inRAx

MVI56-CAS

ControlLogix Platform

Teledyne CA Slave Module

User Manual

December 26, 2007

ProSoft TECHNOLOGY

Please Read This Notice

Successful application of this module requires a reasonable working knowledge of the Rockwell Automation ControlLogix hardware, the MVI56-CAS Module and the application in which the combination is to be used. For this reason, it is important that those responsible for implementation satisfy themselves that the combination will meet the needs of the application without exposing personnel or equipment to unsafe or inappropriate working conditions.

This manual is provided to assist the user. Every attempt has been made to ensure that the information provided is accurate and a true reflection of the product's installation requirements. In order to ensure a complete understanding of the operation of the product, the user should read all applicable Rockwell Automation documentation on the operation of the Rockwell Automation hardware.

Under no conditions will ProSoft Technology be responsible or liable for indirect or consequential damages resulting from the use or application of the product.

Information in this manual is subject to change without notice and does not represent a commitment on the part of ProSoft Technology Improvements and/or changes in this manual or the product may be made at any time. These changes will be made periodically to correct technical inaccuracies or typographical errors.

Warnings

UL 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.

C Suitable for use in Class I, division 2 Groups A, B, C and D Hazardous Locations or Non-Hazardous Locations.

ATEX 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. The devices shall provide external means to prevent the rated voltage being exceeded by transient disturbances of more than 40%. This device must be used only with ATEX certified backplanes.

D DO NOT OPEN WHEN ENERGIZED.

Electrical Ratings

- Backplane Current Load: 800 mA @ 5 V DC; 3mA @ 24V DC
- Operating Temperature: 0 to 60°C (32 to 140°F)
- Storage Temperature: -40 to 85°C (-40 to 185°F)
- Shock: 30g Operational; 50g non-operational; Vibration: 5 g from 10 to 150 Hz
- Relative Humidity 5% to 95% (non-condensing)
- All phase conductor sizes must be at least 1.3 mm(squared) and all earth ground conductors must be at least 4mm(squared).

Markings:

 II 3 G 0C <=Ta<= 60C EEx nA IIC T4 DEMKO 07ATEX0710717X

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 the product, documentation or support, please write or call us.

ProSoft Technology
1675 Chester Avenue, Fourth Floor
Bakersfield, CA 93301
+1 (661) 716-5100
+1 (661) 716-5101 (Fax)
http://www.prosoft-technology.com

MVI56-CAS User Manual
December 26, 2007
PSFT.CAS.MVI56.UM.07.12.26

Contents

Guide to the MVI56-CAS User Manual

Function		Section to Read	Details
Introduction (Must Do)	→	Start Here (page 9)	This Section introduces the customer to the module. Included are: package contents, system requirements, hardware installation, and basic configuration.
Verify Communication, Diagnostic and Troubleshooting	→	Verifying Communication (page 31) Diagnostics and Troubleshooting (page 31)	This section describes how to verify communications with the network. Diagnostic and Troubleshooting procedures.
Reference Product Specifications Functional Overview Glossary	→	Reference (page 47) Functional Overview (page 49) Product Specifications (page 47)	These sections contain general references associated with this product, Specifications, and the Functional Overview.
Support, Service, and Warranty Index	→	Support, Service and Warranty (page 75)	This section contains Support, Service and Warranty information. Index of chapters.

1 Start Here

In This Chapter

Installing the MVI56-CAS module requires a reasonable working knowledge of the Rockwell Automation hardware, the MVI56-CAS Module and the application in which they will be used.

Caution: It is important that those responsible for implementation can complete the application without exposing personnel, or equipment, to unsafe or inappropriate working conditions. Safety, quality and experience are key factors in a successful installation.

1.1 System Requirements

The MVI56-CAS module requires the following minimum hardware and software components:

- Rockwell Automation ControlLogix™ processor, with compatible power supply and one free slot in the rack, for the MVI56-CAS module. The module requires 800mA of available power.
- Rockwell Automation RSLogix 5000 programming software version 2.51 or higher.
- Rockwell Automation RSLinx communication software
- Pentium® II 450 MHz minimum. Pentium III 733 MHz (or better) recommended
- Supported operating systems:
  - 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
- HyperTerminal or other terminal emulator program.

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.

1.2 Package Contents

The following components are included with your MVI56-CAS module, and are all required for installation and configuration.

Important: Before beginning the installation, please verify that all of the following items are present.

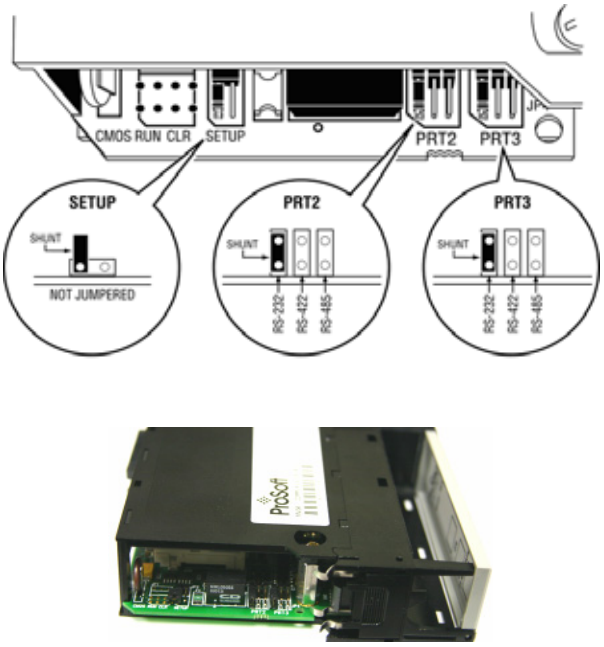
Qty.	Part Name	Part Number	Part Description
1	MVI56-CAS Module	MVI56-CAS	Teledyne CA Slave Module
1	Cable	RS232 Null Modem	For RS232 Connection to the CFG Port
3	Cable	Cable #14, RJ45 to DB9 Male Adapter	For DB9 Connection to Module's Port
2	Adapter	1454-9F	Two Adapters, DB9 Female to Screw Terminal. For RS422 or RS485 Connections to Port 1 and 2 of the Module
1	ProSoft Solutions CD		Contains sample programs, utilities and documentation for the MVI56-CAS module.

If any of these components are missing, please contact ProSoft Technology Support for replacement parts.

1.3 Setting Jumpers

If you use an interface other than RS-232 (default), you must change the jumper configuration to match the interface. There are three jumpers located at the bottom of the module.

The following illustration shows the MVI56-CAS jumper configuration:



1. Set the PRT 2 (for application port 1) and PRT 3 (for application port 2) jumpers for RS232, RS422 or RS485 to match the wiring needed for your application. The default jumper setting for both application ports is 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. Do not jumper the Setup pins together unless you are directed to do so by ProSoft Technical Support.

1.4 Install the Module in the Rack

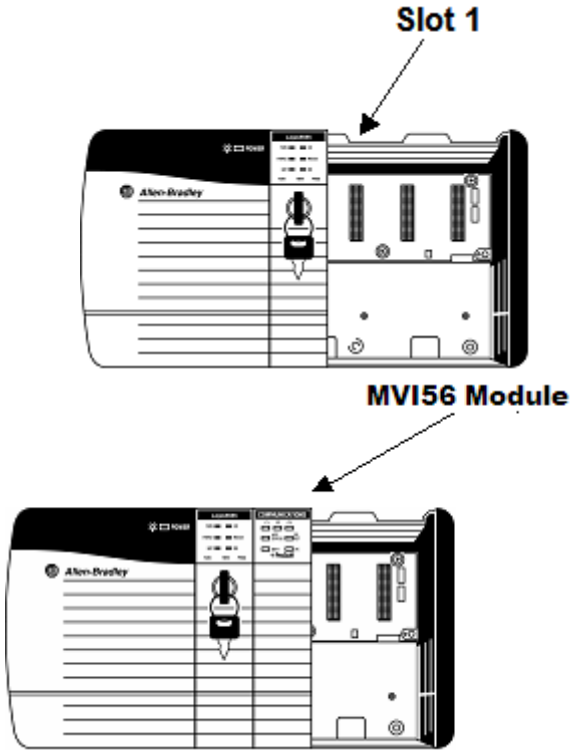
If you have not already installed and configured your ControlLogix processor and power supply, please do so before installing the MVI56-CAS module. Refer to your Rockwell Automation product documentation for installation instructions.

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 MVI56-CAS into the ControlLogix chassis. Use the same technique recommended by Rockwell Automation to remove and install ControlLogix modules.

Warning: When you insert or remove the module while backplane power is on, an electrical arc can occur. This could cause an explosion in hazardous location installations. 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. Turn power OFF.
2. Align the module with the top and bottom guides, and slide it into the rack until the module is firmly against the backplane connector.

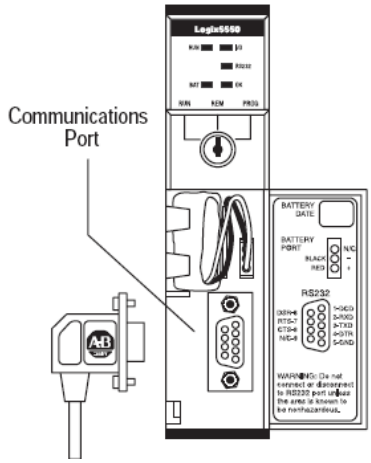


3 With a firm but steady push, snap the module into place.
4 Check that the holding clips on the top and bottom of the module are securely in the locking holes of the rack.
5 Make a note of the slot location. You will need to identify the slot in which the module is installed in order for the sample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

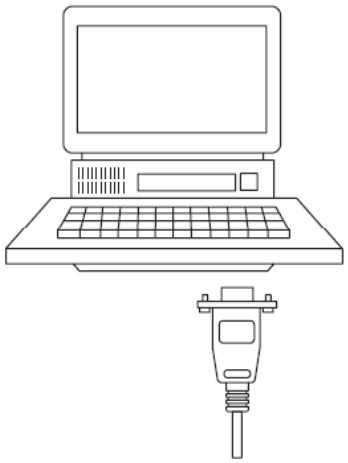
Note: If you insert the module improperly, the system may stop working, or may behave unpredictably.
Note: If you are installing MVI56-CAS with other modules connected to the PCI bus, the peripheral modules will not have holding clips. Make sure all of the modules are aligned with their respective slots before you snap them into place.

1.5 Connect your PC to the Processor

7 Connect the right-angle connector end of the cable to your controller at the communications port.



8 Connect the straight connector end of the cable to the serial port on your computer.



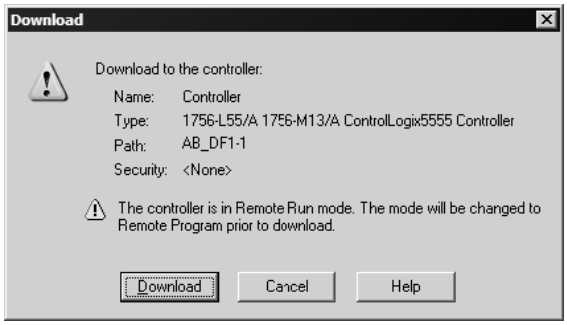
1.6 Download the Sample Program to the Processor

➢ ***To download the sample program from RSLogix 5000 to the ControlLogix processor:***

Note: The key switch on the front of the ControlLogix module must be in the REM position.

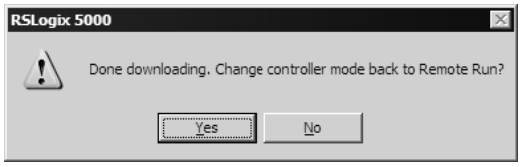
1 If you are not already online to the processor, open the Communications menu, and then choose Download. RSLogix will establish communication with the processor.

2 When communication is established, RSLogix will open a confirmation dialog box. Click the Download button to transfer the sample program to the processor.



3 RSLogix will compile the program and transfer it to the processor. This process may take a few minutes.

4 When the download is complete, RSLogix will open another confirmation dialog box. Click OK to switch the processor from Program mode to Run mode.

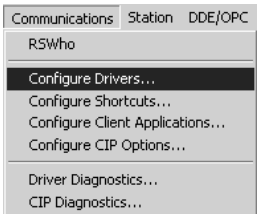


Note: If you receive an error message during these steps, refer to your RSLogix documentation to interpret and correct the error.

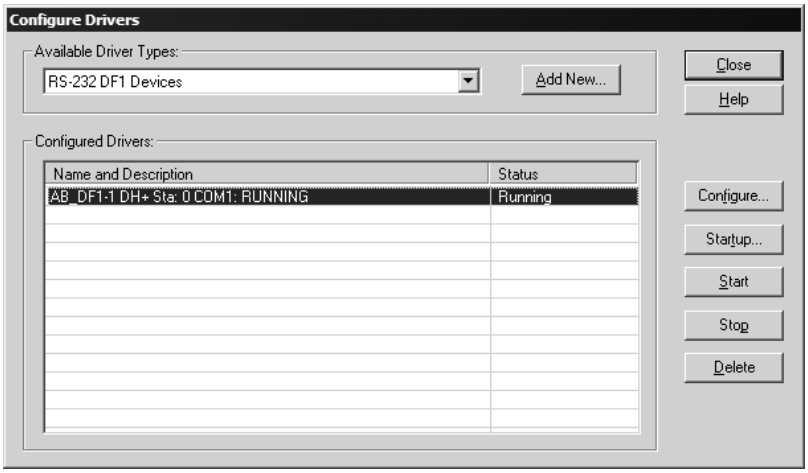
1.6.1 Configuring RSLinx

- ***If RSLogix is unable to establish communication with the processor, follow these steps:***

1 Open RSLinx.
2 Open the Communications menu, and choose Configure Drivers.

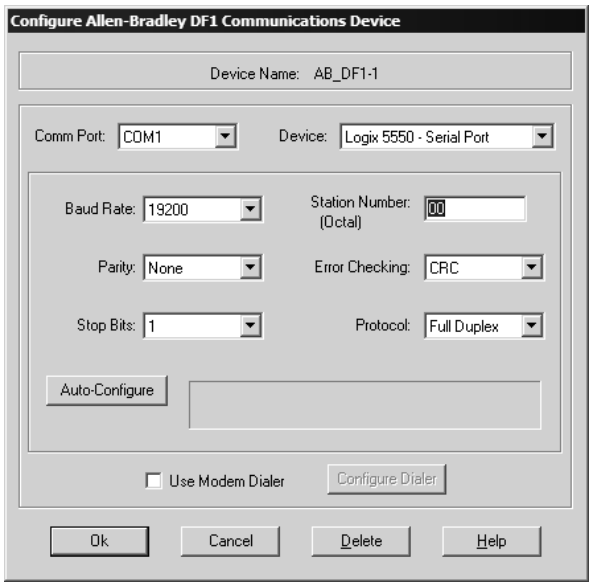


This action opens the Configure Drivers dialog box.



Note: If the list of configured drivers is blank, you must first choose and configure a driver from the Available Driver Types list. The recommended driver type to choose for serial communication with the processor is "RS-232 DF1 Devices".

3 Click to select the driver, and then click Configure. This action opens the Configure Allen-Bradley DF1 Communications Device dialog box.



4 Click the Auto-Configure button. RSLinx will attempt to configure your serial port to work with the selected driver.

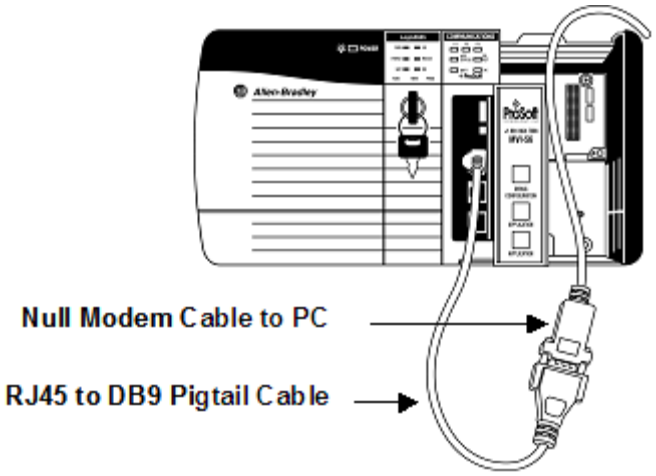
5 When you see the message "Auto Configuration Successful", click the OK button to dismiss the dialog box.

Note: If the auto-configuration procedure fails, verify that the cables are connected correctly between the processor and the serial port on your computer, and then try again. If you are still unable to auto-configure the port, refer to your RSLinx documentation for further troubleshooting steps.

1.7 Connect your PC to the Module

With the module securely mounted, connect your PC to the **Configuration/Debug** port using an RJ45-DB-9 Serial Adapter Cable and a Null Modem Cable.

1. Attach both cables as shown.
2. Insert the RJ45 cable connector into the Configuration/Debug port of the module.
3. Attach the other end to the serial port on your PC or laptop.



2 Installing and Configuring the Module

In This Chapter

This chapter describes how to install and configure the module to work with your application. The configuration process consists of the following steps.

1 Use RSLogix 5000 to identify the module to the processor and add the module to a project.

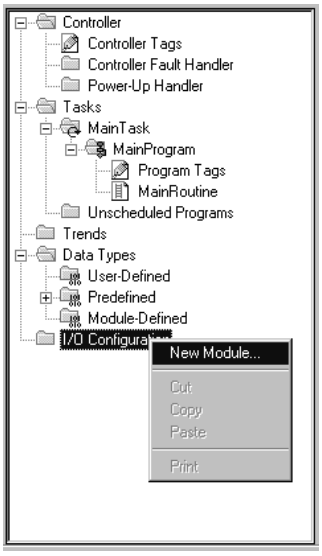
Note: The RSLogix 5000 software must be in "offline" mode to add the module to a project.

2 Modify the example ladder logic to meet the needs of your application, and copy the ladder logic to the processor. Example ladder logic files are provided on the CD-ROM.

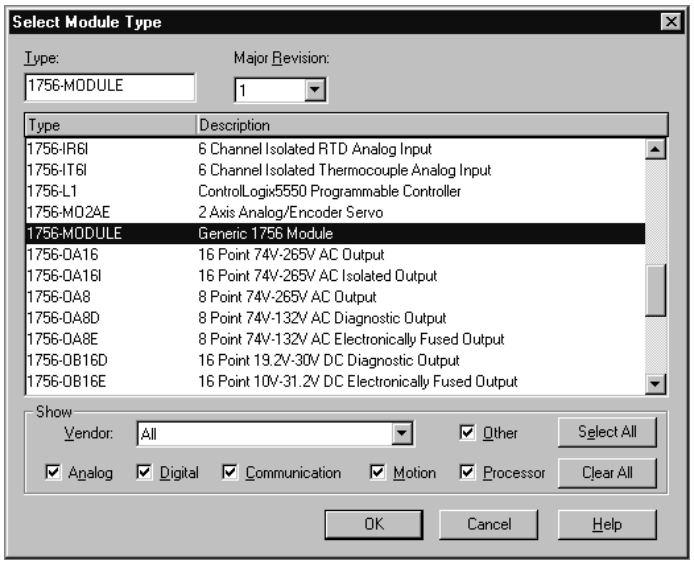
Note: If you are installing this module in an existing application, you can copy the necessary elements from the example ladder logic into your application.

The rest of this chapter describes these steps in more detail.

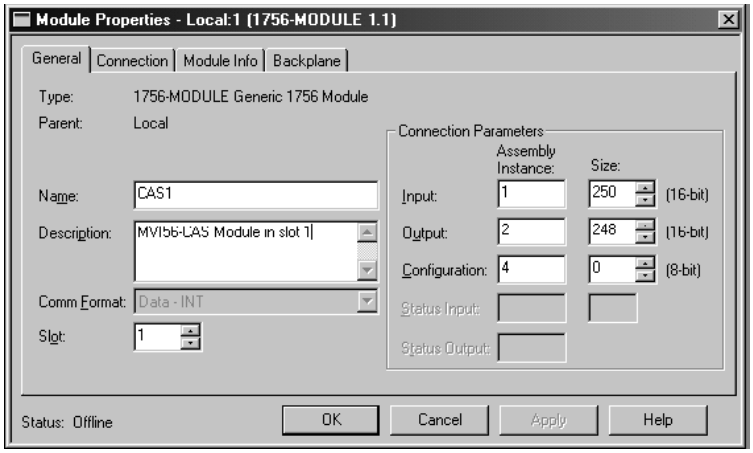
The first step in installing and configuring the module is to define the module to the system. Right-click the mouse button on the I/O Configuration option in the Controller Organization window to display a pop-up menu. Select the New Module… option from the I/O Configuration menu:



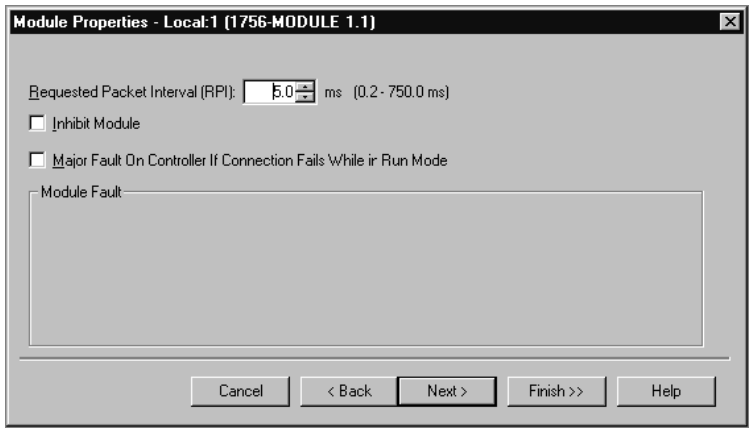
This action opens the following dialog box:



Select the 1756-Module (Generic 1756 Module) from the list and click OK. The following dialog box is displayed:

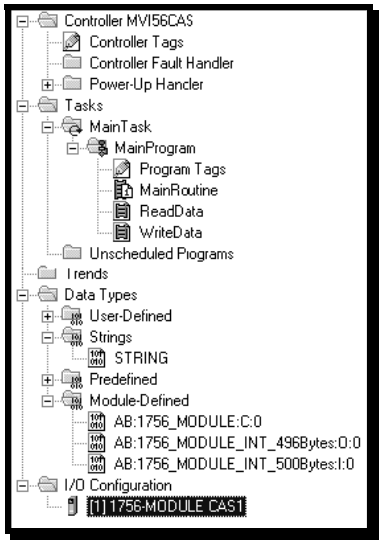


Fill in the dialog boxes as shown adjusting the Name, Description and Slot options for your application. You must select the **Comm Format** as **Data - INT** in the dialog box. Failure to set the **Assembly Instance** and **Size** values correctly will result in a module that will not communicate over the backplane of the ControlLogix rack. Click Next to display the next dialog box.

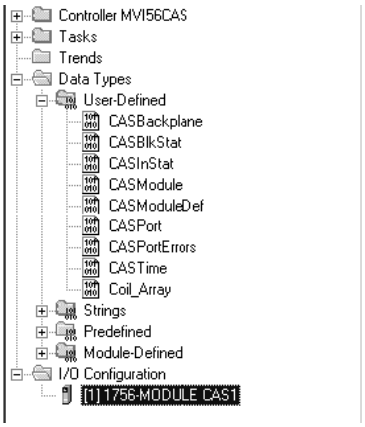


Select the Request Packet Interval value for scanning the I/O on the module. This value represents the minimum frequency the module will handle scheduled events. This value should not be set to less than 1 millisecond. Values between 1 and 10 milliseconds should work with most applications.

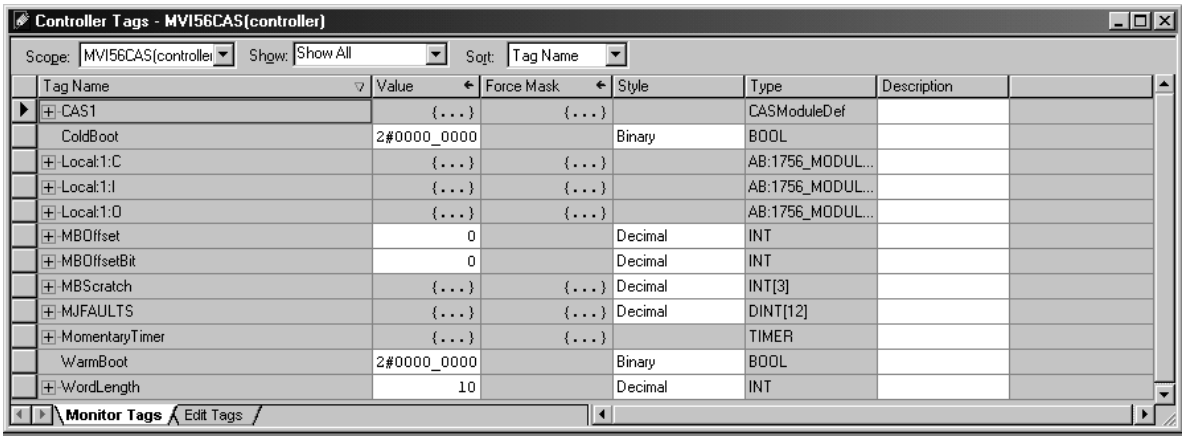
After completing the module setup, the Controller Organization window will display the module's presence. The data required for the module will be defined to the application, and objects will be allocated in the Controller Tags data area. An example of the Controller Organization window is shown below:



The next step in the module's setup is to define the User Defined Data Types to be used with the module. Copy these data types from the example ladder logic if you are not using the example. They will be defined if you are starting from the example ladder logic. The Controller Organization window should display the User Defined Data Types shown below:

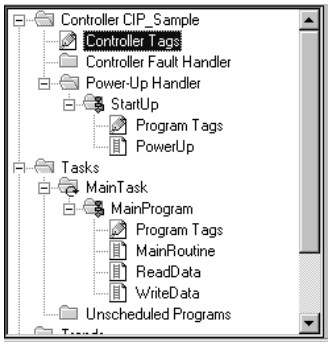


The next step in module setup is to define the data to be used to interface with the module and the ladder logic. Open the Controller Tags Edit Tags dialog box and enter the values shown in the following example. The MVI56-CAS module is defined in the example as CAS1. You can set the tag name to any valid tag name you desire. If you are using the example ladder logic, this step has already been performed.



At this point, take the time to fill in the configuration values in the CAS1 data table and adjust array sizes. Refer to the Module Data Object section of this document for information on configuring the module.

The last step in the module setup is to add the ladder logic. If you are using the example ladder logic, adjust the ladder to fit your application. If you are not using the ladder example, copy the ladder logic shown in the Controller Organization window below to your application.



The module is now set up and ready to be used with your application. Insert the module in the rack and attach the TCA serial communication cables. Download the new application to the controller and place the processor in run mode. If all the configuration parameters are set correctly and the module is attached to a TCA network, the module's Application LED (APP LED) should remain off and the backplane activity LED (BP ACT) should blink very rapidly. Refer to the **Diagnostics and Troubleshooting** section of this manual if you encounter errors. Attach a computer or terminal to Debug/Configuration port on the module and look at the status of the module using the Configuration/Debug Menu in the module.

2.1 Module Configuration

In order for the MVI56-CAS module to function, a minimum amount of configuration data must be transferred to the module. The following table provides an overview of the different types of configuration data that the module will require, depending on the operating modes to be supported.

Module Register Address	Functional Modes Affected	Name	Description
5000 to 5009	Data Transfer	General Module Configuration	This section of the configuration data contains the module configuration data that defines the data transfer between the module and the ControlLogix processor.
5010 to 5039 and 5040 to 5069	Master and Slave	Port Configuration	These sections define the characteristics of each of the TCA serial communication ports on the module. These parameters must be set correctly for proper module operation.

Refer to the Installing and Configuring the Module section for a description of the configuration of the module. The MVI56-CAS module must be configured at least once when the card is first powered, and any time thereafter when the parameters must be changed.

2.1.1 Power Up

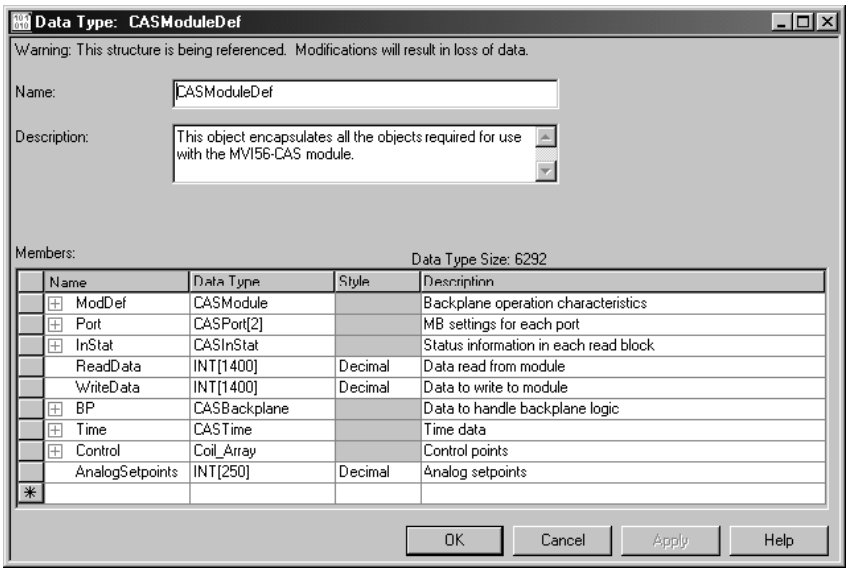
On power up, the module enters into a logical loop waiting to receive configuration data from the processor. Upon receipt, the module will begin execution of the command list if it is present.

2.1.2 Changing Parameters During Operation

A copy of the module's configuration data is mapped in the module's database as displayed in the table above. These values are initialized when the module first receives its configuration from the ControlLogix processor. Any node on the network can change this data. The module will not use this data until it is commanded. Ladder logic can be written to issue a Write Configuration command block (9997) to the module. A remote device can set a value of 9997 at address 6800 in the module to download the configuration to the processor. Alternatively, the configuration/debug port on the module can be used to issue the command directly to the module. All three of these methods will force the module to download the configuration to the ControlLogix processor. Ladder logic must exist in the processor to accept the blocks sent by the module. If everything is configured correctly, the module can receive its configuration from a remote device.

2.1.3 Module Data Object (CASModuleDef)

All data related to the MVI56-CAS is stored in a user defined data type. An instance of the data type is required before the module can be used. This is done by declaring a variable of the data type in the Controller Tags Edit Tags dialog box. The structure of the object is displayed in the following figure.

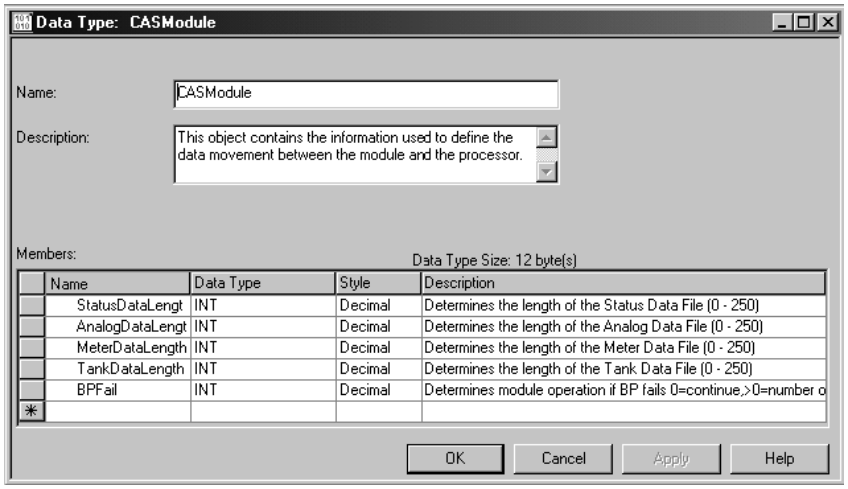


This object contains objects that define the configuration, user data, status and command control data related to the module. Each of these object types is discussed in the following topics of the document.

Configuration Objects

Configuration of the module is performed by filling in the values in the module object defined in the Controller Tags Edit Tags dialog. Each parameter required by the module has a defined location in the object. The tables and discussions below describe the parameters set in the dialog box. You can view these tables by opening the data type under the User Defined Data Type option in the Controller Organization window.

Data Transfer Parameters (CASModule)



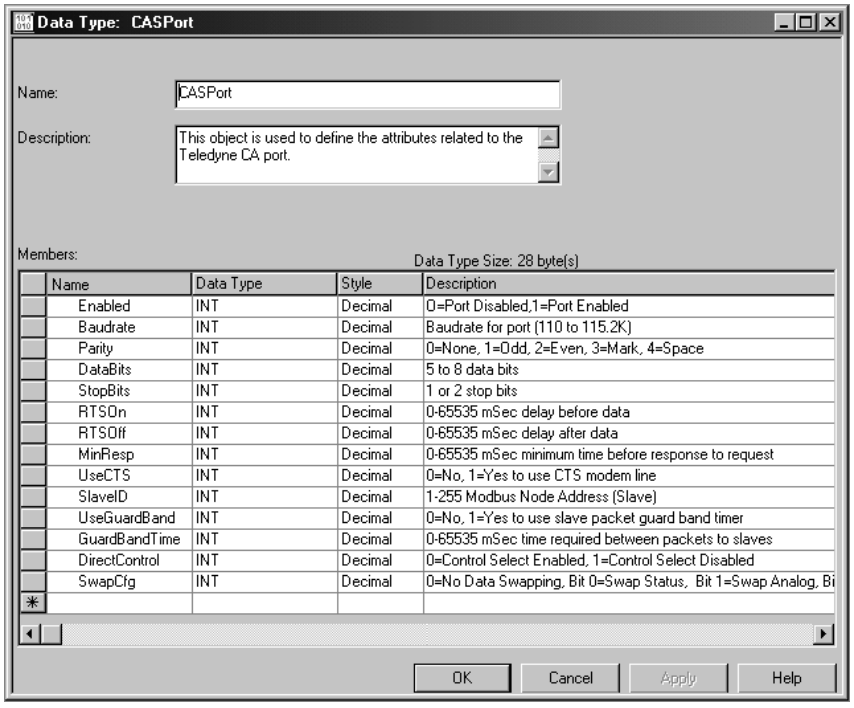
This object defines the parameters for data movement between the module and the processor. Values entered determine the ladder logic and data size required in the application. The ReadData and WriteData arrays must be sized to or larger than the count values entered. The ladder logic must process the number of blocks of data to be transferred. The number of blocks is computed as follows:

BlockCnt = INT(RegCnt/200) + if(MOD(RegCnt,200), 1,0)

If the register count is evenly divisible by 200, the number of blocks is easy to compute and the ladder is much simpler to write. If the number is not evenly divisible by 200, special handling of the last block of data must developed, as it must transfer less than 200 words. **It is recommended that the count values always be set to values evenly divisible by 200.**

The BPFail parameter determines if the module should continue communicating on the TCA network when the backplane transfer operation fails. A value of zero indicates that the module should continue communicating when the backplane is not operational. If the value is greater than zero, the backplane will be retried the entered number of times before a failure will be reported and communication will cease on the ports. When backplane communication is restored, the module will start communicating on the network. For example, if you enter a value of 10 for the parameter, the module will stop all TCA communications if 10 successive backplane errors are recognized. When a successful transfer is recognized, the module will resume communications on the network.

TCA Port Parameters (CASPort)

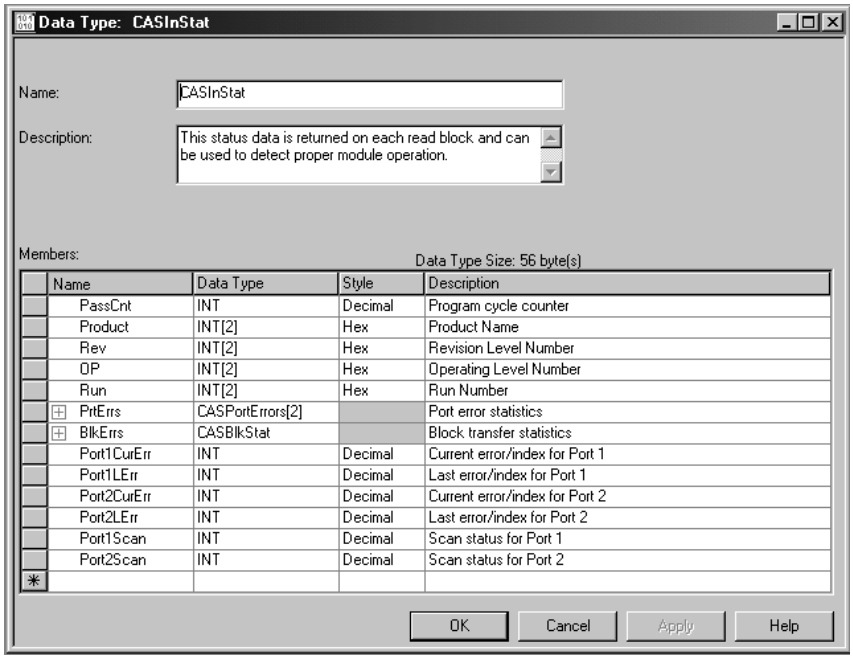


Name	Data Type	Style	Description
Enabled	INT	Decimal	0=Port Disabled,1=Port Enabled
Baudrate	INT	Decimal	Baudrate for port (110 to 115.2K)
Parity	INT	Decimal	0=None, 1=Odd, 2=Even, 3=Mark, 4=Space
DataBits	INT	Decimal	5 to 8 data bits
StopBits	INT	Decimal	1 or 2 stop bits
RTSOn	INT	Decimal	0-65535 mSec delay before data
RTSOff	INT	Decimal	0-65535 mSec delay after data
MinResp	INT	Decimal	0-65535 mSec minimum time before response to request
UseCTS	INT	Decimal	0=No, 1=Yes to use CTS modem line
SlaveID	INT	Decimal	1-255 Modbus Node Address (Slave)
UseGuardBand	INT	Decimal	0=No, 1=Yes to use slave packet guard band timer
GuardBandTime	INT	Decimal	0-65535 mSec time required between packets to slaves
DirectControl	INT	Decimal	0=Control Select Enabled, 1=Control Select Disabled
SwapCfg	INT	Decimal	0=No Data Swapping, Bit 0=Swap Status, Bit 1=Swap Analog, Bi

This object defines the parameters for the operation of each of the TCA ports on the module. Refer to the Reference chapter for the definition of each parameter.

Status Object (CASInStat)

This object views the status of the module. The **CASInStat** object shown below is updated each time a read block is received by the processor. Use this data to monitor the state of the module at a "real-time rate".



Refer to the Reference chapter for a complete listing of the data stored in this object.

2.1.4 User Data Objects

These objects hold data to be transferred between the processor and the MVI56-CAS module. The user data is the read and write data transferred between the processor and the module as "pages" of data up to 200 words long.

ReadData	INT[1400]	Decimal	Data read from module
WriteData	INT[1400]	Decimal	Data to write to module

The read data (**ReadData**) is an array set to match the value entered in the **ReadRegCnt** parameter of the **CASModule** object. For ease of use, this array should be dimensioned as an even increment of 200 words. This data is paged up to 200 words at a time from the module to the processor. The ReadData task is responsible for placing the data received into the proper position in the read data array. Use this data for status and control in the ladder logic of the processor.

The write data (**WriteData**) is an array set to match the value entered in the **WriteRegCnt** parameter of the **CASModule** object. For ease of use, this array should be dimensioned as even increments of 200 words. This data is paged up to 200 words at a time from the processor to the module. The WriteData task is responsible for placing the write data into the output image for transfer to the module. This data is passed from the processor to the module for status and control information for use in other nodes on the network

2.1.5 TCA Message Data

The module's program relies on the pass-through mode. Write messages sent to a slave port are passed directly through to the processor. It is the responsibility of the ladder logic to process the message received using this feature. Three data objects are required for this mode: a variable to hold the length of the message, a variable to hold the address, and a buffer to hold the message. This information is passed from the module to the processor using block identification codes of 9991, 9992, or 9994. Word two of this block contains the length of the message, word three contains the address and the message starts at word 5.

3 Ladder Logic

Ladder logic is required for application of the MVI56-CAS module. Tasks that must be handled by the ladder logic are module data transfer, special block handling and status data receipt. Additionally, a power-up handler may be needed to handle the initialization of the module's data and to clear any processor fault conditions.

The sample ladder logic, on the ProSoft Solutions CD-ROM, is extensively commented, to provide information on the purpose and function of each rung. For most applications, the sample ladder will work without modification.

4 Diagnostics and Troubleshooting

In This Chapter

The module provides information on diagnostics and troubleshooting in the following forms:

- Status data values are transferred from the module to the processor.
- Data contained in the module can be viewed through the Configuration/Debug port attached to a terminal emulator.
- LED status indicators on the front of the module provide information on the module's status.

4.1 Reading Status Data From the Module

The MVI56-CAS module returns a 35-word Status Data block that can be used to determine the module's operating status. This data is located in the module's database at registers 6000 to 6034 and at the location specified in the configuration. This data is transferred to the ControlLogix processor continuously with each read block. For a complete listing of the status data object, refer to the **Module Set Up** section.

4.1.1 The Configuration/Debug Menu

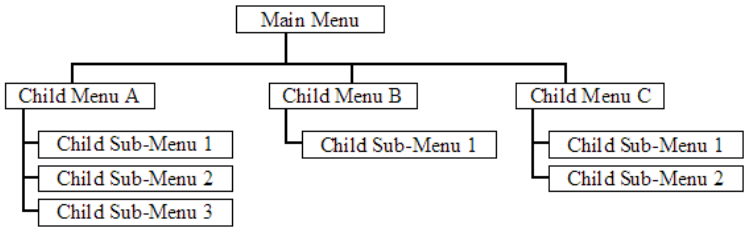
The Configuration and Debug menu for this module is arranged as a tree structure, with the Main Menu at the top of the tree, and one or more sub-menus for each menu command. The first menu you see when you connect to the module is the Main menu.

Because this is a text-based menu system, you enter commands by typing the command letter from your computer keyboard in the terminal application (for example, HyperTerminal). The module does not respond to mouse movements or clicks. The command executes as soon as you press the command letter — you do not need to press **[Enter]**. When you type a command letter, a new screen will be displayed in your terminal application.

Navigation

All of the sub-menus for this module contain commands to redisplay the menu or return to the previous menu. You can always return from a sub-menu to the next higher menu by pressing **[M]** on your keyboard.

The organization of the menu structure is represented in simplified form in the following illustration:



The remainder of this section shows you the menus available for this module, and briefly discusses the commands available to you.

Keystrokes

The keyboard commands on these menus are almost always non-case sensitive. You can enter most commands in lower case or capital letters.

The menus use a few special characters (**[?]**, **[-]**, **[+]**, **[@]**) that must be entered exactly as shown. Some of these characters will require you to use the **[Shift]**, **[Ctrl]** or **[Alt]** keys to enter them correctly. For example, on US English keyboards, enter the **[?]** command as **[Shift][/]**.

Also, take care to distinguish capital letter **[I]** from lower case letter **[l]** (L) and number **[1]**; likewise for capital letter **[O]** and number **[0]**. Although these characters look nearly the same on the screen, they perform different actions on the module.

4.1.2 Required Hardware

You can connect directly from your computer's serial port to the serial port on the module to view configuration information and perform maintenance.

ProSoft Technology recommends the following minimum hardware to connect your computer to the module:

- 80486 based processor (Pentium preferred)
- 1 megabyte of memory
- At least one serial communications port available
- A null modem serial cable.

4.1.3 Required Software

In order to send and receive data over the serial port (COM port) on your computer to the module, you must use a communication program (terminal emulator).

A simple communication program called HyperTerminal is pre-installed with recent versions of Microsoft Windows operating systems. If you are connecting from a machine running DOS, you must obtain and install a compatible communication program. The following table lists communication programs that have been tested by ProSoft Technology.

DOS	ProComm, as well as several other terminal emulation programs
Windows 3.1	Terminal
Windows 95/98	HyperTerminal
Windows NT/2000/XP	HyperTerminal

4.1.4 Using the Configuration/Debug Port

To connect to the module's Configuration/Debug port:

1. Connect your computer to the module's port using a null modem cable.
2. Start the communication program on your computer and configure the communication parameters with the following settings:

Baud Rate	57,600
Parity	None
Data Bits	8
Stop Bits	1
Software Handshaking	None

3. Open the connection. When you are connected, press the **[?]** key on your keyboard. If the system is set up properly, you will see a menu with the module name followed by a list of letters and the commands associated with them.

If there is no response from the module, follow these steps:

1. Verify that the null modem cable is connected properly between your computer's serial port and the module. A regular serial cable will not work.
2. Verify that RSLinx is not controlling the COM port. Refer to Disabling the RSLinx Driver for the Com Port on the PC (page 57).
3. Verify that your communication software is using the correct settings for baud rate, parity and handshaking.
4. On computers with more than one serial port, verify that your communication program is connected to the same port that is connected to the module.

If you are still not able to establish a connection, you can contact ProSoft Technology Technical Support for further assistance.

4.1.5 Main Menu

When you first connect to the module from your computer, your terminal screen will be blank. To activate the main menu, press the [?] key on your computer's keyboard. If the module is connected properly, the following menu will appear on your terminal screen:

```
Main Menu Selected

TELEDYNE CA SLAVE COMMUNICATION MODULE (MVI56-CAS) MENU
  ?=Display Menu
  A=Data Analyzer
  B=Block Transfer Statistics
  C=Module Configuration
  D=CAS Database View
  V=Version Information
  W=Warm Boot Module
  Y=Transfer Module Cfg to Processor
  Communication Status : 1=Port 1   2=Port 2
  Port Configuration   : 6=Port 1   7=Port 2

  Esc=Exit Program
```

Caution: Some of the commands available to you from this menu are designed for advanced debugging and system testing only, and can cause the module to stop communicating with the processor or with other devices, resulting in potential data loss or other failures. Only use these commands if you are specifically directed to do so by ProSoft Technology Technical Support staff. Some of these command keys are not listed on the menu, but are active nevertheless. Please be careful when pressing keys so that you do not accidentally execute an unwanted command.

Opening the Data Analyzer Menu

Press **[A]** to open the Data Analyzer Menu. Use this command to view all bytes of data transferred on each port. Both the transmitted and received data bytes are displayed. Refer to Data Analyzer for more information about this menu.

Important: When in analyzer mode, program execution will slow down. Only use this tool during a troubleshooting session. Before disconnecting from the Config/Debug port, please press **[S]** to stop the data analyzer, and then press **[M]** to return to the main menu. This action will allow the module to resume its normal high speed operating mode.

Viewing Block Transfer Statistics

Press **[B]** from the Main Menu to view the Block Transfer Statistics screen.

Use this command to display the configuration and statistics of the backplane data transfer operations between the module and the processor. The information on this screen can help determine if there are communication problems between the processor and the module.

Tip: To determine the number of blocks transferred each second, mark the numbers displayed at a specific time. Then some seconds later activate the command again. Subtract the previous numbers from the current numbers and divide by the quantity of seconds passed between the two readings.

Viewing Module Configuration

Press **[C]** to view the Module Configuration screen.

Use this command to display the current configuration and statistics for the module.

Opening the Database Menu

Press **[D]** to open the Database View menu. Use this menu command to view the current contents of the module's database.

Viewing Version Information

Press **[V]** to view Version information for the module.

Use this command to view the current version of the software for the module, as well as other important values. You may be asked to provide this information when calling for technical support on the product.

Values at the bottom of the display are important in determining module operation. The Program Scan Counter value is incremented each time a module's program cycle is complete.

Tip: Repeat this command at one-second intervals to determine the frequency of program execution.

Warm Booting the Module

Caution: Some of the commands available to you from this menu are designed for advanced debugging and system testing only, and can cause the module to stop communicating with the processor or with other devices, resulting in potential data loss or other failures. Only use these commands if you are specifically directed to do so by ProSoft Technology Technical Support staff. Some of these command keys are not listed on the menu, but are active nevertheless. Please be careful when pressing keys so that you do not accidentally execute an unwanted command.

Press **[W]** from the Main Menu to warm boot (restart) the module. This command will cause the program to exit and reload, refreshing configuration parameters that must be set on program initialization. Only use this command if you must force the module to re-boot.

Transferring Module Configuration to the Processor

Press **[Y]** to transfer the module's configuration data to the processor. Ladder logic is required in the processor to receive and implement the updated configuration. You will be prompted to confirm the transfer.

If the operation is not successful, an error code will be returned.

Code	Description
0	Transfer successful
-1	Error transferring module configuration data (block -9000)
-2	Error transferring device definition data (blocks -9100 to -9103)
-3	Error transferring master command list data (blocks -6000 to -6007)

After successful data transfer, the module will perform a warm-boot operation to read in the new data.

Viewing Port Communication Status

Press **[1]** or **[2]** from the Main Menu to view the port communication status for Ports 1 and 2.

Use this command to view communication status and statistics for the selected port. This information can be informative when troubleshooting communication problems.

Viewing Port Configuration

Press **[6]** or **[7]** from the Main Menu to view configuration information for ports 1 and 2.

Use this command to display detailed configuration information for the selected port.

Exiting the Program

Caution: Some of the commands available to you from this menu are designed for advanced debugging and system testing only, and can cause the module to stop communicating with the processor or with other devices, resulting in potential data loss or other failures. Only use these commands if you are specifically directed to do so by ProSoft Technology Technical Support staff. Some of these command keys are not listed on the menu, but are active nevertheless. Please be careful when pressing keys so that you do not accidentally execute an unwanted command.

Press **[Esc]** to restart the module and force all drivers to be loaded. The module will use the configuration stored in the module's Flash ROM to configure the module.

4.1.6 Data Analyzer

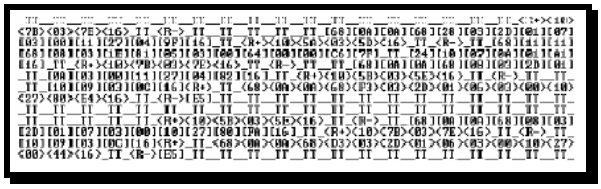
The data analyzer mode allows you to view all bytes of data transferred on each port. Both the transmitted and received data bytes are displayed. Use of this feature is limited without a thorough understanding of the protocol.

Note: The Port selection commands on the Data Analyzer menu differs very slightly in different modules, but the functionality is basically the same. Use the illustration above as a general guide only. Refer to the actual data analyzer menu on your module for the specific port commands to use.

Important: When in analyzer mode, program execution will slow down. Only use this tool during a troubleshooting session. Before disconnecting from the Config/Debug port, please press **[S]** to stop the data analyzer, and then press **[M]** to return to the main menu. This action will allow the module to resume its normal high speed operating mode.

Analyzing Data for the first application port

Press **[1]** to display I/O data for the first application port in the Data Analyzer. The following illustration shows an example of the Data Analyzer output.



Analyzing Data for the second application port

Press **[2]** to display I/O data for the second application port in the Data Analyzer.

Displaying Timing Marks in the Data Analyzer

You can display timing marks for a variety of intervals in the data analyzer screen. These timing marks can help you determine communication-timing characteristics.

Key	Interval
[5]	1 milliseconds ticks
[6]	5 milliseconds ticks
[7]	10 milliseconds ticks
[8]	50 milliseconds ticks
[9]	100 milliseconds ticks
[0]	Turn off timing marks

Removing Timing Marks in the Data Analyzer

Press **[0]** to turn off timing marks in the Data Analyzer screen.

Viewing Data in Hexadecimal Format

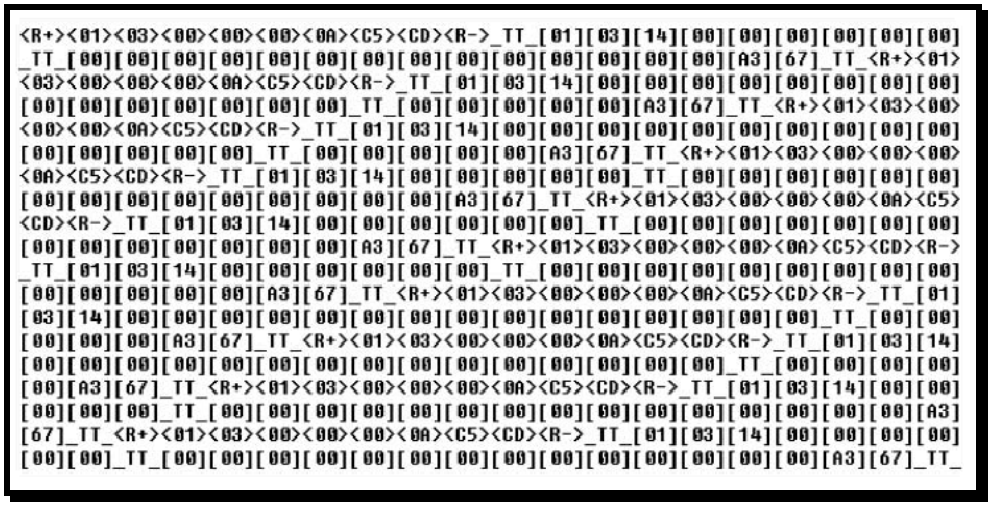
Press **[H]** to display the data on the current page in hexadecimal format.

Viewing Data in ASCII (Text) Format

Press **[A]** to display the data on the current page in ASCII format. This is useful for regions of the database that contain ASCII data.

Starting the Data Analyzer

Press **[B]** to start the data analyzer. After the key is pressed, all data transmitted and received on the currently selected port will be displayed. An example display is shown below:



The Data Analyzer displays the following special characters:

Character	Definition
[ ]	Data enclosed in these characters represent data received on the port.
< >	Data enclosed in these characters represent data transmitted on the port.
<R+>	These characters are inserted when the RTS line is driven high on the port.
<R->	These characters are inserted when the RTS line is dropped low on the port.
<CS>	These characters are displayed when the CTS line is recognized high.
_TT_	These characters are displayed when the timing mark interval has been reached. This parameter is user defined.

Stopping the Data Analyzer

Press **[S]** to stop the data analyzer. Use this option to freeze the display so the data can be analyzed. To restart the analyzer, press **[B]**.

Important: When in analyzer mode, program execution will slow down. Only use this tool during a troubleshooting session. Before disconnecting from the Config/Debug port, please press **[S]** to stop the data analyzer, and then press **[M]** to return to the main menu. This action will allow the module to resume its normal high speed operating mode.

Returning to the Main Menu

Press **[M]** to return to the Main Menu.

4.1.7 Data Analyzer Tips

From the main menu, press **[A]** for the "Data Analyzer". You should see the following text appear on the screen:

```
Data Analyzer Mode Selected
```

After the "Data Analyzer" mode has been selected, press **[?]** to view the Data Analyzer menu. You will see the following menu:

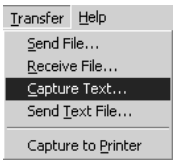
```
DATA ANALYZER VIEW MENU
 ?=Display Menu
 1=Select Port 1
 2=Select Port 2
 5=1 mSec Ticks
 6=5 mSec Ticks
 7=10 mSec Ticks
 8=50 mSec Ticks
 9=100 mSec Ticks
 0=No mSec Ticks
 H=Hex Format
 A=ASCII Format
 B=Start
 S=Stop
 M=Main Menu

 Port = 1, Format=HEX, Tick=10
```

From this menu, you can select the "Port", the "format", and the "ticks" that you can display the data in.

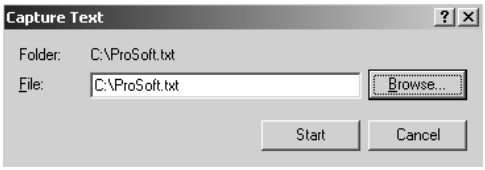
For most applications, HEX is the best format to view the data, and this does include ASCII based messages (because some characters will not display on HyperTerminal and by capturing the data in HEX, we can figure out what the corresponding ASCII characters are supposed to be).

The Tick value is a timing mark. The module will print a _TT for every xx milliseconds of no data on the line. Usually 10milliseconds is the best value to start with.

After you have selected the Port, Format, and Tick, we are now ready to start a capture of this data. The easiest way to do so is to go up to the top of you HyperTerminal window, and do a **Transfer / Capture Text** as shown below:



After selecting the above option, the following window will appear:



Next name the file, and select a directory to store the file in. In this example, we are creating a file ProSoft.txt and storing this file on our root C: drive. After you have done this, press the Start button.

Now you have everything that shows up on the HyperTerminal screen being logged to a file called ProSoft.txt. This is the file that you will then be able to email to ProSoft Technical Support to assist with issues on the communications network.

To begin the display of the communications data, you will then want to press 'B' to tell the module to start printing the communications traffic out on the debug port of the module. After you have pressed 'B', you should see something like the following:

```
[03][00][04][00][05][00][06][00][07][00][08][00][09][FB][B7]_TT__TT_<R+><01><02>
<00><00><00><0A><F8><0D><R->_TT__TT__TT_[01][02][02][00][00][B9][B8]_TT__TT_<R+>
<01><03><00><00><00><0A><C5><CD><R->_TT__TT_[01][03][14][00][00][00][01][00]_TT_
[02][00][03][00][04][00][05][00][06][00][07][00][08][00][09][CD][51]_TT__TT_<R+>
<01><01><00><00><00><A0><3C><72><R->_TT__TT_[01][01][14][00][00][01][00][02]_TT_
[00][03][00][04][00][05][00][06][00][07][00][08][00][09][00][B7][52]_TT__TT_<R+>
<01><04><00><00><00><0A><70><0D><R->_TT__TT_[01][04][14][00][00][00][01][00]_TT_
[02][00][03][00][04][00][05][00][06][00][07][00][08][00][09][FB][B7]_TT__TT_<R+>
<01><02><00><00><00><0A><F8><0D><R->_TT__TT__TT_[01][02][02][00][00][B9][B8]_TT_
_TT_<R+><01><03><00><00><00><0A><C5><CD><R->_TT__TT_[01][03][14][00][00][00][01]
[00]_TT_[02][00][03][00][04][00][05][00][06][00][07][00][08][00][09][CD][51]_TT_
_TT_<R+><01><01><00><00><00><A0><3C><72><R->_TT__TT__TT_[01][01][14][00][00][01]
[00][02]_TT_[00][03][00][04][00][05][00][06][00][07][00][08][00][09][00][B7][52]
_TT__TT_<R+><01><04><00><00><00><0A><70><0D><R->_TT__TT_[01][04][14][00][00][00]
[01][00]_TT_[02][00][03][00][04][00][05][00][06][00][07][00][08][00][09][FB][B7]
_TT__TT_<R+><01><02><00><00><00><0A><F8><0D><R->_TT__TT_[01][02][02][00][00][B9]
[B8]_TT__TT_<R+><01><03><00><00><00><0A><C5><CD><R->_TT__TT_[01][03][14][00][00]
[00][01][00]_TT_[02][00][03][00][04][00][05][00][06][00][07][00][08][00][09][CD]
[51]_TT__TT_<R+><01><01><00><00><00><A0><3C><72><R->_TT__TT__TT_[01][01][14][00]
[00][01][00][02]_TT_[00][03][00][04][00][05][00][06][00][07][00][08][00][09][00]
[B7][52]_TT__TT_<R+><01><04><00><00><00><0A><70><0D><R->_TT__TT_[01][04][14][00]
[00][00][01][00]_TT_[02][00][03][00][04][00][05][00][06][00][07][00][08][00][09]
[FB][B7]_TT__TT_<R+><01><02><00><00><00><0A><F8><0D><R->_TT__TT__TT_[01][02][02]
[00][00][B9][B8]_TT__TT_<R+><01><03><00><00><00><0A><C5><CD><R->_TT__TT_
```

The <R+> means that the module is transitioning the communications line to a transmit state.

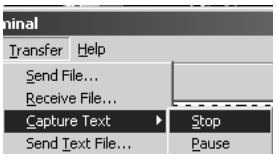
All characters shown in <> brackets are characters being sent out by the module.

The <R-> shows when the module is done transmitting data, and is now ready to receive information back.

And finally, all characters shown in the [ ] brackets is information being received from another device by the module.

After taking a minute or two of traffic capture, you will now want to stop the "Data Analyzer". To do so, press the 'S' key, and you will then see the scrolling of the data stop.

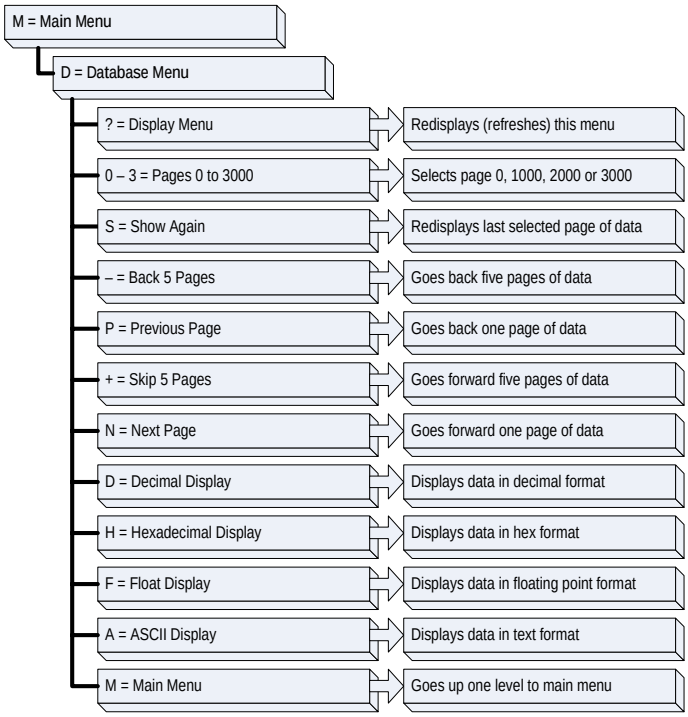
When you have captured the data you want to save, open the Transfer menu and choose Capture Text. On the secondary menu, choose Stop.



You have now captured, and saved the file to your PC. This file can now be used in analyzing the communications traffic on the line, and assist in determining communication errors.

4.1.8 Database View Menu

Press **[D]** from the Main Menu to open the Database View menu. Use this menu command to view the current contents of the module's database. Press **[?]** to view a list of commands available on this menu.



Viewing Register Pages

To view sets of register pages, use the keys described below:

Command	Description
[0]	Display registers 0 to 99
[1]	Display registers 1000 to 1099
[2]	Display registers 2000 to 2099

And so on. The total number of register pages available to view depends on your module's configuration.

Displaying the Current Page of Registers Again

```
DATABASE DISPLAY 0 TO 99 (DECIMAL)
   100   101   102     4     5     6     7     8     9    10
    11    12    13    14    15    16     0     0     0     0
     0     0     0     0     0     0     0     0     0     0
     0     0     0     0     0     0     0     0     0     0
     0     0     0     0     0     0     0     0     0     0
     0     0     0     0     0     0     0     0     0     0
     0     0     0     0     0     0     0     0     0     0
     0     0     0     0     0     0     0     0     0     0
     0     0     0     0     0     0     0     0     0     0
     0     0     0     0     0     0     0     0     0     0
```

This screen displays the current page of 100 registers in the database.

Moving Back Through 5 Pages of Registers

Press **[-]** from the Database View menu to skip back to the previous 500 registers of data.

Viewing the Previous 100 Registers of Data

Press **[P]** from the Database View menu to display the previous 100 registers of data.

Skipping 500 Registers of Data

Hold down **[Shift]** and press **[=]** to skip forward to the next 500 registers of data.

Viewing the Next 100 Registers of Data

Press **[N]** from the Database View menu to select and display the next 100 registers of data.

Viewing Data in Decimal Format

Press **[D]** to display the data on the current page in decimal format.

Viewing Data in Hexadecimal Format

Press **[H]** to display the data on the current page in hexadecimal format.

Viewing Data in Floating Point Format

Press **[F]** from the Database View menu. Use this command to display the data on the current page in floating point format. The program assumes that the values are aligned on even register boundaries. If floating-point values are not aligned as such, they are not displayed properly.

Viewing Data in ASCII (Text) Format

Press **[A]** to display the data on the current page in ASCII format. This is useful for regions of the database that contain ASCII data.

Returning to the Main Menu

Press **[M]** to return to the Main Menu.

4.2 LED Status Indicators

The LEDs indicate the module's operating status as follows:

ProSoft Module	Color	Status	Indication
CFG	Green	On	Data is being transferred between the module and a remote terminal using the Configuration/Debug port.
		Off	No data is being transferred on the Configuration/Debug port.
P1	Green	On	Data is being transferred between the module and the TCA network on its TCA Port 1.
		Off	No data is being transferred on the port.
P2	Green	On	Data is being transferred between the module and the TCA network on its TCA Port 2.
		Off	No data is being transferred on the port.
APP	Amber	On	The MVI56-CAS is working normally.
		Off	The MVI56-CAS module program has recognized a communication error on one of its TCA ports.
BP ACT	Amber	On	The LED is on when the module is performing a write operation on the backplane.
		Off	The LED is off when the module is performing a read operation on the backplane. Under normal operation, the LED should blink rapidly on and off.
OK	Red/ Green	Off	The card is not receiving any power and is not securely plugged into the rack.
		Green	The module is operating normally.
		Red	The program has detected an error or is being configured. If the LED remains red for over 10 seconds, the program has probably halted. Remove the card from the rack and re-insert the card to restart the module's program.
BAT	Red	Off	The battery voltage is OK and functioning.
		On	The battery voltage is low or battery is not present. Allow battery to charge by keeping module plugged into rack for 24 hours. If BAT LED still does not go off, contact ProSoft Technology, as this is not a user serviceable item.

During module configuration, the OK LED will be red and the APP and BP ACT LEDs will be on. If the LEDs are latched in this mode for a long period of time, look at the configuration error words in the configuration request block. The structure of the block is as shown below.

Offset	Description	Length
0	Reserved	1
1	9000	1
2	Module Configuration Errors	1
3	Port 1 Configuration Errors	1
4	Port 2 Configuration Errors	1
5 to 248	Spare	244
249	-2 or -3	1

If the APP, BP ACT and OK LEDs blink at a rate of every one-second, this indicates a serious problem with the module. Call Prosoft Technology support to arrange for repairs.

4.2.1 Clearing a Fault Condition

Typically, if the OK LED on the front of the module turns red for more than ten seconds, a hardware problem has been detected in the module, or the program has exited.

To clear the condition, follow these steps:

1. Turn off power to the rack
2. Remove the card from the rack
3. Verify that all jumpers are set correctly
4. If the module requires a Compact Flash card, verify that the card is installed correctly
5. Re-insert the card in the rack and turn the power back on
6. Verify the configuration data being transferred to the module from the ControlLogix processor.

If the module's OK LED does not turn green, verify that the module is inserted completely into the rack. If this does not cure the problem, contact ProSoft Technology Support.

4.2.2 Troubleshooting

Use the following troubleshooting steps if you encounter problems when the module is powered up. If these steps do not resolve your problem, please contact ProSoft Technology Technical Support.

Processor Errors

Problem Description	Steps to take
Processor Fault	Verify that the module is plugged into the slot that has been configured for the module. Verify that the slot in the rack configuration has been set up correctly in the ladder logic.
Processor I/O LED flashes	This indicates a problem with backplane communications. Verify that all modules in the rack are configured in the ladder logic.

Module Errors

Problem Description	Steps to take
BP ACT LED remains off or blinks slowly	This indicates that backplane transfer operations are failing. Connect to the module's Configuration/Debug port to check this. To establish backplane communications, verify the following items: ▪ The processor is in Run mode. ▪ The backplane driver is loaded in the module. ▪ The module is configured for read and write block data transfer. ▪ The ladder logic handles all read and write block situations. ▪ The module is configured in the processor.
OK LED remains red	The program has halted or a critical error has occurred. Connect to the Configuration/Debug port to see if the module is running. If the program has halted, turn off power to the rack, remove the card from the rack and re-insert the card in the rack, and then restore power to the rack.

5 Reference

In This Chapter

5.1 Product Specifications

The MVI56 Teledyne CA Slave Module allows ControlLogix I/O compatible processors to interface easily with TCA protocol compatible devices.

5.1.1 Features and Benefits

The MVI56-CAS module acts as an input/output module between the TCA network and the ControlLogix backplane. The data transfer from the ControlLogix processor is asynchronous from the actions on the TCA network. A 5000-word register space in the module exchanges data between the processor and the TCA network.

5.1.2 General Specifications

- Single Slot - 1756 backplane compatible
- Local or remote rack
- The module is recognized as an Input/Output module and has access to processor memory for data transfer between processor and module
- Ladder Logic is used for data transfer between module and processor.
- Configuration data obtained through user-defined ladder. Sample ladder file included

5.1.3 Hardware Specifications

Specification	Description
Backplane Current Load	800 mA @ 5 V DC; 3mA @ 24V DC
Operating Temperature	0 to 60°C (32 to 140°F)
Storage Temperature	-40 to 85°C (-40 to 185°F)
Shock	30g Operational
	50g non-operational
	Vibration: 5 g from 10 to 150 Hz
Relative Humidity	5% to 95% (non-condensing)
LED Indicators	Module Status
	Backplane Transfer Status
	Application Status
	Serial Activity
Debug/Configuration port (CFG)	
CFG Port (CFG)	RJ45 (DB-9M with supplied cable)
	RS-232 only
Application ports (PRT1 & PRT2)	
Full hardware handshaking control, providing radio, modem and multi-drop support	
Software configurable communication parameters	Baud rate: 110 to 115,200 baud, depending on protocol
	RS-232 and 422
	Parity: none, odd or even
	Data bits: 5, 6, 7, or 8
	Stop bits: 1 or 2
	RTS on/off delay: 0 to 65535 milliseconds
App Ports (P1,P2) (Serial modules)	RJ45 (DB-9M with supplied cable)
	RS-232 handshaking configurable
	500V Optical isolation from backplane
Shipped with Unit	RJ45 to DB-9M cables for each port
	6-foot RS-232 configuration cable

5.1.4 Functional Specifications

- Function codes
  - 1: Read output data
  - 2: Read analog data
  - 3: Read meter data
  - 10: Read frozen analog values
  - 11: Read frozen meter data
  - 16: Read all data
  - 20: Read tank data from RTU
  - 30: Control momentary select
  - 31: Control momentary operate
  - 32: Control continuous select
  - 33: Control continuous operate

- 34: Setpoint select
- 35: Setpoint operate
- 43: Freeze meters
- 44: Freeze analogs
- 45: Freeze meters/analogs
- 46/6: Special freeze
- 50: Exception data request

- Supports broadcast commands from master
- Register addressing
  - Status: Up to 250 words
  - Analog: Up to 250 words
  - Meter: Up to 125 values (2 words per value)
  - Tank: Up to 250 words
- Supports write commands from host
- Operates in both direct or indirect modes
- Setpoint select and operate
  - Control point continuous and momentary

5.2 Functional Overview

This section provides an overview of how the MVI56-CAS module transfers data using the CAS protocol. You should understand the important concepts in this chapter before you begin installing and configuring the module.

5.2.1 General Concepts

The following topics describe several concepts that are important for understanding the operation of the MVI56-CAS module.

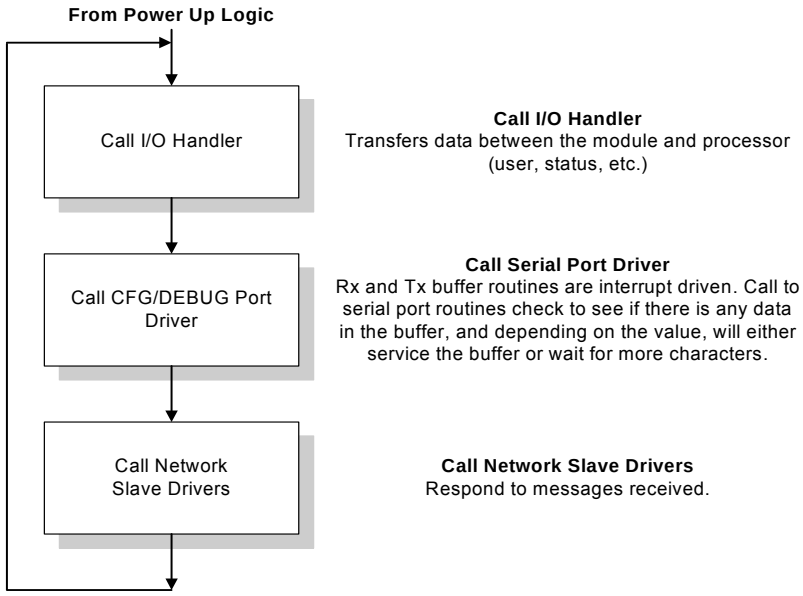
On power up the module begins performing the following logical functions:

1 Initialize hardware components
2 Initialize ControlLogix backplane driver
3 Test and Clear all RAM
4 Initialize the serial communication ports
5 Wait for Module Configuration from ControlLogix processor
6 Initialize Module Register space
7 Enable Slave Driver on selected ports

After the module has received the Module Configuration Block from the processor, the module will begin communicating with other nodes on the network, depending on the configuration.

Main Logic Loop

Upon completing the power up configuration process, the module enters an infinite loop that performs the functions shown in the following diagram.



ControlLogix Processor Not in Run

Whenever the module detects that the processor has gone out of the Run mode (that is, Fault or PGM), the TCA ports can be shut down as prescribed in the user configuration. When the processor is returned to a running state, the module will resume communications on the network.

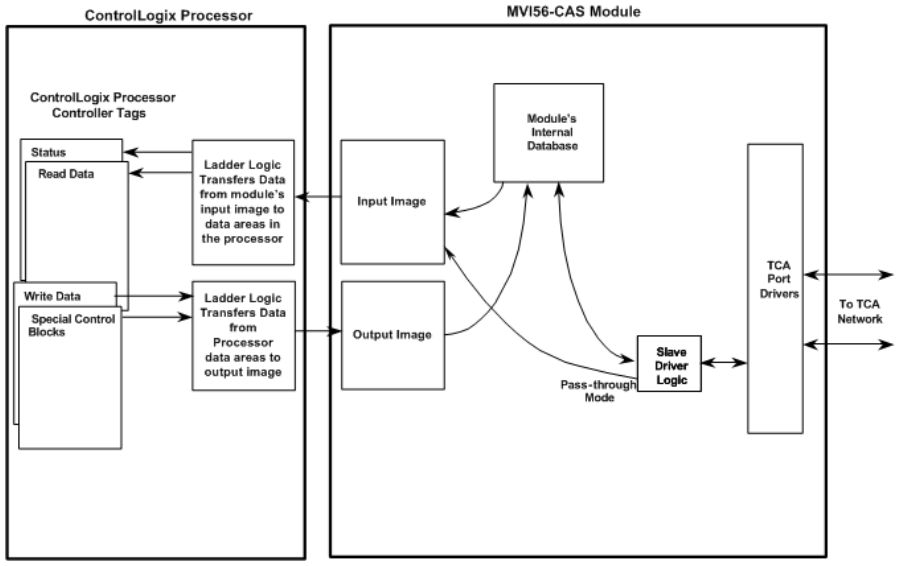
Backplane Data Transfer

The MVI56-CAS module communicates directly over the ControlLogix backplane. Data is paged between the module and the ControlLogix processor across the backplane using the module's input and output images. The update frequency of the images is determined by the scheduled scan rate defined by the user for the module and the communication load on the module. Typical updates are in the range of 2 to 10 milliseconds.

This bi-directional transference of data is accomplished by the module filling in data in the module's input image to send to the processor. Data in the input image is placed in the Controller Tags in the processor by the ladder logic. The input image for the module is set to 250 words. This large data area permits fast throughput of data between the module and the processor.

The processor inserts data to the module's output image to transfer to the module. The module's program extracts the data and places it in the module's internal database. The output image for the module is set to 248 words. This large data area permits fast throughput of data from the processor to the module.

The following illustration shows the data transfer method used to move data between the ControlLogix processor, the MVI56-CAS module and the TCA network.



As shown in the diagram above, all data transferred between the module and the processor over the backplane is through the input and output images. Ladder logic must be written in the ControlLogix processor to interface the input and output image data with data defined in the Controller Tags. All data used by the module is stored in its internal database. This database is defined as a data table with addresses from 0 to 6999. The following illustration shows the layout of the database:

Module's Internal Database Structure

5000 registers for user data

2000 words of configuration and status data

Register Data	0
	4999
Status and Config	5000
	6999

Data contained in this database is paged through the input and output images by coordination of the ControlLogix ladder logic and the MVI56-CAS module's program. Up to 248 words of data can be transferred from the module to the processor at a time. Up to 247 words of data can be transferred from the processor to the module. Each image has a defined structure depending on the data content and the function of the data transfer as defined below.

5.2.2 Normal Data Transfer

Normal data transfer includes the paging of the user data found in the module's internal database in registers 0 to 4999 and the status data. These data are transferred through read (input image) and write (output image) blocks. Refer to the **Module Set Up** section for a description of the data objects used with the blocks and the ladder logic required. The structure and function of each block is discussed below.

Read Block

These blocks of data transfer information from the module to the ControlLogix processor. The structure of the input image used to transfer this data is shown in the following table:

Offset	Description	Length
0	Reserved	1
1	Write Block ID	1
2 to 201	Read Data	200
202	Program Scan Counter	1
203 to 204	Product Code	2
205 to 206	Product Version	2
207 to 208	Operating System	2
209 to 210	Run Number	2
211 to 217	Port 1 Error Status	7
218 to 224	Port 2 Error Status	7
225 to 230	Data Transfer Status	6
231	Port 1 Current Error/Index	1
232	Port 1 Last Error/Index	1
233	Port 2 Current Error/Index	1
234	Port 2 Last Error/Index	1
235	Port 1 Status Scan	1
236	Port 2 Status Scan	1
237 to 248	Spare	12
249	Read Block ID	1

The Read Block ID is an index value used to determine the location of where the data will be placed in the ControlLogix processor controller tag array of module read data. Each transfer can move up to 200 words (block offsets 2 to 201) of data. In addition to moving user data, the block also contains status data for the module. This last set of data is transferred with each new block of data and is used for high-speed data movement.

The Write Block ID associated with the block requests data from the ControlLogix processor. Under normal, program operation, the module sequentially sends read blocks and requests write blocks. For example, if three read and two write blocks are used with the application, the sequence will be as follows:

R1W1→R2W2→R3W1→R1W2→R2W1→R3W2→R1W1→

This sequence will continue until interrupted by other write block numbers sent by the controller or by a command request from a node on the TCA network or operator control through the module's Configuration/Debug port.

Write Block

These blocks of data transfer information from the ControlLogix processor to the module. The structure of the output image used to transfer this data is shown in the following table:

Offset	Description	Length
0	Write Block ID	1
1 to 200	Write Data	200
201 to 206	Time Data	6
207 to 247	Spare	41

The Write Block ID is an index value used to determine the location in the module's database where the data will be placed. Each transfer can move up to 200 words (block offsets 1 to 200) of data.

5.2.3 Configuration Data Transfer

When the module performs a restart operation, it will request configuration information from the ControlLogix processor. This data is transferred to the module in specially formatted write blocks (output image). The module will poll for each block by setting the required write block number in a read block (input image). Refer to the **Module Set Up** section for a description of the data objects used with the blocks and the ladder logic required. The format of the blocks for configuration is given in the following topics.

Module Configuration Data

This block sends general configuration information from the processor to the module. The data is transferred in a block with an identification code of 9000. The structure of the block is displayed in the following table:

Offset	Description	Length
0	9000	1
1 to 5	Backplane Setup	5
6 to 19	Port 1 Configuration	14
20 to 33	Port 2 Configuration	14
34 to 247	Spare	214

The read block used to request the configuration has the following structure:

Offset	Description	Length
0	Reserved	1
1	9000	1
2	Module Configuration Errors	1
3	Port 1 Configuration Errors	1
4	Port 2 Configuration Errors	1
5 to 248	Spare	244
249	-2 or -3	1

If there are any errors in the configuration, the bit associated with the error will be set in one of the three configuration error words. The error must be corrected before the module starts operating.

5.2.4 Write Configuration

This block is sent from the ControlLogix processor to the module to force the module to write its current configuration back to the processor. This function is used when the module's configuration has been altered remotely using database write operations. The write block contains a value of -9000 in the first word. The module will respond with blocks containing the module configuration data. Ladder logic must handle the receipt of these blocks. The blocks transferred from the module are as follows:

Block -9000, General Configuration Data:

Offset	Description	Length
0	Reserved	1
1	-9000	1
2 to 6	Backplane Setup	5
7 to 20	Port 1 Configuration	14
21 to 34	Port 2 Configuration	14
35 to 248	Spare	214
249	-9000	1

Warm Boot

This block is sent from the ControlLogix processor to the module (output image) when the module is required to perform a warm-boot (software reset) operation. This block is commonly sent to the module any time configuration data modifications are made in the controller tags data area. This will force the module to read the new configuration information and to restart. The structure of the control block is shown below:

Offset	Description	Length
0	9998	1
1 to 247	Spare	247

Cold Boot

This block is sent from the ControlLogix processor to the module (output image) when the module is required to perform the cold boot (hardware reset) operation. This block is sent to the module when a hardware problem is detected by the ladder logic that requires a hardware reset. The structure of the control block is shown in the following table:

Offset	Description	Length
0	9999	1
1 to 247	Spare	247

5.2.5 Pass-Through Control Blocks

The module will pass blocks with identification codes of 9991, 9992 and 9994 to the processor for each received write command. Ladder logic must handle the receipt of all TCA write functions to the processor and to respond as expected to commands issued by the remote TCA device. The structure of the pass-through control block is shown in the following table:

Offset	Description	Length
0	0	1
1	9991, 9992, or 9994	1
2	Number of bytes in message	1
3	Write Address 1	1
4	Write Address 2	1
5 to 248	Message Received	244
249	9991, 9992, or 9994	1

The ladder logic will be responsible for parsing and copying the received message and performing the proper control operation as expected by the master device. The processor must then respond to the pass-through control block with a write block with the following format.

Offset	Description	Length
0	9991, 9992, or 9994	1
1 to 247	Spare	247

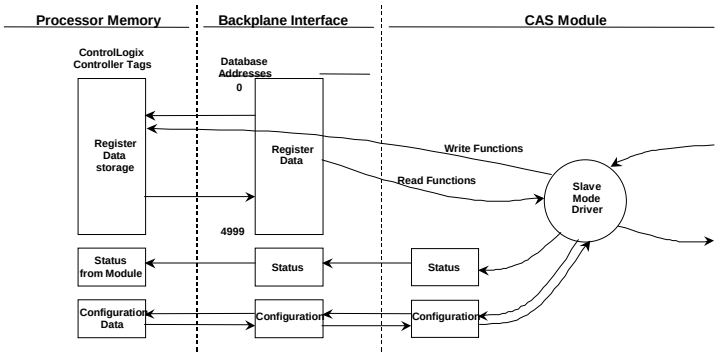
This will inform the module that the command has been processed and can be cleared from the pass-through queue.

5.2.6 Data Flow Between MVI56-CAS Module and ControlLogix Processor

The following topics describe the flow of data between the two pieces of hardware (ControlLogix processor and MVI56-CAS module) and other nodes on the TCA network under the module's different operating modes. Each port on the module is configured to emulate a TCA slave device. The operation of each port is dependent on this configuration. The sections below discuss the operation of each mode.

Slave Driver

The Slave Driver Mode allows the MVI56-CAS module to respond to data read and write commands issued by a master on the TCA network. The following flow chart and associated table describe the flow of data into and out of the module.



Step	Description
1	The TCA slave port driver receives the configuration information from the ControlLogix processor. This information configures the serial port and define the slave node characteristics. Additionally, the configuration information contains data that can be used to offset data in the database to addresses requested in messages received from master units.
2	A Host device issues a read or write command to the module's node address. The port driver qualifies the message before accepting it into the module.
3	After the module accepts the command, the data is immediately transferred to the processor or from the internal database in the module. If the command is a read command, the data is read out of the database and a response message is built. If the command is a write command, the data is written directly to the processor and a response message is built.
4	After the data processing has been completed in Step 2, the response is issued to the originating master node.
5	Counters are available in the Status Block that permit the ladder logic program to determine the level of activity of the Slave Driver.

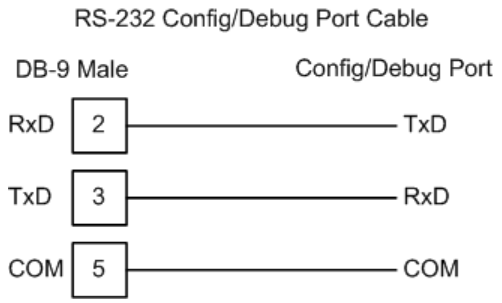
5.3 Cable Connections

The application ports on the MVI56-CAS module support RS-232, RS-422, and RS-485 interfaces. Please inspect the module to ensure that the jumpers are set correctly to correspond with the type of interface you are using.

Note: When using RS-232 with radio modem applications, some radios or modems require hardware handshaking (control and monitoring of modem signal lines). Enable this in the configuration of the module by setting the UseCTS parameter to 1.

5.3.1 RS-232 Configuration/Debug Port

This port is physically an RJ45 connection. An RJ45 to DB-9 adapter cable is included with the module. This port permits a PC based terminal emulation program to view configuration and status data in the module and to control the module. The cable for communications on this port is shown in the following diagram:

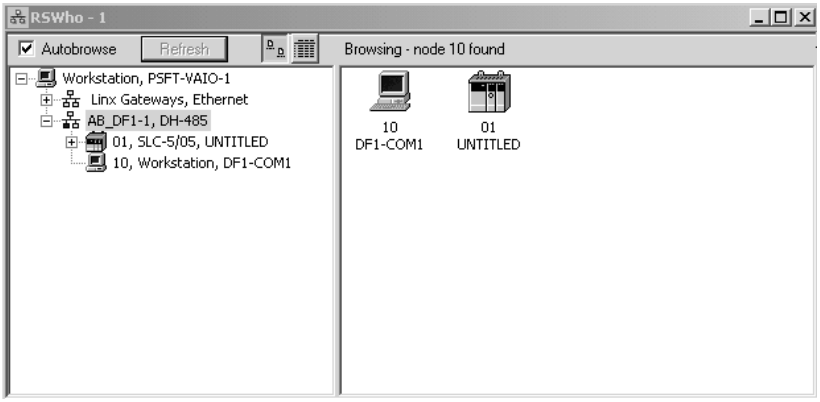


Disabling the RSLinx Driver for the Com Port on the PC

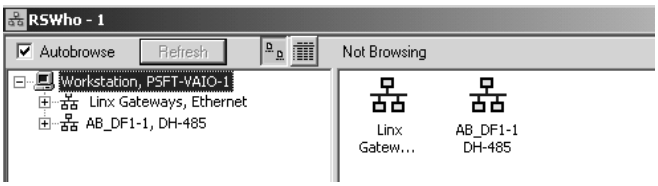
The communication port driver in RSLinx can occasionally prevent other applications from using the PC's COM port. If you are not able to connect to the module's configuration/debug port using ProSoft Configuration Builder (PCB), HyperTerminal or another terminal emulator, follow these steps to disable the RSLinx Driver.

1 Open RSLinx and go to Communications>RSWho

2 Make sure that you are not actively browsing using the driver that you wish to stop. The following shows an actively browsed network:



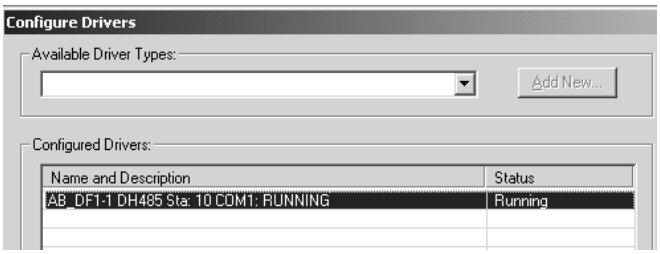
3 Notice how the DF1 driver is opened, and the driver is looking for a processor on node 1. If the network is being browsed, then you will not be able to stop this driver. To stop the driver your RSWho screen should look like this:



Branches are displayed or hidden by clicking on the ⊞ or the ⊟ icons.

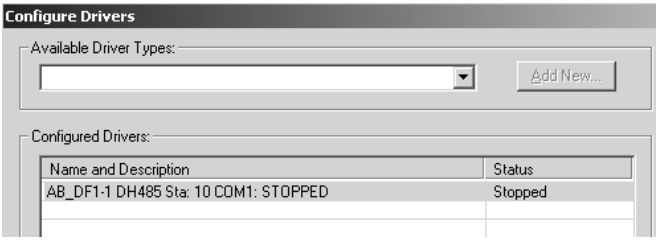
4 When you have verified that the driver is not being browsed, go to **Communications>Configure Drivers**

You may see something like this:



If you see the status as running, you will not be able to use this com port for anything other than communication to the processor. To stop the driver press the "Stop" on the side of the window:

5 After you have stopped the driver you will see the following:

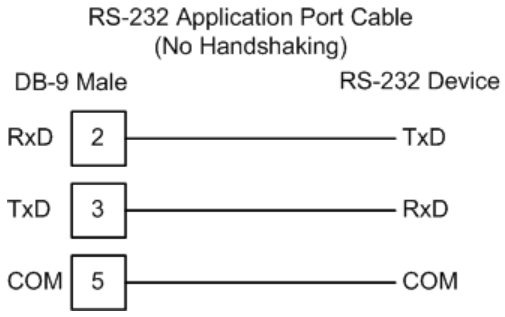


6 Upon seeing this, you may now use that com port to connect to the debug port of the module.

Note: You may need to shut down and restart your PC before it will allow you to stop the driver (usually only on Windows NT machines). If you have followed all of the above steps, and it will not stop the driver, then make sure you do not have RSLogix open. If RSLogix is not open, and you still cannot stop the driver, then reboot your PC.

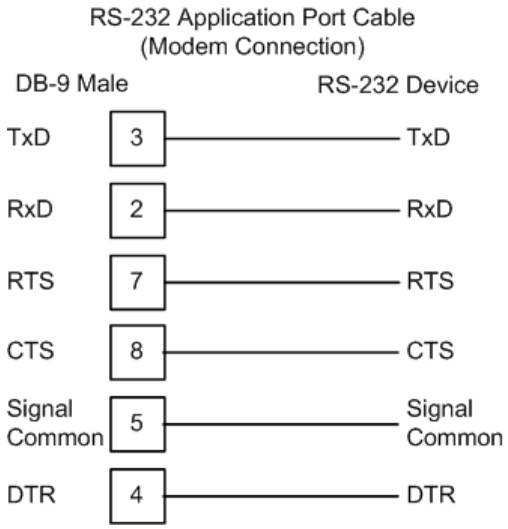
5.3.2 RS-232

When the RS-232 interface is selected, the use of hardware handshaking (control and monitoring of modem signal lines) is user definable. If no hardware handshaking will be used, the cable to connect to the port is as shown below:



RS-232: Modem Connection

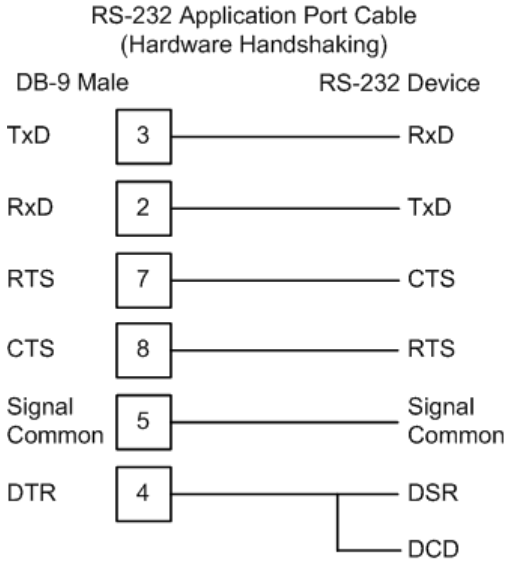
This type of connection is required between the module and a modem or other communication device.



The "Use CTS Line" parameter for the port configuration should be set to 'Y' for most modem applications.

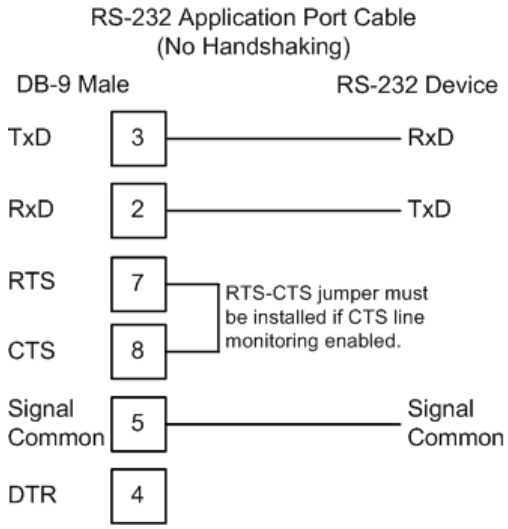
RS-232: Null Modem Connection (Hardware Handshaking)

This type of connection is used when the device connected to the module requires hardware handshaking (control and monitoring of modem signal lines).



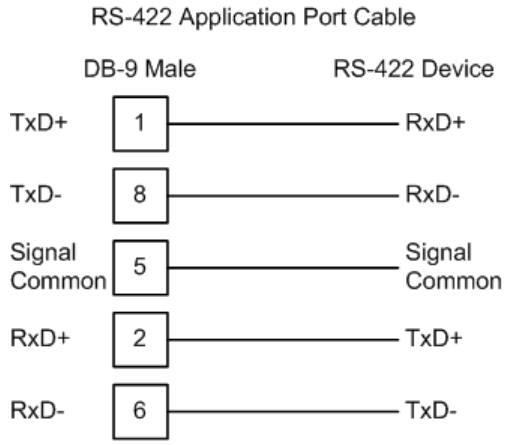
RS-232: Null Modem Connection (No Hardware Handshaking)

This type of connection can be used to connect the module to a computer or field device communication port.



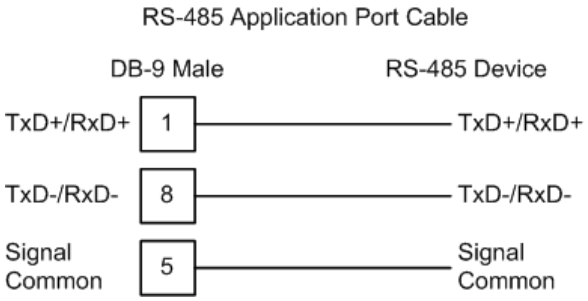
Note: If the port is configured with the "Use CTS Line" set to 'Y', then a jumper is required between the RTS and the CTS line on the module connection.

5.3.3 RS-422



5.3.4 RS-485

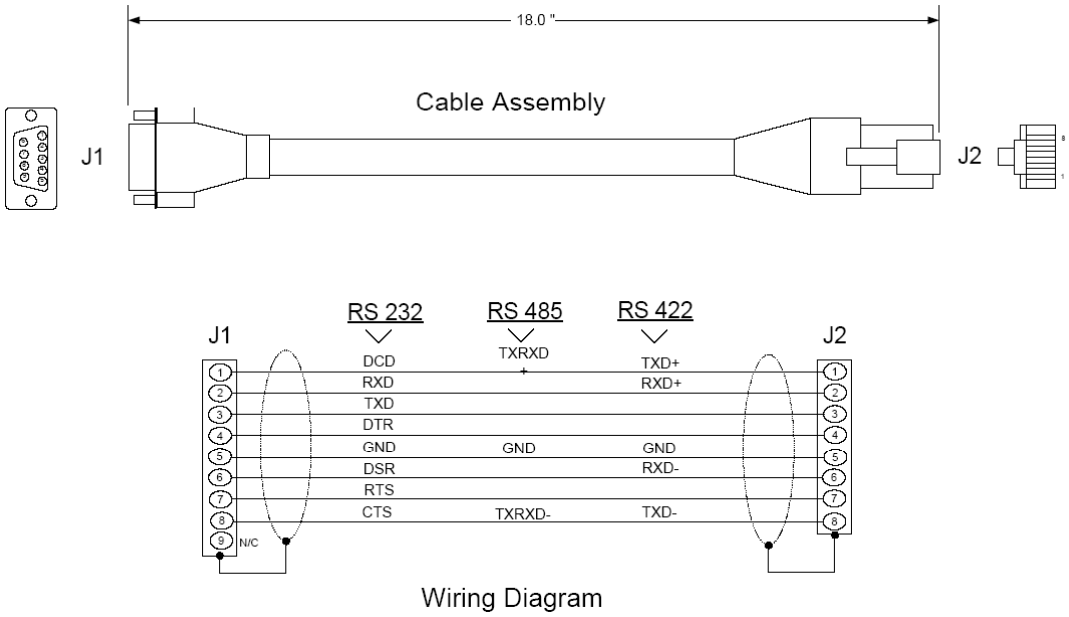
The RS-485 interface requires a single two or three wire cable. The Common connection is optional and dependent on the RS-485 network. The cable required for this interface is shown below:



RS-485 and RS-422 Tip

If communication in the RS-422/RS-485 mode does not work at first, despite all attempts, try switching termination polarities. Some manufacturers interpret +/- and A/B polarities differently.

5.3.5 DB9 to RJ45 Adaptor (Cable 14)



Wiring Diagram

5.4 MVI56-CAS Database Definition

This section contains a listing of the internal database of the MVI56-CAS module. This information can be used to interface other devices to the data contained in the module.

Register Range	TCA Low	TCA High	Content	Size
0 to 4999	40001	45000	User Data	5000
5000 to 5009	45001	45010	Backplane Configuration	10
5010 to 5039	45011	45040	Port 1 Setup	30
5040 to 5069	45041	45070	Port 2 Setup	30
6000 to 6034	47601	47633	Misc. Status Data	335

The User Data area holds data received from the processor (write blocks). Additionally, this data area is used as a data source for the processor (read blocks).

Detailed definition of the miscellaneous status data area can be found in the Reference chapter.

Definition of the configuration data areas can be found in the data definition section of this document and in the Reference chapter.

Refer to the Reference chapter for a discussion of the command control section of the database.

5.5 MVI56-CAS Status Data Definition

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

Offset	Content	Description
202	Program Scan Count	This value is incremented each time a complete program cycle occurs in the module.
203 to 204	Product Code	These two registers contain the product code of "CAS"
205 to 206	Product Version	These two registers contain the product version for the currently running software.
207 to 208	Operating System	These two registers contain the month and year values for the program operating system.
209 to 210	Run Number	These two registers contain the run number value for the currently running software.
211	Port 1 Command List Requests	This field contains the number of requests made from this port to slave devices on the network.
212	Port 1 Command List Response	This field contains the number of slave response messages received on the port.
213	Port 1 Command List Errors	This field contains the number of command errors processed on the port. These errors could be due to a bad response or command.
214	Port 1 Requests	This field contains the total number of messages sent out of the port.

Offset	Content	Description
215	Port 1 Responses	This field contains the total number of messages received on the port.
216	Port 1 Errors Sent	This field contains the total number of error messages sent out of the port.
217	Port 1 Errors Received	This field contains the total number of message errors received on the port.
218	Port 2 Command List Requests	This field contains the number of requests made from this port to slave devices on the network.
219	Port 2 Command List Response	This field contains the number of slave response messages received on the port.
220	Port 2 Command List Errors	This field contains the number of command errors processed on the port. These errors could be due to a bad response or command.
221	Port 2 Requests	This field contains the total number of messages sent out the port.
222	Port 2 Responses	This field contains the total number of messages received on the port.
223	Port 2 Errors Sent	This field contains the total number of message errors sent out the port.
224	Port 2 Errors Received	This field contains the total number of message errors received on the port.
225	Read Block Count	This field contains the total number of read blocks transferred from the module to the processor.
226	Write Block Count	This field contains the total number of write blocks transferred from the processor to the module.
227	Parse Block Count	This field contains the total number of blocks successfully parsed that were received from the processor.
228	Command Event Block Count	This field contains the total number of command event blocks received from the processor.
229	Command Block Count	This field contains the total number of command blocks received from the processor.
230	Error Block Count	This field contains the total number of block errors recognized by the module.
231	Port 1 Current Error	For a slave port, this field contains the value of the current error code returned. For a master port, this field contains the index of the currently executing command.
232	Port 1 Last Error	For a slave port, this field contains the value of the last error code returned. For a master port, this field contains the index of the command with an error.
233	Port 2 Current Error	For a slave port, this field contains the value of the current error code returned. For a master port, this field contains the index of the currently executing command.
234	Port 2 Last Error	For a slave port, this field contains the value of the last error code returned. For a master port, this field contains the index of the command with an error.

Offset	Content	Description
235	Port 1 Status Scan	Contains the port's scan status in the upper byte and the port's current error status in the lower byte. The scan status is encoded in the upper byte as follows: Bit 13 - Status Data Scan Bit 14 - Analog Data Scan Bit 15 - Read All Data Scan
236	Port 2 Status Scan	Contains the port's scan status in the upper byte and the port's current error status in the lower byte. The scan status is encoded in the upper byte as follows: Bit 13 - Status Data Scan Bit 14 - Analog Data Scan Bit 15 - Read All Data Scan

5.6 MVI56-CAS Configuration Data Definition

This section contains listings of the MVI56-CAS module's database that are related to the module's configuration. This data is available to any node on the network and is read from the ControlLogix processor when the module first initializes.

5.6.1 Port 1

Group	Register	Content	Description
Backplane Setup	5000	Status Length	This parameter specifies the number of status words in the module (0 to 250).
	5001	Analog Length	This parameter specifies the number of analog words in the module (0 to 250).
	5002	Meter Length	This parameter specifies the number of meter words in the module (0 to 250).
	5003	Tank Length	This parameter specifies the number of tank words in the module (0 to 250).
	5004	Backplane Fail	This parameter specifies the number of successive transfer errors that must occur before the communication ports are shut down. If the parameter is set to zero, the communication ports will continue to operate under all conditions. If the value is set to larger than zero (1 to 65535), communications will cease if the specified number of failures occur.
	5005	Spare	Spare
	5006	Spare	Spare
	5007	Spare	Spare
	5008	Spare	Spare
	5009	Spare	Spare
	5010	Enabled	This parameter enables the port. Valid values are 0 and 1.

Group	Register	Content	Description
	5011	Baud Rate	This is the baud rate to be used on the port. Enter the baud rate as a value. For example, to select 19K baud, enter 19200. Valid entries for this field include 110, 150, 300, 600, 1200, 2400, 4800, 9600, 19200, 28800, 38400, 57600, and 115.
	5012	Parity	This is the parity code to be used for the port. None, Odd, Even.
	5013	Data Bits	This parameter sets the number of data bits for each word used by the protocol. Valid entries are 5 or 8.
	5014	Stop Bits	This parameter sets the number of stop bits to be used with each data value set. Valid entries are 1 and 2.
	5015	RTS On	This parameter sets the number of milliseconds to delay after RTS is asserted before the data is transmitted. Valid values are 0 to 65535.
	5016	RTS Off	This parameter sets the number of milliseconds to delay after the last byte of data is sent before the RTS modem signal will be set low. Valid values are 0 to 65535.
	5017	Minimum Response Time	This parameter specifies the minimum number of milliseconds to delay before responding to a request message. This pre-send delay is applied before the RTS on time. This may be required when communicating with slow devices.
	5018	Use CTS Line	This parameter specifies if the CTS modem control line is to be used. If the parameter is set to 0, the CTS line will not be monitored. If the parameter is set to 1, the CTS line will be monitored and must be high before the module will send data. This parameter is normally only required when half-duplex modems are used for communication (2-wire).
	5019	Slave ID	This parameter defines the virtual slave address for the internal database. Any request received by the port with this address will be processed by the module. Be certain each device has a unique address on a network. Valid range for this parameter is 1 to 255 (247 on some networks).
	5020	Use Guard Band	Not implemented.
	5021	Guard Band Time	Not Implemented.
	5022	Direct Control	This parameter enables direct control. If enabled, a Select command is not required before a Control command. Valid entries are 0 and 1.
	5023	Swap Config	This parameter sets the data swapping. The swapping is set as a bit field. Bit 0 corresponds to status data, Bit 1 to analog data, and Bit 2 to meter data.

5.6.2 Port 2

Group	Register	Content	Description
	5040	Enabled	This parameter enables the port. Valid entries are 0 and 1.
	5041	Baud Rate	This is the baud rate to be used on the port. Enter the baud rate as a value. For example, to select 19K baud, enter 19200. Valid entries for this field include 110, 150, 300, 600, 1200, 2400, 4800, 9600, 19200, 28800, 38400, 57600, and 115.
	5042	Parity	This is the parity code to be used for the port. None, Odd, Even.
	5043	Data Bits	This parameter sets the number of data bits for each word used by the protocol. Valid entries are 5 or 8.
	5044	Stop Bits	This parameter sets the number of stop bits to be used with each data value set. Valid entries are 1 and 2.
	5045	RTS On	This parameter sets the number of milliseconds to delay after RTS is asserted before the data is transmitted. Valid values are 0 to 65535.
	5046	RTS Off	This parameter sets the number of milliseconds to delay after the last byte of data is sent before the RTS modem signal will be set low. Valid values are 0 to 65535.
	5047	Minimum Response Time	This parameter specifies the minimum number of milliseconds to delay before responding to a request message. This pre-send delay is applied before the RTS on time. This may be required when communicating with slow devices.
	5048	Use CTS Line	This parameter specifies if the CTS modem control line is to be used. If the parameter is set to 0, the CTS line will not be monitored. If the parameter is set to 1, the CTS line will be monitored and must be high before the module will send data. This parameter is normally only required when half-duplex modems are used for communication (2-wire).
	5049	Slave ID	This parameter defines the virtual slave address for the internal database. Any request received by the port with this address will be processed by the module. Be certain each device has a unique address on a network. Valid range for this parameter is 1 to 255 (247 on some networks).
	5050	Use Guard Band	Not implemented.
	5051	Guard Band Time	Not implemented.
	5052	Direct Control	This parameter enables direct control. If enabled, a Select command is not required before a Control command. Valid entries are 0 and 1.
	5053	Swap Config	This parameter sets data swapping. The swapping is set as a bit field. Bit 0 corresponds to status data, Bit 1 to analog data, and Bit 2 to meter data.

5.6.3 Status

Group	Register	Content	Description
Misc. Status	6000	Program Scan Count	This value is incremented each time a complete program cycle occurs in the module.
	6001, 6002	Product Code	These two registers contain the product code of "CAS"
	6003, 6004	Product Version	These two registers contain the product version for the currently running software.
	6005, 6006	Operating System	These two registers contain the month and year values for the program operating system.
	6007, 6008	Run Number	These two registers contain the run number value for the currently running software.
	6009	Port 1 Command List Requests	This field contains the number of requests made from this port to slave devices on the network.
	6010	Port 1 Command List Response	This field contains the number of slave response messages received on the port.
	6011	Port 1 Command List Errors	This field contains the number of command errors processed on the port. These errors could be due to a bad response or command.
	6012	Port 1 Requests	This field contains the total number of messages sent out of the port.
	6013	Port 1 Responses	This field contains the total number of messages received on the port.
	6014	Port 1 Errors Sent	This field contains the total number of error messages sent out of the port.
	6015	Port 1 Errors Received	This field contains the total number of message errors received on the port.
	6016	Port 2 Command List Requests	This field contains the number of requests made from this port to slave devices on the network.
	6017	Port 2 Command List Response	This field contains the number of slave response messages received on the port.
	6018	Port 2 Command List Errors	This field contains the number of command errors processed on the port. These errors could be due to a bad response or command.
	6019	Port 2 Requests	This field contains the total number of messages sent out the port.
	6020	Port 2 Responses	This field contains the total number of messages received on the port.
	6021	Port 2 Errors Sent	This field contains the total number of message errors sent out the port.

Group	Register	Content	Description
	6022	Port 2 Errors Received	This field contains the total number of message errors received on the port.
	6023	Read Block Count	This field contains the total number of read blocks transferred from the module to the processor.
	6024	Write Block Count	This field contains the total number of write blocks transferred from the processor to the module.
	6025	Parse Block Count	This field contains the total number of blocks successfully parsed that were received from the processor.
	6026	Command Event Block Count	This field contains the total number of command event blocks received from the processor.
	6027	Command Block Count	This field contains the total number of command blocks received from the processor.
	6028	Error Block Count	This field contains the total number of block errors recognized by the module.
	6029	Port 1 Current Error/Index	For a slave port, this field contains the value of the current error code returned. For a master port, this field contains the index of the currently executing command.
	6030	Port 1 Last Error	For a slave port, this field contains the value of the last error code returned. For a master port, this field contains the index of the command with an error.
	6031	Port 2 Current Error	For a slave port, this field contains the value of the current error code returned. For a master port, this field contains the index of the currently executing command.
	6032	Port 2 Last Error	For a slave port, this field contains the value of the last error code returned. For a master port, this field contains the index of the command with an error.
	6033	Port 1 Status Scan	This register contains the port's Scan Status in the upper byte and the port's current Error Status in the lower byte. The Scan Status is encoded in the upper byte as follows: Bit 13 - Status Data scan Bit 14 - Analog Data scan Bit 15 - Read All Data scan
	6034	Port 2 Status Scan	This register contains the port's Scan Status in the upper byte and the port's current Error Status in the lower byte. The Scan Status is encoded in the upper byte as follows: Bit 13 - Status Data scan Bit 14 - Analog Data scan Bit 15 - Read All Data scan
	6703	Spare	Spare
	-	-	-
	6799	Spare	Spare

5.7 MVI56-CAS Command Control

Command Control data is received from other nodes on the network that can control the MVI56-CAS module. Specific values are written to regions of this block to control the module. Currently, the module is programmed to handle the receipt of the following requests: write configuration to processor, warm boot and cold boot.

The remote node controls the module by writing one of the following values to register 6800:

Value	Description
9997	Write configuration in database to the processor and warm boot the module.
9998	Warm boot the module.
9999	Cold boot the module.

The control register is cleared (a value of 0) after the operation is executed with the exception of the 9997 command. If the module fails to successfully transfer the configuration to the processor, an error code will be returned in the control register as follows:

Value	Description
0	No error, transfer successful
-1	Error transferring general configuration information.

Ladder logic must handle the 9997 command. No ladder logic is required when using the warm or cold boot commands.

5.8 Protocol Commands

The ProSoft Technology CAS module communication driver supports several data read and write commands. When configuring an application, it may be important to understand how the commands function in order to determine how to structure the application data.

5.8.1 Data Read Functions

As stated in earlier sections, the data sent to the master is taken directly out of the module's memory at the time a response is sent.

Function	Function Name	Description
1	Read Status Data	The module supports access to a total of up to 250 words of status data from the ladder processor. In TCAP vernacular, the "Start Card Address" identifies the starting word, and the "Number of Cards" identifies the number of words.
2	Read Analog Data	The module supports access to a total of up to 250 words of analog data from the ladder processor. In TCAP vernacular, the "Start Card No." and the "Start Point No." (upper and lower four bits respectively) are treated as one byte to develop a starting word address, while the "Number of Analogs" determines the number of words to be returned to the master.

Function	Function Name	Description
3	Read Meter Data	The module supports access to a total of up to 250 words of Meter data (125 meter readings) from the ladder processor. In TCAP vernacular, the "Starting Meter Number" determines the first word to be accessed, while the "Number of meters to be returned" value determines the number of words to be returned to the master.
10	Read Frozen Analog Data	This function provides the master access to the contents of up to a 250 word buffer maintained in the module. This buffer contains the analog values stored as a result of the last "Freeze Analogs" command (44/45) received from the Master. It is addressed in the same fashion as Function 2.
11	Read Frozen Meter Data	This function provides the Master access to the contents of up to a 250 word buffer maintained in the module. This buffer contains the meter values stored as a result of the last "Freeze Meters" command (43/45) received from the Master. It is addressed in the same fashion as Function 3.
16	Read All Data	This general-purpose command is supported in the ProSoft module. The following Data Select Types are available from the module: Status Inputs (Data Select 1, bit 7) Analog Inputs (Data Select 1, bit 6) Meter Accumulator Inputs (Data Select 1, bit 5) Current Tank Data (Data Select 1, bit 2) Maximum Analogs Register (Data Select 1, bit 1) Local Freeze Analog Data (Data Select 1, bit 0) Local Freeze Meter Data (Data Select 2, bit 7) Local Freeze Tank Data (Data Select 2, bit 6) Hourly Freeze Analog Data (Data Select 2, bit 5) Hourly Freeze Meter Data (Data Select 2, bit 4) Hourly Freeze Tank Data (Data Select 2, bit 3) Periodic Analog Data (Data Select 2, bit 2) Periodic Meter Data (Data Select 2, bit 1) Periodic Tank Data (Data Select 2, bit 0) The command allows up to 63 analog values to be requested and up to 7 meters. If no data quantity is requested (length fields are zero), all of the available data, as defined in the Configuration Parameters, will be returned. When the Status Inputs are requested, the full status table is returned.

5.8.2 Control Commands From Master

When Control commands are received from a Master, their action is immediately communicated to the PLC/SLC for action. No modification is made directly to the ProSoft Module's memory as a result of a Control command. In order for and Control action to be reflected in the module's memory, it must be transferred within the Status Data block.

Function	Function Name	Description
30/31	Momentary Control Select/Operate	These control commands are recognized by the module. If the Operate command is received out of sequence (must be received in next communication sequence after Select command), the command is disregarded. If the module is configured for Direct Control, then the Master need only send the Operate command for action to occur. In TCAP vernacular, the "Card Number" and the "Point Number" are combined to identify the bit number. The action value is always set to 1, so the action will always be to set a bit true momentarily.
32/33	Continuous Control Select/Operate	These control commands are recognized by the module. If the Operate command is received out of sequence (must be received in next communication sequence after Select command), the command is disregarded. If the module is configured for Direct control, then the master need only send the Operate command for action to occur. In TCAP vernacular, the "Card Number" and the "Point Number" are combined to identify the bit number.
34/35	Setpoint Select/Operate	These control commands are recognized by the module. If the Operate command is received out of sequence (must be received in next communication sequence after Select command), the command is disregarded. If the module is configured for Direct Control, then the Master need only send the Operate command for action to occur. In TCAP vernacular, the "Card Number" (upper 7 bits) is combined with the "Point Number" (low 1 bit) to obtain an effective addressing range of 256 registers. The value written to the ladder logic is limited to a 12 bit value by the protocol specification.

5.8.3 Freeze Data Commands

These general purpose commands initiate the movement of data within buffers in the ProSoft Technology module. These buffers are maintained for access by related read commands, discussed previously.

The module one-shots the freeze commands from the Master, enable the movement of the buffers only if the Master has previously issued a read to the previously frozen buffer.

Function	Function Name	Description
43	Freeze Meters	Moves the Meter Data buffer to the Freeze Meter buffer.
44	Freeze Analogs	Moves the Analog Data buffer to the Freeze Analog buffer.
45	Freeze Meters and Analogs	Moves the Meter Data buffer and the Analog Data buffer to their respective Freeze buffers.
46/6	Special Freeze	Moves current data to hourly, local, and periodic registers.

5.8.4 Exception Data Commands

The polled exception functionality is implemented in the module. The module copies data that changes enough to meet exception requirements into an exception table. The data must change more than the default dead-bands in order to record an exception. The default dead-bands are:

- Analog Dead-Band: 1000
- Meter Dead-Band: 100
- Tank Dead-Band: 0
- Status Dead-Band: 0

Function	Function Name	Description
50	Exception Data Request	Two modes are supported; Current Exceptions and All Exception Data. The mode is selected by the update bit in the master message.

6 Support, Service & Warranty

In This Chapter

Be sure and read the full Warranty that can be found on our web site at www.prosoft-technology.com for details and other terms and conditions. The content in this summary is subject to change without notice. The content is current at date of publication.

ProSoft Technology, Inc. strives to provide meaningful support to its customers. Should any questions or problems arise, please feel free to contact us at:

Internet	Web Site: http://www.prosoft-technology.com/support
	E-mail address: support@prosoft-technology.com

Those of us at ProSoft Technology, Inc. want to provide the best and quickest support possible, so before calling please have the following information available. You may wish to fax this information to us prior to calling.

1. Product Version Number
2. System architecture
3. Network details

In the case of hardware, we will also need the following information:

1. Module configuration and contents of file
2. Module Operation
3. Configuration/Debug status information
4. LED patterns
5. Information about the processor and user data files as viewed through the development software and LED patterns on the processor
6. Details about the networked devices interfaced, if any

For technical support calls within the United States, an after-hours answering system allows pager access to one of our qualified technical and/or application support engineers at any time to answer your questions.

6.1 How to Contact Us: Sales and Support

All ProSoft Technology Products are backed with full technical support. Contact our worldwide Technical Support team and Customer Service representatives directly by phone or email:

USA / Latin America (excluding Brasil) (Office in California)

+1(661) 716-5100
+1(661) 716-5101 (Fax)
1675 Chester Avenue, 4th Floor
Bakersfield, California 93301
U.S.A.
+1.661.716.5100, support@prosoft-technology.com
Languages spoken include: English, Spanish

Asia Pacific (office in Malaysia)

+603.7724.2080
+603.7724.2090 (Fax)
C210, Damansara Intan,
1 Jalan SS20/27, 47400 Petaling Jaya
Selangor, Malaysia
+603.7724.2080, asiapc@prosoft-technology.com
Languages spoken include: Chinese, Japanese, English

China Pacific (office in China)

+86.21.64518356 x 8011
+86.21.64756957 (Fax)
4/F, No. 16 Hongcao Road
Shanghai, China 200233
China
+86.21.64518356 x 8011, zhang@prosoft-technology.com
Languages spoken include: Chinese, English

Europe / Middle East / Africa (office in Toulouse, France)

+33 (0) 5.34.36.87.20
+33 (0) 5.61.78.40.52 (Fax)
Zone d'activité de Font Grasse
17, rue des Briquetiers
F-31700 Blagnac
France
+33 (0) 5.34.36.87.20. support. EMEA@prosoft-technology.com
Languages spoken include: French, English

Brasil (office in Sao Paulo)

+55-11-5084-5178
+55-11-5083-3776 (Fax)
Rua Vergueiro, 2949 - sala 182 - Edifício Vergueiro Work Center
Vila Mariana - São Paulo
Cep: 04101-300 - Brasil
+55-11-5084-5178, eduardo@prosoft-technology.com
Languages spoken include: Portuguese, English

6.2 Return Material Authorization (RMA) Policies and Conditions

The following RMA Policies and Conditions apply to any returned product. These RMA Policies are subject to change by ProSoft without notice. For warranty information, see Section C below entitled "Limited Warranty". In the event of any inconsistency between the RMA Policies and the Warranty, the Warranty shall govern.

6.2.1 All Product Returns

1 In order to return a Product for repair, exchange or otherwise, the Customer must obtain a Returned Material Authorization (RMA) number from ProSoft and comply with ProSoft shipping instructions.
2 In the event that the Customer experiences a problem with the Product for any reason, Customer should contact ProSoft Technical Support at one of the telephone numbers listed above in Section A. A Technical Support Engineer will request several tests in an attempt to isolate the problem. If after these tests are completed, the Product is found to be the source of the problem, ProSoft will issue an RMA.
3 All returned Products must be shipped freight prepaid, in the original shipping container or equivalent, to the location specified by ProSoft, and be accompanied by proof of purchase. The RMA number is to be prominently marked on the outside of the shipping box. Customer agrees to insure the Product or assume the risk of loss or damage in transit. Products shipped to ProSoft without an RMA number will be returned to the Customer, freight collect. Contact ProSoft Technical Support for further information.
4 Out of warranty returns are not allowed on RadioLinx accessories such as antennas, cables, and brackets.

The following policy applies for Non-Warranty Credit Returns:

A 10% Restocking Fee if Factory Seal is *not* broken
B 20% Restocking Fee if Factory Seal is broken

ProSoft retains the right, in its absolute and sole discretion, to reject any non-warranty returns for credit if the return is not requested within three (3) months after shipment of the Product to Customer, if the Customer fails to comply with ProSoft's shipping instructions, or if the Customer fails to return the Product to ProSoft within six (6) months after Product was originally shipped.

6.3 Procedures for Return of Units Under Warranty

1 A Technical Support Engineer must pre-approve all product returns.
2 Module is repaired or replaced after a Return Material Authorization Number is entered and a replacement order is generated.
3 Credit for the warranted item is issued within 10 business days after receipt of product and evaluation of the defect has been performed by ProSoft. The credit will only be issued provided the product is returned with a valid Return Material Authorization Number and in accordance with ProSoft's shipping instructions.

a) If no defect is found, a credit is issued.
b) If a defect is found and is determined to be customer generated or if the defect is otherwise not covered by ProSoft's Warranty, or if the module is not repairable, a credit is not issued and payment of the replacement module is due.

6.4 Procedures for Return of Units Out of Warranty

1 Customer sends unit in for evaluation.
2 If no defect is found, Customer will be charged the equivalent of US $100 plus shipping, duties and taxes that may apply. A new Purchase Order will be required for this evaluation fee.

If the unit is repaired the charge to the Customer will be 30%* of the list price plus any shipping, duties and taxes that may apply. A new Purchase Order will be required for a product repair.

3 For an immediate exchange, a new module may be purchased and sent to Customer while repair work is being performed. Credit for purchase of the new module will be issued when the new module is returned in accordance with ProSoft's shipping instructions and subject to ProSoft's policy on non-warranty returns. This is in addition to charges for repair of the old module and any associated charges to Customer.
4 If, upon contacting ProSoft Customer Service, the Customer is informed that unit is believed to be unrepairable, the Customer may choose to send unit in for evaluation to determine if the repair can be made. Customer will pay shipping, duties and taxes that may apply. If unit cannot be repaired, the Customer may purchase a new unit.

6.4.1 Un-repairable Units

- 3150-All
- 3750
- 3600-All
- 3700
- 3170-All
- 3250
- 1560 can be repaired, if defect is the power supply
- 1550 can be repaired, if defect is the power supply
- 3350
- 3300
- 1500-All

*** 30% of list price is an estimated repair cost only. The actual cost of repairs will be determined when the module is received by ProSoft and evaluated for needed repairs.**

6.4.2 Purchasing Warranty Extension

As detailed below in ProSoft's Warranty, the standard Warranty Period is one year (or in the case of RadioLinx modules, three years) from the date of delivery. The Warranty Period may be extended for an additional charge, as follows:

- Additional 1 year = 10% of list price
- Additional 2 years = 20% of list price
- Additional 3 years = 30% of list price

6.5 LIMITED WARRANTY

This Limited Warranty ("Warranty") governs all sales of hardware, software and other products (collectively, "Product") manufactured and/or offered for sale by ProSoft, and all related services provided by ProSoft, including maintenance, repair, warranty exchange, and service programs (collectively, "Services"). By purchasing or using the Product or Services, the individual or entity purchasing or using the Product or Services ("Customer") agrees to all of the terms and provisions (collectively, the "Terms") of this Limited Warranty. All sales of software or other intellectual property are, in addition, subject to any license agreement accompanying such software or other intellectual property.

6.5.1 What Is Covered By This Warranty

A *Warranty On New Products*: ProSoft warrants, to the original purchaser only, that the Product that is the subject of the sale will (1) conform to and perform in accordance with published specifications prepared, approved, and issued by ProSoft, and (2) will be free from defects in material or workmanship; provided these warranties only cover Product that is sold as new. This Warranty expires one year (or in the case of RadioLinx modules, three years) from the date of shipment (the "Warranty Period"). If the Customer discovers within the Warranty Period a failure of the Product to conform to specifications, or a defect in material or workmanship of the Product, the Customer must promptly notify ProSoft by fax, email or telephone. In no event may that notification be received by ProSoft later than 15 months (or in the case of RadioLinx modules, 39 months) from the date of delivery. Within a reasonable time after notification, ProSoft will correct any failure of the Product to conform to specifications or any defect in material or workmanship of the Product, with either new or used replacement parts. Such repair, including both parts and labor, will be performed at ProSoft's expense. All warranty service will be performed at service centers designated by ProSoft. If ProSoft is unable to repair the Product to conform to this Warranty after a reasonable number of attempts, ProSoft will provide, at its option, one of the following: a replacement product, a full refund of the purchase price or a credit in the amount of the purchase price. All replaced product and parts become the property of ProSoft. These remedies are the Customer's only remedies for breach of warranty.

B *Warranty On Services*: Material and labor used by ProSoft to repair a verified malfunction or defect are warranted on the terms specified above

for new Product, provided said warranty will be for the period remaining on the original new equipment warranty or, if the original warranty is no longer in effect, for a period of 90 days from the date of repair.

C The Warranty Period for RadioLinx accessories (such as antennas, cables, brackets, etc.) are the same as for RadioLinx modules, that is, three years from the date of shipment.

6.5.2 What Is Not Covered By This Warranty

A ProSoft makes no representation or warranty, expressed or implied, that the operation of software purchased from ProSoft will be uninterrupted or error free or that the functions contained in the software will meet or satisfy the purchaser's intended use or requirements; the Customer assumes complete responsibility for decisions made or actions taken based on information obtained using ProSoft software.

B With the exception of RadioLinx accessories referenced in paragraph 1(c) this Warranty does not cover any product, components, or parts not manufactured by ProSoft.

C This Warranty also does not cover the failure of the Product to perform specified functions, or any other non-conformance, defects, losses or damages caused by or attributable to any of the following: (i) shipping; (ii) improper installation or other failure of Customer to adhere to ProSoft's specifications or instructions; (iii) unauthorized repair or maintenance; (iv) attachments, equipment, options, parts, software, or user-created programming (including, but not limited to, programs developed with any IEC 61131-3 programming languages, or "C") not furnished by ProSoft; (v) use of the Product for purposes other than those for which it was designed; (vi) any other abuse, misapplication, neglect or misuse by the Customer; (vii) accident, improper testing or causes external to the Product such as, but not limited to, exposure to extremes of temperature or humidity, power failure or power surges outside of the limits indicated on the product specifications; or (viii) disasters such as fire, flood, earthquake, wind or lightning.

D The information in this Agreement is subject to change without notice. ProSoft shall not be liable for technical or editorial errors or omissions made herein; nor for incidental or consequential damages resulting from the furnishing, performance or use of this material. The user guides included with your original product purchased by you from ProSoft, contains information protected by copyright. No part of the guide may be duplicated or reproduced in any form without prior written consent from ProSoft.

6.5.3 DISCLAIMER REGARDING HIGH RISK ACTIVITIES

PRODUCT MANUFACTURED OR SUPPLIED BY PROSOFT IS NOT FAULT TOLERANT AND IS NOT DESIGNED, MANUFACTURED OR INTENDED FOR USE IN HAZARDOUS ENVIRONMENTS REQUIRING FAIL-SAFE PERFORMANCE (INCLUDING, WITHOUT LIMITATION, THE OPERATION OF NUCLEAR FACILITIES, AIRCRAFT NAVIGATION OF COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL, DIRECT LIFE SUPPORT MACHINES OR WEAPONS SYSTEMS), IN WHICH THE FAILURE OF THE PRODUCT COULD LEAD DIRECTLY OR INDIRECTLY TO DEATH, PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE (COLLECTIVELY, "HIGH RISK ACTIVITIES"). PROSOFT SPECIFICALLY DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.

6.5.4 DISCLAIMER OF ALL OTHER WARRANTIES

THE WARRANTIES SET FORTH IN PARAGRAPH 1 ABOVE ARE IN LIEU OF ALL OTHER WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

6.5.5 LIMITATION OF REMEDIES**

IN NO EVENT WILL PROSOFT (OR ITS DEALER) BE LIABLE FOR ANY SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES BASED ON BREACH OF WARRANTY, BREACH OF CONTRACT, NEGLIGENCE, STRICT TORT, OR ANY OTHER LEGAL THEORY. DAMAGES THAT PROSOFT AND ITS DEALER WILL NOT BE RESPONSIBLE FOR INCLUDE, BUT ARE NOT LIMITED TO: LOSS OF PROFITS; LOSS OF SAVINGS OR REVENUE; LOSS OF USE OF THE PRODUCT OR ANY ASSOCIATED EQUIPMENT; LOSS OF DATA; COST OF CAPITAL; COST OF ANY SUBSTITUTE EQUIPMENT, FACILITIES, OR SERVICES; DOWNTIME; THE CLAIMS OF THIRD PARTIES, INCLUDING CUSTOMERS OF THE PURCHASER; AND INJURY TO PROPERTY.

** Some areas do not allow time limitations on an implied warranty, or allow the exclusion or limitation of incidental or consequential damages. In such areas the above limitations may not apply. This Warranty gives you specific legal rights, and you may also have other rights which vary from place to place.

6.5.6 Time Limit for Bringing Suit

Any action for breach of warranty must be commenced within 15 months (or in the case of RadioLinx modules, 39 months) following shipment of the Product.

6.5.7 No Other Warranties

Unless modified in writing and signed by both parties, this Warranty is understood to be the complete and exclusive agreement between the parties, suspending all oral or written prior agreements and all other communications between the parties relating to the subject matter of this Warranty, including statements made by salesperson. No employee of ProSoft or any other party is authorized to make any warranty in addition to those made in this Warranty. The Customer is warned, therefore, to check this Warranty carefully to see that it correctly reflects those terms that are important to the Customer.

6.5.8 Intellectual Property

A Any documentation included with Product purchased from ProSoft is protected by copyright and may not be photocopied or reproduced in any form without prior written consent from ProSoft.

B ProSoft's technical specifications and documentation that are included with the Product are subject to editing and modification without notice.

C Transfer of title shall not operate to convey to Customer any right to make, or have made, any Product supplied by ProSoft.

D Customer is granted no right or license to use any software or other intellectual property in any manner or for any purpose not expressly permitted by any license agreement accompanying such software or other intellectual property.

E Customer agrees that it shall not, and shall not authorize others to, copy software provided by ProSoft (except as expressly permitted in any license agreement accompanying such software); transfer software to a third party separately from the Product; modify, alter, translate, decode, decompile, disassemble, reverse-engineer or otherwise attempt to derive the source code of the software or create derivative works based on the software; export the software or underlying technology in contravention of applicable US and international export laws and regulations; or use the software other than as authorized in connection with use of Product.

6.5.9 Additional Restrictions Relating To Software And Other Intellectual Property

In addition to complying with the Terms of this Warranty, Customers purchasing software or other intellectual property shall comply with any license agreement accompanying such software or other intellectual property. Failure to do so may void this Warranty with respect to such software and/or other intellectual property.

6.5.10 Allocation of risks

This Warranty allocates the risk of product failure between ProSoft and the Customer. This allocation is recognized by both parties and is reflected in the price of the goods. The Customer acknowledges that it has read this Warranty, understands it, and is bound by its Terms.

6.5.11 Controlling Law and Severability

This Warranty shall be governed by and construed in accordance with the laws of the United States and the domestic laws of the State of California, without reference to its conflicts of law provisions. If for any reason a court of competent jurisdiction finds any provisions of this Warranty, or a portion thereof, to be unenforceable, that provision shall be enforced to the maximum extent permissible and the remainder of this Warranty shall remain in full force and effect. Any cause of action with respect to the Product or Services must be instituted in a court of competent jurisdiction in the State of California.

Index

N

O

P

R

S

T

U

V

W

Y