



Where Automation Connects.



## **MVI56E-GSCXT**

**ControlLogix Platform**

Generic ASCII Serial Communication  
Module

July 13, 2011

**SETUP GUIDE**

## **Your Feedback Please**

We always want you to feel that you made the right decision to use our products. If you have suggestions, comments, compliments or complaints about our products, documentation, or support, please write or call us.

## **How to Contact Us**

### **ProSoft Technology**

5201 Truxtun Ave., 3rd Floor

Bakersfield, CA 93309

+1 (661) 716-5100

+1 (661) 716-5101 (Fax)

[www.prosoft-technology.com](http://www.prosoft-technology.com)

[support@prosoft-technology.com](mailto:support@prosoft-technology.com)

**Copyright © 2011 ProSoft Technology, Inc., all rights reserved.**

MVI56E-GSCXTXT Setup Guide

July 13, 2011

ProSoft Technology<sup>®</sup>, ProLinx<sup>®</sup>, inRAx<sup>®</sup>, ProTalk<sup>®</sup>, and RadioLinx<sup>®</sup> are Registered Trademarks of ProSoft Technology, Inc. All other brand or product names are or may be trademarks of, and are used to identify products and services of, their respective owners.

## **ProSoft Technology<sup>®</sup> Product Documentation**

In an effort to conserve paper, ProSoft Technology no longer includes printed manuals with our product shipments. User Manuals, Datasheets, Sample Ladder Files, and Configuration Files are provided on the enclosed CD-ROM, and are available at no charge from our web site: [www.prosoft-technology.com](http://www.prosoft-technology.com)

## Important Safety Information

### North America Warnings

- A** Warning - Explosion Hazard - Substitution of components may impair suitability for Class I, Division 2.
- B** Warning - Explosion Hazard - When in hazardous locations, turn off power before replacing or rewiring modules.  
Warning - Explosion Hazard - Do not disconnect equipment unless power has been switched off or the area is known to be nonhazardous.

### Warnings and Conditions of Safe Usage

Power, Input, and Output (I/O) wiring must be in accordance with the authority having jurisdiction

- A** Warning - Explosion Hazard - When in hazardous locations, turn off power before replacing or wiring modules.
- B** Warning - Explosion Hazard - Do not disconnect equipment unless power has been switched off or the area is known to be non-hazardous.
- C** These products are intended to be mounted in an IP54 enclosure.
- D** DO NOT OPEN WHEN ENERGIZED.

### Electrical Ratings

- Backplane Current Load: 800 mA @ 5 Vdc; 3 mA @ 24 Vdc
- Operating Temperature: -13°F to 158°F (-25°C to 70°C)
- Storage Temperature: -40°C to 85°C (-40°F to 185°F)
- Shock: 30 g operational; 50 g non-operational; Vibration: 5 g from 10 Hz to 150 Hz
- Relative Humidity 5% to 95% (without condensation)
- All phase conductor sizes must be at least 1.3 mm (squared) and all earth ground conductors must be at least 4mm (squared).

## Agency Approvals and Certifications

| Agency | Applicable Standards |
|--------|----------------------|
|--------|----------------------|

There are currently no certifications for the MVI56E-GSCXTXT For more information, contact ProSoft Technology at +1 661.716.5100 or [www.prosoft-technology.com](http://www.prosoft-technology.com).

## Battery Life Advisory

**Note:** Modules manufactured after April 1<sup>st</sup>, 2011 do not contain a battery. For modules manufactured before that date the following applies:

The module uses a rechargeable Lithium Vanadium Pentoxide battery to back up the real-time clock and CMOS settings. The battery itself should last for the life of the module. However, if left in an unpowered state for 14 to 21 days, the battery may become fully discharged and require recharging by being placed in a powered-up ControlLogix chassis. The time required to fully recharge the battery may be as long as 24 hours.

Once it is fully charged, the battery provides backup power for the CMOS setup and the real-time clock for approximately 21 days. Before you remove a module from its power source, ensure that the battery within the module is fully charged (the BATT LED on the front of the module goes OFF when the battery is fully charged). If the battery is allowed to become fully discharged, the module will revert to the default BIOS and clock settings.

**Note:** The battery is not user-replaceable or serviceable.

# Contents

|  |           |
|--|-----------|
| Your Feedback Please.....  | 2         |
| How to Contact Us.....   | 2         |
| ProSoft Technology® Product Documentation.....                           | 2         |
| Important Safety Information.....  | 3         |
| Battery Life Advisory.....   | 4         |
| <b>1 Scope</b>   | <b>7</b>  |
| 1.1 Learning Objectives.....   | 8         |
| 1.2 ProSoft Technology Documentation.....                                | 9         |
| 1.3 Prerequisites.....   | 10        |
| <b>2 Before You Begin</b>  | <b>11</b> |
| 2.1 What's New?.....   | 12        |
| 2.2 What's Different?.....   | 13        |
| 2.3 System Requirements.....   | 14        |
| 2.4 Required Items.....  | 15        |
| 2.5 Sample Files.....  | 16        |
| <b>3 The Sample Application</b>  | <b>17</b> |
| 3.1 Overview.....  | 18        |
| 3.2 Architecture.....  | 19        |
| 3.3 Data Flow.....   | 20        |
| <b>4 Procedures</b>  | <b>23</b> |
| 4.1 Required Steps.....  | 24        |
| 4.2 Physical Setup.....  | 25        |
| 4.2.1 Setting Module Jumpers.....  | 25        |
| 4.2.2 Installing the Module in the Rack.....                             | 26        |
| 4.2.3 Connecting Your PC to the Module's Ethernet Port.....              | 27        |
| 4.2.4 Connecting the MVI56E-GSCXT and ControlLogix PAC Serial Ports..... | 28        |
| 4.3 Using ProSoft Configuration Builder Software.....                    | 29        |
| 4.3.1 Setting Up the Project.....  | 30        |
| 4.3.2 Setting Up a Temporary IP Address.....                             | 32        |
| 4.3.3 Setting Up a Permanent IP Address.....                             | 35        |
| 4.4 Loading the Sample Application.....                                  | 47        |
| 4.5 Sample Application Setup.....  | 48        |
| 4.5.1 Configuring the ControlLogix (CLX) Serial Port.....                | 48        |
| 4.5.2 Configuring the MVI56E-GSCXT Serial Port.....                      | 50        |
| 4.5.3 What's In the Sample Application?.....                             | 51        |
| 4.6 Downloading the Sample Program to the Processor.....                 | 52        |
| 4.7 Rebooting the Module to Transfer New Parameter Values.....           | 53        |
| 4.8 Verifying Communication.....   | 54        |
| 4.8.1 Viewing Exchanged Data.....  | 54        |
| 4.8.2 Checking Module Status in RSLogix5000 Controller Tags.....         | 63        |

---

|          |  |           |
|----------|--|-----------|
| 4.8.3    | Scrolling LED Status Indicators .....  | 65        |
| <b>5</b> | <b>Building on Success</b>   | <b>67</b> |
| 5.1      | Frequently Asked Questions.....  | 68        |
| 5.1.1    | What are the differences between the MVI56 and the MVI56E modules? What does the "E" stand for?.....                   | 68        |
| 5.1.2    | Is the MVI56E product a direct replacement to my existing MVI56 product?.....  | 68        |
| 5.1.3    | How is the MVI56E-GSCXT configured?.....   | 68        |
| 5.1.4    | What is ProSoft Configuration Builder (PCB)? .....   | 69        |
| 5.1.5    | What is ProSoft Discovery Service (PDS)? .....   | 69        |
| 5.1.6    | What is the purpose of the MVI56E-GSCXT Ethernet Config (E1) Port? .....   | 69        |
| 5.1.7    | How do I change the module's IP address? .....   | 69        |
| 5.1.8    | Does the MVI56E-GSCXT module require processor logic? .....  | 69        |
| 5.1.9    | What is the purpose of the Optional MVI56E-GSCXT Add-On Instruction? .....   | 70        |
| 5.1.10   | How do I monitor MVI56E-GSCXT operation? .....   | 70        |
| 5.1.11   | Are there any other ways to monitor module diagnostics besides being connected to the module's network (subnet)? ..... | 70        |
|          | <b>Glossary of Terms</b>   | <b>71</b> |
|          | <b>Index</b>   | <b>75</b> |

---

# 1 Scope

## *In This Chapter*

- ❖ Learning Objectives..... 8
- ❖ ProSoft Technology Documentation..... 9
- ❖ Prerequisites ..... 10

This document acts as a tutorial, providing step-by-step instructions on how to read and write bi-directional data from one network device to another network device using the MVI56E-GSCXT.

## 1.1 Learning Objectives

When you have completed all the steps in this Setup Guide, you will have learned how to:

- Install the MVI56E-GSCXT setup and diagnostic software
- Install the MVI56E-GSCXT module (page 25)
- Load the Sample Application file (page 47)
- Verify Sample Application Setup (page 48)
- Download the Sample Application to the processor (page 52)
- Verify the MVI56E-GSCXT module communication status (page 54)

## 1.2 ProSoft Technology Documentation

ProSoft Technology provides the following documentation (manuals) with your MVI56E-GSCXT.

### **Documentation, tools, and product support**

- **Web Site Support:** Visit the ProSoft Technology web site at [www.prosoft-technology.com](http://www.prosoft-technology.com) to download additional documentation, tools, and application information.
- **Email Technical Support:** Send your support questions to [Support@prosoft-technology.com](mailto:Support@prosoft-technology.com).
- **Telephone Support:** Please call ProSoft Technology Technical Support at: (Country Code 1+) 661-716-5100. Support is available 24 hours a day, 7 days a week. ProSoft Technology telephone support is free and unlimited.

### 1.3 Prerequisites

To get the most benefit from this *Setup Guide*, you should have the following skills:

- **Rockwell Automation® RSLogix™ 5000 software:** launch the program, configure, and transfer the Add-On Instruction (or ladder logic) Sample Application program to the processor
- **Microsoft Windows®:** install and launch programs, execute menu commands, navigate dialog boxes and enter data.
- **Serial data communication:** correctly configure data communication parameters such as baud rate, parity, data bits, and so on, using the documentation for the devices connected to the network
- **Ethernet networking:** connect the MVI56E-GSCXT module to an Ethernet network using a valid IP address and subnet mask
- **Hardware installation and wiring:** install the module and safely connect Generic ASCII Serial and ControlLogix devices to a power source and to the MVI56E-GSCXT module's serial ports

## 2 Before You Begin

### *In This Chapter*

|                             |    |
|-----------------------------|----|
| ❖ What's New? .....         | 12 |
| ❖ What's Different? .....   | 13 |
| ❖ System Requirements ..... | 14 |
| ❖ Required Items .....      | 15 |
| ❖ Sample Files.....         | 16 |

## 2.1 What's New?

MVI56E products are **backward compatible** with existing MVI56 products, ladder logic, and module configuration files already in use. Easily swap and upgrade to benefit from an array of new features designed to improve interoperability and enhance ease of use.

- **ProSoft Configuration Builder (PCB):** Microsoft Windows®-based utility software for diagnostics. Connect through the module's Ethernet port or use CIPconnect® to access troubleshooting features and functions.
- **ProSoft Discovery Service (PDS):** New Windows-based utility software to find and display a list of MVI56E modules on the network and to temporarily change a module's IP address to be able to connect with a module's web page.
- **CIPconnect-enabled:** Allows PC-to-module diagnostics from the Ethernet network through a ControlLogix® 1756-ENxT EtherNet/IP™ module.
- **Personality Card:** An industrial-grade compact flash memory card storing the module's Ethernet settings, allowing quick and easy replacement.
- **LED Scrolling Diagnostic Display:** 4-character, alphanumeric display, providing English messages for status and alarm data, and for processor and network communication status.
- **XT series for Extreme Environments:** The MVI56E-GXT is part of the new XT series, designed to work at extreme temperatures and in harsh or caustic environments. XT series modules operate over a wider temperature range than the standard MVI56E series. The XT series also comes with conformal coating to protect module components from corrosive environmental elements.

## 2.2 What's Different?

The MVI56E-GSCXT Generic ASCII Serial Communication module is configured in RSLogix™ 5000 software using the sample ladder or Add-On Instruction (AOI). The product CD-ROM in the box contains two installation programs for ProSoft Discovery Service (PDS) and ProSoft Configuration Builder (PCB), as well as all required product documentation.

- PDS is the software utility used to allow your PC to connect to the module to set a temporary Ethernet IP address. Then you can connect to the module's web page to retrieve or change the module's firmware through an Ethernet link.
- PCB is the software used to provide access to the module's diagnostic menus and application serial port communication data analyzer features.

## 2.3 System Requirements

The MVI56E-GSCXT module requires the following minimum hardware and software components:

- Rockwell Automation ControlLogix<sup>®</sup> processor (firmware version 10 or higher), with compatible power supply, and one free slot in the rack for the MVI56E-GSCXT module. The module requires 800 mA of available 5 Vdc power
- Rockwell Automation RSLogix 5000 programming software
  - Version 16 or higher required for Add-On Instruction
  - Version 15 or lower must use Sample Ladder, available from [www.prosoft-technology.com](http://www.prosoft-technology.com)
- Rockwell Automation RSLinx<sup>®</sup> communication software version 2.51 or higher
- ProSoft Configuration Builder (PCB) (included)
- ProSoft Discovery Service (PDS) (included in PCB)
- Pentium<sup>®</sup> II 450 MHz minimum. Pentium III 733 MHz (or better) recommended
- Supported operating systems:
  - Microsoft Windows<sup>®</sup> Vista
  - Microsoft Windows XP Professional with Service Pack 1 or 2
  - Microsoft Windows 2000 Professional with Service Pack 1, 2, or 3
  - Microsoft Windows Server 2003
- 128 Mbytes of RAM minimum, 256 Mbytes of RAM recommended
- 100 Mbytes of free hard disk space (or more based on application requirements)
- 256-color VGA graphics adapter, 800 x 600 minimum resolution (True Color 1024 × 768 recommended)
- CD-ROM drive

**Note:** The Hardware and Operating System requirements in this list are the minimum recommended to install and run software provided by ProSoft Technology<sup>®</sup>. Other third party applications may have different minimum requirements. Refer to the documentation for any third party applications for system requirements.

**Note:** You can install the module in a local or remote rack. For remote rack installation, the module requires EtherNet/IP or ControlNet communication with the processor.

## 2.4 Required Items

This Setup Guide uses a sample application that shows you how to establish communication between the MVI56E-GSCXT module application port and the serial port of a ControlLogix Programmable Automation Controller (CLX PAC). The sample application uses the following equipment.

| Item  | Description  |
|---|--|
| MVI56E-GSCXT  | Generic ASCII Serial Communicaton Module   |
| 1756-L63  | ControlLogix processor   |
| 1756-ENET, EWEB, ENBT or EN2T   | ControlLogix EtherNet/IP Communication Module  |
| 1756-A4/A, B<br>1756-A7/A, B<br>1756-A10/A, B<br>1756-A13/A, B<br>1756-A17/A, B | ControlLogix Chassis with four or seven slots  |
| 1756-PA7 5/A, B, C<br>1756-PA7 2/A, B, C  | ControlLogix Chassis Power Supply  |
| CPU434 12A  | Quantum processor (Modbus device)  |
| Ethernet cable  | Ethernet cable to connect MVI56E-GSCXT module to Ethernet network for diagnostics (supplied with the module) |
| Null-modem serial cable with RJ45-to-DB9M adapter cable                         | Serial and adapter cables to connect MVI56E-GSCXT module to serial port of the CLX PAC                       |

## 2.5 Sample Files

The sample application file has all the logic required for the procedures found in this Setup Guide.

| Item                          | Description   |
|-------------------------------|---|
| MVI56EGSC_AddOn_Rung_v1_3.L5X | Sample RSLogix500 .ACD project file with pre-configured Sample Application tags and logic. (Which also includes basic Sample AOI) |

The sample application assumes you are using RSLogix 5000 version 16 (or later), which supports Add-On Instructions. The sample file contains all elements required for the MVI56E-GSCXT module to perform the communication verification tests found later in this Setup Guide.

- User-defined Data Types (UDTs)
- Add-On Instruction (AOI)
- Ladder rung with AOI
- Controller tags
- Additional ladder rungs to control the ControlLogix processor ASCII serial port

The sample application .ACD project file is located on the distribution CD-ROM, along with the other generic ladder samples.

**Note:** For RSLogix v15 (or older) applications, please refer to the *MVI56E-GSCXT User Manual* for information on how to use the generic sample ladder logic. If you must use an older version of RSLogix and want to create the equivalent of this AOI Sample Application logic, you can add the controller tags and ladder logic rungs (page 48) found later in this chapter to the generic Version 13 or 15 samples found on the CD-ROM.

## 3 The Sample Application

### *In This Chapter*

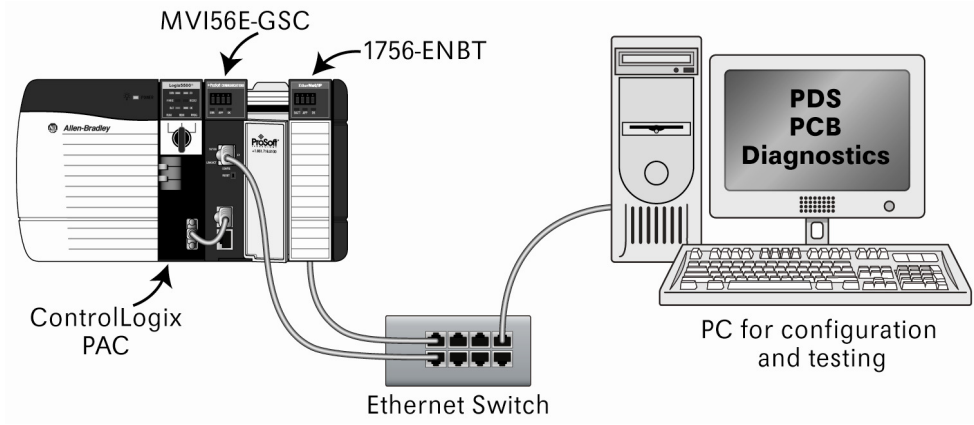
|                      |    |
|----------------------|----|
| ❖ Overview .....     | 18 |
| ❖ Architecture ..... | 19 |
| ❖ Data Flow .....    | 20 |

### 3.1 Overview

The sample application shows how to exchange data between the MVI56E-GSCXT application serial port (P1) and a remote ASCII device. For this application example, we will use the ControlLogix Programmable Automation Controller (CLX PAC) to act as the remote ASCII device. The CLX serial port will be connected directly to the MVI56E-GSCXT through an RS-232 connection using the provided null modem serial cable and RJ45-to-DB9M adapter.

### 3.2 Architecture

For this sample application, you will need to install these components and make the connections, as shown.

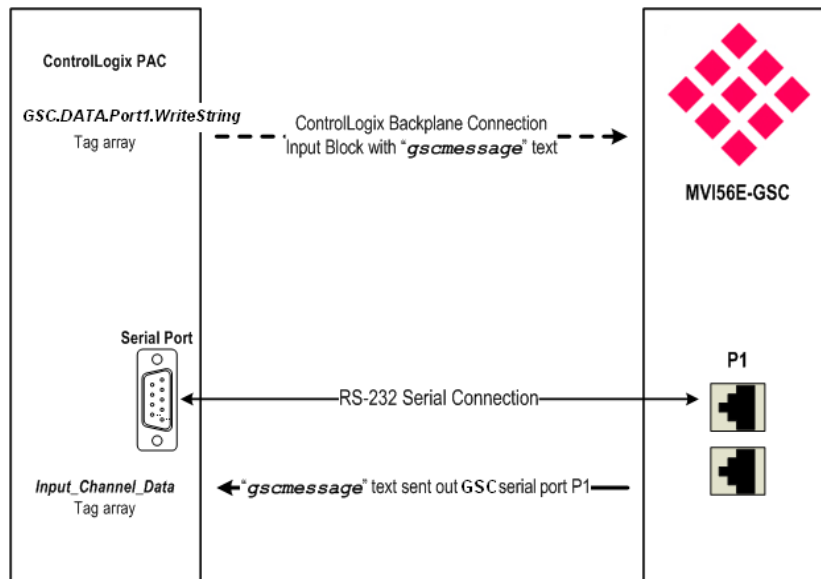


### 3.3 Data Flow

The sample application exchanges data between the ControlLogix processor (CLX) and the MVI56E-GSCXT module (GSC). The exchange is organized into two stages.

#### STAGE 1. ASCII String sent FROM GSC Serial Port To CLX Serial Port

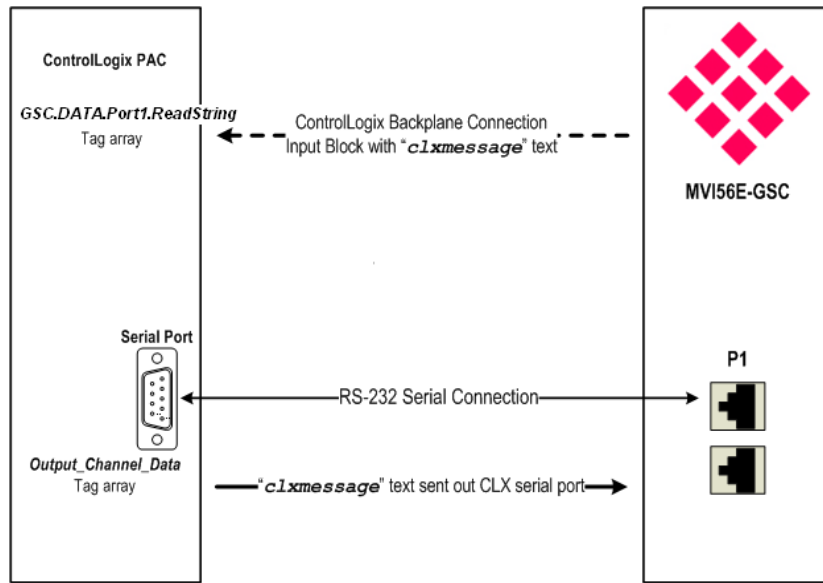
- 1 The CLX transfers the ASCII string "gscmessage" from the **GSC.DATA.Port1.WriteString** tag array to the GSC module through the CLX backplane.
- 2 The CLX requests the GSC to send the string out through the GSC's Port 1.
- 3 The CLX receives the string from the GSC through the CLX's serial port.
- 4 The sample ladder then copies the ten characters from the ASCII port buffer into a controller tag array named **INPUT\_CHANNEL\_DATA**.



#### STAGE 2. ASCII String sent FROM CLX Serial Port TO GSC Serial Port

- 1 The CLX receives the data from Step 1.
- 2 The sample ladder sends the sequence of characters "clxmessage", from the tag array **OUTPUT\_CHANNEL\_DATA**, out through the CLX's serial port.
- 3 The module receives the string on GSC Port 1. The configuration settings recognize the character "t" at the end of the string as the termination character, which indicates that the message is complete.

- 4 The GSC sends the message to the CLX through the backplane, where it appears in the **GSC.DATA.PORT1.READSTRING** tag array.





## 4 Procedures

### *In This Chapter*

|   |    |
|---|----|
| ❖ Required Steps.....   | 24 |
| ❖ Physical Setup.....   | 25 |
| ❖ Using ProSoft Configuration Builder Software.....           | 29 |
| ❖ Loading the Sample Application .....                        | 47 |
| ❖ Sample Application Setup .....                              | 48 |
| ❖ Downloading the Sample Program to the Processor.....        | 52 |
| ❖ Rebooting the Module to Transfer New Parameter Values ..... | 53 |
| ❖ Verifying Communication.....                                | 54 |

## 4.1 Required Steps

This Setup Guide will take you through the following steps:

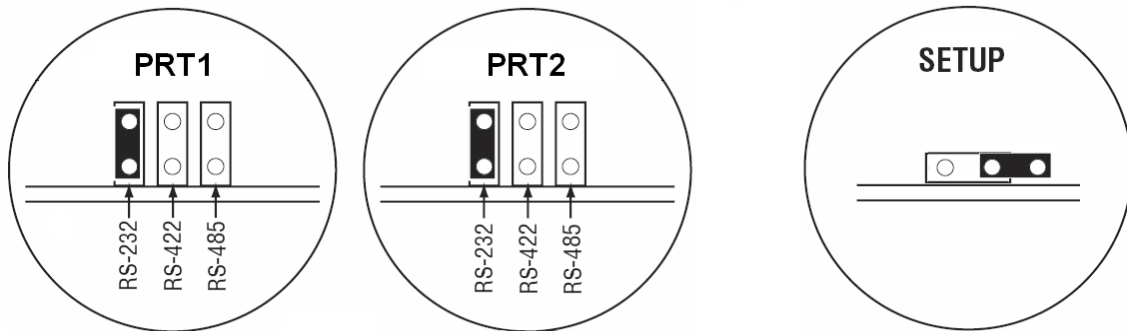
- 1 Physical Setup - installing and connecting the module (page 25)
- 2 Load the Sample Application - using the provided .ACD file (page 47)
- 3 Verify Sample Application Setup - check controller tag values (page 48)
- 4 Download the Sample Program to the Processor (page 52)
- 5 Reboot the Module - transfers configuration to the module
- 6 Use the Sample Application to Verify Communication (page 54)

## 4.2 Physical Setup

### 4.2.1 Setting Module Jumpers

There are three jumpers located at the bottom of the module. The first two jumpers (PRT1 and PRT2) set the serial communication mode: RS-232, RS-422, or RS-485.

The following illustration shows the MVI56E-GSCXT jumper configuration.



- 1 The sample application will connect the MVI56E-GSCXT application port P1 to the target device using the supplied null-modem cable (RS-232). Set the PRT1 jumper for RS-232.
- 2 The Setup Jumper acts as "write protection" for the module's flash memory. In "write protected" mode, the Setup pins are not connected, and the module's firmware cannot be overwritten. The module is shipped with the Setup pins jumpered, so that you can update the module's firmware if necessary. As you will not be updating the firmware for this procedure, remove the setup jumper and store it in a safe place (for example, on only one of the pins).

**Note:** If you are installing the module in a remote rack, you may prefer to leave the Setup pins jumpered. That way, you can update the module's firmware without requiring physical access to the module.

### **4.2.2 Installing the Module in the Rack**

If you have not already installed and configured your ControlLogix processor and power supply, please do so before installing the MVI56E-GSCXT module. Refer to your Rockwell Automation product documentation for installation instructions.

**Warning:** You must follow all safety instructions when installing this or any other electronic devices. Failure to follow safety procedures could result in damage to hardware or data, or even serious injury or death to personnel. Refer to the documentation for each device you plan to connect to verify that suitable safety procedures are in place before installing or servicing the device.

After you have checked the placement of the jumpers, insert the MVI56E-GSCXT into the ControlLogix chassis. Use the same technique recommended by Rockwell Automation to remove and install ControlLogix modules.

You can install or remove ControlLogix system components while chassis power is applied and the system is operating. However, please note the following warning.

**Warning:** When you insert or remove the module while backplane power is on, an electrical arc can occur. An electrical arc can cause personal injury or property damage by sending an erroneous signal to your system's actuators. This can cause unintended machine motion or loss of process control. Electrical arcs may also cause an explosion when they happen in a hazardous environment. Verify that power is removed or the area is non-hazardous before proceeding. Repeated electrical arcing causes excessive wear to contacts on both the module and its mating connector. Worn contacts may create electrical resistance that can affect module operation.

- 1 Align the module with the top and bottom guides, and then slide it into the rack until the module is firmly against the backplane connector.

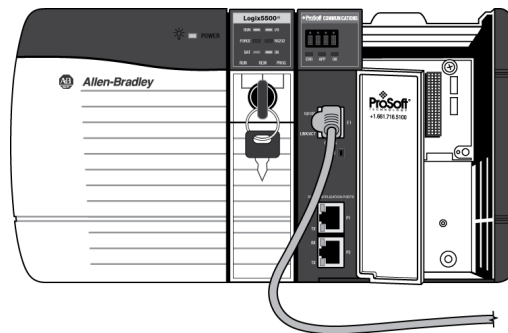


- 2 With a firm, steady push, snap the module into place.
- 3 Check that the holding clips on the top and bottom of the module are securely in the locking holes of the rack.
- 4 Make a note of the slot location. You must identify the slot in which the module is installed in order for the sample program to work correctly. Slot numbers are identified on the green circuit board (backplane) of the ControlLogix rack.
- 5 Turn power ON.

**Note:** When using the MVI56E-GSCXTXT, you must use the 1756-A5XT or 1756-A7LXT chassis. In these chassis, modules are spaced further apart than in standard ControlLogix chassis. Blank spacers are inserted between active modules.

#### 4.2.3 Connecting Your PC to the Module's Ethernet Port

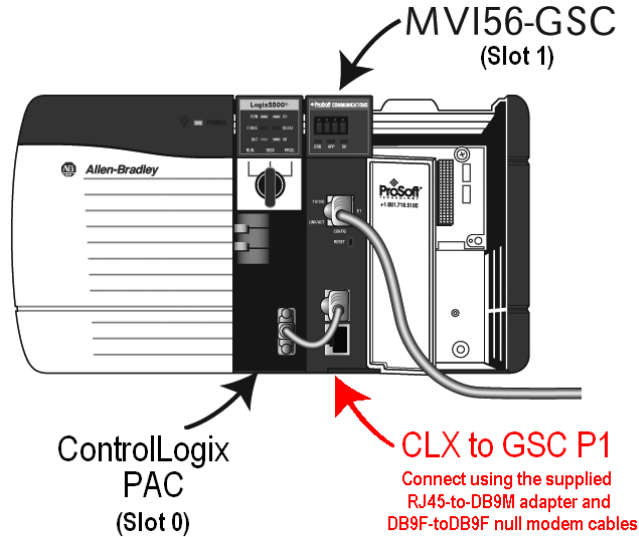
With the module securely mounted, connect one end of the Ethernet cable to the **CONFIG (E1)** Port, and the other end to an Ethernet hub or switch accessible from the same network as your PC. Or, you can connect directly from the Ethernet Port on your PC to the **CONFIG (E1)** Port on the module.



#### 4.2.4 Connecting the MVI56E-GSCXT and ControlLogix PAC Serial Ports

There are three RJ45 jacks on the MVI56E-GSCXT. The top-most RJ45 port is the E1 Ethernet configuration port. The lower two RJ45 ports, labeled P1 and P2 are serial communication ports, not Ethernet ports.

Use the RJ45-to-DB9M adapter and DB9F-to-DB9F null modem cable provided with the module to connect between the module's application port P1 and the DB9M serial port on the front of the ControlLogix processor.



### 4.3 Using ProSoft Configuration Builder Software

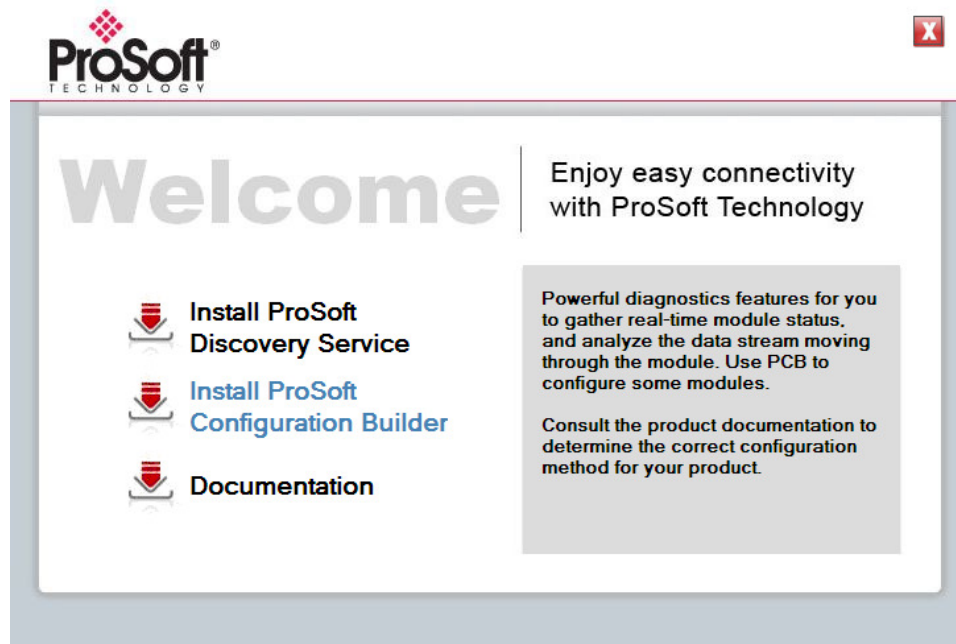
*ProSoft Configuration Builder (PCB)* provides a quick and easy way to manage module configuration files customized to meet your application needs. *PCB* is not only a powerful solution for new configuration files, but also allows you to import information from previously installed (known working) configurations to new projects.

The ProSoft Discovery Service (PDS) is available as a stand-alone application, or as part of ProSoft Configuration Builder. ProSoft Discovery Service shows you all the MVI56E modules available on your local area network.

**Note:** The MVI56E-GSCXT module receives its protocol and backplane configuration information from the Ladder Logic. Use ProSoft Configuration Builder to configure the module's Ethernet settings.

#### To install ProSoft Configuration Builder from the CD-ROM

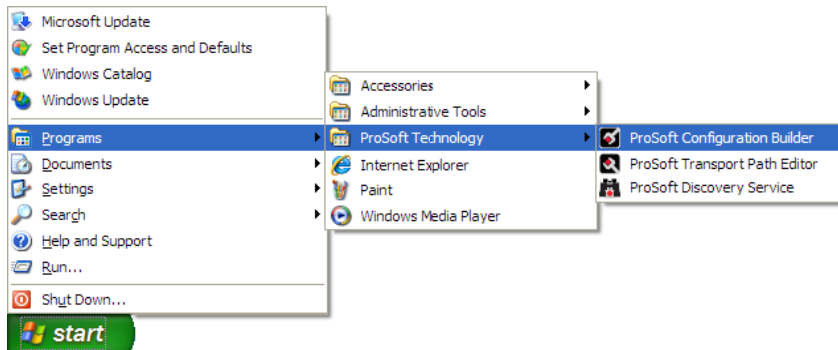
- 1 Insert the ProSoft Solutions CD-ROM into the CD drive of your PC. Wait for the startup screen to appear.



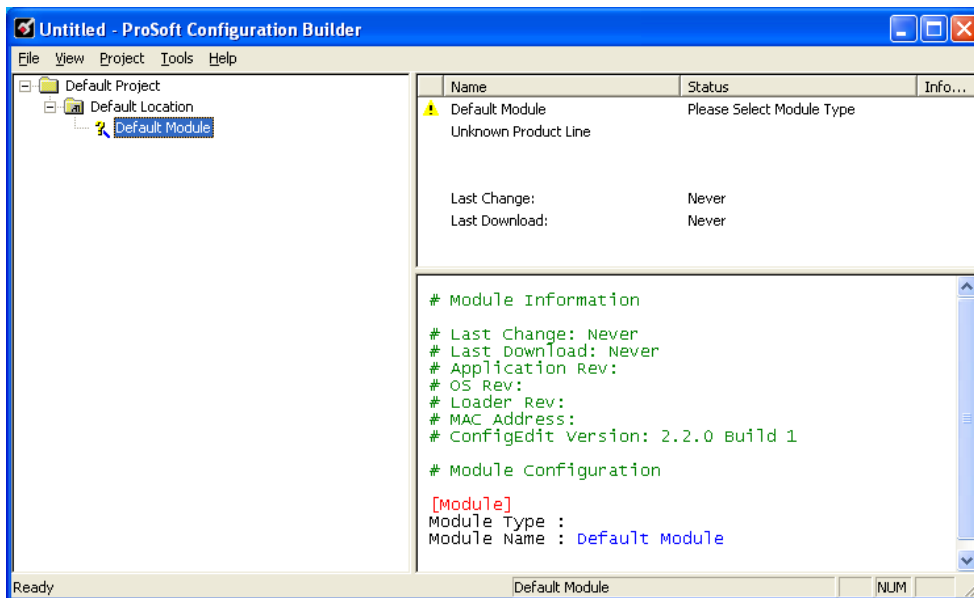
- 2 On the startup screen, click **INSTALL PROSOFT CONFIGURATION BUILDER**. This action starts the installation wizard for ProSoft Configuration Builder.
- 3 Click **NEXT** on each page of the installation wizard. Click **FINISH** on the last page of the wizard.

### 4.3.1 Setting Up the Project

To begin, start **PROSOFT CONFIGURATION BUILDER (PCB)**.



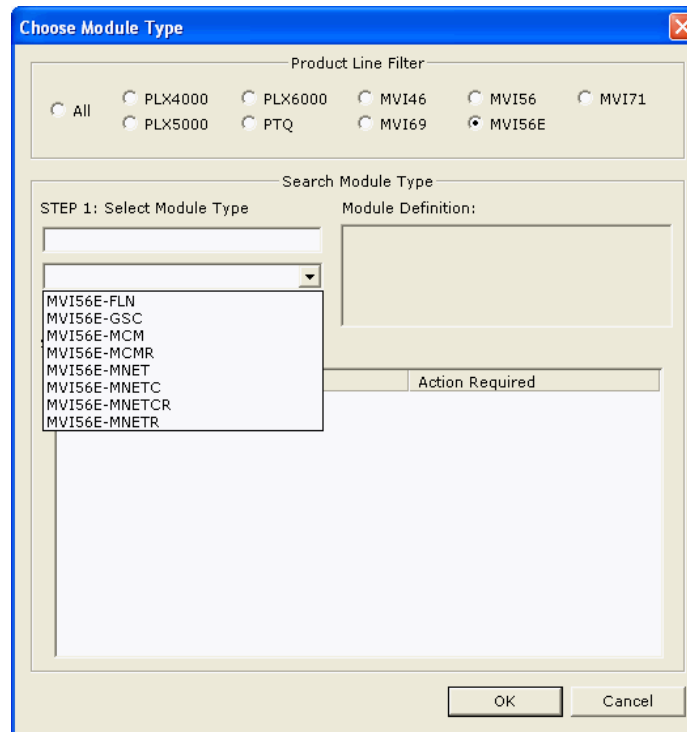
If you have used other Windows configuration tools before, you will find the screen layout familiar. *PCB's* window consists of a tree view on the left, and an information pane and a configuration pane on the right side of the window. When you first start *PCB*, the tree view consists of folders for *Default Project* and *Default Location*, with a *Default Module* in the *Default Location* folder. The following illustration shows the *PCB* window with a new project.



Your first task is to add the MVI56E-GSCXT module to the project.

- 1 Use the mouse to select **DEFAULT MODULE** in the tree view, and then click the right mouse button to open a shortcut menu.

- 2 On the shortcut menu, select **CHOOSE MODULE TYPE**. This action opens the *Choose Module Type* dialog box.

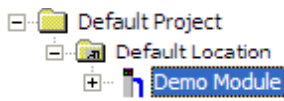


- 3 In the *Product Line Filter* area of the dialog box, select **MVI56E**. In the *Select Module Type* dropdown list, select **MVI56E-GSCXT**, and then click **OK** to save your settings and return to the *ProSoft Configuration Builder* window.

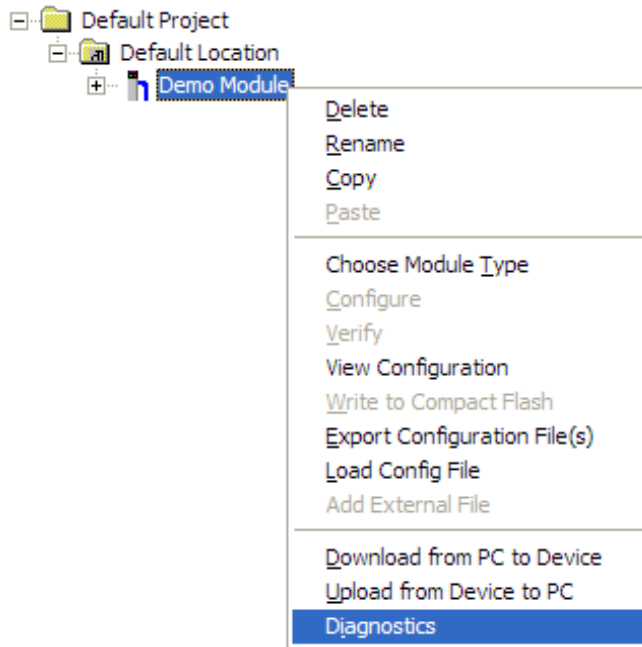
### 4.3.2 Setting Up a Temporary IP Address

**Important:** *ProSoft Configuration Builder* locates MVI56E-GSCXT modules through UDP broadcast messages. These messages may be blocked by routers or layer 3 switches. In that case, *ProSoft Discovery Service* will be unable to locate the modules. To use *ProSoft Configuration Builder*, arrange the Ethernet connection so that there is no router/layer 3 switch between the computer and the module OR reconfigure the router/layer 3 switch to allow routing of the UDP broadcast messages.

- 1 In the tree view in *ProSoft Configuration Builder*, select the **MVI56E-GSCXT** module.



- 2 Click the right mouse button to open a shortcut menu. On the shortcut menu, choose **DIAGNOSTICS**.

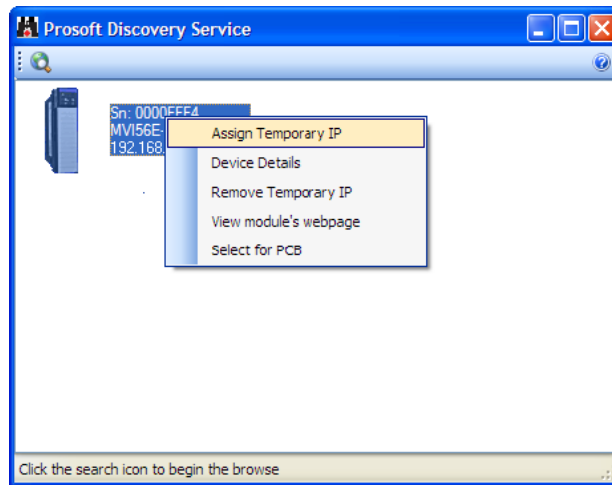


- 3 In the *Diagnostics* window, click the **SET UP CONNECTION** button.

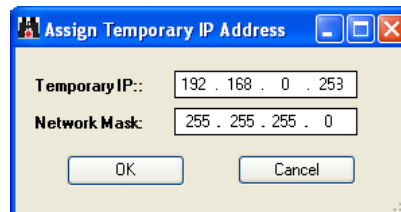


**Click to set up connection**

- 4 In the *Connection Setup* dialog box, click the **BROWSE DEVICE(S)** button to open the *ProSoft Discovery Service*. Select the module, then right-click and choose **ASSIGN TEMPORARY IP**.

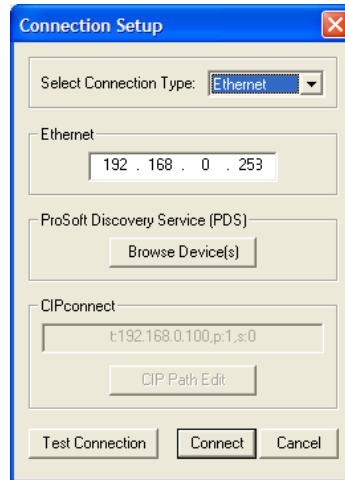


- 5 The module's default IP address is usually 192.168.0.250. Choose an unused IP within your subnet, and then click **OK**.

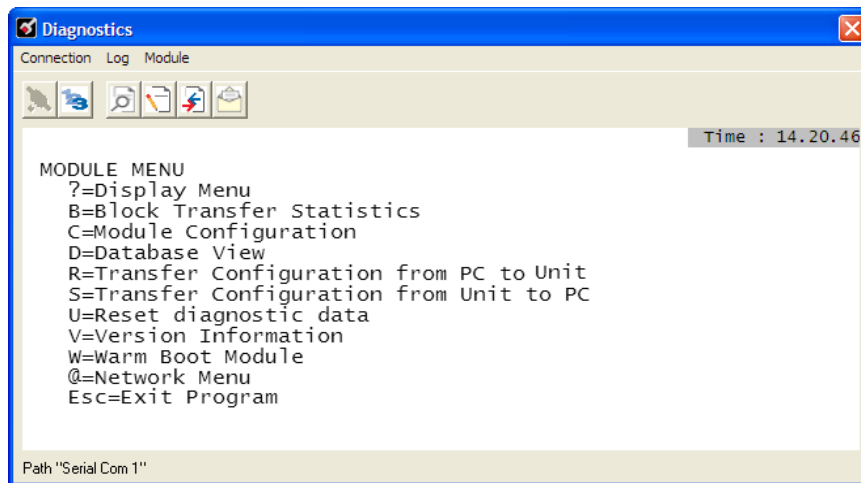


**Important:** The temporary IP address is only valid until the next time the module is initialized. For information on how to set the module's permanent IP address, see *Setting Up a Permanent IP Address* (page 35).

- 6 Close the *ProSoft Discovery Service* window. Enter the temporary IP in the Ethernet address field of the *Connection Setup* dialog box, then click the **TEST CONNECTION** button to verify that the module is accessible with the current settings.



- 7 If the *Test Connection* is successful, click **CONNECT**. The *Diagnostics* menu will display in the *Diagnostics* window.

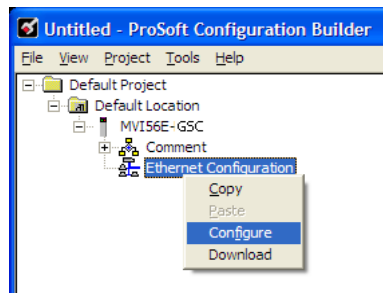


### 4.3.3 Setting Up a Permanent IP Address

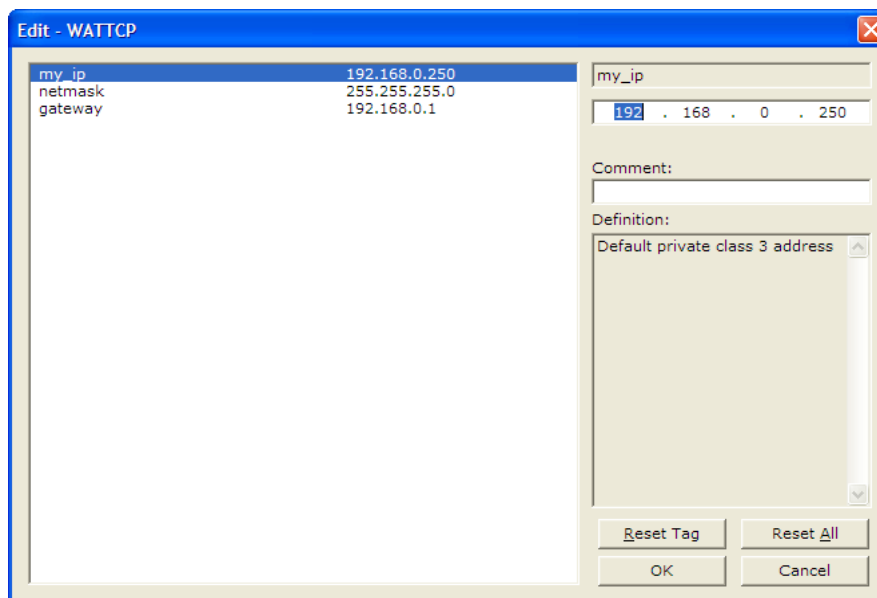
**Note:** For alternative methods of connecting to the module with your PC, refer to Using CIPconnect to Connect to the Module (page 37) or Using RSWho to Connect to the Module (page 46).

These steps show you how to set a permanent IP address on the module. This example assumes module's default IP address is 192.168.0.250.

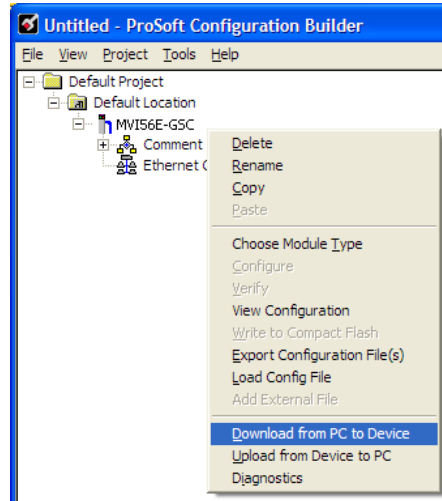
- 1 Start ProSoft Configuration Builder.
- 2 Select the **MVI56E-GSCXT** icon, and then click the [+] symbol to expand the **MVI56E-GSCXT** tree.
- 3 Select **ETHERNET CONFIGURATION**, and then click the right mouse button to open a shortcut menu.
- 4 On the shortcut menu, select **CONFIGURE**.



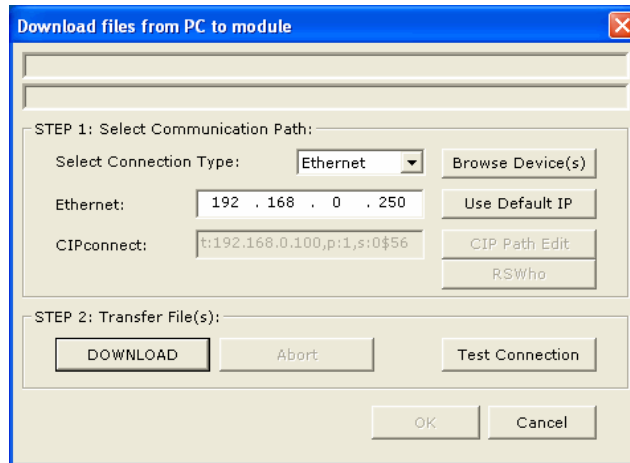
This action opens the *Edit-WATTCP* dialog box. Use this dialog box to enter the MVI56E-GSCXT module's permanent IP Address (**MY\_IP**), **SUBNET MASK (NETMASK)** and **DEFAULT GATEWAY (GATEWAY)**.



- 5 Click **OK** to save the updated Ethernet configuration.
- 6 Next, select the MVI56E-GSCXT icon, and then click the right mouse button to open a shortcut menu. On the shortcut menu, choose **DOWNLOAD FROM PC TO DEVICE**.



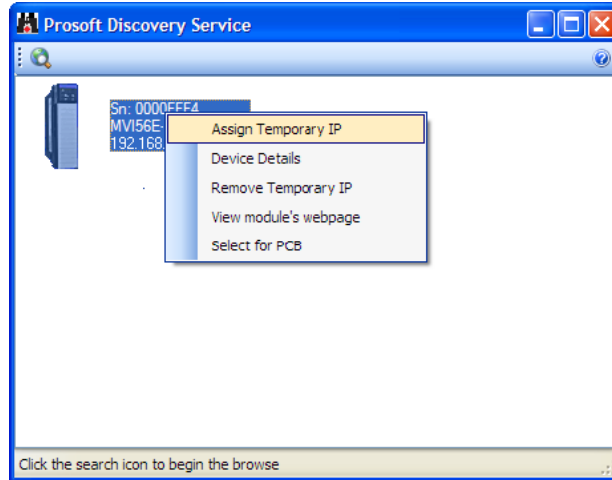
This action opens the **DOWNLOAD** dialog box. Notice that the Ethernet address field contains the temporary IP address you assigned previously. ProSoft Configuration Builder will use this temporary IP address to connect to the module.



- Click **TEST CONNECTION** to verify that the temporary IP address is correct.
- 7 If the connection succeeds, click **DOWNLOAD** to transfer the Ethernet configuration to the module.

If the Test Connection procedure fails, you will see an error message. To correct the error, follow these steps.

- 1 Click **OK** to dismiss the error message.
- 2 On the **DOWNLOAD** dialog box, click **BROWSE DEVICES** to open **PROSOFT DISCOVERY SERVICE**.



- 3 Select the module, and then click the right mouse button to open a shortcut menu. On the shortcut menu, choose **SELECT FOR PCB**.
- 4 Close Prosoft Discovery Service.
- 5 Click **DOWNLOAD** to transfer the Ethernet configuration to the module.

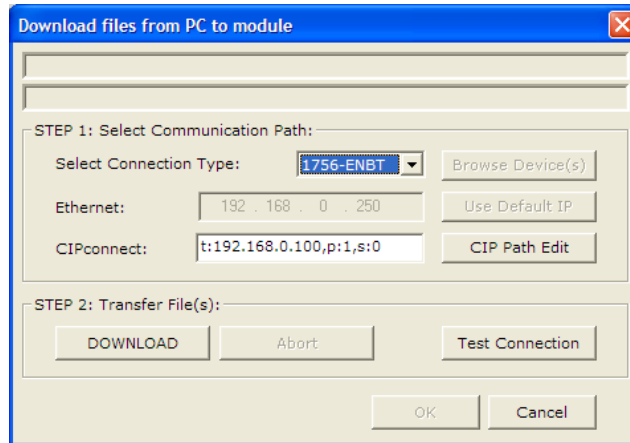
#### Using CIPconnect® to Connect to the Module

You can use CIPconnect® to connect a PC to the ProSoft Technology MVI56E-GSCXT module over Ethernet using Rockwell Automation's 1756-ENBT EtherNet/IP® module. This allows you to configure the MVI56E-GSCXT network settings and view module diagnostics from a PC. RSLinx is not required when you use CIPconnect. All you need are:

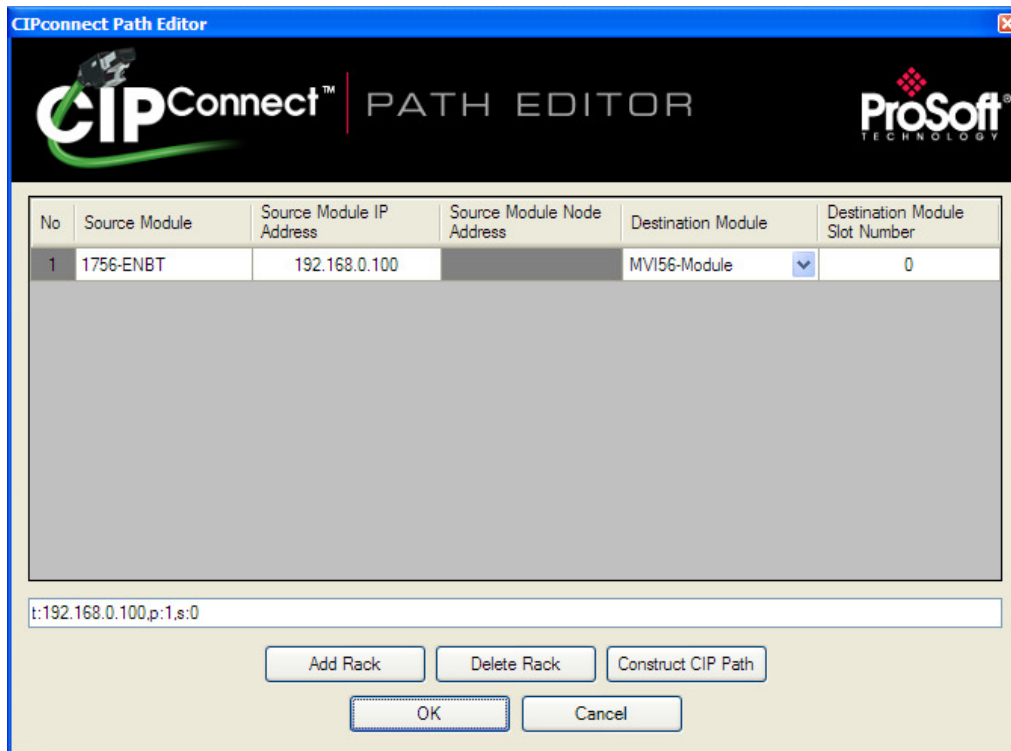
- The IP addresses and slot numbers of any 1756-ENBT modules in the path
- The slot number of the MVI56E-GSCXT in the destination ControlLogix chassis (the last ENBTx and chassis in the path).

To use CIPconnect, follow these steps.

- 1 In the *Select Port* dropdown list, choose **1756-ENBT**. The default path appears in the text box, as shown in the following illustration.



- 2 Click **CIP PATH EDIT** to open the *CIPconnect Path Editor* dialog box.



The *CIPconnect Path Editor* allows you to define the path between the PC and the MVI56E-GSCXT module. The first connection from the PC is always a 1756-ENBT (Ethernet/IP) module.

Each row corresponds to a physical rack in the CIP path.

- If the MVI56E-GSCXT module is located in the same rack as the first 1756-ENBT module, select **RACK NO. 1** and configure the associated parameters.

- If the MVI56E-GSCXT is available in a remote rack (accessible through ControlNet or Ethernet/IP), include all racks (by using the **ADD RACK** button).

| Parameter                      | Description  |
|--------------------------------|--|
| Source Module                  | Source module type. This field is automatically selected depending on the destination module of the last rack (1756-CNB or 1756-ENBT).                                   |
| Source Module IP Address       | IP address of the source module (only applicable for 1756-ENBT)  |
| Source Module Node Address     | Node address of the source module (only applicable for 1756-CNB)   |
| Destination Module             | Select the destination module associated to the source module in the rack. The connection between the source and destination modules is performed through the backplane. |
| Destination Module Slot Number | The slot number where the destination MVI56E module is located.  |

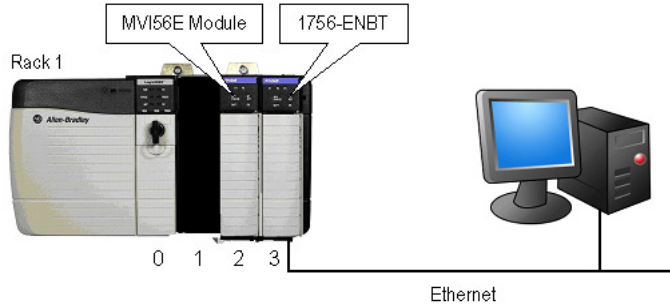
To use the CIPconnect Path Editor, follow these steps.

- 1 Configure the path between the 1756-ENBT connected to your PC and the MVI56E-GSCXT module.
  - If the module is located in a remote rack, add more racks to configure the full path.
  - The path can only contain ControlNet or Ethernet/IP networks.
  - The maximum number of supported racks is six.
- 2 Click **CONSTRUCT CIP PATH** to build the path in text format
- 3 Click **OK** to confirm the configured path.

The following examples should provide a better understanding on how to set up the path for your network.

**Example 1: Local Rack Application**

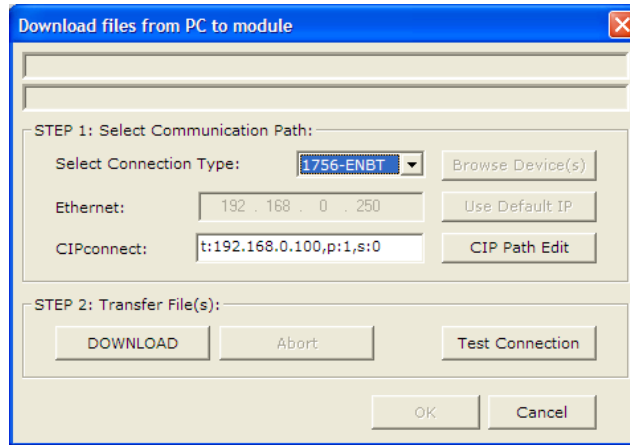
For this example the MVI56E-GSCXT module is located in the same rack as the 1756-ENBT that is connected to the PC.



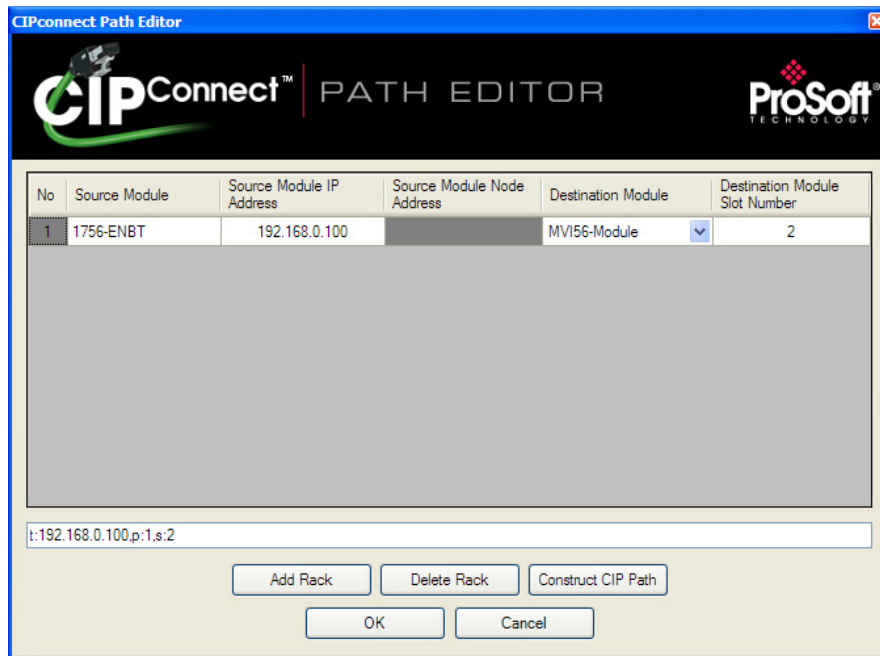
**Rack 1**

| Slot | Module                 | Network Address  |
|------|------------------------|------------------|
| 0    | ControlLogix Processor | -                |
| 1    | Any                    | -                |
| 2    | MVI56E-GSCXT           | -                |
| 3    | 1756-ENBT              | IP=192.168.0.100 |

- 1 In the *Download* window, click **CIP PATH EDIT**.

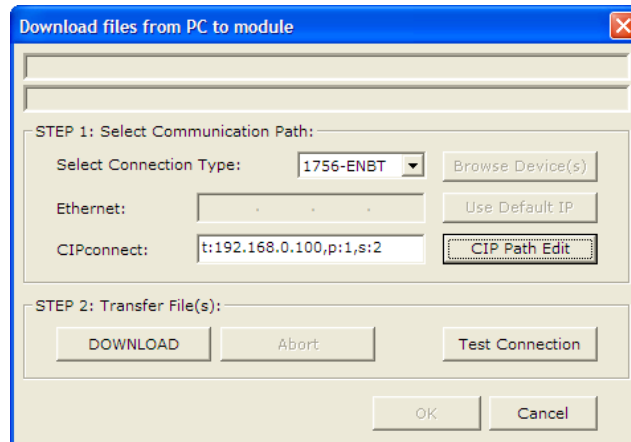


- 2 Configure the path as shown in the following illustration, and click **CONSTRUCT CIP PATH** to build the path in text format.

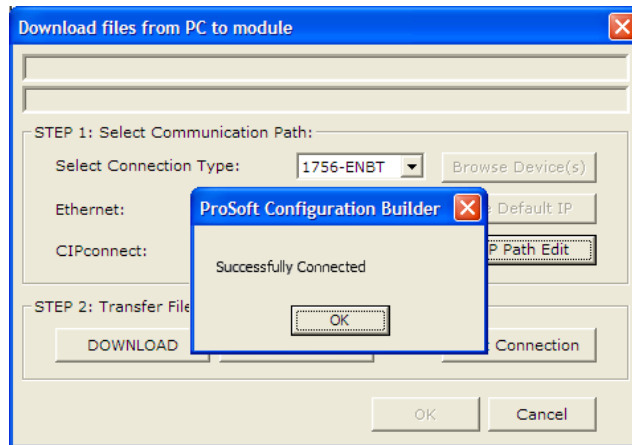


Click **OK** to close the **CIP PATH EDITOR** and return to the *Download* dialog box.

- 3 Check the new path in the download text box.



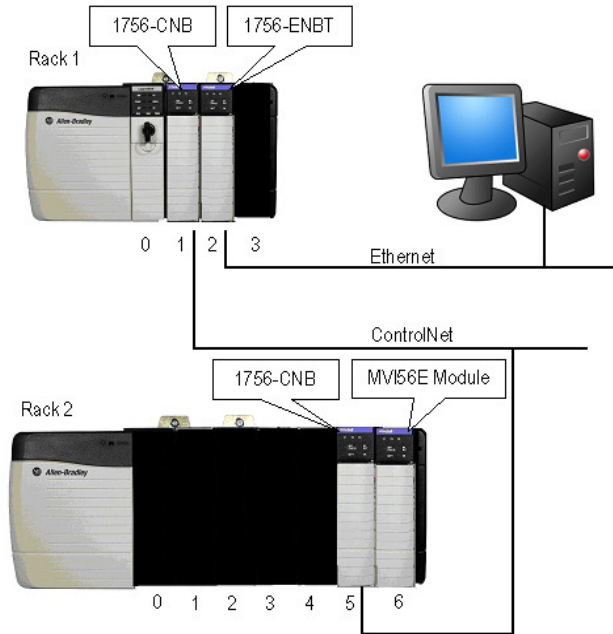
- 4 Click **TEST CONNECTION** to verify that the physical path is available. The following message should be displayed upon success.



- 5 Click **OK** to close the *Test Connection* pop-up. You are now ready to use CIPconnect for downloading changes to the module's Ethernet configuration settings. You may also use this connection to access module diagnostics screens and menus. For more information, see the chapter on Diagnostics and Troubleshooting. For more information, see the chapter on Diagnostics and Troubleshooting.

**Example 2: Remote Rack Application - CIPconnect No Download**

For this example, the MVI56E-GSCXT module is located in a remote rack accessible through ControlNet, as shown in the following illustration.



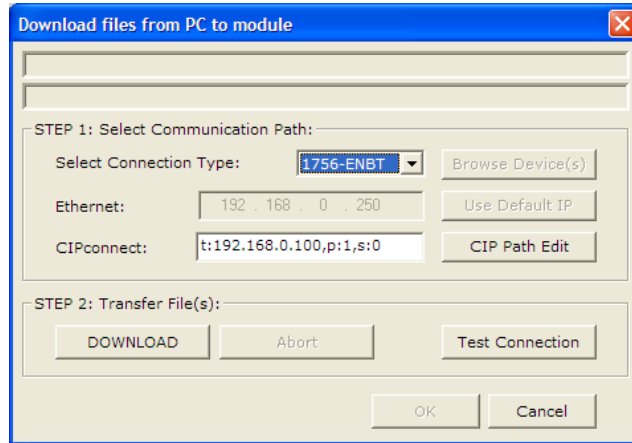
**Rack 1**

| Slot | Module                 | Network Address  |
|------|------------------------|------------------|
| 0    | ControlLogix Processor | -                |
| 1    | 1756-CNB               | Node = 1         |
| 2    | 1756-ENBT              | IP=192.168.0.100 |
| 3    | Any                    | -                |

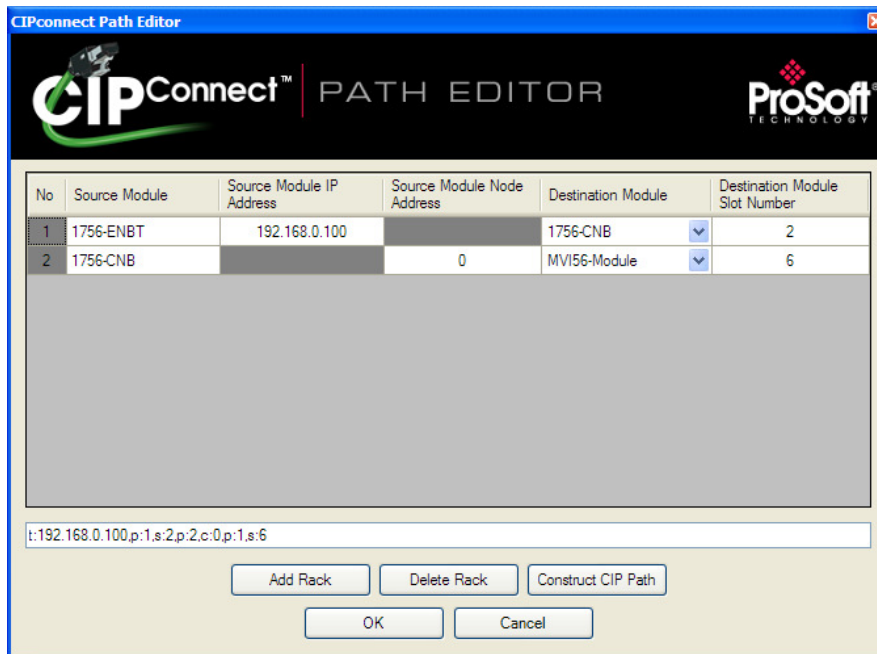
**Rack 2**

| Slot | Module       | Network Address |
|------|--------------|-----------------|
| 0    | Any          | -               |
| 1    | Any          | -               |
| 2    | Any          | -               |
| 3    | Any          | -               |
| 4    | Any          | -               |
| 5    | 1756-CNB     | Node = 2        |
| 6    | MVI56E-GSCXT | -               |

- 1 In the Download window, click **CIP PATH EDIT**.

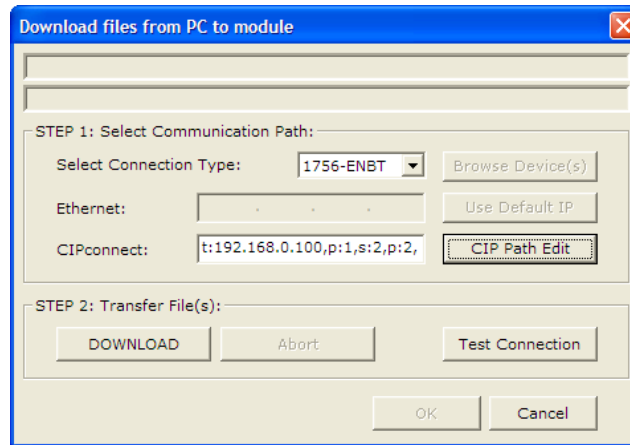


- 2 Configure the path as shown in the following illustration for this example and click **CONSTRUCT CIP PATH** to build the path in text format.

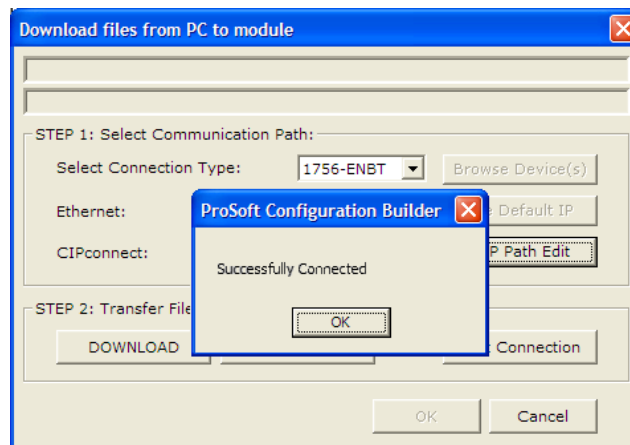


Click **OK** to close the **CIP PATH EDITOR** and return to the **DOWNLOAD** dialog box.

- 3 Check the new path in the download text box.



- 4 Click **TEST CONNECTION** to verify that the physical path is available. The following message should be displayed upon success.

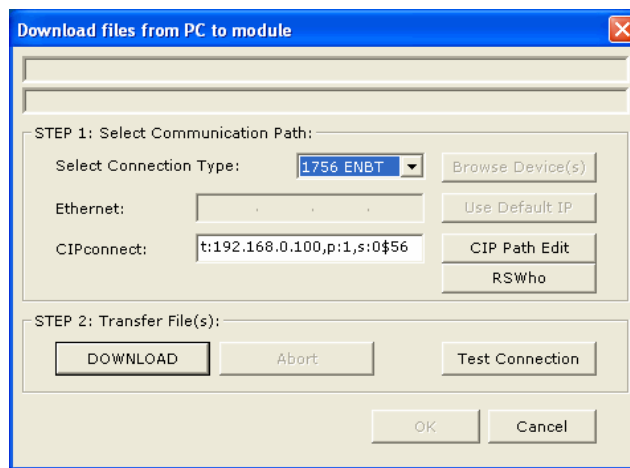


- 5 Click **OK** to close the *Test Connection* pop-up. You are now ready to use CIPconnect for downloading changes to the module's Ethernet configuration settings. You may also use this connection to access module diagnostics screens and menus. For more information, see the chapter on Diagnostics and Troubleshooting. For more information, see the chapter on Diagnostics and Troubleshooting.

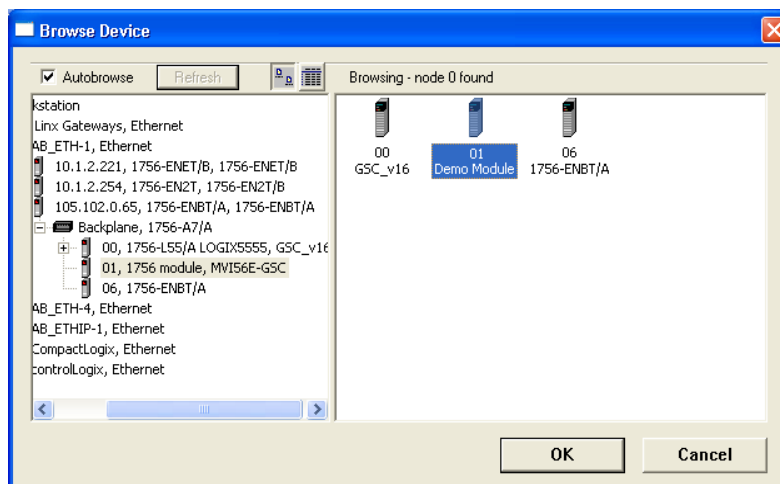
### Using RSWHo to Connect to the Module

You need to have RSLinx installed on your PC to use this feature. You also need an ENBT module set up in the rack. For information on setting up the ENBT module, see *Using CIPconnect to Connect to the Module*.

- 1 In the tree view in *ProSoft Configuration Builder*, right-click the **MVI56E-GSCXT** module.
- 2 From the shortcut menu, choose **DOWNLOAD FROM PC TO DEVICE**.
- 3 In the *Download* dialog box, choose **1756 ENBT** from the *Select Connection Type* dropdown box.



- 4 Click **RSWHO** to display modules on the network. The MVI56E-GSCXT module will automatically be identified on the network.



- 5 Select the module, and then click **OK**.
- 6 In the *Download* dialog box, click **DOWNLOAD**.

#### 4.4 Loading the Sample Application

ProSoft Technology has provided a Sample Application file for RSLogix5000, version 16 or higher, that provides all you need to test the basic sending and receiving capability of the MVI56E-GSCXT module. This special sample adds a few controller tags, tag arrays, and rungs of ladder logic to the basic Add-Instruction (AOI) sample to create a simple working application.

- 1 Put the distribution CD-ROM that came with the module into the CD-ROM or DVD-ROM drive of your personal computer (PC). If you do not have access to the CD-ROM, you may download the ladder logic samples from the product web page at: **[www.prosoft-technology.com](http://www.prosoft-technology.com)**.
- 2 Use Microsoft Windows (Drive) Explorer to navigate to MVI56E-GSCXT ladder logic folder. The path should be similar to "**X:\LadderLogic\ControlLogix\MVI56E\MVI56E\_GSC**", where "**X**" is the drive letter of your CD/DVD-ROM drive.
- 3 Locate the file, named "**MVI56(E)GSC\_SetupGuide\_SampleApplication.ACD**", and copy it to a temporary or working directory on your PC hard drive. You will need to have it on your hard drive to be able to open it in RSLogix5000 because that program needs to create two temporary working files for every project file it opens. Since the CD-ROM is read-only, these temporary files cannot be created on the CD-ROM, so any attempt to open the Sample Application directly from the CD-ROM will fail.
- 4 Start RSLogix5000.
- 5 Use the **FILE|OPEN** menu option to navigate to the temporary or working folder where you stored the Sample Application file and open it in RSLogix5000.

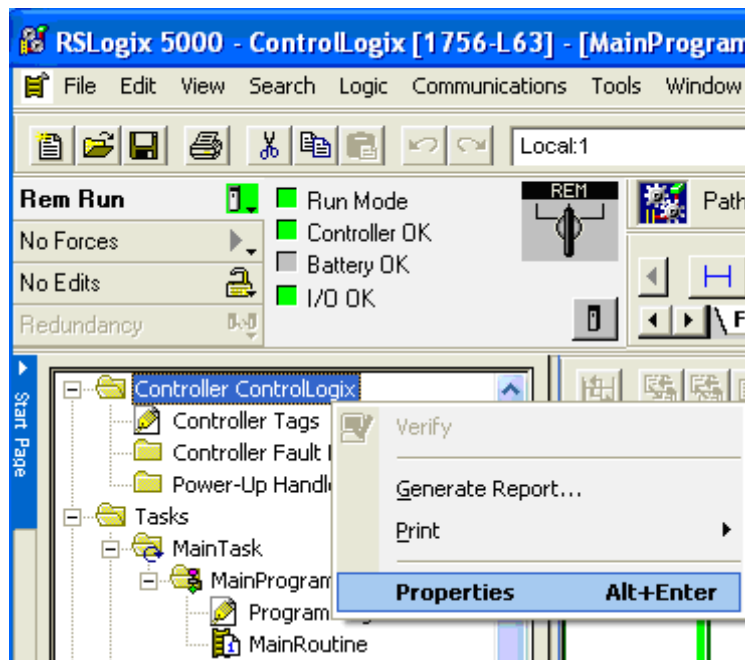
## 4.5 Sample Application Setup

### 4.5.1 Configuring the ControlLogix (CLX) Serial Port

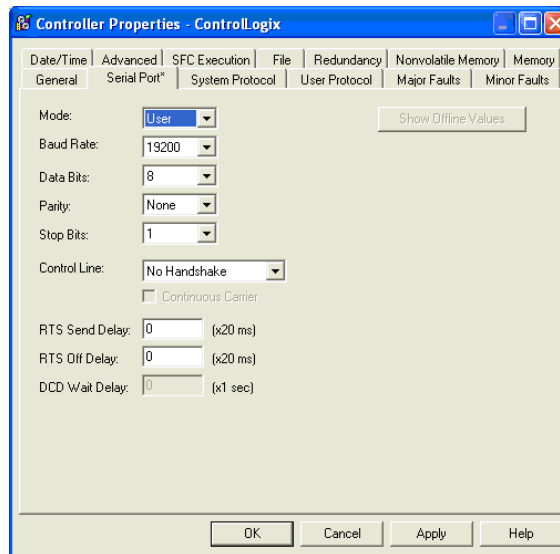
In this step, you will configure the ControlLogix serial port with the following settings.

| Setting   | Value |
|-----------|-------|
| Baud Rate | 19200 |
| Parity    | None  |
| Data Bits | 8     |
| Stop Bits | 1     |

- 1 Open the **CONTROLLER PROPERTIES** dialog box.

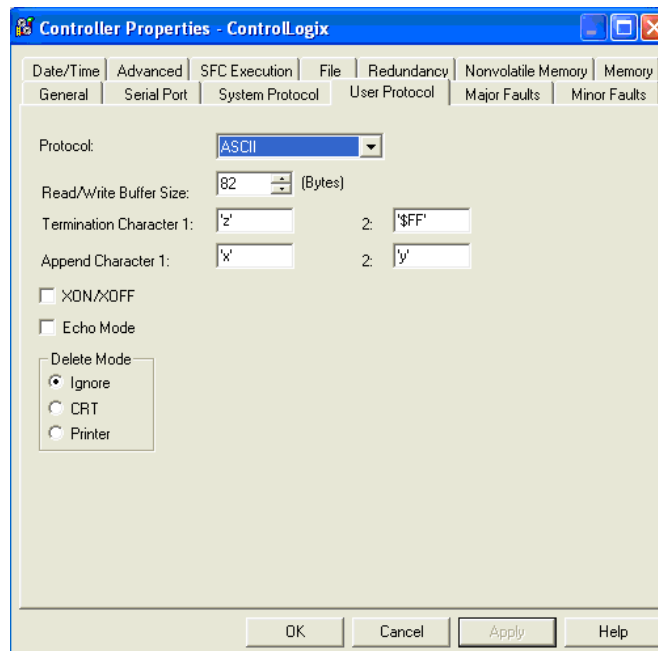


- 2 On the **SERIAL PORT** tab, set the port parameters as shown in the following illustration.



- 3 On the **USER PROTOCOL** tab, select the Protocol dropdown list, and then select **ASCII**.

**Note:** The sample application does not use the settings for **TERMINATION CHARACTERS** and **APPEND CHARACTERS** parameters, so you can leave these settings unchanged.



- 4 Click **OK** to save your changes and dismiss the *Controller Properties* dialog box.

### 4.5.2 Configuring the MVI56E-GSCXT Serial Port

- 1 In the **CONTROLLER ORGANIZER** pane, expand the **CONTROLLER** folder, and then double-click **CONTROLLER TAGS**.
- 2 Click the **MONITOR TAGS** tab at the bottom of the Controller Tags pane.
- 3 Navigate to the tag **GSC.CONFIG.PORT1**, and then expand the tag, as shown in the following illustration.

|                                  |       |         |                   |
|----------------------------------|-------|---------|-------------------|
| [-] GSC                          | {...} |         | GSCMODULEDEF      |
| [-] GSC.CONFIG                   | {...} |         | GSCPORTCONFIG     |
| [-] GSC.CONFIG.Port1             | {...} |         | GSCPORTCONFIGPARA |
| [-] GSC.CONFIG.Port1.Enabled     | 1     | Decimal | INT               |
| [-] GSC.CONFIG.Port1.Type        | 1     | Decimal | INT               |
| [-] GSC.CONFIG.Port1.Baudrate    | 9600  | Decimal | INT               |
| [-] GSC.CONFIG.Port1.Parity      | 0     | Decimal | INT               |
| [-] GSC.CONFIG.Port1.DataBits    | 8     | Decimal | INT               |
| [-] GSC.CONFIG.Port1.StopBits    | 1     | Decimal | INT               |
| [-] GSC.CONFIG.Port1.RTSOn       | 0     | Decimal | INT               |
| [-] GSC.CONFIG.Port1.RTSOff      | 0     | Decimal | INT               |
| [-] GSC.CONFIG.Port1.Handshaking | 0     | Decimal | INT               |
| [-] GSC.CONFIG.Port1.RTermCnt    | 2     | Decimal | INT               |
| [-] GSC.CONFIG.Port1.RTermChar   | {...} | ASCII   | SINT[12]          |
| [-] GSC.CONFIG.Port1.RPacketLen  | 240   | Decimal | INT               |
| [-] GSC.CONFIG.Port1.RTimeout    | 2000  | Decimal | INT               |
| [-] GSC.CONFIG.Port1.RDelay      | 0     | Decimal | INT               |
| [-] GSC.CONFIG.Port1.WTermCnt    | 0     | Decimal | INT               |
| [-] GSC.CONFIG.Port1.WTermChar   | {...} | ASCII   | SINT[12]          |
| [-] GSC.CONFIG.Port1.WPacketLen  | 0     | Decimal | INT               |
| [-] GSC.CONFIG.Port1.WTimeout    | 500   | Decimal | INT               |
| [-] GSC.CONFIG.Port1.WDelay      | 0     | Decimal | INT               |
| [-] GSC.CONFIG.Port1.WMinDelay   | 0     | Decimal | INT               |
| [-] GSC.CONFIG.Port2             | {...} |         | GSCPORTCONFIGPARA |
| [-] GSC.DATA                     | {...} |         | GSCDATA           |
| [-] GSC.STATUS                   | {...} |         | GSCSTATUS         |
| [-] GSC.CONTROL                  | {...} |         | GSCCONTROL        |
| [-] GSC.UTIL                     | {...} |         | GSCUTIL           |

- 4 The Sample Application configures the port to use a termination character to detect the end of a transmitted string and trigger the data transfer to the processor (**GSC.CONFIG.PORT1.TYPE = 1**).

For this termination type, you must also configure the number of termination characters (**GSC.CONFIG.PORT1.RTERMCNT = 1**) and input the character that indicates the end of the string (**GSC.CONFIG.PORT1.RTERMCHAR[0] = "t"**). Note that the termination character value is case sensitive. To enter the termination character, expand the **GSC.CONFIG.PORT1.RTERMCHAR** tag, set the *Style* to **ASCII**, and then type a "t" character into the **RTERMCHAR[0]** array element.

|                                   |        |         |          |
|-----------------------------------|--------|---------|----------|
| [-] GSC.CONFIG.Port1.RTermCnt     | 1      | Decimal | INT      |
| [-] GSC.CONFIG.Port1.RTermChar    | {...}  | ASCII   | SINT[12] |
| [-] GSC.CONFIG.Port1.RTermChar[0] | 't'    | ASCII   | SINT     |
| [-] GSC.CONFIG.Port1.RTermChar[1] | '\$00' | ASCII   | SINT     |
| [-] GSC.CONFIG.Port1.RTermChar[2] | '\$00' | ASCII   | SINT     |
| [-] GSC.CONFIG.Port1.RTermChar[3] | '\$00' | ASCII   | SINT     |
| [-] GSC.CONFIG.Port1.RTermChar[4] | '\$00' | ASCII   | SINT     |

### 4.5.3 What's In the Sample Application?

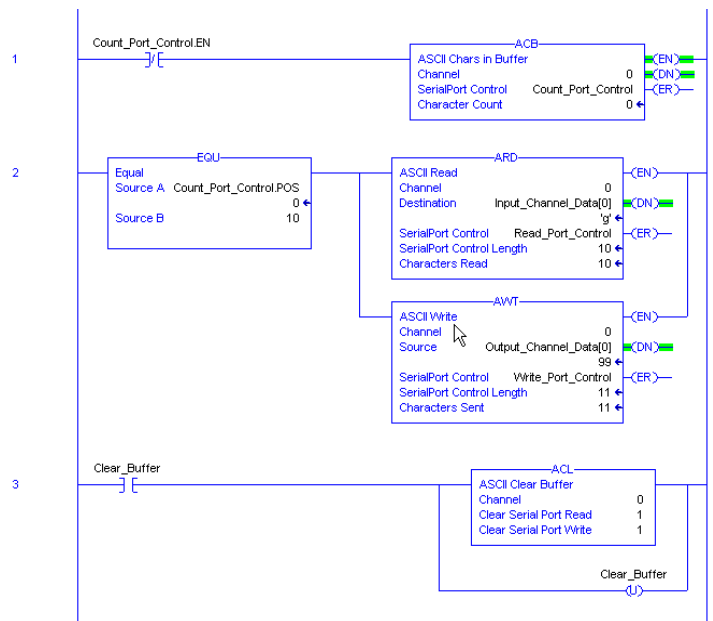
The Sample Application ladder program was created by starting with the version 16 Add-On Instruction (AOI) sample and adding a few controller tags, tags arrays, and logic rungs. These additional items have already been added to the "**MVI56(E)GSC\_SetupGuide\_SampleApplication.ACD**" file for you. Once you load that file, you have all you need to verify communication.

However, if you need to create a Sample Application for an earlier version of RSLogix5000, you can start with the version 13 or 15 sample (ladder logic only, no AOI) and add the following items.

- 1 Start by adding the controller tags shown here to the generic sample ladder program. All data types used in the Sample Application are standard RSLogix types, so no special User-Defined Data Types need to be created. To add tags or arrays, open the Controller Tags dialog box, click on the Edit tab, and enter the tags, using the tag names, styles, data types, and array sizes, as shown. .

|                          |       |         |                     |
|--------------------------|-------|---------|---------------------|
| Clear_Buffer             | 0     | Decimal | BOOL                |
| Count_Port_Control       | {...} |         | SERIAL_PORT_CONTROL |
| Input_Channel_Chars_Read | 0     | Decimal | DINT                |
| Input_Channel_Data       | {...} | ASCII   | SINT[82]            |
| Output_Channel_Data      | {...} | ASCII   | SINT[82]            |
| Read_Port_Control        | {...} |         | SERIAL_PORT_CONTROL |
| Write_Port_Control       | {...} |         | SERIAL_PORT_CONTROL |

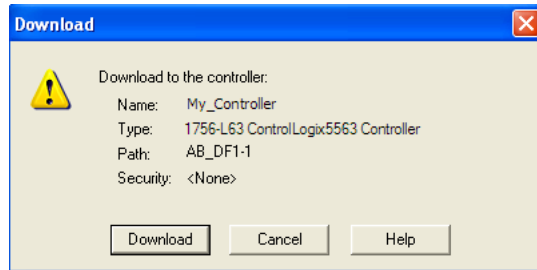
- 2 Next, ladder logic rungs can be added to the basic sample logic. The most convenient place to add them would be in the default *Main Task - Main Program - Main Routine* in the sample, just below any existing rungs. The following ladder code can perform the tasks required to pass data on the processor's serial port.



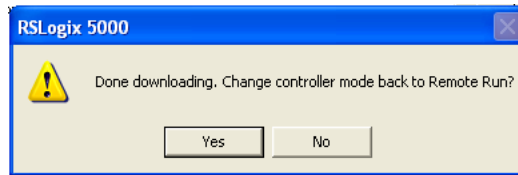
## 4.6 Downloading the Sample Program to the Processor

**Note:** The key switch on the front of the ControlLogix processor must be in the REM or PROG position.

- 1 If you are not already online with the processor, open the *Communications* menu, and then choose **DOWNLOAD**. RSLogix 5000 will establish communication with the processor. You do not have to download through the processor's serial port, as shown here. You may download through any available network connection.
- 2 When communication is established, RSLogix 5000 will open a confirmation dialog box. Click the **DOWNLOAD** button to transfer the sample program to the processor.



- 3 RSLogix 5000 will compile the program and transfer it to the processor. This process may take a few minutes.
- 4 When the download is complete, RSLogix 5000 will open another confirmation dialog box. If the key switch is in the REM position, click **OK** to switch the processor from PROGRAM mode to RUN mode.



**Note:** If you receive an error message during these steps, refer to your RSLogix documentation to interpret and correct the error.

## 4.7 Rebooting the Module to Transfer New Parameter Values

Any changes you may make in the sample program do not take effect until you reboot the module.

- 1 Locate the **GSC.CONTROL.WARMBOOT** controller tag.

|   |       |         |            |
|---|-------|---------|------------|
| <input type="checkbox"/> GSC.CONTROL          | {...} |         | GSCCONTROL |
| <input type="checkbox"/> GSC.CONTROL.Warmboot | 0     | Decimal | BOOL       |

- 2 Enter a value of 1 into this controller tag.

|   |       |         |            |
|---|-------|---------|------------|
| <input type="checkbox"/> GSC.CONTROL          | {...} |         | GSCCONTROL |
| <input type="checkbox"/> GSC.CONTROL.Warmboot | 1     | Decimal | BOOL       |

- 3 Verify that the tag value is automatically reset to zero (0) after a short delay.

|   |       |         |            |
|---|-------|---------|------------|
| <input type="checkbox"/> GSC.CONTROL          | {...} |         | GSCCONTROL |
| <input type="checkbox"/> GSC.CONTROL.Warmboot | 0     | Decimal | BOOL       |

The module is now configured with the values currently entered into the configuration parameters and will begin exchanging data with the processor.

## 4.8 Verifying Communication

There are several ways to verify that the MVI56E-GSCXT module is communicating with the processor and with the Generic ASCII Serial network. You can:

- View Exchanged Data
- View the Module Status in the RSLogix 5000 Controller Tags
- View the LED Status Indicators

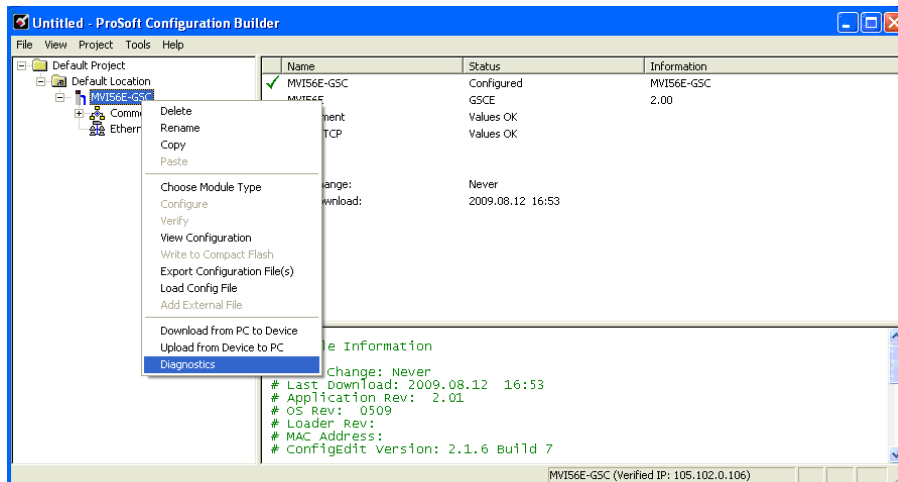
### 4.8.1 Viewing Exchanged Data

The *Data Analyzer*, part of the Diagnostics menu in ProSoft Configuration Builder (PCB), shows you the data being sent and received by the MVI56E-GSCXT. In this step, you will populate the *GSC.DATA.Port1.WriteString* and *GSC.DATA.Port1.WriteLength* controller tags in RSLogix, and then verify that the data travels from the ControlLogix processor to the MVI56E-GSCXT and from the MVI56E-GSCXT to the ControlLogix processor by viewing the *Data Analyzer* and the affected controller tags.

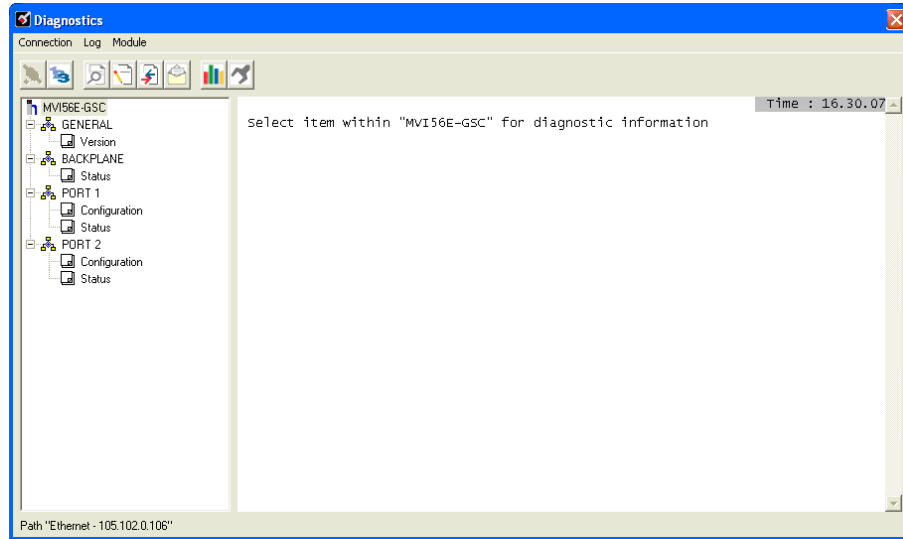
#### Starting the Data Analyzer


Turn on the *Data Analyzer* feature so you can monitor the data exchanged between the processor and the module. Start ProSoft Configuration Builder (PCB) and perform the following sequence.

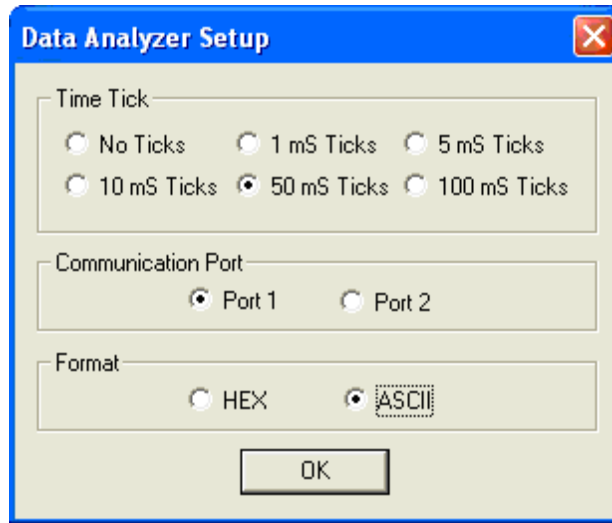
- 1 Start by right-clicking on the module name and left-clicking **DIAGNOSTICS** on the context menu.




This opens the main Diagnostics screen.



- 2 Left-click on the  button to setup the *Data Analyzer* feature as shown.



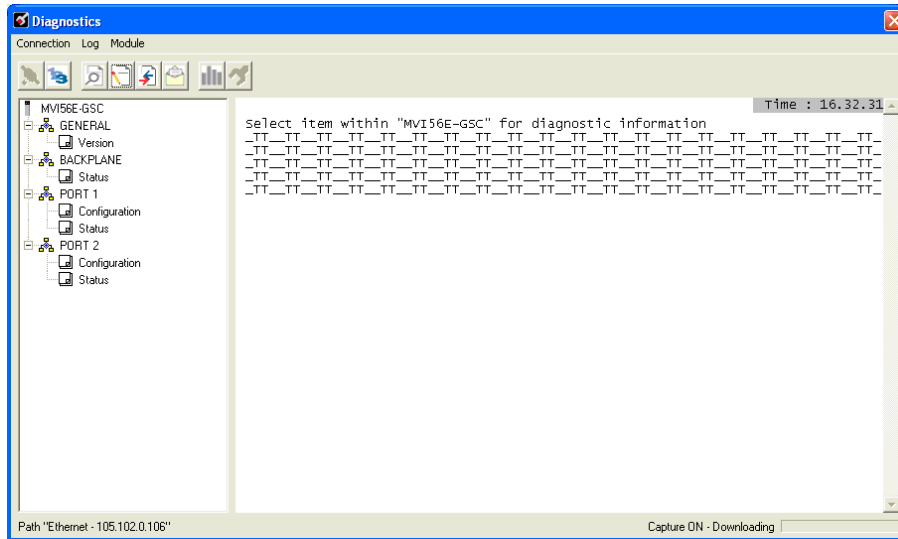
- 3 Left-click on the  button to start the *Data Analyzer* feature.



Click to start Data Analyzer

The following illustration shows an example of the Data Analyzer output with no traffic being passed on the serial network. With no transmitted data to display, the Data Analyzer shows only the "\_TT\_" timing markers, called "Time Ticks". The time between Time Ticks is adjustable and can be set in the Data Analyzer Setup dialog box, as previously seen.

In this example the Time Ticks have been set to 50 milliseconds. During actual transmission of data, the Time Ticks will appear interspersed within the data bytes and indicate the configured time interval. This information can be valuable for troubleshooting certain kinds of communication problems.



The Data Analyzer can display the following special characters.

| Character | Definition  |
|-----------|---|
| [ ]       | Data enclosed in [square brackets] represent data received on the port.   |
| < >       | Data enclosed in <angle brackets> represent data transmitted on the port.   |
| <R+>      | These characters are inserted when the Ready To Send (RTS) line is driven high on the port, just before data transmission begins.   |
| <R->      | These characters are inserted when the RTS line is dropped low on the port, indicating the end of transmission.   |
| <CS>      | These characters are displayed when the Clear to Send (CTS) line is recognized high. These characters will appear only when hardware handshaking is enabled in the port configuration (Use CTS = 1). Most applications do not require the use of hardware handshaking.  |
| _TT_      | These characters are displayed when the "Time Tick" is set to any value other than "No Ticks". Time Ticks will be displayed at the interval selected in the Data Analyzer Setup dialog box and will be interspersed with any data received on the port. ASCII Characters or Decimal byte values shown between any two Time Ticks will indicate the characters received on the port during that time interval. |

Checking Write Data

- 1 Initialize the *Input\_Channel\_Data* array by setting the first 10, or so, elements to ASCII character 00, the Null character.

|                            |        |
|----------------------------|--------|
| [-] Input_Channel_Data     | {...}  |
| [+] Input_Channel_Data[0]  | '\$00' |
| [+] Input_Channel_Data[1]  | '\$00' |
| [+] Input_Channel_Data[2]  | '\$00' |
| [+] Input_Channel_Data[3]  | '\$00' |
| [+] Input_Channel_Data[4]  | '\$00' |
| [+] Input_Channel_Data[5]  | '\$00' |
| [+] Input_Channel_Data[6]  | '\$00' |
| [+] Input_Channel_Data[7]  | '\$00' |
| [+] Input_Channel_Data[8]  | '\$00' |
| [+] Input_Channel_Data[9]  | '\$00' |
| [+] Input_Channel_Data[10] | '\$00' |
| [+] Input_Channel_Data[11] | '\$00' |

- 2 Initialize the *GSC.DATA.Port1.ReadString* array by setting the first 11, or so, elements to ASCII character 00, the Null character.

|                                   |        |       |             |
|-----------------------------------|--------|-------|-------------|
| [-] GSC.DATA                      | {...}  |       | GSCDATA     |
| [-] GSC.DATA.Port1                | {...}  |       | GSCPORTDATA |
| [-] GSC.DATA.Port1.ReadString     | {...}  | ASCII | SINT[4096]  |
| [+] GSC.DATA.Port1.ReadString[0]  | '\$00' | ASCII | SINT        |
| [+] GSC.DATA.Port1.ReadString[1]  | '\$00' | ASCII | SINT        |
| [+] GSC.DATA.Port1.ReadString[2]  | '\$00' | ASCII | SINT        |
| [+] GSC.DATA.Port1.ReadString[3]  | '\$00' | ASCII | SINT        |
| [+] GSC.DATA.Port1.ReadString[4]  | '\$00' | ASCII | SINT        |
| [+] GSC.DATA.Port1.ReadString[5]  | '\$00' | ASCII | SINT        |
| [+] GSC.DATA.Port1.ReadString[6]  | '\$00' | ASCII | SINT        |
| [+] GSC.DATA.Port1.ReadString[7]  | '\$00' | ASCII | SINT        |
| [+] GSC.DATA.Port1.ReadString[8]  | '\$00' | ASCII | SINT        |
| [+] GSC.DATA.Port1.ReadString[9]  | '\$00' | ASCII | SINT        |
| [+] GSC.DATA.Port1.ReadString[10] | '\$00' | ASCII | SINT        |
| [+] GSC.DATA.Port1.ReadString[11] | '\$00' | ASCII | SINT        |
| [+] GSC.DATA.Port1.ReadString[12] | '\$00' | ASCII | SINT        |

- Initialize the *Output\_Channel\_Data* array by setting the first eleven elements of the array to the string value, "clxmessage" (do not include the quotation marks <"> in the string; see below).

|                          |        |
|--------------------------|--------|
| -Output_Channel_Data     | {...}  |
| +Output_Channel_Data[0]  | 'c'    |
| +Output_Channel_Data[1]  | 'l'    |
| +Output_Channel_Data[2]  | 'x'    |
| +Output_Channel_Data[3]  | 'm'    |
| +Output_Channel_Data[4]  | 'e'    |
| +Output_Channel_Data[5]  | 's'    |
| +Output_Channel_Data[6]  | 's'    |
| +Output_Channel_Data[7]  | 'a'    |
| +Output_Channel_Data[8]  | 'g'    |
| +Output_Channel_Data[9]  | 'e'    |
| +Output_Channel_Data[10] | 't'    |
| +Output_Channel_Data[11] | '\$00' |
| +Output_Channel_Data[12] | '\$00' |

- Initialize the *GSC.DATA.Port1.WriteString* array. This sets up the data to be written from the MVI56E-GSCXT Port 1 to the ControlLogix ASCII Serial Port 0. For this test, enter the string "gscmessage" (do not include the quotation marks <"> in the string; see below).

|                                 |        |        |             |
|---------------------------------|--------|--------|-------------|
| -GSC.DATA                       | {...}  |        | GSCDATA     |
| -GSC.DATA.Port1                 | {...}  |        | GSCPORTDATA |
| +GSC.DATA.Port1.ReadString      | {...}  | ASCII  | SINT[4096]  |
| +GSC.DATA.Port1.WriteLength     | 0      | Decima | INT         |
| -GSC.DATA.Port1.WriteString     | {...}  | ASCII  | SINT[4096]  |
| +GSC.DATA.Port1.WriteString[0]  | 'g'    | ASCII  | SINT        |
| +GSC.DATA.Port1.WriteString[1]  | 's'    | ASCII  | SINT        |
| +GSC.DATA.Port1.WriteString[2]  | 'c'    | ASCII  | SINT        |
| +GSC.DATA.Port1.WriteString[3]  | 'm'    | ASCII  | SINT        |
| +GSC.DATA.Port1.WriteString[4]  | 'e'    | ASCII  | SINT        |
| +GSC.DATA.Port1.WriteString[5]  | 's'    | ASCII  | SINT        |
| +GSC.DATA.Port1.WriteString[6]  | 's'    | ASCII  | SINT        |
| +GSC.DATA.Port1.WriteString[7]  | 'a'    | ASCII  | SINT        |
| +GSC.DATA.Port1.WriteString[8]  | 'g'    | ASCII  | SINT        |
| +GSC.DATA.Port1.WriteString[9]  | 'e'    | ASCII  | SINT        |
| +GSC.DATA.Port1.WriteString[10] | '\$00' | ASCII  | SINT        |
| +GSC.DATA.Port1.WriteString[11] | '\$00' | ASCII  | SINT        |
| +GSC.DATA.Port1.WriteString[12] | '\$00' | ASCII  | SINT        |

- 5 Since there are 10 characters in the string, "gscmessage", set the number of characters to be written out of the MVI56E-GSCXT port by entering the value of **10** into the tag, *GSC.DATA.Port1.WriteStringLength*. Entering a non-zero value into this tag will trigger the write operation from the module to the controller.

|                                      |        |        |             |
|--------------------------------------|--------|--------|-------------|
| [-] GSC.DATA                         | {...}  |        | GSCDATA     |
| [-] GSC.DATA.Port1                   | {...}  |        | GSCPORTDATA |
| [+] GSC.DATA.Port1.ReadString        | {...}  | ASCII  | SINT[4096]  |
| [+] GSC.DATA.Port1.WriteStringLength | 10     | Decima | INT         |
| [-] GSC.DATA.Port1.WriteString       | {...}  | ASCII  | SINT[4096]  |
| [+] GSC.DATA.Port1.WriteString[0]    | 'g'    | ASCII  | SINT        |
| [+] GSC.DATA.Port1.WriteString[1]    | 's'    | ASCII  | SINT        |
| [+] GSC.DATA.Port1.WriteString[2]    | 'c'    | ASCII  | SINT        |
| [+] GSC.DATA.Port1.WriteString[3]    | 'm'    | ASCII  | SINT        |
| [+] GSC.DATA.Port1.WriteString[4]    | 'e'    | ASCII  | SINT        |
| [+] GSC.DATA.Port1.WriteString[5]    | 's'    | ASCII  | SINT        |
| [+] GSC.DATA.Port1.WriteString[6]    | 's'    | ASCII  | SINT        |
| [+] GSC.DATA.Port1.WriteString[7]    | 'a'    | ASCII  | SINT        |
| [+] GSC.DATA.Port1.WriteString[8]    | 'g'    | ASCII  | SINT        |
| [+] GSC.DATA.Port1.WriteString[9]    | 'e'    | ASCII  | SINT        |
| [+] GSC.DATA.Port1.WriteString[10]   | '\$00' | ASCII  | SINT        |
| [+] GSC.DATA.Port1.WriteString[11]   | '\$00' | ASCII  | SINT        |
| [+] GSC.DATA.Port1.WriteString[12]   | '\$00' | ASCII  | SINT        |

The *GSC.DATA.Port1.WriteLength* controller tag will be automatically reset to **0** once the data is sent out from the port.

|                                    |        |        |             |
|------------------------------------|--------|--------|-------------|
| [-] GSC.DATA                       | {...}  |        | GSCDATA     |
| [-] GSC.DATA.Port1                 | {...}  |        | GSCPORTDATA |
| [+] GSC.DATA.Port1.ReadString      | {...}  | ASCII  | SINT[4096]  |
| [+] GSC.DATA.Port1.WriteLength     | 0      | Decima | INT         |
| [-] GSC.DATA.Port1.WriteString     | {...}  | ASCII  | SINT[4096]  |
| [+] GSC.DATA.Port1.WriteString[0]  | 'g'    | ASCII  | SINT        |
| [+] GSC.DATA.Port1.WriteString[1]  | 's'    | ASCII  | SINT        |
| [+] GSC.DATA.Port1.WriteString[2]  | 'c'    | ASCII  | SINT        |
| [+] GSC.DATA.Port1.WriteString[3]  | 'm'    | ASCII  | SINT        |
| [+] GSC.DATA.Port1.WriteString[4]  | 'e'    | ASCII  | SINT        |
| [+] GSC.DATA.Port1.WriteString[5]  | 's'    | ASCII  | SINT        |
| [+] GSC.DATA.Port1.WriteString[6]  | 's'    | ASCII  | SINT        |
| [+] GSC.DATA.Port1.WriteString[7]  | 'a'    | ASCII  | SINT        |
| [+] GSC.DATA.Port1.WriteString[8]  | 'g'    | ASCII  | SINT        |
| [+] GSC.DATA.Port1.WriteString[9]  | 'e'    | ASCII  | SINT        |
| [+] GSC.DATA.Port1.WriteString[10] | '\$00' | ASCII  | SINT        |
| [+] GSC.DATA.Port1.WriteString[11] | '\$00' | ASCII  | SINT        |
| [+] GSC.DATA.Port1.WriteString[12] | '\$00' | ASCII  | SINT        |

The *WriteData* message will be passed across the backplane to the MVI56E-GSCXT and the string, "gscmessage", will be transmitted out of Application Port 1, which is connected to the ControlLogix processor ASCII Serial Port 0. The Controllogix processor will receive the data on its serial port and, based on the sample ladder rungs (see Sample Application (page 51)), it will copy the data received through its ASCII port to the *Input\_Channel\_Data* controller tag array.

|                            |        |
|----------------------------|--------|
| [-] Input_Channel_Data     | {...}  |
| [+] Input_Channel_Data[0]  | 'g'    |
| [+] Input_Channel_Data[1]  | 's'    |
| [+] Input_Channel_Data[2]  | 'c'    |
| [+] Input_Channel_Data[3]  | 'm'    |
| [+] Input_Channel_Data[4]  | 'e'    |
| [+] Input_Channel_Data[5]  | 's'    |
| [+] Input_Channel_Data[6]  | 's'    |
| [+] Input_Channel_Data[7]  | 'a'    |
| [+] Input_Channel_Data[8]  | 'g'    |
| [+] Input_Channel_Data[9]  | 'e'    |
| [+] Input_Channel_Data[10] | '\$00' |
| [+] Input_Channel_Data[11] | '\$00' |

Checking Read Data

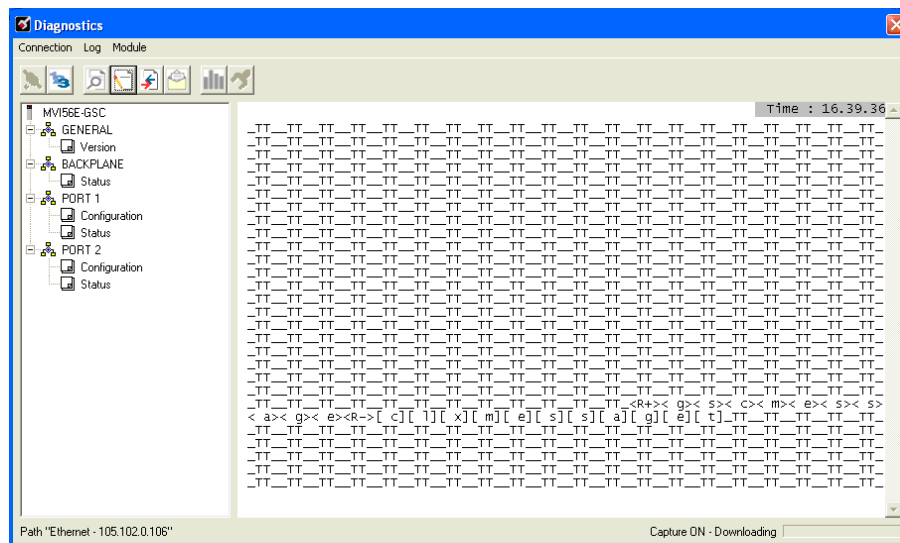
The sample application ladder logic (see Sample Application (page 51)) waits for the "gscmessage" string to be received on the processor's ASCII Serial Port 0. Once received, the Sample Application logic sends data from the *Output\_Channel\_Data* array to the Controllogix ASCII port for transmission back to the MVI56E-GSCXT Application Port 1.

|                             |        |
|-----------------------------|--------|
| [-] Output_Channel_Data     | {...}  |
| [+] Output_Channel_Data[0]  | 'c'    |
| [+] Output_Channel_Data[1]  | 'l'    |
| [+] Output_Channel_Data[2]  | 'x'    |
| [+] Output_Channel_Data[3]  | 'm'    |
| [+] Output_Channel_Data[4]  | 'e'    |
| [+] Output_Channel_Data[5]  | 's'    |
| [+] Output_Channel_Data[6]  | 's'    |
| [+] Output_Channel_Data[7]  | 'a'    |
| [+] Output_Channel_Data[8]  | 'g'    |
| [+] Output_Channel_Data[9]  | 'e'    |
| [+] Output_Channel_Data[10] | 't'    |
| [+] Output_Channel_Data[11] | '\$00' |
| [+] Output_Channel_Data[12] | '\$00' |

The module receives the "clxmessage" string. During configuration, the module was told to look for ASCII character "t" as the terminating or last character of an incoming string (see Configure the MVI56E-GSCXT Serial Port (page 50)). When the module sees the "t" character, it sends the entire message string across the backplane in an Input Data Block to the processor, where the standard logic copies the data to the *GSC.DATA.Port1.ReadString* tag array.

|                                 |        |       |             |
|---------------------------------|--------|-------|-------------|
| - GSC.DATA                      | {...}  |       | GSCDATA     |
| - GSC.DATA.Port1                | {...}  |       | GSCPORTDATA |
| - GSC.DATA.Port1.ReadString     | {...}  | ASCII | SINT[4096]  |
| + GSC.DATA.Port1.ReadString[0]  | 'c'    | ASCII | SINT        |
| + GSC.DATA.Port1.ReadString[1]  | 'l'    | ASCII | SINT        |
| + GSC.DATA.Port1.ReadString[2]  | 'x'    | ASCII | SINT        |
| + GSC.DATA.Port1.ReadString[3]  | 'm'    | ASCII | SINT        |
| + GSC.DATA.Port1.ReadString[4]  | 'e'    | ASCII | SINT        |
| + GSC.DATA.Port1.ReadString[5]  | 's'    | ASCII | SINT        |
| + GSC.DATA.Port1.ReadString[6]  | 's'    | ASCII | SINT        |
| + GSC.DATA.Port1.ReadString[7]  | 'a'    | ASCII | SINT        |
| + GSC.DATA.Port1.ReadString[8]  | 'g'    | ASCII | SINT        |
| + GSC.DATA.Port1.ReadString[9]  | 'e'    | ASCII | SINT        |
| + GSC.DATA.Port1.ReadString[10] | 't'    | ASCII | SINT        |
| + GSC.DATA.Port1.ReadString[11] | '\$00' | ASCII | SINT        |
| + GSC.DATA.Port1.ReadString[12] | '\$00' | ASCII | SINT        |

At this point you can check the *Data Analyzer* and see that it shows the data sent by the module, <g><s><c><m><e><s><s><a><g><e>, and the data received by the module, [c][l][x][m][e][s][s][a][g][e][t], has passed through Application Port 1.



Seeing data in the Data Analyzer and in the controller tags verifies that the module configuration is valid and that communication has been established.

### 4.8.2 Checking Module Status in RSLogix5000 Controller Tags

General module status and port communication status can be viewed in the controller tag array called **GSC.STATUS**. This tag array holds all available status data, as defined in the **GSCSTATUS** User-defined Data Type (UDT) object. Definitions of the values found in this array are provided below. See the *MVI56E-GSCXT User Manual* for additional details on the **GSCSTATUS** object.

#### Status Data Definition

This section contains a description of the members present in the **GSC.STATUS** controller tags. This data is transferred from the module to the processor as part of each read block.

| Status Data Object in GSCInStat | Input Data Block Offset | Description   |
|---------------------------------|-------------------------|---|
| PassCnt                         | 210 to 211              | Program cycle counter   |
| Product                         | 212 to 215              | Product name as ASCII string  |
| Rev                             | 216 to 219              | Revision level as ASCII string  |
| OP                              | 220 to 223              | Operating system level as ASCII string  |
| Run                             | 224 to 227              | Run number as ASCII string  |
| BlkErrs.Read                    | 228 to 229              | Number of blocks transferred from module to processor   |
| BlkErrs.Write                   | 230 to 231              | Number of blocks transferred from processor to module   |
| BlkErrs.Parse                   | 232 to 233              | Number of blocks parsed by module   |
| BlkErrs.Err                     | 234 to 235              | Number of block errors in module  |
| Port1.RxState                   | 236 to 237              | Port 1 receive state:<br>-1 = Listening for data<br>1 = Receiving Port Data<br>2 = Waiting for Backplane transfer   |
| Port1.RXCharCnt                 | 238 to 239              | Port 1 receive character count  |
| Port1.RxMsgCnt                  | 240 to 241              | Port 1 receive block count  |
| Port1.TxState                   | 242 to 243              | Port 1 transmit state:<br>0 = Waiting for Data to Send<br>1 = RTS On<br>2 = RTS Timeout<br>3 = Sending data<br>4 = Waiting for RTS Off<br>5 = RTS turned off<br>30 = Intercharacter Delay<br>31 = Intercharacter Delay<br>32 = Intercharacter Delay<br>100 = Message Delay before Transmit<br>101 = Message Delay before Transmit |
| Port1.TxCharCnt                 | 244 to 245              | Port 1 transmit character count   |
| Port1.TxMsgCnt                  | 246 to 247              | Port 1 transmit block count   |
| Port1.ErrorWord                 | 248 to 249              | Port 1 Error Word   |

| <b>Status Data Object<br/>in GSCInStat</b> | <b>Input Data<br/>Block Offset</b> | <b>Description</b>  |
|--|------------------------------------|---|
| Port2.RxState                              | 456 to 457                         | Port 2 receive state:<br>-1 = Listening for data<br>1 = Receiving Port Data<br>2 = Waiting for Backplane transfer   |
| Port2.RXCharCnt                            | 458 to 459                         | Port 2 receive character count  |
| Port2.RxMsgCnt                             | 460 to 461                         | Port 2 receive block count  |
| Port2.TxState                              | 462 to 463                         | Port 2 transmit state:<br>0 = Waiting for Data to Send<br>1 = RTS On<br>2 = RTS Timeout<br>3 = Sending data<br>4 = Waiting for RTS Off<br>5 = RTS turned off<br>30 = Intercharacter Delay<br>31 = Intercharacter Delay<br>32 = Intercharacter Delay<br>100 = Message Delay before Transmit<br>101 = Message Delay before Transmit |
| Port2.TxCharCnt                            | 464 to 465                         | Port 2 transmit character count   |
| Port2.TxMsgCnt                             | 466 to 467                         | Port 2 transmit block count   |
| Port2.ErrorWord                            | 468 to 469                         | Port 2 Error Word   |

### 4.8.3 Scrolling LED Status Indicators

The scrolling LED display indicates the module's operating status as follows:

#### Initialization Messages

| Message  | Meaning  |
|--|--|
| "Boot"   | Module is initializing its operating system  |
| "Waiting for Processor Connection..."  | Module is not able to establish a backplane connection with the ControlLogix PAC. Possible causes: <ul style="list-style-type: none"> <li>▪ Ladder logic or AOI is not loaded on processor</li> <li>▪ Module is located in a different slot than the one configured in the ladder logic/AOI</li> <li>▪ Module I/O properties are invalid</li> <li>▪ Processor is not in RUN or REM RUN mode</li> </ul>   |
| "INIT"   | Module is beginning its firmware initialization  |
| "Ladder Logic Configuration Required"  | Module is waiting for valid module configuration data from ladder logic. Check that module configuration parameters are set to valid values in the ladder logic controller tags  |
| "UPDATING"   | Module is performing a firmware update, initiated from the web page.   |
| "STOP"   | Module firmware is stopping after receiving a "Cold Boot" request, which forces an automatic re-start, same as from power-up   |
| "MVI-56E-GSC <Version#> Last Config: <LCfgDate> Config P1: <TermType> <BaudRate>, <Parity>, <DataBits>, <StopBits>, <RS_Type>, P2: <TermType> <BaudRate>, <Parity>, <DataBits>, <StopBits>, <RS_Type>" | After power up and every reconfiguration reboot, the module will display the configuration of both ports. This message will scroll through once and not be repeated until the next reboot. The message contains: <ul style="list-style-type: none"> <li>▪ &lt;Version#&gt; Firmware revision number, as in "V2.01"</li> <li>▪ &lt;LCfgDate&gt; Date of last configuration change (reboot)</li> <li>▪ &lt;TermType&gt;: Termination Type; options are:<br/>               STREAM - Streaming<br/>               TERM - Terminating Characters<br/>               DELAY - Intercharacter Delay<br/>               PACKET - Packet Length<br/>               MSGTO - Message Timeout</li> <li>▪ &lt;BaudRate&gt;: 115200 / 57600 / 38400 / 19200 / 9600 / 4800 / 2400 / 1200 / 600 / 300</li> <li>▪ &lt;Parity&gt;: None / Even / Odd</li> <li>▪ &lt;DataBits&gt;: 7 / 8</li> <li>▪ &lt;StopBits&gt;: 1 / 2</li> <li>▪ &lt;RS_Type&gt;: RS-232 / RS-422 / RS-485</li> </ul> |

### Initialization Error Messages

During initialization, if the module detects a unrecoverable fault that will prevent communication, an error message will be displayed. If any of these errors persist after repeated reboot attempts, contact ProSoft Technology Technical Support for further assistance.

| Message | Meaning   |
|---------|---|
| "E003"  | Startup sequence failed, module failed to initialize  |
| "E005"  | Displayed during startup if there is an error opening application serial Port 1 (P1)                        |
| "E006"  | Displayed during startup if there is an error opening application serial Port 2 (P2)                        |
| "E008"  | Displayed during startup if either of the two application serial ports fails to open after a second attempt |
| "E009"  | Displayed during startup if there is an error setting up the signal handlers)                               |

### Operation Message

After the initialization step, the following message pattern will be repeated on the scrolling LED display.

```
<Backplane Status> <IP Address> <Backplane Status> <Port1 Status> <Port2
Status>
```

| Message Component  | Meaning  |
|--------------------|--|
| <Backplane Status> | "OK": Module is communicating with processor                   |
| <IP Address>       | Module IP address  |
| <Port1 Status>     | "Enabled": Port is enabled, sending and receiving possible     |
| <Port2 Status>     | "Disabled": Port is disabled, no sending or receiving possible |

## 5 Building on Success

### In This Chapter

- ❖ Frequently Asked Questions ..... 68

Now that you have successfully installed, configured, and verified operation of the MVI56E-GSCXT module, you should have a better understanding of how to make it work for your specific application. The following resources are available to help you build on your success.

- For more information about the MVI56E-GSCXT module, including detailed hardware and software configuration, troubleshooting, and application information, refer to the *MVI56E-GSCXT User Manual*.
- For technical support and warranty information for your MVI56E-GSCXT module, refer to *Support, Service, and Warranty* in the *MVI56E-GSCXT User Manual*.
- For more information on ProSoft Technology products and services, please visit [www.prosoft-technology.com](http://www.prosoft-technology.com).

## 5.1 Frequently Asked Questions

### 5.1.1 *What are the differences between the MVI56 and the MVI56E modules? What does the "E" stand for?*

The "E" stands for *Enhanced with Ethernet* communication capabilities. The new enhancements are:

- **PCB:** MVI56E-GSCXT products now use PCB (ProSoft Configuration Builder) software; a Windows-based configuration utility providing a new graphic user interface for module diagnostics with screen navigation, improving interoperability with the module.
- **Seamless Migration:** MVI56E products are backward compatible with existing ladder logic and module configuration files, allowing for a smooth "plug and play" transition when replacing the earlier version MVI56-GSC product.
- **Personality Memory Module:** The module incorporates a non-volatile compact flash (CF) memory card for storing the module's network configuration data (or personality). This feature allows quick replacement of faulted modules and restoration of systems by a simple exchange of the Personality Memory Module, with absolutely no other PC or configuration requirements.
- **Ethernet Configuration Port:** Allows for remote module connectivity. Replaces serial communication and becomes the primary configuration port, as well as providing a web server interface.
- **LED Display:** The 4-character, scrolling, alphanumeric LED display provides additional detailed plain English diagnostic and error information, backplane communication, and network conditions for the module, .
- **Discovery Service:** Allows PCB configuration software (or separate utility) to find and display products located on the network using key product attributes such as name, serial number, and IP address. The user will be able to change IP address, upload/download, and enter into diagnostics from the list.
- **CIPconnect<sup>®</sup> enabled:** Allows end-users to connect from remote locations to local- and remote-chassis-installed MVI56E modules from anywhere on a Rockwell Automation<sup>®</sup> EtherNet/IP<sup>™</sup> or ControlNet<sup>™</sup> process network, up to six remote links away.

### 5.1.2 *Is the MVI56E product a direct replacement to my existing MVI56 product?*

Yes

### 5.1.3 *How is the MVI56E-GSCXT configured?*

The module is configured with ladder logic controller tags. A warm boot operation is required to transfer all parameters from the processor to the module.

### **5.1.4 What is ProSoft Configuration Builder (PCB)?**

ProSoft Configuration Builder (PCB) provides a quick and easy way to manage module diagnostics and troubleshooting operations. Built-in module diagnostics menus and the serial port data stream analyzer can be accessed using PCB through the module's high-speed Ethernet configuration port (E1) or through Rockwell Automation 1756-ENxT or 1756-CNBx communications interfaces using CIPconnect®.

### **5.1.5 What is ProSoft Discovery Service (PDS)?**

ProSoft Discovery Service (PDS) is Windows-based software that connects to the Ethernet port of the module for the following purposes:

- Automatic module discovery on the Ethernet network
- Set a temporary IP address for the module for easy commissioning
- Allow PCB to select the module for monitoring and IP address reconfiguration

The ProSoft Discovery Service software is supplied as a stand-alone utility, as well as being integrated into PCB.

### **5.1.6 What is the purpose of the MVI56E-GSCXT Ethernet Config (E1) Port?**

The MVI56E-GSCXT Ethernet Port (E1) allows a remote PC to set the module's IP address, as well as monitor module operation using ProSoft Configuration Builder (PCB) diagnostics.

You can also set the IP address with ladder logic. Refer to the *MVI56E-GSCXT User Manual* for more information on this option.

### **5.1.7 How do I change the module's IP address?**

- 1 Use ProSoft Configuration Builder to edit and download the Ethernet configuration to the module.
- 2 Use the Optional Add-On Instruction (AOI). The AOI file is available on the *ProSoft Solutions CD-ROM*, or can be downloaded from [www.prosoft-technology.com](http://www.prosoft-technology.com).

### **5.1.8 Does the MVI56E-GSCXT module require processor logic?**

Yes, ladder logic is required for data transfer between the MVI56E module and the ControlLogix® processor.

- For RSLogix™ 5000 version 16 applications (or later), the included Add-On Instruction encapsulates the entire ladder logic into one single instruction.
- For RSLogix 5000 version 15 and older, sample ladder logic is available from the ProSoft Technology® website at [www.prosoft-technology.com](http://www.prosoft-technology.com).

### ***5.1.9 What is the purpose of the Optional MVI56E-GSCXT Add-On Instruction?***

The Optional Add-On Instruction (AOI) allows the processor to perform the following tasks:

- 1** Set the MVI56E Ethernet settings
- 2** Read the MVI56E Ethernet settings
- 3** Set MVI56E date/time information
- 4** Read the MVI56E date/time information

Items 1 and 2 can also be performed through ProSoft Configuration Builder (PCB) using ProSoft Discovery Service. Items 3 and 4 can also be performed through the module's built-in web page.

The Optional AOI is needed only for specific applications where Ethernet or CIPconnect access from a programmer's personal computer (PC) to the module is not possible.

### ***5.1.10 How do I monitor MVI56E-GSCXT operation?***

Module operation can be monitored either through the processor controller tags or through the ProSoft Configuration Builder diagnostic window. Available status information includes number of messages sent, number of messages received, number of errors, and error codes.

### ***5.1.11 Are there any other ways to monitor module diagnostics besides being connected to the module's network (subnet)?***

PCB can monitor the module via ControlLogix backplanes and process networks using CIPconnect. The PC running PCB can use its Ethernet port to connect to any 1756-ENxT EtherNet/IP™ interface module which is on the same Ethernet subnet. Through this connection, PCB can use CIPconnect to route through the ControlLogix backplane to other 1756-ENxT or 1756-CNBx modules, for up to five more route links, to reach an MVI56E module in a chassis connected on EtherNet/IP or ControlNet™ process networks.

## Glossary of Terms

### A

#### **ASCII**

American Standard Code for Information Interchange. A communication mode in which each eight-bit byte in a message contains one ASCII character code. ASCII characters (or hexadecimal characters) are sometimes used as a key to encrypt data and ensure its secure transmission.

### B

#### **Baud Rate**

The speed of communication between devices on the network. All devices must communicate at the same rate.

### D

#### **DCE**

Data communications equipment. A modem, for example.

#### **Default Gateway**

The IP address of a network router where data is sent if the destination IP address is outside the local subnet. The gateway is the device that routes the traffic from the local area network to other networks such as the Internet.

#### **DTE**

Data terminal equipment. A computer or terminal, for example.

### E

#### **ESD**

Electrostatic Discharge. Can cause internal circuit damage to the coprocessor.

#### **Ethernet**

A set of network cabling and network access (CSMA/CD) protocol standards for bus topology computer networks invented by Xerox but now controlled by the 802.3 subcommittee of the IEEE.

### F

#### **Firmware**

Software for embedded computers.

### **Full-Duplex**

A communications circuit or system designed to simultaneously transmit and receive two different streams of data. Telephones are an example of a full-duplex communication system. Both parties on a telephone conversation can talk and listen at the same time. If both talk at the same time, their two signals are not corrupted.

## **H**

### **Half-Duplex**

A communications circuit or system designed to transmit and receive data, but not both simultaneously. CB or walkie-talkie radios are an example of a half-duplex communication system. Either parties on a radio conversation may talk or listen; but both cannot talk at the same time without corrupting each other's signal. If one operator is "talking", the other must be "listening" to have successful communication.

## **I**

### **IP Address**

A 32-bit identification number for each node on an Internet Protocol network. These addresses are represented as four sets of 8-bit numbers (numbers from 0 to 255), separated by periods ("dots").

Networks using the TCP/IP Protocol route messages based on the IP address of the destination. Each number can be 0 to 255. For example, 192.168.0.100 could be an IP address. Each node on the network must have a unique IP address.

## **L**

### **LED**

Light-emitting diode.

## **M**

### **MAC ID**

A hexadecimal number that uniquely identifies an Ethernet device.

### **Master**

A Master is a device that makes requests for information from a software program, or the device on which that program runs, in a Master-Slave relationship.

A Client on an Ethernet network is equivalent to a Master on a Serial network.

## **N**

### **Network**

A series of stations or nodes connected by some type of communication medium. A network may consist of a single link or multiple links.

### **Node**

An address or software location on the network.

## P

### **Peer-to-Peer**

A network relationship between devices where each device can send commands as a master or client, and respond to commands as a slave or server.

### **Power Supply**

Device that supplies electrical power to the I/O chassis containing the processor, coprocessor, or other modules.

### **Protocol**

The language or packaging of information that is transmitted between nodes on a network.

## R

### **RS-232**

Recommended Standard 232; the standard for serial binary signals between DTE and DCE devices.

## S

### **Serial Data/Serial Data Transmission**

Data that is transferred one bit at a time. Serial data transmission involves changing a carrier signal line between two possible states to indicate a binary 0 or binary 1 value. Successive data bits are rapidly transmitted one after the other with a fixed time allowed for each bit. Data bits are usually grouped into "packets", which contain a specific amount of data bits, along with extra bits included to provide error-checking capability.

### **Simplex**

A communications circuit or system designed to either transmit data or receive data, but not both. Broadcast television is an example of simplex communication system. A television station sends a TV signal but cannot receive responses back from the television sets to which it is transmitting. The TV sets can receive the signal from the TV station but cannot transmit back to the station.

### **Slave**

A Slave is a software program, or the device on which that program runs, which provides a specific kind of service to a Master software program, or the device on which that program runs, on a serial network.

A Slave on a Serial network is equivalent to a Server on an Ethernet network.

**Subnet Mask**

A mask used to determine what subnet an IP address belongs to. An IP address has two components: the network address, and the host (node or device) address. For example, consider the IP address 150.215.017.009. Assuming this is part of a Class B network (with a subnet mask of 255.255.0.0), the first two numbers (150.215) represent the Class B network address, and the second two numbers (017.009) identify a particular host on this network.

## Index

### A

Architecture • 19  
Are there any other ways to monitor module  
diagnostics besides being connected to the  
module's network (subnet)? • 70  
ASCII • 71

### B

Battery Life Advisory • 4  
Baud Rate • 71  
Before You Begin • 11  
Building on Success • 67

### C

Checking Module Status in RSLogix5000 Controller  
Tags • 62  
Checking Read Data • 60  
Checking Write Data • 56  
Configuring the ControlLogix (CLX) Serial Port • 47  
Configuring the MVI56E-GSCXT Serial Port • 49, 61  
Connecting the MVI56E-GSCXT and ControlLogix  
PAC Serial Ports • 27  
Connecting Your PC to the Module's Ethernet Port • 27

### D

Data Flow • 20  
DCE • 71  
Default Gateway • 71  
Does the MVI56E-GSCXT module require processor  
logic? • 69  
Downloading the Sample Program to the Processor •  
8, 24, 51  
DTE • 71

### E

ESD • 71  
Ethernet • 71  
Example 1  
Local Rack Application • 39  
Example 2  
Remote Rack Application - CIPconnect No  
Download • 42

### F

Firmware • 71  
Frequently Asked Questions • 68  
Full-Duplex • 72

### H

Half-Duplex • 72

How do I change the module's IP address? • 69  
How do I monitor MVI56E-GSCXT operation? • 70  
How is the MVI56E-GSCXT configured? • 68  
How to Contact Us • 2

### I

Important Safety Information • 3  
Installing the Module in the Rack • 26  
IP Address • 72  
Is the MVI56E product a direct replacement to my  
existing MVI56 product? • 68

### L

Learning Objectives • 8  
LED • 72  
Loading the Sample Application • 8, 24, 46

### M

MAC ID • 72  
Master • 72

### N

Network • 72  
Node • 72

### O

Overview • 18

### P

Peer-to-Peer • 73  
Physical Setup • 8, 24, 25  
Power Supply • 73  
Prerequisites • 10  
Procedures • 23  
ProSoft Technology Documentation • 9  
ProSoft Technology® Product Documentation • 2  
Protocol • 73

### R

Rebooting the Module to Transfer New Parameter  
Values • 52  
Required Items • 15  
Required Steps • 24  
RS-232 • 73

### S

Sample Application Setup • 8, 16, 24, 47  
Sample Files • 16  
Scope • 7  
Scrolling LED Status Indicators • 64  
Serial Data/Serial Data Transmission • 73  
Setting Module Jumpers • 25  
Setting Up a Permanent IP Address • 32, 34  
Setting Up a Temporary IP Address • 31  
Setting Up the Project • 29  
Simplex • 73  
Slave • 73

Starting the Data Analyzer • 53  
Status Data Definition • 62  
Subnet Mask • 74  
System Requirements • 14

## **T**

The Sample Application • 17

## **U**

Using CIPconnect® to Connect to the Module • 34, 36  
Using ProSoft Configuration Builder Software • 28  
Using RSWho to Connect to the Module • 34, 45

## **V**

Verifying Communication • 8, 24, 53  
Viewing Exchanged Data • 53

## **W**

What are the differences between the MVI56 and the  
MVI56E modules? What does the • 68  
What is ProSoft Configuration Builder (PCB)? • 69  
What is ProSoft Discovery Service (PDS)? • 69  
What is the purpose of the MVI56E-GSCXT Ethernet  
Config (E1) Port? • 69  
What is the purpose of the Optional MVI56E-GSCXT  
Add-On Instruction? • 70  
What's Different? • 13  
What's In the Sample Application? • 50, 59, 60  
What's New? • 12

## **Y**

Your Feedback Please • 2