30%)
ESD Susceptibility (IEC 1000-4-2)	6KV contact, 8KV open air
Regulatory Agencies	UL 508 and CUL European Union EMC and Low Voltage Directives
DIN Rail Mounting Standard	1.38 x 0.30 in. (35 x 7.5 mm)

ATTENTION: The 1203-CN1 ControlNet communications module contains ESD (Electrostatic Discharge) sensitive parts. Static control precautions are required when installing, testing, or servicing this assembly. Device malfunction may occur if you do not follow ESD control procedures. If you are not familiar with static control procedures, refer to Allen-Bradley publication 8000-4.5.2, Guarding Against Electrostatic Damage, or other applicable ESD protection handbook.

Important: To meet EMC Emission (Class A, Group 1) requirements, you must place the module in a full metal enclosure. It is the end user's responsibility to choose a full metal enclosure that will provide adequate emission protection.

Notes:

1203-CN1 Module Parameters

Appendix Objectives

Appendix B provides information on the 1203-CN1 SCANport to ControlNet module's parameters. The information includes:

- Discussion of datalinks.
- Discussion of Fault Configurable inputs.
- List of parameters.

What Are Datalinks?

Datalinks let you increase the scheduled I/O values to and from a SCANport device (provided the SCANport device supports datalinks). By enabling datalinks, you can continuously change or monitor the value of a parameter without using the ControlNet to SCANport messaging function.

Each datalink consists of two 16-bit words of input and two 16-bit words of output when enabled. Up to eight words in and eight words out of data are available if supported in the connected SCANport device.

SCANport devices that support this function have a group of parameters for datalink configuration. These parameters are *Data In A1 – D2* and *Data Out A1 – D2*.

If you intend to use datalinks, you must do the following:

1. Enable desired *DataLink* parameters within the 1203-CN1 module. Refer to Chapter 3, *Getting Started*, for instructions on editing parameters.
2. Configure or link the *Data In A1 – D2* and *Data Out A1 – D2* parameters in the SCANport device. Refer to the documentation for your SCANport product.
3. Set the Input Size and Output Size to the correct values in RSNetWorx. Refer to Chapter 4, *Configuring a Controller to Communicate with the 1203-CN1 Module*, for instructions on setting these.

What Are Fault Configurable Inputs?

You can select constant values that your 1203-CN1 module will maintain in the event of a controller mode change or error. These constant values are referred to as Fault Configurable inputs. When the controller is placed in program mode or a ControlNet network fault occurs, the control outputs from the module to the SCANport product can be set to automatically switch to the constant values set in the *Fault Cfg In* parameters. This lets you define a safe operating state for

controlled devices that depend on pre-programmed output from the module.

ATTENTION: Risk of severe bodily injury or loss of life exists. The *Idle Flt Action* (9) and *Comm Flt Action* (10) parameters allow the user to change the default configuration that would allow the module and associated drive to continue to operate if communication is lost. Precautions should be taken to assure that your settings for these parameters and your application do not create a hazard of bodily injury or loss of life.

If you intend to use Fault Configurable inputs, you must do the following:

1. Set desired values for the *Fault Cfg Logic*, *Fault Cfg Ref*, and *Fault Cfg In* parameters.
2. Set the *Idle Fault Config* parameter and/or the *Comm Flt Action* parameter to **Fault Cfg**.

Refer to Chapter 3, *Getting Started*, for instructions on editing parameters.

Parameters

The following table provides information on the parameters for the 1203-CN1 module.

#	Name	Valid Values/ Settings	Default	Writable	Reset or Power Cycle to Take Effect	Description
1	*SCANport Adapter*	0 – 7	0	No	No	Identifies the port number to which the module is connected on the SCANport product.
2	*CN Node Address*	0 – 99	0	No	Yes	Provides the ControlNet node address for the 1203-CN1 module.
3	*CN Data Rate*	0 = 5 Mbps	0	No	No	Speed of ControlNet.
4	*Cmd/Stat*	0 = Disabled 1 = Enabled	1=Enabled	Yes	Yes	Determines whether to pass logic command and analog reference control data from a ControlNet connection to a SCANport product.
5	*DataLnk A Config*	0 = Disabled 1 = Enabled	0=Disabled	Yes	Yes	Determines whether to pass control data contained in datalink A from a ControlNet connection to the SCANport product.
6	*DataLnk B Config*	0 = Disabled 1 = Enabled	0=Disabled	Yes	Yes	Determines whether to pass control data contained in datalink B from a ControlNet connection to the SCANport product.

Important: For information on accessing and editing parameters, refer to Chapter 3, *Getting Started.*

#	Name	Valid Values/ Settings	Default	Writable	Reset or Power Cycle to Take Effect	Description
7	*DataLnk C Config*	0 = Disabled 1 = Enabled	0=Disabled	Yes	Yes	Determines whether to pass control data contained in datalink C from a ControlNet connection to the SCANport product.
8	*DataLnk D Config*	0 = Disabled 1 = Enabled	0=Disabled	Yes	Yes	Determines whether to pass control data contained in datalink D from a ControlNet connection to the SCANport product.
9	*Idle Flt Action*	0 = Fault 1 = Zero Data 2 = Hold Last 3 = Fault Cfg	0=Fault	Yes	No	Determines the action the module should instruct the SCANport product to take if the module detects that the PLC is set to program mode.
						**ATTENTION:** Risk of severe bodily injury or loss of life exists. The *Idle Flt Action* (9) parameter allows the user to change the default configuration that would allow the module and associated drive to continue to operate if communication is lost. Precautions should be taken to assure that your settings for these parameters and your application do not create a hazard of bodily injury or loss of life.
10	*Comm Flt Action*	0 = Fault 1 = Zero Data 2 = Hold Last 3 = Fault Cfg	0=Fault	Yes	No	Determines the action the module should instruct the SCANport product to take if the module detects a network failure.
						**ATTENTION:** Risk of severe bodily injury or loss of life exists. The *Comm Flt Action* (10) parameter allows the user to change the default configuration that would allow the module and associated drive to continue to operate if communication is lost. Precautions should be taken to assure that your settings for these parameters and your application do not create a hazard of bodily injury or loss of life.
11	*Fault Cfg Logic*	0 – 65535	0	Yes	No	Provides the logic command data to the SCANport product when the module is instructed to use the values for the Fault Cfg parameters.
12	*Fault Cfg Ref*	0 – 65535	0	Yes	No	Provides the analog reference data to the SCANport product when the module is instructed to use the values for the Fault Cfg parameters.
13	*Fault Cfg A1 In*	0 – 65535	0	Yes	No	Provides the first word of datalink A data to the SCANport product when the module is instructed to use the values for the Fault Cfg parameters.
14	*Fault Cfg A2 In*	0 – 65535	0	Yes	No	Provides the second word of datalink A data to the SCANport product when the module is instructed to use the values for the Fault Cfg parameters.
15	*Fault Cfg B1 In*	0 – 65535	0	Yes	No	Provides the first word of datalink B data to the SCANport product when the adapter is instructed to use the values for the Fault Cfg parameters.

Important: For information on accessing and editing parameters, refer to Chapter 3, *Getting Started.*

#	Name	Valid Values/ Settings	Default	Writable	Reset or Power Cycle to Take Effect	Description
16	*Fault Cfg B2 In*	0 – 65535	0	Yes	No	Provides the second word of datalink B data to the SCANport product when the module is instructed to use the values for the Fault Cfg parameters.
17	*Fault Cfg C1 In*	0 – 65535	0	Yes	No	Provides the first word of datalink C data to the SCANport product when the adapter is instructed to use the values for the Fault Cfg parameters.
18	*Fault Cfg C2 In*	0 – 65535	0	Yes	No	Provides the second word of datalink C data to the SCANport product when the module is instructed to use the values for the Fault Cfg parameters.
19	*Fault Cfg D1 In*	0 – 65535	0	Yes	No	Provides the first word of datalink D data to the SCANport product when the adapter is instructed to use the values for the Fault Cfg parameters.
20	*Fault Cfg D2 In*	0 – 65535	0	Yes	No	Provides the second word of datalink D data to the SCANport product when the module is instructed to use the values for the Fault Cfg parameters.
21	*Serial Port Rate*	0 = 2400 1 = 4800 2 = 9600 3 = 19.2K 4 = 38.4K	2=9600	Yes	Yes	Sets the baud rate for the 1203-CN1 module's serial/ DF1 port. **Important:** These settings are valid for v1.004 or higher CN1s. v1.003 or lower CN1s only have two settings: 0 = 9600 and 1 = 19.2K. **Important:** If you change the baud rate in the module, you must also change it in your terminal emulation software or terminal.
22	*Reset Adapter*	0 = Ready 1 = Enable 2 = Set Defaults	0=Ready	Yes	No	1 = Resets the module if set to Enable. 2 = Sets all parameters to their default values and then resets the module.
23	*Active I/O Cfg*	b0 = Logic/Ref b1 = Datalink A b2 = Datalink B b3 = Datalink C b4 = Datalink D	0	No	No	Determines whether each of the control data connections (logic command / analog reference (LSB), datalink A, B, C, and D (MSB) are currently logged into the SCANport product.

Important: For information on accessing and editing parameters, refer to Chapter 3, *Getting Started.*

N-File Structure

Appendix Objectives

Appendix C details the N-file structure in the 1203-CN1 module. The N-files contain data for the 1203-CN1 module and the connected SCANport product. If you need to access information in the product using messages or program a PLC or PC, you will need to know what data is stored in each N-file.

N-File Structure

The following table lists the N-files and a description of each N-file's content.

N-File	Description
N10:0	Number of SCANport Product Parameters
N10:1 – N10:999 N11:0 – N11:999 N12:0 – N12:999	SCANport Product Parameter Value Read/Write (Parameters 1 – 2999)
N13:0	Number of 1203-CN1 Module Parameters
N13:1 – N13:999	1203-CN1 Module Parameter Value Read/Write (Parameters 1 – 999)
N20:0 – N20:127	Status of last SCANport parameter write (into N10 – N19) (0-good, 1 – 7 = SCANport error code) If ten parameter values are written to N10 with a single PCCC message, each of the first ten elements will contain a status code for the respective SCANport write operation. If twenty parameter values are written, the first twenty elements of this file will contain these codes. It makes no difference if the parameter write began at some element other than 0 because the resulting status codes will still begin at N20:0.
N30:1 – N30:999 N31:0 – N31:999 N32:0 – N32:999	SCANport Product Parameter Full/All Info Read (Parameters 1 – 2999)
N33:1 – N33:999	1203-CN1 Module Parameter Full/All Info Read (Parameters 1 – 999)
N40:0 – N40:63	Block Transfer Emulation
N42:6	Max Network Node (Read Only)
N42:7	1203-CN1 Module's Port Number (Read Only)

N-File	Description
N50:1 – N50:249 N51:0 – N51:249 N52:0 – N52:249 N53:0 – N53:249 N54:0 – N54:249 N55:0 – N55:249 N56:0 – N56:249 N57:0 – N57:249 N58:0 – N58:249 N59:0 – N59:249 N60:0 – N60:249 N61:0 – N61:249	SCANport Product Parameter Value Read/Write (Parameters 1 – 2999)
N90:1 – N90:249 N91:0 – N91:249 N92:0 – N92:249 N93:0 – N93:249 N94:0 – N94:249 N95:0 – N95:249 N96:0 – N96:249 N97:0 – N97:249 N98:0 – N98:249 N99:0 – N99:249 N100:0 – N100:249 N101:0 – N101:249	SCANport Product Parameter Full/All Info Read (Parameters 1 – 2999) (Each file contains a maximum of 250 parameters)

① SCANport Product refers to a drive or other SCANport product connected to the 1203-CN1 module.

ControlNet Objects

Appendix Objectives

Appendix D defines the ControlNet object classes, class services, and attributes that are supported by the 1203-CN1 ControlNet communications module. These objects can be used to develop programs for the module.

This appendix assumes that you have experience in object programming.

Object Classes

The 1203-CN1 module supports the following object classes.

Class Code 0x01 — Identity Object

The identity object provides identification and general information about the device.

Class Attributes

Attribute ID	Access Rule	Name	Data Type	Description
1	Get	Revision	UINT	Revision of this object. First revision, value=1.
2	Get	Max Instance	UINT	Maximum instance number of an object currently created in this class level of the device.
6	Get	Max ID (Number of Class Attributes)	UINT	The attribute ID number of the last class attribute of the class definition implemented in the device.
7	Get	Max ID (Number of Instance Attributes)	UINT	The attribute ID number of the last instance attribute of the class definition implemented in the device.

Instances

The total number of instances depends on the number of microprocessors in the SCANport product connected to the module.

- Instance 1 includes information on both the module and the product.
- The instances for the SCANport product's microprocessors start at instance 2.
- The instances for the module start after all the instances for the SCANport product's microprocessors.

Instance	Description
1	Total Product
2 through n - 5 ①	Product Microprocessor(s)
n - 4 ①	Total Module
n - 3 ①	Module Application Code
n - 2 ①	Module Boot Code
n -1 ①	Module ASIC Code
n ①	Module application and ASIC code

① The value of n is the maximum instance in the object. This value is obtainable via class attribute 2.

Instance Attributes

Attribute ID	Access Rule	Name	Data Type	Description
1	Get	Vendor ID	UINT	Identification of each vendor by number. 1 = Allen-Bradley
2	Get	Device Type	UINT	Indication of general type of product. 0x69 = Sub-Component 0x6E = SCANport Device
3	Get	Product Code	UINT	Identification of a particular product of an individual vendor. 0x0002 = 1336 PLUS 0.5 – 10 HP 0x0003 = 1336 PLUS 7.5 – 800 HP 0x0007 = 1336 PLUS II 0x0010 = 1336 FORCE w/ PLC Adapter 0x0011 = 2364F RGU 0x0012 = 1394 Motion Drive 0x0013 = 1557 Medium Voltage AC Drive 0x0014 =193 SMP-3 0x0015 = 150 SMC Dialog Plus 0x0017 = 1305 AC Drive 0x0018 = 1397 DC Drive 0x0019 = 1336 VSC 0x0020 = 1336T Force w/ Std Adapter 0x0022 = 1336 IMPACT
4	Get	Revision	STRUCT of	Revision of the item that this instance of the Identity Object represents. Value varies based on product.
		Major Revision	USINT	
		Minor Revision	USINT	
5	Get	Status	WORD	Summary status of product. Value varies based on product.
6	Get	Serial Number	UDINT	Serial number of product. Value varies based on product.
7	Get	Product Name	SHORT_ STRING	Human readable identification. Value varies based on product.

Common Services

Service Code	Implemented for:		Service Name
	Class	Instance	
0x01	Yes	Yes	Get_Attribute_All
0x05	Yes	Yes	Reset
0x0E	Yes	Yes	Get_Attribute_Single

Get_Attribute_All Response

At the Class level the order of the attributes returned in the "Object/ service specific reply data" portion of the Get_Attribute_All response is defined as follows:

Attribute ID	Definition
1	Revision (low BYTE) Default = 1. Revision (high BYTE) Default = 0.
2	Max Instance (low BYTE) Default = 1. Max Instance (high BYTE) Default = 0.
6	Max ID Number of Class Attributes (low BYTE) Default = 0. Max ID Number of Class Attributes (high BYTE) Default = 0.
7	Max ID Number of Instance Attributes (low BYTE) Default = 0. Max ID Number of Instance Attributes (high BYTE) Default = 0.

At the instance level, the order of the attributes returned in the "Object/service specific reply data" portion of the Get_Attribute_All response is as follows:

Attribute ID	Definition
1	Vendor
2	Device Type
3	Product Code
4	Major Revision
5	Status
6	Serial Number
7	Product Name Length

Class Code 0x02 — Message Router Object

The Message Router Object provides a messaging connection point through which a client may address to any object class or instance residing in the physical devices.

Class Attributes

Attribute ID	Access Rule	Name	Data Type	Description
1	Get	Revision	UINT	Revision of this object. First revision, value=1.
4	Get	Optional Attribute List	STRUCT of	List of optional instance attributes utilized in an object class implementation.
		Number Attributes	UINT	Number of attributes in the optional attribute list.
		Optional Attributes	ARRAY of UINT	List of optional attribute numbers.
5	Get	Optional Service List	STRUCT of	A list of service codes specifying the optional services implemented in the device for this class.
		Number Services	UINT	Number of services in the optional service list.
		Optional Services	ARRAY of UINT	List of optional service codes.
6	Get	Max ID (Number of Class Attributes)	UINT	The attribute ID number of the last class attribute of the class definition implemented in the device.
7	Get	Max ID (Number of Instance Attributes)	UINT	The attribute ID number of the last instance attribute of the class definition implemented in the device.

Instances

Instance	Description
1	Message Router Object

Instance Attributes

Attribute ID	Access Rule	Name	Data Type	Description
1	Get	Object_List	STRUCT of	A list of supported objects. Structure with an array of object class codes supported by the device via the message router.
		Number	UINT	Number of supported classes in the classes array.
		Classes	ARRAY of UINT	List of supported class codes.
2	Get	Number available	UINT	Maximum number of connections supported.
3	Get	Number active	UINT	Number of connections currently used by system components. Not used. Reserved for compatibility purposes. Value = 0
4	Get	Active connections	ARRAY of UINT	A list of the connection IDs of the currently active connections. This attribute not used. Reserved for compatibility purposes.

Common Services

Service Code	Implemented for:		Service Name
	Class	Instance	
0x01	Yes	Yes	Get_Attribute_All
0x0E	Yes	Yes	Get_Attribute_Single

Get_Attribute_All Response

At the class level, the order of the attributes returned in the Get_Attribute_All response is as follows:

Class Attribute ID	Definition
1	Revision Default = 1
4	Optional attribute list, number of attributes default = 0
5	Optional service list, number of services default = 0
6	Max ID number of class attributes default = 0
7	Max ID number of instance attributes default = 0

At the instance level, the order of attributes returned in the Get_Attribute_All response is as follows:

Instance Attribute ID	Definition
1	Object list number, number of supported classes default = 0
2	Number available, maximum number of connections default = 0
3	Always = 0x000

Class Code 0x04 — Assembly Object

The Assembly Object binds attributes of multiple objects, allowing data to or from each object to be sent or received over a single connection. Assembly objects are used to produce and/or consume data to/from the network. An instance of the assembly object can both produce and consume data from the network if designed to do so.

Important: Setting of an assembly attribute can only be accomplished through a connection. This feature is to prevent accidental control of the SCANport product.

Class Attributes

Attribute ID	Access Rule	Name	Data Type	Description
1	Get	Revision	UINT	Revision of this object. First revision, value=1.
2	Get	Max Instance	UINT	Maximum instance number of an object currently created in this class level of the device.

Instances

Instance	Description
1	PLC Input Data from the SCANport Product
2	PLC Output Data to the SCANport Product
3	Controlling Heartbeat
4	Listen Only Heartbeat
5	Redundant Output
6	Dummy Configuration

Instance Attributes

Attribute ID	Access Rule	Name	Data Type	Description
1	Get	Number of Members in List	UINT	Number of members in attribute 2.
2	Get	Member List	ARRAY of STRUCT	The member list is an array of paths.
		Member Data Size	UINT	Size of member data (in bits).
		Member Path Size	UINT	Size of member path (in BYTEs). 0 = Empty Path
		Member Path	ARRAY of BYTES	Path to/from data for this member.①
3①	Get	Data	ARRAY of BYTES	All of the member data packed into one array.

① Instance attribute 3, Data, contains all of the member data packed into one array. This data may contain many different data types. For efficiency it is best to keep this data word aligned by packing it on word boundaries and adding padding as needed. This can be accomplished by using “empty paths” (Member Path Size = 0).

Common Services

Service Code	Implemented for:		Service Name
	Static Assembly		
	Class	Instance	
0x0E	Yes	Yes	Get_Attribute_Single

Class Code 0x06 — Connection Manager Object

The Connection Manager Object is used to manage the establishment and maintenance of communication connections.

Class Attributes

Attribute ID	Access Rule	Name	Data Type	Description
1	Get	Revision	UINT	Revision of this object. First revision, value=1.
2	Set	Max Instance	UINT	Maximum instance number of an object currently created in this class level of the device.
4	Get	Optional Attribute List	STRUCT of	List of optional instance attributes used in an object class implementation.
		Number Attributes	UINT	Number of attributes in the optional attribute list.
		Optional Attributes	ARRAY of UINT	List of optional attribute numbers.

Instances

Instance	Description
1	Connection Manager

Instance Attributes

Attribute ID	Access Rule	Name	Data Type	Description
1	Set	OpenReqs	UINT	Number of Open Requests received including Null Open Requests.
2	Set	OpenFormat Rejects	UINT	Number of open requests rejected by this node due to format errors.
3	Set	OpenResource Rejects	UINT	Number of open requests rejected by this node.
4	Set	OpenOther Rejects	UINT	Number of open requests rejected or timed out by downstream nodes.
5	Set	CloseReqs	UINT	Number of close requests received.
6	Set	CloseFormat Rejects	UINT	Number of close requests rejected by this node due to format errors.
7	Set	CloseOther Rejects	UINT	Number of close requests received rejected or timed out by downstream nodes.
8	Set	Conn Timeouts	UINT	Number of connections which have been timed out by this node after they were opened.

Attribute ID	Access Rule	Name	Data Type	Description
9	Get	Connection Entry List	STRUCT of	List of connections
		NumConnEntries	UINT	Number of Conn Array Entries (bit field). This attribute, divided by 8 and incremented for any remainder, gives the length of the array in the Conn Open Bits field.
		ConnOpenBits	ARRAY of BYTE	List of connection data which may be individually queried by the Get/Search Connection Data Services. Each bit represents a possible connection. 0 = No Connection 1 = Connection Established Query for more information
11	Get	CpuUtilization	UINT	CPU Utilization in tenths of a percent. Range of 0 – 1000 representing 0 to 100%.
12	Get	MaxBuffSize	UDINT	Amount of buffer space originally available. Size is in BYTEs.
13	Get	BufSize Remaining	UDINT	Amount of buffer space available at this time. Size is in BYTEs.

Important: Instance attribute number 10 is not used.

Common Services

Service Code	Implemented for:		Service Name
	Class	Instance	
0x01	Yes	Yes	Get_Attribute_All
0x02	No	Yes	Set_Attribute_All
0x0E	Yes	Yes	Get_Attribute_Single
0x10	No	Yes	Set_Attribute_Single

Class Code 0x07 — Register Object

The Register Object is used to address individual bits or a range of bits. It may operate as either a producer (input) register or a consumer (output) register. A producer register object produces data onto the network. A consumer register object consumes data from the network.

Message writes to the Register Object can perform control functions. Therefore, message writes are only allowed when the controller is not actively controlling the module and the message write is done through a connection with a time-out value not equal to zero. Writes cannot be performed through an unconnected message. After a write, any time-out or closure of the connection may cause the SCANport product to fault.

Important: For firmware 1.001, connections to this object are not supported. Therefore, it is a read only object.

Class Attributes

Attribute ID	Access Rule	Name	Data Type	Description
1	Get	Revision	UINT	Revision of this object. First revision, value = 1.
2	Get	Number of Instances	UINT	Number of instances of object. Value = 14.

Instances

Instance	Description
1	All output data configured from the SCANport device
2	All input data configured to the SCANport device
3	Status/Feedback data
4	Command/Reference data
5	Datalink A input data
6	Datalink A output data
7	Datalink B input data
8	Datalink B output data
9	Datalink C input data
10	Datalink C output data
11	Datalink D input data
12	Datalink D output data
13	Status/Feedback Data
14	Logic Command/Command Mask

① The command word is set to the value of the first word of the data where there are ones in the second word of the data. command = (word 1 and not word 2) or (word 1 and word 2). This only controls specified bits in the logic command data to the SCANport product and does not change the reference value.

Instance Attributes

Setting of an assembly attribute can only be accomplished through a connection. This feature is to prevent accidental control of the SCANport product.

Attribute ID	Access Rule	Name	Data Type	Description
1	Get	Bad Flag	BOOL	If set to 1, then attribute 4 may contain invalid, bad or otherwise corrupt data. 0 = good 1 = bad
2	Set	Direction	BOOL	Direction of data transfer 0 = Producer Register 1 = Consumer Register
3	Set	Size	UINT	Size of register data in bits
4	Conditional①	Data	ARRAY of BITS	Data to be transferred

① The access rule of Set is optional if attribute 2, Direction = 1. If Direction = 0, the access rule is Get.

Common Services

Service Code	Implemented for:		Service Name
	Class	Instance	
0x0E	Yes	Yes	Get_Attribute_Single
0x10	No	Yes	Set_Attribute_Single

Class Code 0x0F — Parameter Object

The Parameter Object provides a known, public interface for device configuration data. This object also provides all the information necessary to define and describe each individual configuration parameter of a device.

Class Attributes

Attribute ID	Access Rule	Name	Data Type	Description
1	Get	Revision	UINT	Revision of this object. First revision, value=1.
2	Get	Max Instance	UINT	Maximum instance number of an object currently created in this class level of the device.
8	Get	Parameter Class Descriptor	WORD	Bits that describe parameters.
9	Get	Configuration Assembly Instance	UINT	Instance number of the configuration assembly. This attribute should be set to zero if a configuration assembly is not supported.
10	Set	Native Language	USINT	Language ID for all character array accesses. 0 = English 1 = French 2 = Spanish 3 = Italian 4 = German 5 = Japanese 6 = Portuguese 7 = Mandarin Chinese

Instances

The number of instances varies based on the number of parameters in your SCANport product. The 23 module parameters immediately follow the SCANport product parameters.

Instance	Description
1 through n - 23 ①	SCANport Product Parameters
n - 22 through n ①	Module Parameters

① The value of n is the maximum instance in the object. This value is obtainable via class attribute 2.

Instance Attributes

Attribute ID	Access Rule	Stub/Full	Name	Data Type	Description
1	①	Stub	Parameter Value	Specified in Descriptor, Data Type and Data Size attributes.	Actual value of parameter. Data type specified in descriptor, data type, and data size.① ②
2	Set	Stub	Link Path Size	USINT	Size of Link Path attribute. If this attribute is 0, then no link is specified. Number of BYTEs in attribute 3.
3	Set	Stub	Link Path	ARRAY of path segments	Path to the object from where this parameter value is retrieved. The link path is limited to 255 BYTEs.
			Segment type/port	BYTE	
			Segment Address	Path (format depends on data contained in segment type/port)	
4	Get	Stub	Descriptor	WORD	Descriptor of parameter. Refer to the table on page D-15.
5	Get	Stub	Data Type	USINT	Data type code. Refer to the table on page D-16.
6	Get	Stub	Data Size	USINT	Number of BYTEs in attribute 1, Parameter Value.
7	Get	Full	Parameter Name String	SHORT_ STRING	A human readable string representing the parameter name. For example, “frequency #1.” The maximum number of characters is 16. (The first byte is a length code.)
8	Get	Full	Units String	SHORT_ STRING	Engineering unit string. The maximum number of characters is 4. (The first byte is a length code.)
9	Get	Full	Help String	SHORT_ STRING	Help string. The maximum number of characters is 64. (The first byte is a length code.)
10	Get	Full	Minimum Value	Same as attribute 1	The minimum valid actual value to which attribute 1, Parameter Value can be set.
11	Get	Full	Maximum Value	Same as attribute 1	The maximum valid actual value to which attribute 1, Parameter Value can be set.
12	Get	Full	Default Value	Same as attribute 1	The actual value attribute 1, Parameter Value should be set to when the user wants the default for the parameter.
13	Get	Full	Scaling Multiplier	UINT	Multiplier for scaling formula.
14	Get	Full	Scaling Divisor	UINT	Divisor for scaling formula.
15	Get	Full	Scaling Base	UINT	Base for scaling formula.

Attribute ID	Access Rule	Stub/Full	Name	Data Type	Description
16	Get	Full	Scaling Offset	UINT	Offset for scaling formula.
17	Get	Full	Multiplier Link	UINT	Parameter object instance number of multiplier source.
18	Get	Full	Divisor Link	UINT	Parameter object instance number of base source.
19	Get	Full	Base Link	UINT	Parameter object instance number of offset source.
20	Get	Full	Offset Link	UINT	Parameter object instance number of offset source.
21	Get	Full	Decimal Precision	USINT	Specifies number of decimal places to use when displaying the scaled engineering value. Also used to determine actual increment value so that incrementing a value causes a change in scaled engineering value to this precision.

① The access rule is defined in bit 4 of instance attribute 4, the Descriptor. If bit 4 is 0 the access rule is Set and the Parameter Value can be read and written. If bit 4 is 1, the access rule is Get and the Parameter Value can only be read.

② Data type specified in instance attributes 4 (Descriptor), 5 (Data Type) and 6 (Data Size).

Bit definitions for Instance Attribute 4

Bit	Definition	Value
0	Supports settable path	0 = Link path can not be set. 1 = Link path can be set.
1	Supports enumerated strings	0 = Enumerated strings are not supported. 1 = Enumerated strings are supported and may be read with the Get_Enum_String service.
2	Supports scaling	0 = Scaling not supported. 1 = Scaling is supported. The scaling attributes are implemented and the value presented is in engineering units.
3	Supports scaling links	0 = Scaling links not supported. 1 = The values for the scaling attributes may be retrieved from other parameter object instances.
4	Read only parameter	0 = Parameter value attribute can be written (set) and read (get). Access rule is set. 1 = Parameter value attribute can only be read. Access rule is get.
5	Monitor parameter	0 = Parameter value attribute is not updated in real time by the device. 1 = Parameter value attribute is updated in real time by the device.
6	Supports extended precision scaling	0 = Extended precision scaling is not supported. 1 = Extended precision scaling should be implemented and the value presented to the user in engineering units.
7	Support non-consecutive enumerated strings	0 = Non-consecutive enumerated strings are not supported. 1 = Non-consecutive enumerated strings are supported.
8	Allow both enumeration and individual values	0 = Both enumeration and individual values are not supported. 1 = Both enumeration and individual values are supported.
9 – 15	Not Defined	These bits have yet to be defined. They should be set to 0.

Data Types for Instance Attribute 5

Attribute ID Value	Definition	Data Type Description	Scaling Supported on this Data Type
1	WORD	16-bit word	No
2	UINT	16-bit unsigned integer	Yes
3	INT	16-bit signed integer	Yes
4	BOOL	Boolean	No
5	SINT	Short integer	Yes
6	DINT	Double integer	Yes
7	LINT	Long integer	Yes
8	USINT	Unsigned short integer	Yes
9	UDINT	Unsigned double integer	Yes
10	ULINT	Unsigned long integer	Yes
11	REAL	Single floating point format (IEEE 754)	Yes
12	LREAL	Double floating point format (IEEE 754)	Yes
13	ITIME	Duration (short)	Yes
14	TIME	Duration	Yes
15	FTIME	Duration (high resolution)	Yes
16	LTIME	Duration (long)	Yes
17	DATE	Date	No
18	TIME_OF_DAY	Time of Day	No
19	DATE_AND_TIME	Date and time	No
20	STRING	8-bit per character string	No
21	STRING2	16-bit per character string	No
22	STRINGN	N-byte per character string	No
23	SHORT_STRING	Short N-byte character string	No
24	BYTE	8-bit string	No
25	DWORD	32-bit string	No
26	LWORD	64-bit string	No

Common Services

Service Code	Implemented for:		Service Name
	Class	Instance	
0x01	Yes	Yes	Get_Attribute_All
0x0E	Yes	Yes	Get_Attribute_Single
0x10	Yes	Yes	Set_Attribute_Single

Get_Attribute_All Response

At the class level, the order of the attributes returned in the "object/ services specific reply data" portion of the Get_Attribute_All response is as follows:

Class Attribute ID	Definition
1	Revision (low BYTE) Default = 1
	Revision (high BYTE) Default = 0
2	Max Instance (low BYTE)
	Max Instance (high BYTE)
8	Parameter class descriptor (low BYTE)
	Parameter class descriptor (high BYTE)
9	Configuration Assembly Instance (low BYTE)
	Configuration Assembly Instance (high BYTE)
10	Native Language Default = 0

At the instance level, the order of attributes returned in the Get_Attributes_All response is as follows:

Class Attribute ID	Attribute Name and default Value
1	Parameter Value
2	Link Path Size
3	Link Path
4	Descriptor
5	Data Type
6	Data Size
7	Parameter Name String, default character count = 0
8	Units String, default character count = 0
9	Help String, default character count = 0
10	Minimum Value default = 0
11	Maximum Value default = 0
12	Default Value default = 0
13	Scaling Multiplier Default = 1

Class Attribute ID	Attribute Name and default Value
14	Scaling Divisor Default = 1
15	Scaling Base Default = 1
16	Scaling Offset Default = 0
17	Multiplier Link Default = 0
18	Divisor Link Default = 0
19	Base Link Default = 0
20	Offset Link Default = 0
21	Decimal Precision Default = 0

Object Specific Services

Service Code	Implemented for:		Service Name
	Class	Instance	
0x4B	No	Yes	Get_Enum_String

Enumerated strings are human-readable strings that describe either a bit or a value depending on the data type of instance attribute 1, the Parameter Value. If the data type is a BYTE or WORD, the enumerated string is a bit enumerated string. If the data type is INT or UINT, the enumerated string is a value enumerated string. Any other data type does not have enumerated strings.

The table below lists the parameters for the Get_Enum_String request service.

Name	Data Type	Description of Attribute
Enumerated String Number	USINT	Number of enumerated string to retrieve (MAX value is 255).

- If the string to be returned is a bit enumerated string, then the enumerated string number represents a bit position and the Get_Enum_String service returns a string from that bit.
- If the string to be returned is a value enumerated string, then the enumerated string number represents a value and the Get_Enum_String service returns a string for that value.

The enumerated string is returned in the form of a SHORT_STRING and is 16 characters long plus the preceding length byte.

Class Code 0x10 — Parameter Group Object

The Parameter Group Object identifies and provides access to groups of parameters in a device grouping. The Parameter Group Object provides convenient access to related sets of parameters.

Class Attributes

Attribute ID	Access Rule	Name	Data Type	Description
1	Get	Revision	UINT	Revision of this object. First revision, value=1.
2	Get	Max Instance	UINT	Maximum instance number of an object currently created in this class level of the device.
8	Set	Native Language	USINT	Language ID for all STRING accesses. 0 = English 1 = French 2 = Spanish (Mexican) 3 = Italian 4 = German 5 = Japanese 6 = Portuguese 7 = Mandarin Chinese

Instances

The number of instances varies based on the number of groups in the SCANport product. One additional group is added for the module.

Instance	Description
1 – (n - 1)	SCANport product groups
n	Module group

① N is the value returned by a get from class attribute 2 (max instance).

Instance Attributes

Attribute ID	Access Rule	Name	Data Type	Description
1	Get	Group Name String	SHORT_ STRING	A human-readable string representing the group name (e.g., set-up, frequency set). Maximum number of characters = 16
2	Get	Number of Members in Group	UINT	Number of parameters in group.
3	Get	1st Parameter Number in Group	UINT	Parameter instance number.
4	Get	2nd Parameter Number in Group	UINT	Parameter instance number.
n	Get	(n-2)th Parameter Number in Group	UINT	Parameter instance number.

Common Services

Service Code	Implemented for:		Service Name
	Class	Instance	
0x01	Yes	Yes	Get_Attribute_All
0x10	Yes	No	Set_Attribute_Single
0x0E	Yes	Yes	Get_Attribute_Single

Get_Attribute_All Response

At the class level, the order of the attributes returned in the "object/ service specific reply data" portion of the Get_Attribute_All response is as follows:

Class Attribute ID	Definition
0	Revision (low BYTE) Default = 1
1	Revision (high BYTE) Default = 0
2	Max Instance (low BYTE)
3	Max Instance (high BYTE)
4	Native language default

In the instance level, the order of the attributes returned in the "object/ service reply data" portion of the Get_Attribute_All response is as follows:

Class Attribute ID	Definition
0	Group name (character count)
1	Group name (1st character)
n	Group name (last character)
n + 1	Number of members in group (low BYTE)
.	Number of members in group (high BYTE)
.	1st parameter number in group (low BYTE)
.	1st parameter number in group (high BYTE)
.	last parameter number in group (low BYTE)
.	last parameter number in group (high BYTE)

Class Code 0xF0 — ControlNet Object

The ControlNet Object provides a consistent Station Management interface to the Physical and Data Link Layers. This object makes diagnostic information from these layers available to client applications.

Class Attributes

Attribute ID	Access Rule	Name	Data Type	Description
0x01	Get	Revision	UINT	Revision of this object. First revision, value=1.
0x02	Get	Max Instance	UDINT	Maximum instance number. Value determined by node specifics.

Instances

Instance	Description
1	ControlNet Status

Instance Attributes

Attribute ID	Access Rule	Name	Data Type	Description
0x81	Get	current_link_config	STRUCT of 34 BYTEs	Current link configuration parameters.
0x82	Get/Get_and_Clear	diagnostic_counters	STRUCT of 42 BYTEs	Diagnostic counters.
		buffer_errors	UINT	Buffer event counters.
		error_log	BYTE [8]	Bad MAC frame log.
		event_counters	STRUCT of 32 BYTEs	Diagnostic counters.
		good_frames_transmitted	BYTE [3]	Good MAC frames transmitted (LSB first).
		good_frames_received	BYTE [3]	Good MAC frames received (LSB first).
		selected_channel_frame_errors	USINT	Framing errors detected on active receive channel.
		channel_A_frame_errors	USINT	Framing errors detected on active receive channel.
		channel_B_frame_errors	USINT	MAC frames aborted during transmission (transmit underflows).
		aborted_frames_transmitted	USINT	MAC frames aborted during transmission (transmit underflows).
		highwaters	USINT	LLC transmit underflow and LLC receive overflow.

Attribute ID	Access Rule	Name	Data Type	Description
0x82 (continued)	Get/Get_and_Clear (continued)	NUT_overloads	USINT	No unscheduled time in NUT (all time used for scheduled transmissions).
		slot_overloads	USINT	More scheduled data queued for one NUT than allowed by sched_max_frame parameter.
		blockages	USINT	Single Lpacket size exceeds sched_max_frame parameter.
		non-concurrence	USINT	Two or more nodes could not agree whose turn it is to transmit.
		aborted_frames_received	USINT	Incomplete MAC frames received.
		lonely_counter	USINT	Number of times nothing heard on network for 8 or more NUTs.
		duplicate_node	USINT	MAC frame received from node with local node's MAC ID.
		noise hits	USINT	Noise detected that locked the modem rx PLL.
		collisions	USINT	Rx data heard just as we were going to transmit.
		mod_MAC_ID	USINT	MAC ID of the current moderator node.
		non_lowman_mods	USINT	Moderator frames heard from non-lowman nodes.
		rogue_count	USINT	Rogue events detected.
		unheard_moderator	USINT	MAC frames being heard by no moderators being heard.
		vendor_specific	USINT	Vendor specific.
		reserved	BYTE [4]	Reserved.
		vendor_specific	USINT	Vendor specific.
		vendor_specific	USINT	Vendor specific.
		reserved	BYTE	Reserved.
0x83	Get	station_status	STRUCT of 6 BYTEs	Station status.
		smac_ver	USINT	MAC implementation.
		vendor_specific	BYTE [4]	Vendor specific.
		channel_state	BYTE	Channel state LEDs, redundancy warning, and active channel bits.
0x84	Set/Get	MAC_ID	STRUCT of 4 BYTEs	MAC ID switch and current settings.
		MAC_ID_current	USINT	Current MAC ID.
		MAC_ID_switches	USINT	MAC ID switch settings.
		MAC_ID_changed	BOOL	MAC ID switches changed since reset.
		reserved	USINT	Reserved.

Attribute ID	Access Rule	Name	Data Type	Description
0x86	Get	Error_log	STRUCT of 10 BYTEs	Driver firmware buffer error counts and troublesome node list.
		Buffer_errors	UINT	Buffer event counter.
		Error_log	BYTE [8]	Bad MAC frame log.

Common Services

Service Code	Implemented for:		Service Name
	Class	Instance	
0x05	No	Yes	Reset
0x0E	Yes	Yes	Get_Attribute_Single
0x4C	No	Yes	①
0x4D	No	Yes	Enter/Listen Only

① Used only for diagnostic counters.

Class Code 0xA1 — Non-Volatile Storage Object

The Non-Volatile Storage (NVS) Object stores information during the loss of power to a module. The object is an abstraction of EEPROM, FLASH EPROM, and Battery Backed RAM.

Class Attributes

Attribute ID	Access Rule	Name	Data Type	Description
1	Get	Revision	UINT	Revision of this object. (Important: All class definitions are required to include this class attribute.) Value is 02.
2	Get	Max Instance	UDINT	Maximum instance number of an object currently created in this class level of the device.
3	Get	Number of Instances	UDINT	Number of object instances currently created at this class level of the device.
4	Get	Optional attribute list	STRUCT of	List of optional instance attributes used in an object class implementation.
		number attributes	UINT	Number of attributes in the optional attribute list.
		optional attributes	ARRAY of UINT	List of optional attribute numbers.
5	Get	Optional service list	STRUCT of	List of optional services used in an object class implementation.
		Number services	UINT	Number of services in the optional service list.
		Optional services	ARRAY of UINT	List of optional service codes.
6	Get	Maximum ID Number Class Attributes	UINT	The attribute ID number of the last class attribute of the class definition implemented in the device.
7	Get	Maximum ID Number Instance Attributes	UINT	The attribute ID number of the last instance attribute of the class definition implemented in the device.

Instances

Instance	Description
1	Boot Code
2	Application Code
3	ASIC Code
4	Combined Application and ASIC Code

Instance Attributes

Attribute ID	Access Rule	Name	Data Type	Description
1	Get	Status	UINT	Current status of the NVS object.
2	Get	Major Instance Revision	USINT	Current Major Revision number of this NVS instance. Start at 1.
		Minor Instance Revision	USINT	Current Minor Revision number of this NVS instance. Start at 1.

① The Status attribute reports the current status based upon the state of an instance of the object. The assignment of values to "Status" follows:
0 = nothing new/no update
1 = success on transfer
2 = success on programming
3 = failure on transfer
4 = failure on programming
5 = faulted

Common Services

Service Code	Implemented for:		Service Name
	Class	Instance	
0x03	Yes	Yes	Get_Attribute_List
0x01	Yes	Yes	Get_Attributes_All
0x4E	Yes	Yes	Get_Attributes_Single

Class Specific Services

Service Code	Implemented for:		Service Name
	Class	Instance	
0x4B	No	Yes	Update
0x4D	No	Yes	Transfer

Class Code 0x93 — SCANport Pass-Through Parameter Object

The SCANport Pass-Through Parameter Object lets you perform a scattered read or write.

Class Attributes

None

Instance Attributes

None

Common Services

None

Object-Specific Services

Service Code	Implemented for:		Service Name
	Class	Instance	
0x32	Yes	No	Scattered_Parameter_Value_Read①
0x34	Yes	No	Scattered_Parameter_Value_Write①

① Must be directed to Attribute 0, Instance 0.

The table below lists the parameters for the Scattered_Parameter_Value_Read and Scattered_Parameter_Value_Write object specific services:

Name	Data Type	Description
Scattered Parameters	STRUCT of	
Parameter Number	WORD	Parameter to read or write
Parameter Value	WORD	Parameter value to write (zero when reading)

Important: The STRUCT may repeat up to 124 times in a single message.

Class Code 0x97 — SCANport Pass-Through Fault Object

The SCANport Pass-Through Fault Object provides information on the product's fault queue.

Class Attributes

Attribute ID	Access Rule	Name	Data Type	Description
0	Set	Write Fault Command	BYTE	1 = Clear Faults 2 = Clear Fault Queue 3 = Reset Product
1	Get	Read Number of Fault Queue Entries	BYTE	Reads the number of fault queue entries.
2	Get	Read Fault Queue Trip Index	BYTE	Reads the index of the fault that tripped the product.

Instance Attributes

Attribute ID	Access Rule	Name	Data Type	Description
0	Get	Read Fault Queue Entry Full/All Info	STRUCT of	
		Fault Text	ARRAY of BYTE	16 character string (no length information, no terminating null).
		Fault Code	WORD	Fault Code.
		Fault Time Stamp	STRUCT	
			BYTE	1/100 Second (0 – 99).
			BYTE	Second (0 – 59).
			BYTE	Minute (0 – 59).
			BYTE	Hour (0 – 23).
			BYTE	Day of Week (0 – 6).①
			BYTE	Date (1 – 31).
			BYTE	Month (1 – 12).
			BYTE	Year (0 – 99②).
128	Get	Fault Code and Time Stamp	STRUCT of	
		Fault Code	WORD	Fault Code.
		Fault Time Stamp	STRUCT of	
			BYTE	1/100 Second (0 – 99).
			BYTE	Second (0 – 59).
			BYTE	Minute (0 – 59).
			BYTE	Hour (0 – 23).
			BYTE	Day of Week (0 – 6).①
			BYTE	Date (1 – 31).
			BYTE	Month (1 – 12).
			BYTE	Year (0 – 99②).
129	Get	Read Fault Text String Only	ARRAY of BYTE	16 character string (no length information, no terminating null).

① Sunday is a value of zero.

② Year is an offset from 1990.

Class Code 0x98 — SCANport Pass-Through Warning Object

The SCANport Pass-Through Warning Object provides information on the product's warning queue.

Class Attributes

Attribute ID	Access Rule	Name	Data Type	Description
0	Set	Write Warning Command	BYTE	Write Warning Command. 1 = Clear Warnings 2 = Clear Warning Queue 3 - Reset Product
1	Set	Read Number of Warning Queue Entries	BYTE	

Instance Attributes

Attribute ID	Access Rule	Name	Data Type	Description
0	Get	Read Warning Queue Entry Full/All Info	STRUCT of	
		Warning Text	ARRAY of BYTE	16 character string (no length information, no terminating null).
		Warning Code	WORD	Fault Code.
		Warning Time Stamp	STRUCT	
			BYTE	1/100 Second (0 – 99).
			BYTE	Second (0 – 59).
			BYTE	Minute (0 – 59).
			BYTE	Hour (0 – 23).
			BYTE	Day of Week (0 – 6).①
			BYTE	Date (1 – 31).
			BYTE	Month (1 – 12).
			BYTE	Year (0 – 99②).
128	Get	Warning Code and Time Stamp	STRUCT of	
		Warning Code	WORD	Fault Code.
		Warning Time Stamp	STRUCT of	
			BYTE	1/100 Second (0 – 99).
			BYTE	Second (0 – 59).
			BYTE	Minute (0 – 59).
			BYTE	Hour (0 – 23).
			BYTE	Day of Week (0 – 6).①
			BYTE	Date (1 – 31).
			BYTE	Month (1 – 12).
			BYTE	Year (0 – 99②).
129	Get	Read Warning Text String Only	ARRAY of BYTE	16 character string (no length information, no terminating null).

① Sunday is a value of zero.

② Year is an offset from 1990.

Class Code 0x99 — SCANport Pass-Through Link Object

The SCANport Pass-Through Link Object lets you perform a scattered read or write of a number of links or a single read or write of a link.

Class Attributes

Attribute ID	Access Rule	Name	Data Type	Description
0	Set	Link Command	BYTE	1 = Clear all links.
1	Get	NVS Link Diagnostic Value	WORD	Checksum.

Instance Attributes①

Attribute ID	Access Rule	Name	Data Type	Description
0	Set	Parameter Link Reference②	WORD	

① An instance in this class is the number of a parameter that is to get its value from another parameter.

② The Parameter Link Reference value is the number of the parameter whose value is to be transferred.

Common Services

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

Object-Specific Services

Service Code	Implemented for:		Service Name
	Class	Instance	
0x32	Yes	No	Scattered_Link_ Reference_Value_Read①
0x34	Yes	No	Scattered_Link_ Reference_Value_Write①

① Must be directed to Attribute 0, Instance 0.

The table below lists parameters for Scattered_Link_Reference_Read and Scattered_Link_Reference_Write object-specific services.

Name	Data Type	Description
Scattered Link Read/ Write	STRUCT of	
Parameter Number	WORD	Parameter Link Reference to read or write.
Parameter Link Reference	WORD	Link Reference value to write (zero when reading).

Important: The STRUCT may repeat up to 124 times in a single message.

Class Code 0x67 — PCCC Object

The PCCC Object is used to process encapsulated PCCC messages from ControlNet. The PCCC Object does not implement any specific class or instance attributes, so the instance field for any received messages is ignored.

Class Attributes

Attribute ID	Access Rule	Name	Data Type	Description
0	Set	PCCC Message	STRUCT of	
		CMD	BYTE	CMD
		STS	BYTE	STS
		TNS	WORD	TNS
		FNC or EXT STS	BYTE	Optional FNC or EXT STS
		Data	ARRAY of BYTE	Optional Data

Instance Attributes

None

Common Services

None

Object Specific Services

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

Message Structure

Command	Response
Command BYTE	Extended Status BYTE (Replies)
STS BYTE	TNSW - High BYTE
TNSW - Low BYTE	TNSW - Low BYTE
TNSW - High BYTE	STS BYTE
Optional Extended Functional Code (FNC)	Command BYTE
Optional Data: Up to 243 or 244 BYTEs	Optional Data: Up to 243 or 244 BYTEs

More Information

For more information on PCCC messages, refer to Appendix E, *Supported PCCC Messages.*

Supported PCCC Messages

Appendix Objectives

Appendix E lists PCCC messages and whether they are supported by the 1203-CN1 ControlNet Communications Module. This information is provided for those writing software to perform monitoring and configuration functions from a PC.

Supported PCCC Messages

The following table lists PCCC messages and whether each message is supported.

Message Type	Supported
Identify Host message	Yes
Uses 0x7F Processor Type code	Yes
PLC-5 Typed Read (CMD=0x0F, FNC=0x68)	Yes
Logical Binary Addressing	Yes
Symbolic	No
Logical ASCII Addressing	Yes
PLC-2 System Address	No
PLC-5 Typed Write (CMD=0x0F, FNC=0x67)	Yes
Logical Binary Addressing	Yes
Symbolic	No
Logical ASCII Addressing	Yes
PLC-2 System Address	No
Encapsulate Other Protocol message (CMD=0x0F, FNC= 0x95)	Yes

PCCC Error Response Codes

The following table lists the PCCC error response codes.

Messages	STS	EXT STS	Problem
All Messages	0x10		Illegal command or format. The adapter does not support this command.
	0x30		Remote station host is not present, disconnected, or shutdown. The SCANport product is disconnected or cannot communicate for some other reason.
All Messages where CMD=0x0F	0xF0	0x01	Illegal Address Format. A field has an illegal value in a logical binary system address.
	0xF0	0x02	Illegal Address Format. Not enough fields specified in a logical binary system address.
	0xF0	0x03	Illegal Address Format. Too many fields specified in a logical binary system address.
	0xF0	0x06	Illegal Address. Address does not exist, or does not point to something usable by this command.
	0xF0	0x0A	Request is too large. Transaction size plus word address is too large. For example, a write of 10 elements to N50:248 would attempt to write data beyond the end of the file.
	0xF0	0x10	No Access. For example, your message tried to write to a read-only location.
	0xF0	0x11	Illegal data type information. For example, an attempt to write floating point data to a numeric [integer] file.

Related documentation

For more information on PCCC messages, refer to the DF1 Protocol and Command Set Reference Manual, publication 1770-6.5.16.

Supported Emulated Block Transfer Commands

Appendix Objectives

Appendix F provides information about supported emulated block transfer commands. You may want to use these to set or obtain information about parameters in the SCANport product connected to the 1203-CN1. This appendix contains the following:

- List of supported emulated block transfer commands.
- Description of the emulated block transfer status word.
- Header and data configurations that you need to set up the data files for the each of the emulated block transfer commands. The header and data values depend on the operation you want to perform.
- Examples of each emulated block transfer command.

This appendix assumes that you have experience using emulated block transfer commands. For more information on messaging, refer to Chapter 6, *Using Messages*.

Supported Emulated Block Transfer Commands

The following table lists the supported emulated block transfer commands and pages in this appendix where you can find more information on them.

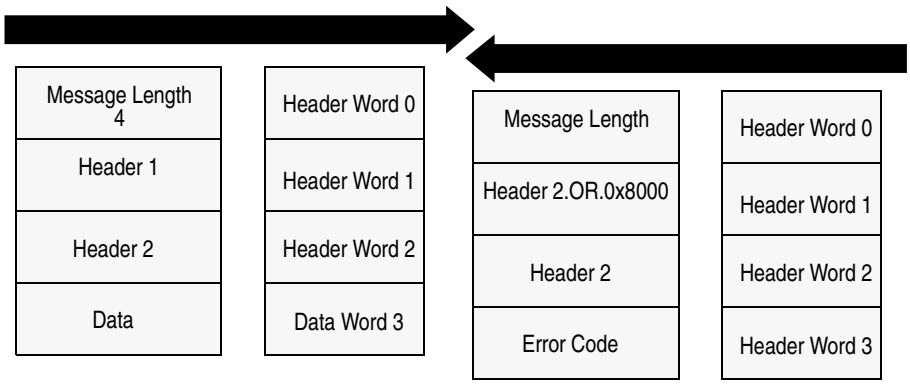
[1] This function can be accessed in the module and product. The following examples describe how to access it in the product. To do so in the module, add 16384 to the decimal value of header word 2.

Emulated Block Transfer Status Word

When an operation is unsuccessful, header word 2 of the drive response contains a negative value (bit 15 = 1).

If an error occurs, the drive also returns a status word to indicate the reason for the failure. The location of the status word is typically header word 3 in the drive response, but will depend on the message.

Example Message Structure and Error Message Reply



The following table lists the error codes.

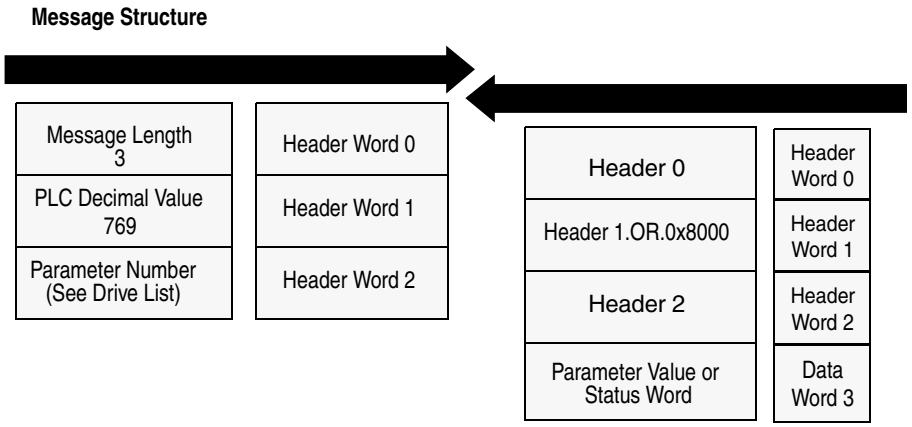
Value	Description
0	No error occurred.
1	The service failed due to an internal reason, and the drive could not perform the request (some messages are read only or write only).
2	The requested service is not supported.
3	An invalid value in the block transfer emulation request header word 2.
4	An invalid value in the block transfer emulation request header word 3.
5	An invalid value in the block transfer emulation request header word 2.
6	The data value is out of range.
7	There is a drive state conflict. The drive is in an incorrect state to perform the function. The drive cannot be running when you perform certain functions.

Parameter Value Read

Parameter Value Read reads the 16-bit parameter data value for the selected parameter number.

PLC Block Transfer Emulation Instruction Data

PLC request instruction length: 3 words
Drive response instruction length: 4 word



Message Operation

Parameter Value Read reads parameter values from the drive and places that value (or an error code) in word 3 of the drive response data file. The value is shown in device units. Device units are listed in the user manual for the device you are using.

If an error occurs:

- Word 3 of the response contains the status code.
- The status area of the data file is non-zero.

Example

In this example, the value of parameter 20 was requested from a 1336 PLUS drive and a value of 4096 was returned. 4096 is the internal drive unit value for the *Maximum Rated Voltage* Parameter. This corresponds to a value of 100% Drive Rated Volts in Display Units.

Data File Format

	0	1	2	3	4	5	6	7	8	9
PLC request	3	769	20*							
Drive response	6	769	20*	4069						

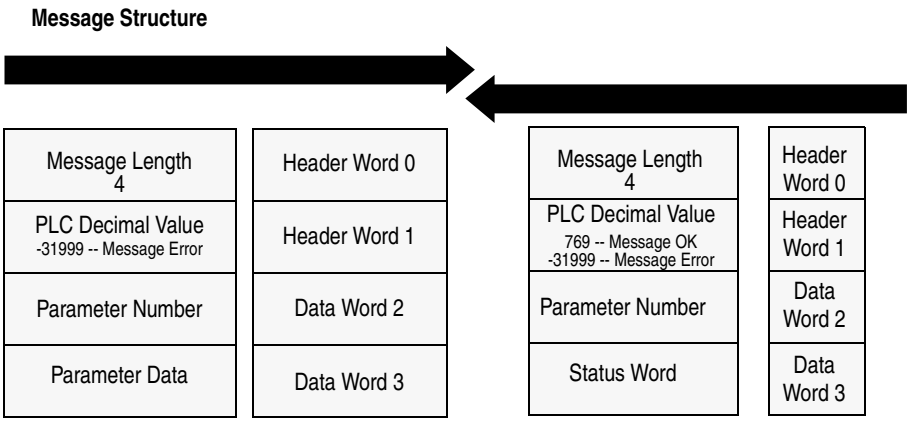
* Example only - These values vary depending on parameters and products.

Parameter Value Write

Parameter Value Write writes a 16-bit parameter data value to the selected parameter number.

PLC Block Transfer Emulation Instruction Data

PLC request instruction length: 4 word
Drive response instruction length: 4 words



Message Operation

Parameter Value Write sends a new value to the specified parameter. The value must be in device units. Units for each parameter are listed in the device manual.

If an error has occurred, word 1 of the response returns a value of -31999, and word 3 contains a status code.

Example

In this example, a value of 4096 was sent to Parameter 20. 4096 is in drive units and indicates a value of 100% Drive Rated Volts, as defined in P147, *Drive Rated Volts*.

Data File Format

	0	1	2	3	4	5	6	7	8	9
PLC request	4	-31999	20*	4096*						
Drive response	3	769	20*	0						

* Example only - These values vary depending on parameters and products.

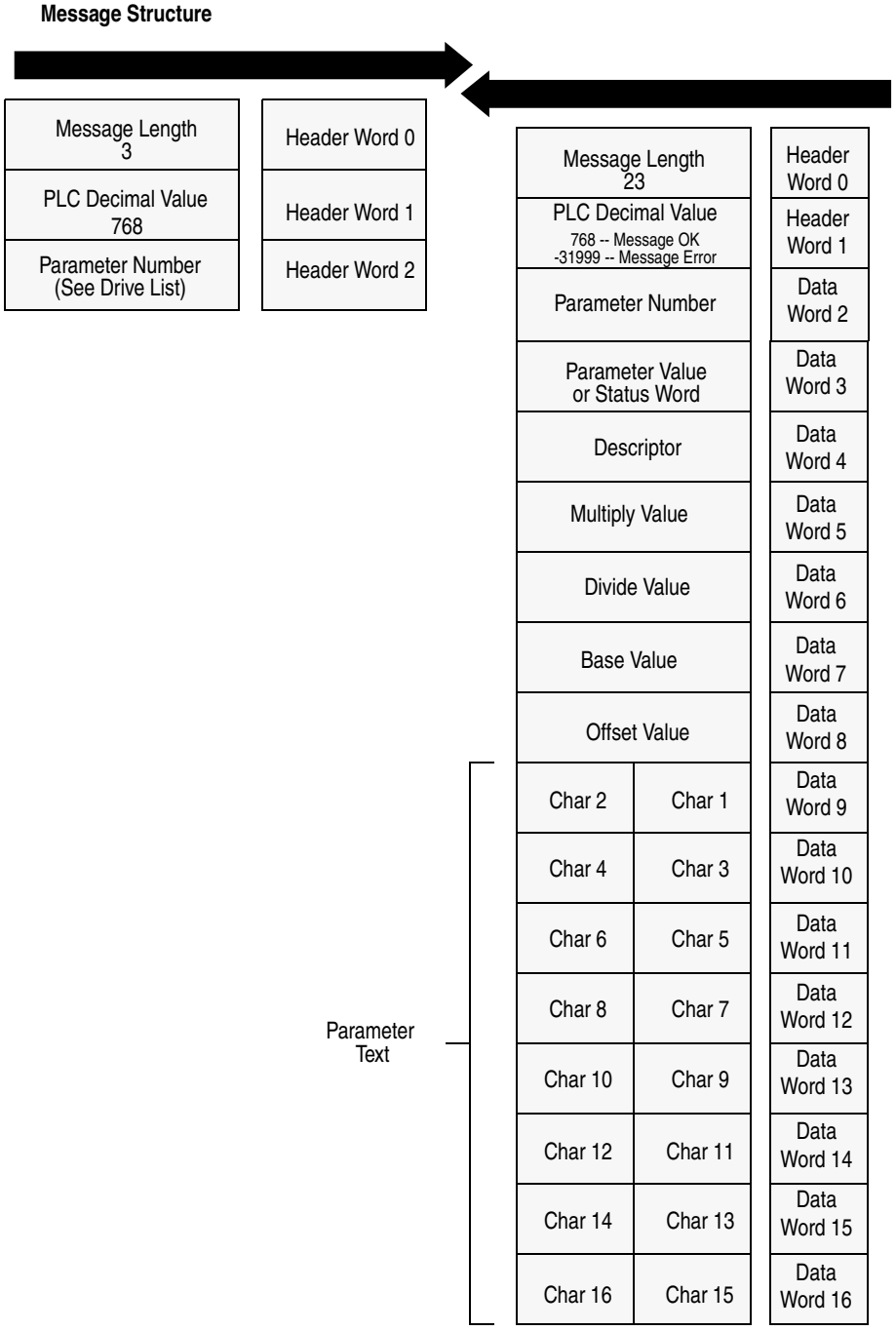
Parameter Read Full

Parameter Read Full provides all known attributes for the parameters requested. This information includes the parameter's current value, descriptor, multiply and divide value, base value, offset value, text string, group element reference, minimum value, maximum value, default value, and unit text string.

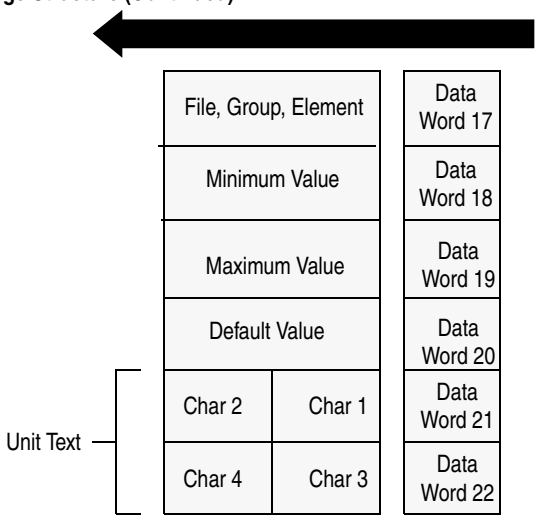
PLC Block Transfer Emulation Instruction Data

PLC request instruction length: 3 words
Drive response instruction length: 23 words

Message Structure



Message Structure (Continued)



Message Operation

Parameter Read Full retrieves the attributes of the specified parameter. The attributes for each parameter include the data, minimum and maximum values, and the parameter text. The response message returns this information.

If an error has occurred in reading any of the values, word 3 contains the status word.

The parameter text is returned with each data word containing two ASCII characters per word. The first and second characters are in opposite order.

Example

In this example, a **Parameter Read Full** was performed through block transfer on a 1336 PLUS drive. N10:10 shows the header message for the request. The data is returned in the response data file, starting with word 3, for parameter 20. Word 3 shows the present value in drive units. Word 4 through word 8 provide scaling information, used to convert drive units to engineering units for the Human Interface Module (HIM). Word 9 through word 16 provide the parameter name.

This example shows the response message in both binary and ASCII. Note the ASCII information beginning with word 9. The parameter name characters return in reverse order for each word. Word 9 has the ASCII value of (aM). To read this, reverse the word to read (Ma). The next word (ix), reversed, gives you (xi). These words, along with the following two words, form the word *Maximum*. You can see the parameter name *Maximum Voltage* in word 9 through word 16 of the response message. In addition, words 21 – 22 are also returned in this format. These words provide the units in which the parameter is defined. In this example it is *vlts*.

Word 17 contains the file, group, and element which are used to reference the parameter.

Words 18 – 20 contain the minimum, maximum, and default values of this parameter.

Data File Format

	0	1	2	3	4	5	6	7	8	9
PLC request	3	768	20*							
Drive response	23	768	20	4096*	355*	1*	4096*	460*	0*	24909*
(Decimal)	27000*	30061*	8301*	28502*	29804*	26465*	8293*	1794*	1024*	4915*
	4096*	27734*	29556*							
Drive response	\00\17	\03\00	\00\14	\10\00	\01 c	\00\01	\10\00	\01\CC	\00\00	a M
(ASCII)	i x	u m	m	o V	t l	g a	e	07 02	04 00	\13 0
	\10\00	l V	s t							

* Example only - These values vary depending on parameters and products.

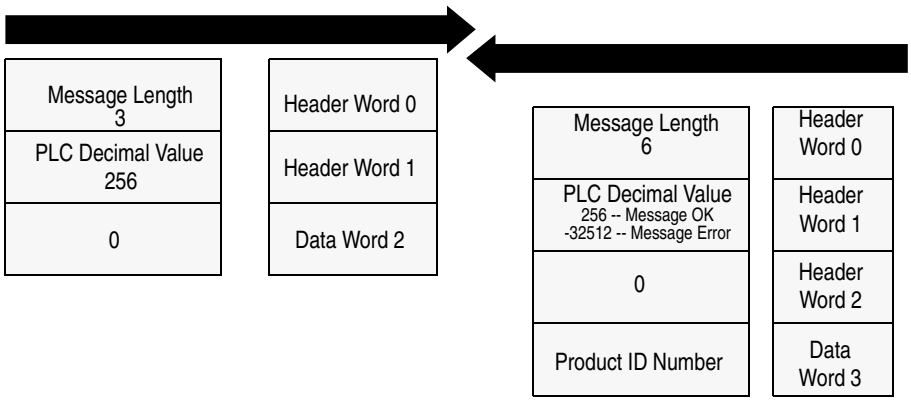
Product ID Number Read

Product ID Number Read returns the product ID of the device to which the 1203-CN1 module is connected.

PLC Block Transfer Emulation Instruction Data

PLC request instruction length: 3 words
Drive response instruction length: 4 words

Message Structure



Product Code (Hex)	Product Code (Decimal)	Bulletin Number	Product
0x02	2	1336S	1336 PLUS Fractional HP
0x03	3	1336S	1336 PLUS
0x07	7	1336F	1336 PLUS II
0x10	16	1336T	1336 FORCE w/PLC Adapter Board
0x11	17	2364F	2364 RGU DC Bus Regen Front End
0x12	18	1394	1394 Motion Drive
0x13	19	1557	1557 Medium Voltage AC Drive
0x14	20	193	SMP-3
0x15	21	150	SMC Dialog Plus
0x17	23	1305	1305 AC Drive
0x18	24	1397	1397 DC Drive
0x19	25	1336R	1336 Line Regeneration Package
0x20	32	1336T	1336 FORCE w/Standard Adapter Board
0x22	34	1336E	1336 IMPACT

Message Operation

Product ID Number Read, through the drive response message word 3, indicates the type of device the 1203-CN1 module is connected to. This value is defined in the message response chart shown above.

If an error has occurred, word 1 of the response returns a negative value of -32512.

Example

In this example, the **Product ID Number Read** was requested. The drive response contained a value of 3 in word 3 of its message response, indicating a connection to a 1336 PLUS drive.

Data File Format

	0	1	2	3	4	5	6	7	8	9
PLC request	3	256	0							
Drive Response	6	256	0	3*						

* Example only - These values vary depending on parameters and products.

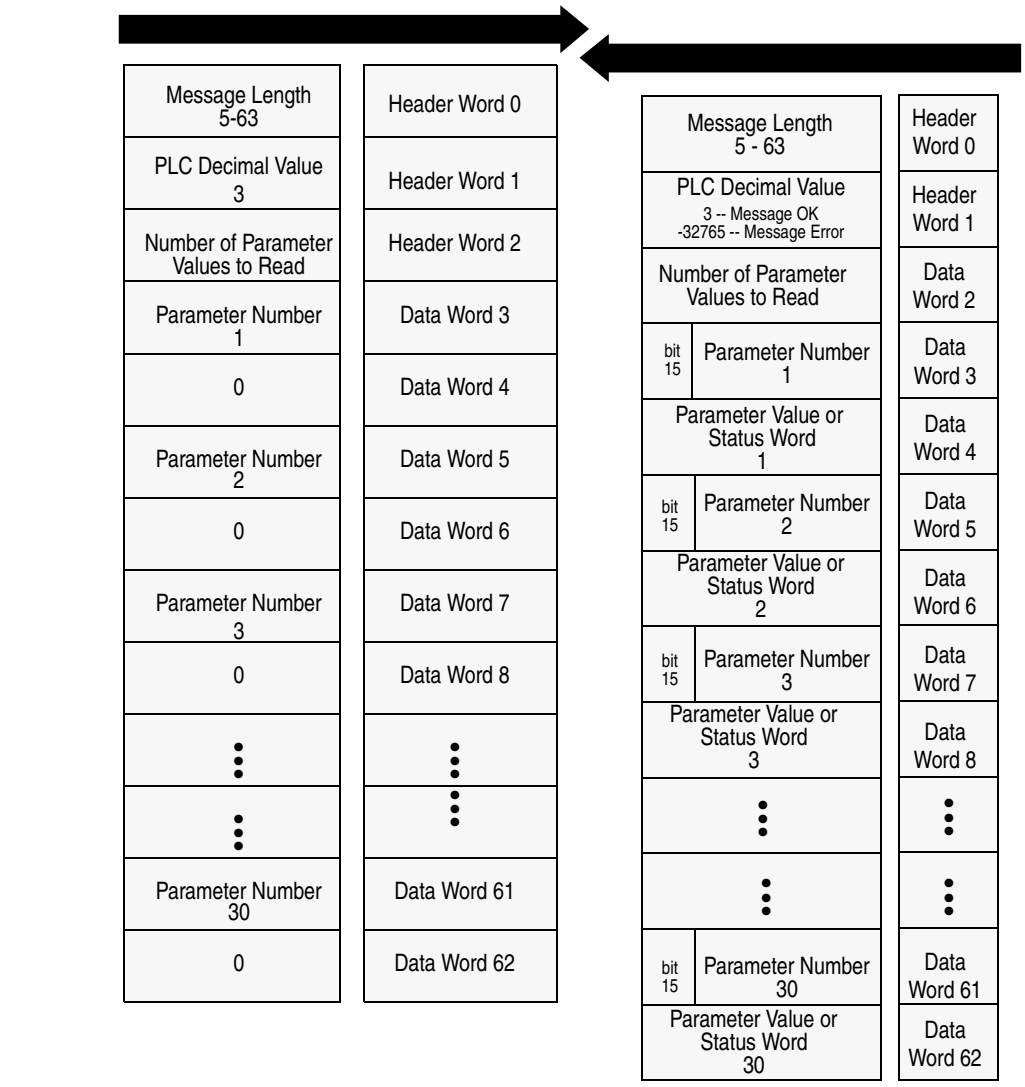
Scattered Parameter Value Read

Scattered Parameter Value Read reads a scattered list of parameters.

PLC Block Transfer Emulation Instruction Data

PLC request instruction length: 5-63 words
Drive response instruction length: 5-63 words

Message Structure



Message Operation

Scattered Parameter Value Read reads a pre-defined group of parameter values, in any order, from the device. You define the number of parameters to read in word 2 of the request. The parameters to be read and their order are defined starting with word 3. An unused word is left between each parameter request, so the drive can respond with the parameter value, as shown.

If an error has occurred in reading any of the parameters:

- Word 1 of the drive response returns a value of -32765.
- Bit 15 of the drive response word for the number of that parameter is set.
- The drive response word for the value of that parameter returns a status word instead of returning the parameter value.

Example

In this example, eight parameters were read from a 1336 PLUS drive, as defined in word 2 of the request. Parameter numbers 5, 7, 8, 20, 18, 17, 19, and 36 were requested. The drive response returned the values of these parameters in the data file. These values are in drive units.

Data File Format

	0	1	2	3	4	5	6	7	8	9
PLC request	19	3	8*	5*	0	7*	0	8*	0	20*
	0	18*	0*	17*	0	19*	0	36*	0	
Drive response	19	3	8*	5*	6*	7*	1000*	8*	1000*	20*
	4096*	18*	4096*	17*	51*	19*	60*	36*	6144*	

* Example only - These values vary depending on parameters and products.

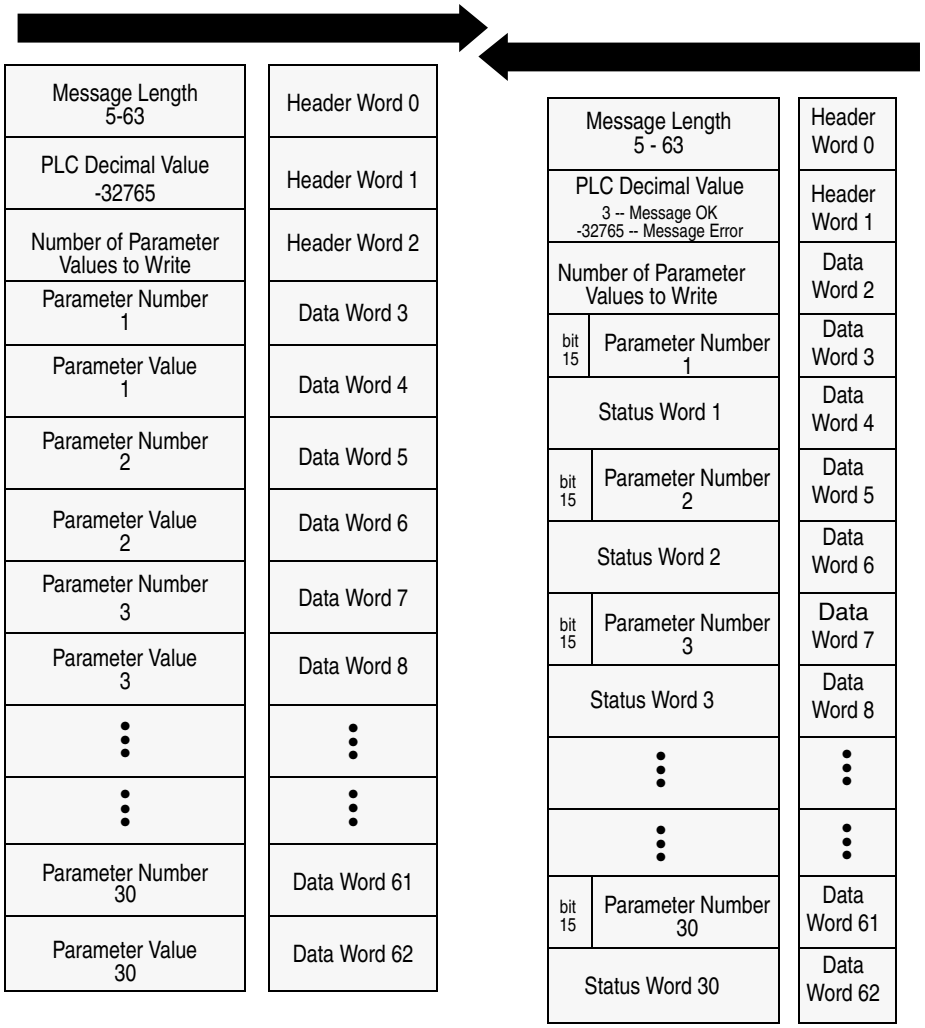
Scattered Parameter Value Write

Scattered Parameter Value Write writes to a scattered list of parameters and returns the status of each parameter. If any of the states have errors, the parameter number is negative.

PLC Block Transfer Emulation Instruction Data

PLC request instruction length: 5-63 words
Drive response instruction length: 5-63 words

Message Structure



Message Operation

Scattered Parameter Value Write writes data values to a pre-defined group of device parameters in any order. You define the number of parameters to write in word 2. The parameters to be written to and their order are defined starting with word 3.

If an error occurs while writing to any of the parameters:

- Word 1 of the drive response returns a value of -32765.
- Bit 15 of the drive response word for that parameter's number is set.
- The drive response word for that parameter's status word is non-zero.

If no error has occurred:

- Word 1 of the drive response returns a value of 3.
- Each of the drive response's parameter numbers are the same as in the request.
- Each of the drive response status words returns a value of 0.

Example

In this example, six parameters were written to in a 1336 PLUS drive. Word 2 of the request defines the number of parameter values that are transferred. Beginning with word 3, the message lists each parameter number followed by the value of the parameter. The values are entered in device units.

The drive response returns the status of each parameter write. If the request was successful, a zero is returned. If an error has occurred, the response returns a status word code for the error.

Data File Format

	0	1	2	3	4	5	6	7	8	9
PLC request	15	-32765	6*	90*	1*	150*	4*	30*	20*	31*
	10*	10*	2*	12*	5*					
Drive response	15	3	6*	90*	0*	150*	0*	30*	0*	31*
	0*	10*	0*	12*	0*					

* Example only - These values vary depending on parameters and products.

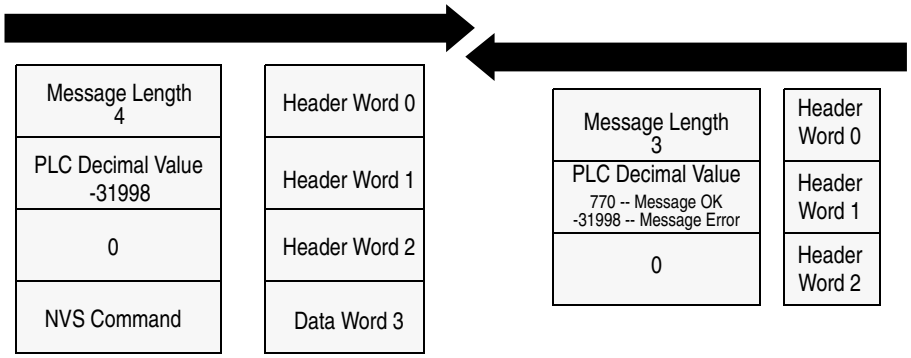
NVS Functions

NVS (Non-Volatile Storage) Functions activates an NVS function.

PLC Block Transfer Emulation Instruction Data

PLC request instruction length: 4 words
Drive response instruction length: 3 words

Message Structure



Value	NVS Command
00	Not Used
01	NVS Save
02	NVS Recall
03	NVS Default Initialize

Message Operation

The NVS storage function allows three different message requests:

- NVS Save saves parameter information from the working memory or RAM to NVS Storage.
- NVS Recall retrieves the last saved data from NVS Storage and places it in the working memory or RAM.
- NVS Default Initialize clears the RAM and NVS Storage and sets all parameter values to default.

If an error has occurred, response word 1 returns a value of -31998.

Example

This example requests the NVS Storage Save function be performed.

Data File Format

	0	1	2	3	4	5	6	7	8	9
PLC request	4	-31998	0*	0, 1, 2, or 3						
Drive response	3	770	0*							

* Example only - These values vary depending on parameters and products.

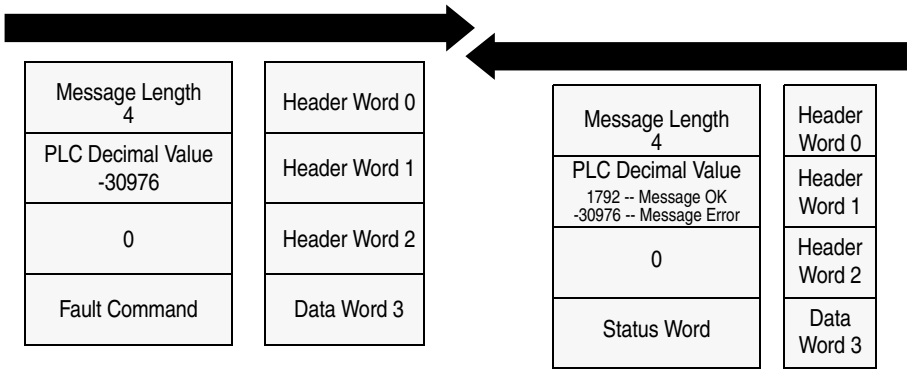
Fault Command Write

Fault Command Write activates the Clear Fault, Clear Fault Queue, and Drive Reset functions.

PLC Block Transfer Emulation Instruction Data

PLC request instruction length: 4 words
Drive response instruction length: 4 words

Message Structure



Value	Fault Command
00	Not Used
01	Clear Fault
02	Clear Fault Queue
03	Drive Reset (1336 FORCE Only)

Message Operation

The specified fault Clear/Reset function sends a fault handling request to the device.

- A Clear Fault request clears the last fault that occurred.
- A Clear Fault Queue clears the entire fault buffer. Certain devices may store more than one fault.
- A Drive Reset is used with the 1336 FORCE drive product only. This function resets the drive, clears the fault queue, and writes the parameter information stored in NVS Storage to RAM.

Data File Format

	0	1	2	3	4	5	6	7	8	9
PLC request	4	-30976	0	0, 1, 2, or 3						
Drive response	4	1792	0	0*						

* Example only - These values vary depending on parameters and products.

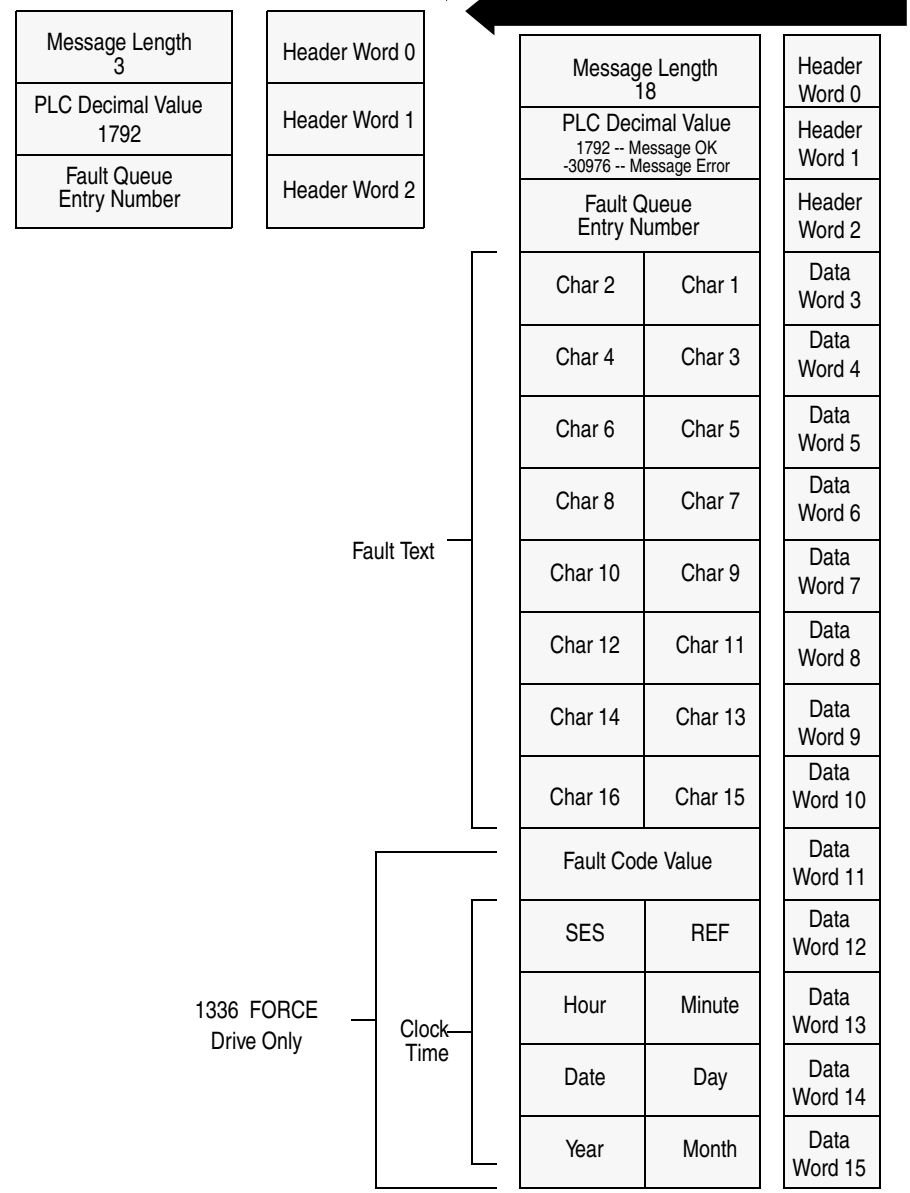
Fault Queue Entry Read Full

Fault Queue Entry Read Full reads the contents of the specified fault queue entry. A message is returned which includes the fault text and fault code associated with the specified fault queue entry. The 1336 FORCE drive also returns the time stamp associated with the fault.

PLC Block Transfer Emulation Instruction Data

PLC request instruction length: 3 words
Drive response instruction length: 12 or 16 words

Message Structure



Message Operation

Fault Queue Entry Read Full reads the contents of the fault queue specified in word 3 of the request. The response returns the fault text which can be ASCII text. Every two characters of text are in reverse order. Also, the 1336 FORCE drive returns a time stamp, indicating the day and time the fault occurred.

If an error has occurred, word 1 of the response returns a negative value.

Example

In this example, Fault Queue Entry number 3 was retrieved from a 1336 PLUS drive. The drive response returned the ASCII text *Drive Reset Flt*, with each character reversed. The fault code for this example is 22.

Data File Format

	0	1	2	3	4	5	6	7	8	9
PLC request	3	1792	3*							
Drive response	18	1792	3*	29252*	20313*	8293*	25938*	25971*	8308*	27718*
	8303*	22*								
Drive response	\00\12	\07\00	\03\00	r D	v i	e	e R	e s	t	l F
	t	\00\16								

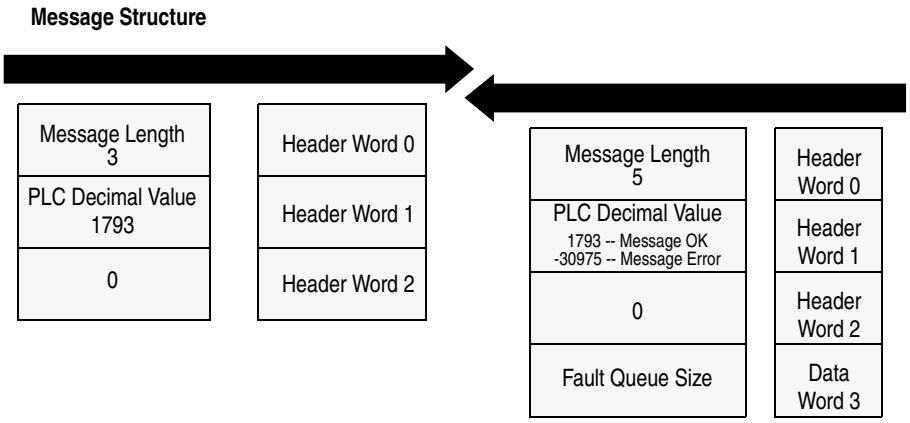
* Example only - These values vary depending on parameters and products.

Fault Queue Size

Fault Queue Size gets the number of fault entries allowed in the fault queue.

PLC Block Transfer Emulation Instruction Data

PLC request instruction length: 3 words
Drive response instruction length: 4 words



Message Operation

Fault Queue Size reads back the size of the fault queue available in the product. Each product may have a different number of fault queue entries available for storage.

If an error has occurred, word 1 of the response returns a value of -30975.

Example

In this example, a 1336 PLUS drive was used. This product has a fault queue of four storage locations available to store faults. This value is seen in word 3 of the response header message.

Data File Format

	0	1	2	3	4	5	6	7	8	9
PLC request	3	1793	0							
Drive response	6	1793	0	4*						

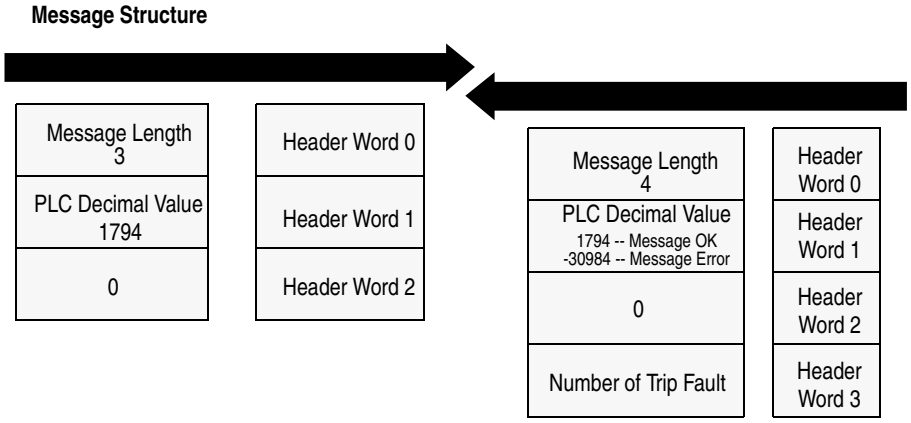
* Example only - These values vary depending on parameters and products.

Trip Fault Queue Number

Trip Fault Queue Number provides the fault queue number of the fault that caused the device to trip.

PLC Block Transfer Emulation Instruction Data

PLC request instruction length: 3 words
Drive response instruction length: 4 words



Message Operation

Trip Fault Queue Number provides the number of the entry in the fault queue that tripped the device in word 3 of the drive response. The fault queue number is 0 when the device is not faulted.

If an error has occurred in the block transfer, word 1 of the response is negative.

Example

In this example, the device has stored a fault in the first entry of the fault queue that caused the drive to trip. Word 3 of the response indicates the entry number.

Data File Format

	0	1	2	3	4	5	6	7	8	9
PLC request	3	1794	0							
Drive response	6	1794	0	1*						

* Example only - These values vary depending on parameters and products.

Notes:

Notes:

We Want Our Manuals to be the Best!

You can help! Our manuals must meet the needs of you, the user. This is your opportunity to make sure they do just that. By filling out this form you can help us provide the most useful, thorough, and accurate manuals available. Please take a few minutes to tell us what you think - then mail this form or FAX it.
FAX: your local Allen-Bradley Sales Office or 414/512-8579

PUBLICATION NAME _____

PUBLICATION NUMBER, DATE AND PART NUMBER (IF PRESENT) _____

✔ **CHECK THE FUNCTION THAT MOST CLEARLY DESCRIBES YOUR JOB.**

- ❑ SUGGEST/RESPONSIBLE FOR THE PURCHASE OF EQUIPMENT
- ❑ DESIGN/IMPLEMENT ELECTRICAL SYSTEMS
- ❑ SUPERVISE FLOOR OPERATIONS
- ❑ MAINTAIN/OPERATE PROGRAMMABLE MACHINERY
- ❑ TRAIN/EDUCATE MACHINE USERS

✔ **WHAT LEVEL OF EXPERIENCE DO YOU HAVE WITH EACH OF THE FOLLOWING PRODUCTS?**

	NONE	LITTLE	MODERATE	EXTENSIVE
PROGRAMMABLE CONTROL	❑	❑	❑	❑
AC/DC DRIVES	❑	❑	❑	❑
PERSONAL COMPUTERS	❑	❑	❑	❑
NC/CNC CONTROLS	❑	❑	❑	❑
DATA COMMUNICATIONS/LAN	❑	❑	❑	❑

✔ **RATE THE OVERALL QUALITY OF THIS MANUAL BY CIRCLING YOUR RESPONSE BELOW. (1) = POOR (5) = EXCELLENT**

HELPFULNESS OF INDEX/TABLE OF CONTENTS	1	2	3	4	5
CLARITY	1	2	3	4	5
EASE OF USE	1	2	3	4	5
ACCURACY AND COMPLETENESS	1	2	3	4	5
QUALITY COMPARED TO OTHER COMPANIES' MANUALS	1	2	3	4	5
QUALITY COMPARED TO OTHER ALLEN-BRADLEY MANUALS	1	2	3	4	5

✔ **WHAT DID YOU LIKE MOST ABOUT THIS MANUAL?**

✔ **WHAT DID YOU LIKE LEAST ABOUT THIS MANUAL?**

✔ **PLEASE LIST ANY ERRORS YOU FOUND IN THIS MANUAL (REFERENCE PAGE, TABLE, OR FIGURE NUMBERS).**

✔ **DO YOU HAVE ANY ADDITIONAL COMMENTS?**

✔ **COMPLETE THE FOLLOWING.**

NAME _____ COMPANY _____

TITLE _____ DEPARTMENT _____

STREET _____ CITY _____ STATE _____ ZIP _____

TELEPHONE _____ DATE _____

CUT ALONG DOTTED LINE

FOLD HERE

FOLD HERE

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 413 MEQUON, WI

POSTAGE WILL BE PAID BY ADDRESSEE

ALLEN-BRADLEY

Attn: Marketing Communications
P.O. Box 760
Mequon, WI 53092-9907

www.rockwellautomation.com

Corporate Headquarters
Rockwell Automation, 777 East Wisconsin Avenue, Suite 1400, Milwaukee, WI, 53202-5302 USA, Tel: (1) 414.212.5200, Fax: (1) 414.212.5201

Headquarters for Allen-Bradley Products, Rockwell Software Products and Global Manufacturing Solutions
Americas: Rockwell Automation, 1201 South Second Street, Milwaukee, WI 53204-2496 USA, Tel: (1) 414.382.2000, Fax: (1) 414.382.4444
Europe: Rockwell Automation SA/NV, Vorstlaan/Boulevard du Souverain 36-BP 3A/B, 1170 Brussels, Belgium, Tel: (32) 2 663 0600, Fax: (32) 2 663 0640
Asia Pacific: Rockwell Automation, 27/F Citicorp Centre, 18 Whitfield Road, Causeway Bay, Hong Kong, Tel: (852) 2887 4788, Fax: (852) 2508 1846

Headquarters for Dodge and Reliance Electric Products
Americas: Rockwell Automation, 6040 Ponders Court, Greenville, SC 29615-4617 USA, Tel: (1) 864.297.4800, Fax: (1) 864.281.2433
Europe: Rockwell Automation, Brühlstraße 22, D-74834 Elztal-Dallau, Germany, Tel: (49) 6261 9410, Fax: (49) 6261 1774
Asia Pacific: Rockwell Automation, 55 Newton Road, #11-01/02 Revenue House, Singapore 307987, Tel: (65) 351 6723, Fax: (65) 355 1733

Publication 1203-5.13 – February, 2002
Supercedes 1203-5.13 – July, 1998
P/N 189939-P02