ProSoft TECHNOLOGY

Where Automation Connects.

inRAx®

MVI56-DFCMR

ControlLogix Platform

DF1 Interface Module with Reduced Data Block

10/20/2008

USER MANUAL

Please Read This Notice

Successful application of this module requires a reasonable working knowledge of the Rockwell Automation ControlLogix hardware, the MVI56-DFCMR 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

Battery Life Advisory

All modules in the MVI series use a rechargeable Lithium Vanadium Pentoxide battery to backup the 512K SRAM memory, real-time clock, and CMOS. The battery should last for the life of the module.
The module must be powered for approximately twenty hours before it becomes fully charged. After it is fully charged, the battery provides backup power for the CMOS setup and configuration data, the real-time clock, and the 512K SRAM memory for approximately 21 days.
Before you remove a module from its power source, ensure that the battery within the module is fully charged. A fully charged battery will hold the BIOS settings (after being removed from its power source) for a limited number of days. When the battery is fully discharged, the module will revert to the default BIOS settings.

Note: The battery is not user replaceable.

ProSoft® Product Documentation

In an effort to conserve paper, ProSoft Technology no longer includes printed manuals with our product shipments. User Manuals, Datasheets, Sample Ladder Files, and Configuration Files are provided on the enclosed CD and are available at no charge from our web site: http://www.prosoft-technology.com
Printed documentation is available for purchase. Contact ProSoft Technology for pricing and availability.
Asia Pacific: +603.7724.2080
Europe, Middle East, Africa: +33.5.34.36.87.20
Latin America: +1.281.298.9109
North America: +1.661.716.5100

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-DFCMR User Manual
10/20/2008

Contents

Guide to the MVI56-DFCMR 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 60) Diagnostics and Troubleshooting (page 45)	This section describes how to verify communications with the network. Diagnostic and Troubleshooting procedures.
Reference Product Specifications Functional Overview Glossary	→	Reference (page 63) Functional Overview (page 65) Product Specifications (page 63)	These sections contain general references associated with this product, Specifications, and the Functional Overview.
Support, Service, and Warranty Index	→	Support, Service and Warranty (page 103)	This section contains Support, Service and Warranty information. Index of chapters.

1 Start Here

In This Chapter

Installing the MVI56-DFCMR module requires a reasonable working knowledge of the Rockwell Automation hardware, the MVI56-DFCMR 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-DFCMR 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-DFCMR 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
- ProSoft Configuration Builder, 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-DFCMR module, and are all required for installation and configuration.

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

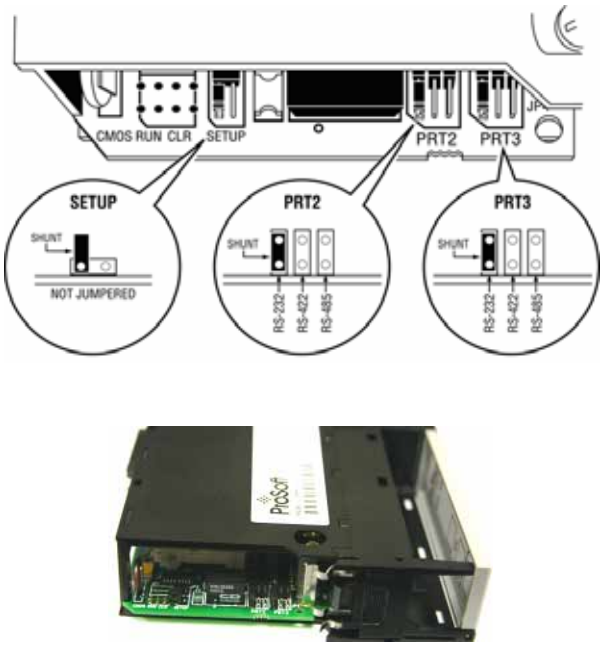
Qty.	Part Name	Part Number	Part Description
1	MVI56-DFCMR Module	MVI56-DFCMR	DF1 Interface Module with Reduced Data Block
1	Cable	Cable #15, RS232 Null Modem	For RS232 Connection to the CFG Port
3	Cable	Cable #14, RJ45 to DB9 Male Adapter cable	For DB9 Connection to Module's Port
2	Adapter	1454-9F	Two 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-DFCMR 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-DFCMR 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

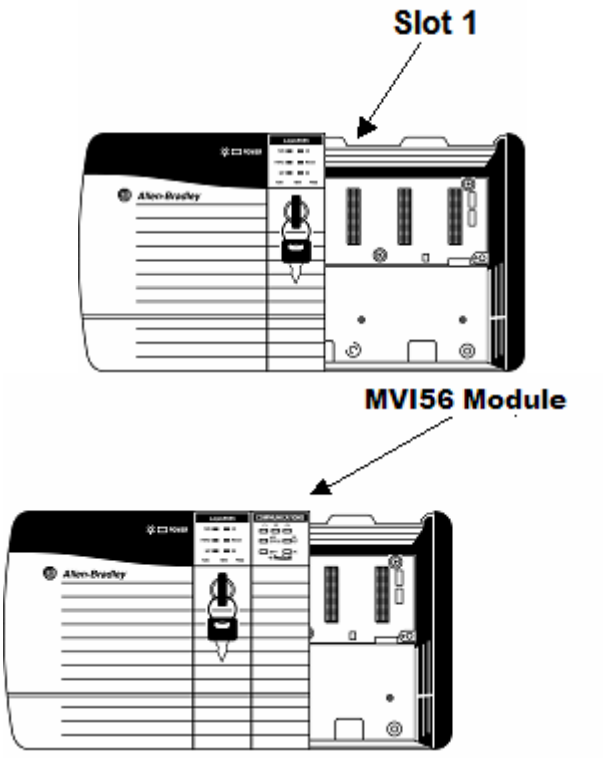
If you have not already installed and configured your ControlLogix processor and power supply, please do so before installing the MVI56-DFCMR module. Refer to your Rockwell Automation product documentation for installation instructions.

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-DFCMR 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.



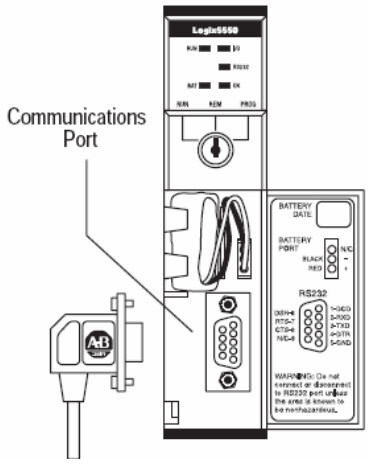
3. With a firm but steady push, snap the module into place.

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.

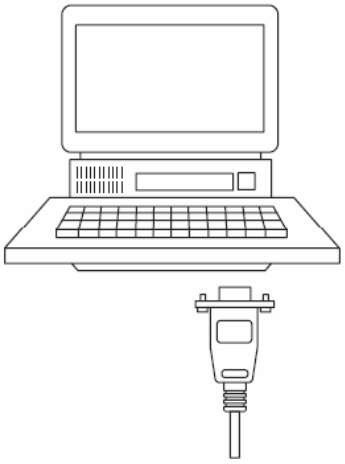
Note: If you insert the module improperly, the system may stop working, or may behave unpredictably.
Note: If you are installing MVI56-DFCMR with other modules connected to the PCI bus, the peripheral modules will not have holding clips. Make sure all of the modules are aligned with their respective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

1.5 Connect your PC to the Processor

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



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



1.6 Open the Sample Ladder Logic

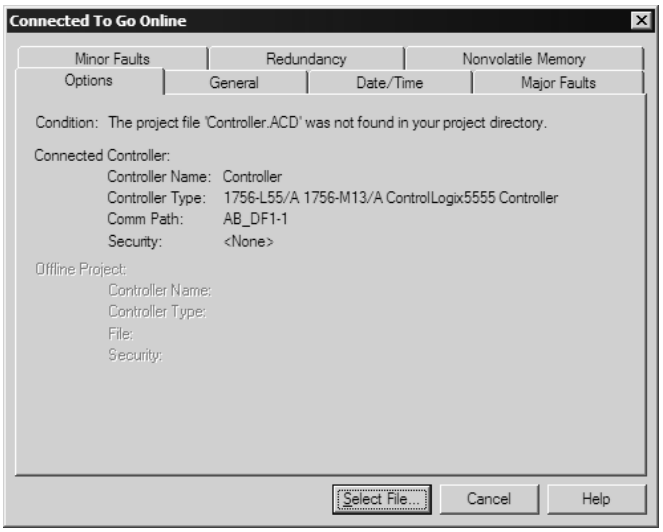
The sample program for your MVI56-DFCMR module includes custom tags, data types and ladder logic for data I/O and status monitoring. For most applications, you can run the sample ladder program without modification, or, for advanced applications, you can incorporate the sample program into your existing application.

The inRAx Solutions CD provides one or more versions of the sample ladder logic. The version number appended to the file name corresponds with the firmware version number of your ControlLogix processor. The firmware version and sample program version must match.

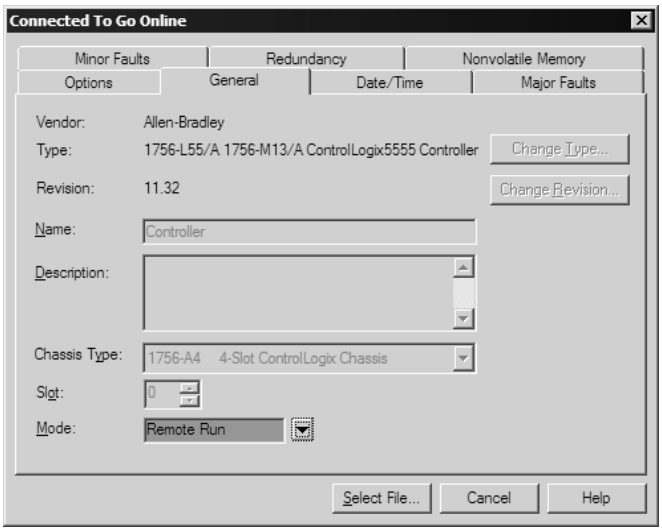
1.6.1 To Determine the Firmware Version of your Processor

Important: The RSLinx service must be installed and running on your computer in order for RSLogix to communicate with the processor. Refer to your RSLinx and RSLogix documentation for help configuring and troubleshooting these applications.

1. Connect an RS-232 serial cable from the COM (serial) port on your PC to the communication port on the front of the processor.
2. Start RSLogix 5000 and close any existing project that may be loaded.
3. Open the Communications menu and choose **Go Online**. RSLogix will establish communication with the processor. This may take a few moments.
4. When RSLogix has established communication with the processor, the Connected To Go Online dialog box will open.



5 On the Connected To Go Online dialog box, click the General tab. This tab shows information about the processor, including the Revision (firmware) version. In the following illustration, the firmware version is 11.32

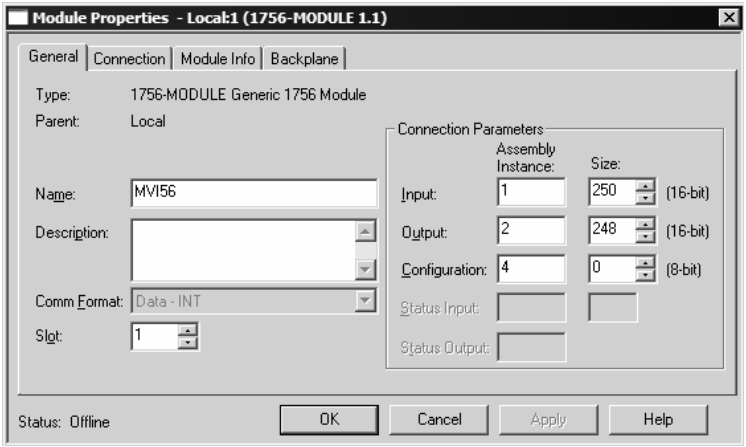


1.6.2 Select the Slot Number for the Module

The sample application is for a module installed in Slot 1 in a ControlLogix rack. The ladder logic uses the slot number to identify the module. If you are installing the module in a different slot, you must update the ladder logic so that program tags and variables are correct, and do not conflict with other modules in the rack.

To change the slot number

1 In the Controller Organization list, select the module [1] 1756-MODULE MVI56, and then click the right mouse button to open a shortcut menu.
2 On the shortcut menu, choose **Properties**. This action opens the Module Properties dialog box.



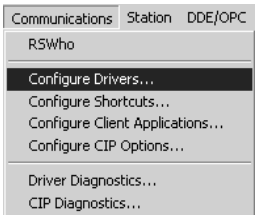
3 In the **Slot:** field, use the spinners on the right side of the field to select the slot number where the module will reside in the rack, and then click OK.

RSLogix will automatically apply the slot number change to all tags, variables and ladder logic rungs that use the MVI56-DFCMR slot number for computation.

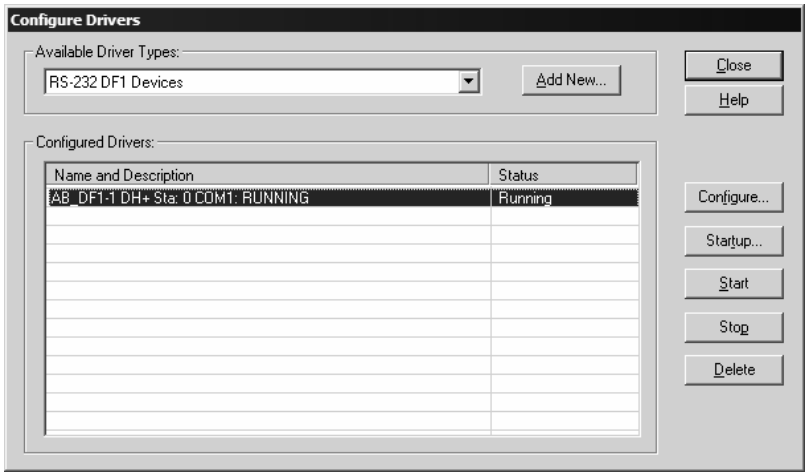
1.6.3 Configuring RSLinx

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.

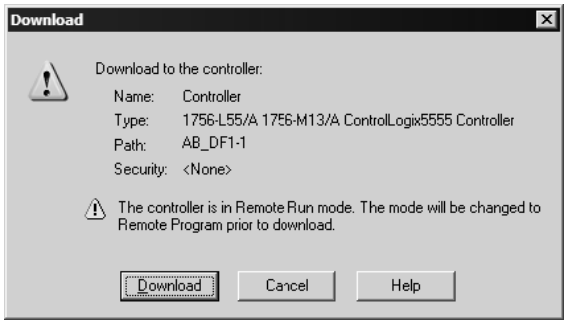
Note: If the auto-configuration procedure fails, verify that the cables are connected correctly between the processor and the serial port on your computer, and then try again. If you are still unable to auto-configure the port, refer to your RSLinx documentation for further troubleshooting steps.

1.7 Download the Sample Program to the Processor

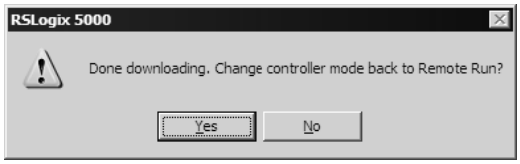
Note: The key switch on the front of the ControlLogix module must be in the REM position.

To download the sample program from RSLogix 5000 to the ControlLogix processor

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.

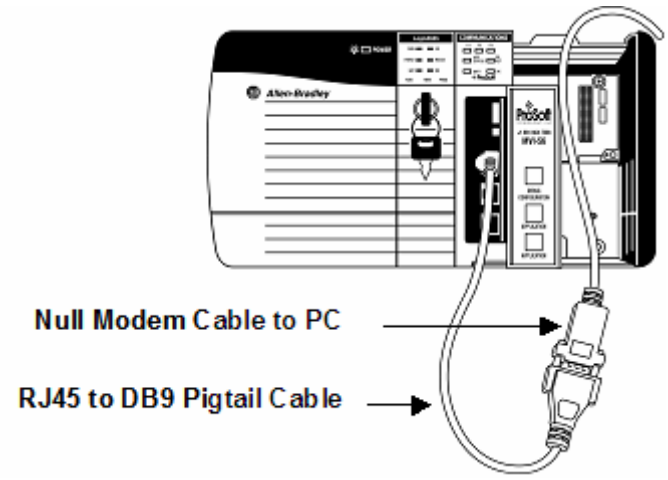


Note: If you receive an error message during these steps, refer to your RSLogix documentation to interpret and correct the error.

1.8 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.

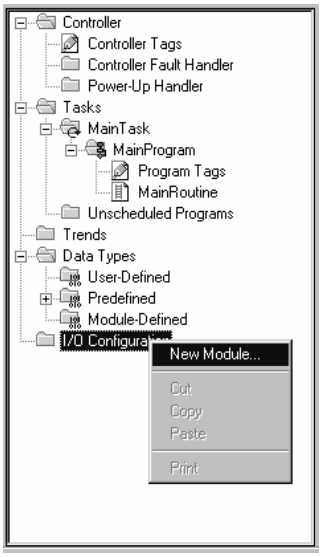
Note: The RSLogix 5000 software must be in "offline" mode to add the module to a project.

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.

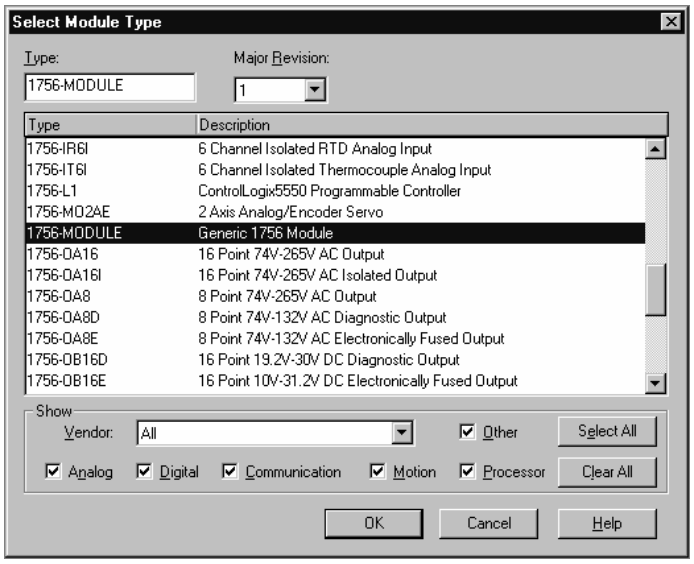
Note: If you are installing this module in an existing application, you can copy the necessary elements from the example ladder logic into your application.

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

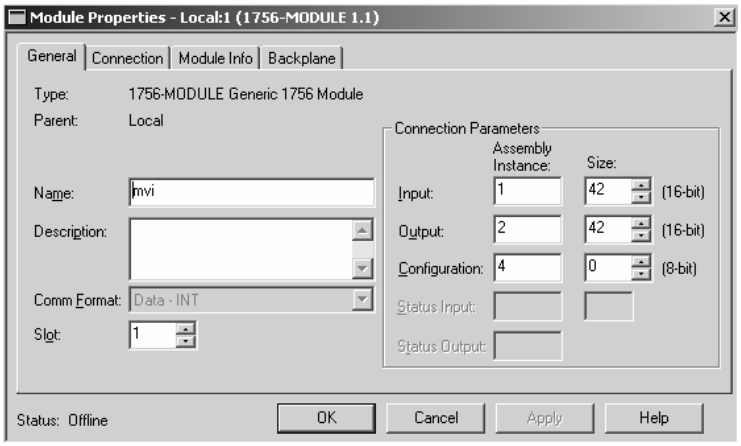
First, define the module to the system. Right-click the mouse button on the I/O Configuration option in the Controller Organization window to display a pop-up menu. Select the **New Module** option from the I/O Configuration menu.



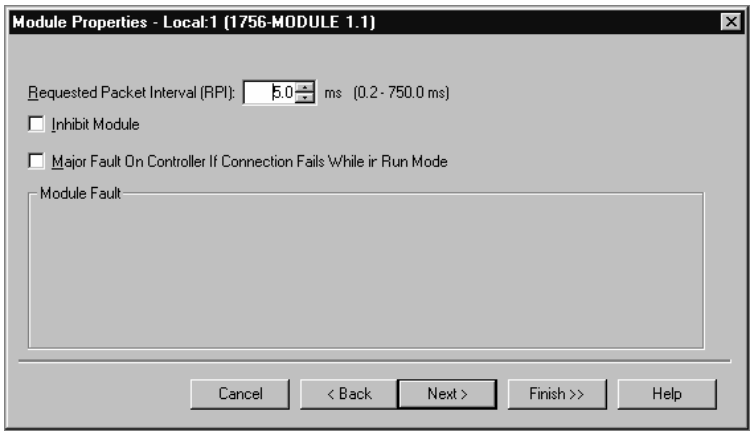
This action opens the following dialog box.



Select the **1756-Module (Generic 1756 Module)** from the list and click the **OK** button. The following dialog box appears:

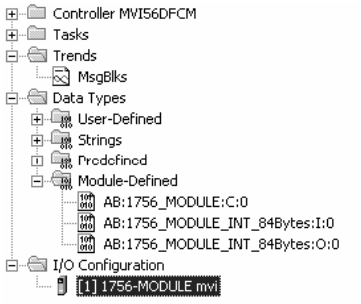


Fill in the dialog box as shown adjusting the Name, Description and Slot options for your application. You must select the **Comm Format** as **Data - INT** in the dialog box. Failure to set the **Assembly Instance** and **Size** values correctly will result in a module that will not communicate over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

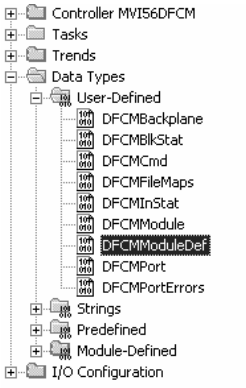


Select the Request Packet Interval value for scanning the I/O on the module. This value represents the minimum frequency that the module will handle scheduled events. This value should not be set to less than 1 millisecond. Values between 1 and 10 milliseconds should work with most applications.

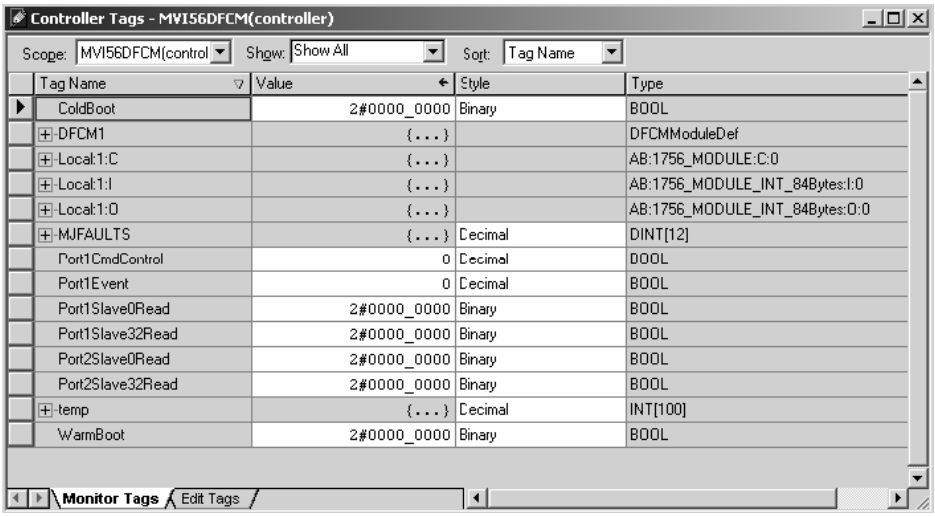
After completing the module setup, the Controller Organization window displays the module's presence. The data required for the module will be defined to the application, and objects will be allocated in the Controller Tags data area.



Next, 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 in the following example:

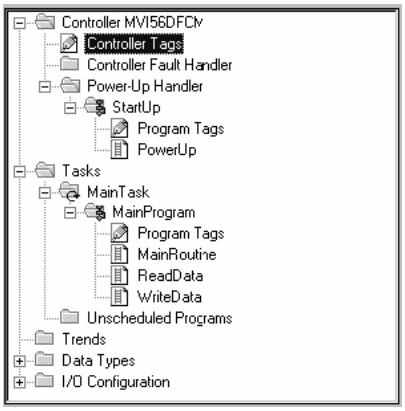


Next, define the data tag to be used to interface with the module and the ladder logic. Open the Controller Tags Edit Tags dialog box and enter the values shown in the following example. The MVI56-DFCMR module is defined in the example as DFCM1. You can set the tag name to any valid tag name you desire. 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 DFCM1 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 the example ladder logic is used, adjust the ladder to fit the application. When the ladder example is not used, copy the ladder logic shown in the Controller Organization window below to the application.



The module is now set up and ready to be used with your application. Insert the module in the rack and attach the DF1 serial communication cables. Download the new application to the controller and places the processor in run mode. If all the configuration parameters are set correctly and the module is attached to a DF1 network, the module's Application LED should remain on and the backplane activity LED (BP ACT) should blink very rapidly. Refer to the Diagnostics and Troubleshooting section if you encounter errors.

Attach a computer or terminal to Port 1 on the module and look at the status of the module using the Configuration/Debug Menu in the module.

2.1 Modifying the Configuration Data

In order for the MVI56-DFCMR 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.

Block Range	Functional Modes Affected	Name	Description
9000	Data Transfer	Backplane Configuration	This section of the configuration data contains the module configuration data that defines the data transfer between the module and the ControlLogix processor.
9001	Master and Slave	Port 1 Configuration	This section defines the characteristics of the DF1 serial communication port 1 on the module. These parameters must be set correctly for proper module operation.

Block Range	Functional Modes Affected	Name	Description
9002	Master and Slave	Port 2 Configuration	This section defines the characteristics of the DF1 serial communication port 2 on the module. These parameters must be set correctly for proper module operation.
6000 to 6033	Master	Port 1 Command List	If the module's Master Mode functionality is to be supported on a port, the Master Command List must be set up.
6100 to 6133	Master	Port 2 Command List	If the module's Master Mode functionality is to be supported on a port, the Master Command List must be set up.
7000 to 7004	Master and Slave	Port 1 Override File Map	Overrides default file maps. Used when the module is a DF1 slave device.
7100 to 7104	Master and Slave	Port 2 Override File Map	Overrides default file maps

Refer to the Modifying the Module Configuration section for a description of the configuration of the module. The MVI56-DFCMR 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 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. A master port on the module may poll a slave for the data or a slave port could receive the data from a remote master unit. 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 7800 (N46:0) 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.

2.2 Configuration Data

This section contains listings of the MVI56-DFCMR 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. Additionally, this section contains the miscellaneous status data and command control database layout.

2.2.1 Backplane Setup

Register	Content	Description
5000	Write Start Reg	Not used in this version of the software
5001	Write Reg Count	Not used in this version of the software
5002	Read Start Reg	Not used in this version of the software
5003	Read Reg Count	Not used in this version of the software
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 0, the communication ports will continue to operate under all conditions. If the value is set larger than 0 (1 to 65535), communications will cease if the specified number of failures occur.
5005	Error Status Pointer	This parameter specifies the register location in the module's database where module status data will be stored. If a value less than 0 is entered, the data will not be stored in the database. If the value specified is in the range of 0 to 4940, the data will be placed in the user data area.
5006 to 5009	Spare	

2.2.2 Port 1 Setup

Register	Content	Description
5010	Enable	This parameter is used to define if this port will be utilized. If the parameter is set to 0, the port is disabled. A value of 1 will enable the port.
5011	Type	This parameter defines if the port will emulate a master or slave device. Enter 0 to emulate a master device and 1 to emulate a slave device.
5012	Local Station ID	This parameter specifies the local station ID for all DF1 messages sent from this master port. A value of 255 is not permitted as this is the broadcast address. Enter a value in the range of 0 to 254.
5013	Protocol	0=full duplex, 1=half-duplex
5014	Termination Type	This parameter specifies the error checking for all DF1 messages. 0=BCC, 1=CRC

<table>
<tr><th>Register</th><th>Content</th><th>Description</th></tr>
<tr><td>5015</td><td>Baud Rate</td><td>This is the baud rate to be used on the port. Enter the baud rate as a value.
<table>
<tr><th>Baud Rate</th><th>Parameter Value</th></tr>
<tr><td>110</td><td>110</td></tr>
<tr><td>150</td><td>150</td></tr>
<tr><td>300</td><td>300</td></tr>
<tr><td>600</td><td>600</td></tr>
<tr><td>1200</td><td>12 or 1200</td></tr>
<tr><td>2400</td><td>24 or 2400</td></tr>
<tr><td>4800</td><td>48 or 4800</td></tr>
<tr><td>9600</td><td>96 or 9600</td></tr>
<tr><td>14,400</td><td>14, 114 or 14400</td></tr>
<tr><td>19,200</td><td>19, 192 or 19200</td></tr>
<tr><td>28,800</td><td>28, 288 or 28800</td></tr>
<tr><td>38,400</td><td>38, 384 or 38400</td></tr>
<tr><td>57,600</td><td>57 or 576</td></tr>
<tr><td>115,200</td><td>115 or 1152</td></tr>
</table></td></tr>
<tr><td>5016</td><td>Parity</td><td>This is the Parity code to be used for the port. The coded values are as follows: 0=None, 1=Odd, 2=Even.</td></tr>
<tr><td>5017</td><td>Data Bits</td><td>This parameter sets the number of data bits for each word used by the protocol. Enter a value in the range of 5 to 8.</td></tr>
<tr><td>5018</td><td>Stop Bits</td><td>This parameter sets the number of stop bits to be used with each data value sent. Enter a value of 1 or 2.</td></tr>
<tr><td>5019</td><td>Minimum Response Delay</td><td>This parameter sets the number of milliseconds to wait before a response message is sent out of the port. This parameter is required when interfacing to a slow responding device. Enter a value in the range of 0 to 65535.</td></tr>
<tr><td>5020</td><td>RTS On Delay</td><td>This parameter sets the number of milliseconds to delay after RTS is asserted before the data will be transmitted. Enter a value in the range of 0 to 65535.</td></tr>
<tr><td>5021</td><td>RTS Off Delay</td><td>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. Enter a value in the range of 0 to 65535.</td></tr>
<tr><td>5022</td><td>Use CTS Line</td><td>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. Normally, this parameter is required when half-duplex modems are used for communication (2-wire).</td></tr>
</table>

Register	Content	Description
5023	ENQ Delay	This parameter specifies the number of milliseconds to wait after a DLE-ACK is received from a slave using half-duplex mode before the DLE-ENQ request is made for data. Enter a value in the range of 0 to 65535.
5024	Command Count	This parameter specifies the number of commands to be processed for the port. Enter a value of 0 to 100.
5025	Minimum Command Delay	This parameter specifies the number of milliseconds to wait between the initial issuance of a command. This parameter can be used to delay all commands sent to slaves to avoid "flooding" commands on the network. This parameter does not affect retries of a command as they will be issued when failure is recognized. Enter a value in the range of 0 to 65535.
5026	Command Error Pointer	This parameter sets the address in the internal DF1 database where the command error data will be placed. If the value is set to -1, the data will not be transferred to the database. Enter a value of 0 to 4999.
5027	Response Timeout	This parameter represents the message response timeout period in 1-ms increments. This is the time that a port configured as a master will wait before re-transmitting a command if no response is received from the addressed slave. The value is set depending upon the communication network used and the expected response time of the slowest device on the network.
5028	Retry Count	This parameter specifies the number of times a command will be retried if it fails. Enter a value in the range of 0 to 10.
5029	Error Delay Count	This parameter specifies the number of polls to be skipped on the slave before trying to re-establish communications. After the slave fails to respond, the master will skip commands to be sent to the slave the number of times entered in this parameter. Enter a value in the range of 0 to 65535.
5030	Slave List Pointer	This parameter specifies the starting address in the virtual database where the 256 slave status values will be written. If the parameter is set to -1, the slave data will not be placed in the database. Enter a value in the range of -1 to 4743.
5031	Slave List Frequency	Not used in this version of the software
5032	First File	This parameter is used when a request for a file is received on the communication port. This field is required when responding to PLC5 and SLC DF1 commands. Use this parameter to define the virtual file(s) to be simulated on the module. This value is fixed at 7 for N7.
5033	File Size	This parameter is used to specify the size of each file to be simulated on the module. All files simulated are defined to have the same assigned size. This value is fixed at 200.

Register	Content	Description
5034	File Offset	This parameter sets the database register location of the first element in the first file simulated in the module. All offsets in the first file and subsequent files will be computed using the address specified. This value is fixed at 0. Therefore, the module assumes N7:0 starts at the database offset of 0 and each file is 200 words in size.
5035	Map Count	Number of file map overrides.
5036 to 5039	Spare	

2.2.3 Port 2 Setup

Register	Content	Description
5040	Enable	This parameter is used to define if this port will be utilized. If the parameter is set to 0, the port is disabled. A value of 1 will enable the port.
5041	Type	This parameter defines if the port will emulate a master or slave device. Enter 0 to emulate a master device and 1 to emulate a slave device.
5042	Local Station ID	This parameter specifies the local station ID for all DF1 messages sent from this master port. A value of 255 is not permitted as this is the broadcast address. Enter a value in the range of 0 to 254.
5043	Protocol	0=full duplex, 1=half-duplex
5044	Termination Type	This parameter specifies the error checking for all DF1 messages. 0=BCC, 1=CRC
5045	Baud Rate	This is the baud rate to be used on the port. Enter the baud rate as a value.
5046	Parity	This is the Parity code to be used for the port. The coded values are as follows: 0=None, 1=Odd, 2=Even.

Baud Rate	Parameter Value
110	110
150	150
300	300
600	600
1200	12 or 1200
2400	24 or 2400
4800	48 or 4800
9600	96 or 9600
14,400	14, 114 or 14400
19,200	19, 192 or 19200
28,800	28, 288 or 28800
38,400	38, 384 or 38400
57,600	57 or 576
115,200	115 or 1152

Register	Content	Description
5047	Data Bits	This parameter sets the number of data bits for each word used by the protocol. Enter a value in the range of 5 to 8.
5048	Stop Bits	This parameter sets the number of stop bits to be used with each data value sent. Enter a value of 1 or 2.
5049	Minimum Response Delay	This parameter sets the number of milliseconds to wait before a response message is sent out of the port. This parameter is required when interfacing to a slow responding device. Enter a value in the range of 0 to 65535.
5050	RTS On Delay	This parameter sets the number of milliseconds to delay after RTS is asserted before the data will be transmitted. Enter a value in the range of 0 to 65535.
5051	RTS Off Delay	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. Enter a value in the range of 0 to 65535.
5052	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. Normally, this parameter is required when half-duplex modems are used for communication (2-wire).
5053	ENQ Delay	This parameter specifies the number of milliseconds to wait after a DLE-ACK is received from a slave using half-duplex mode before the DLE-ENQ request is made for data. Enter a value in the range of 0 to 65535.
5054	Command Count	This parameter specifies the number of commands to be processed for the port. Enter a value of 0 to 100.
5055	Minimum Command Delay	This parameter specifies the number of milliseconds to wait between the initial issuance of a command. This parameter can be used to delay all commands sent to slaves to avoid "flooding" commands on the network. This parameter does not affect retries of a command as they will be issued when failure is recognized. Enter a value in the range of 0 to 65535.
5056	Command Error Pointer	This parameter sets the address in the internal DF1 database where the command error data will be placed. If the value is set to -1, the data will not be transferred to the database. Enter a value of 0 to 4999.
5057	Response Timeout	This parameter represents the message response timeout period in 1-ms increments. This is the time that a port configured as a master will wait before re-transmitting a command if no response is received from the addressed slave. The value is set depending upon the communication network used and the expected response time of the slowest device on the network.

Register	Content	Description
5058	Retry Count	This parameter specifies the number of times a command will be retried if it fails. Enter a value in the range of 0 to 10.
5059	Error Delay Count	This parameter specifies the number of polls to be skipped on the slave before trying to re-establish communications. After the slave fails to respond, the master will skip commands to be sent to the slave the number of times entered in this parameter. Enter a value in the range of 0 to 65535.
5060	Slave List Pointer	This parameter specifies the starting address in the virtual database where the 256 slave status values will be written. If the parameter is set to -1, the slave data will not be placed in the database. Enter a value in the range of -1 to 4743.
5061	Slave List Frequency	Not used in this version of the software
5062	First File	This parameter is used when a request for a file is received on the communication port. This field is required when responding to PLC5 and SLC DF1 commands. Use this parameter to define the virtual file(s) to be simulated on the module. This value is fixed at 7 for N7.
5063	File Size	This parameter is used to specify the size of each file to be simulated on the module. All files simulated are defined to have the same assigned size. This value is fixed at 200.
5064	File Offset	This parameter sets the database register location of the first element in the first file simulated in the module. All offsets in the first file and subsequent files will be computed using the address specified. This value is fixed at 0. Therefore, the module assumes N7:0 starts at the database offset of 0 and each file is 200 words in size.
5065	Map Count	Number of file map overrides.
5066 to 5069	Spare	

2.2.4 Port 1 Commands

Register	Content	Description
5200 to 5211	Command # 1	This set of registers contains the parameters for the first command in the master command list. The structure of this data area is as described in the data object section of the documentation.
5212 to 5223	Command # 2	Command #2 data set
-		
6388 to 6399	Command # 100	Command #100 data set

2.2.5 Port 2 Commands

Register	Content	Description
6400 to 6411	Command # 1	This set of registers contains the parameters for the first command in the master command list. The structure of this data area is as described in the data object section of the documentation.
6412 to 6423	Command # 2	Command #2 data set
-		
7588 to 7599	Command # 100	Command #100 data set

2.2.6 Miscellaneous Status

Register	Content	Description
7600	Program Scan Count	This value is incremented each time a complete program cycle occurs in the module.
7601 to 7602	Product Code	These two registers contain the product code of "DFCM"
7603 to 7604	Product Version	These two registers contain the product version for the currently running software.
7605 to 7606	Operating System	These two registers contain the month and year values for the program operating system.
7607 to 7608	Run Number	These two registers contain the run number value for the currently running software.
7609	Port 1 Command List Requests	This field contains the number of requests made from this port to slave devices on the network.
7610	Port 1 Command List Response	This field contains the number of slave response messages received on the port.
7611	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.
7612	Port 1 Requests	This field contains the total number of messages sent out the port.
7613	Port 1 Responses	This field contains the total number of messages received on the port.
7614	Port 1 Errors Sent	This field contains the total number of message errors sent out the port.
7615	Port 1 Errors Received	This field contains the total number of message errors received on the port.
7616	Port 2 Command List Requests	This field contains the number of requests made from this port to slave devices on the network.
7617	Port 2 Command List Response	This field contains the number of slave response messages received on the port.
7618	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.
7619	Port 2 Requests	This field contains the total number of messages sent out the port.
7620	Port 2 Responses	This field contains the total number of messages received on the port.
7621	Port 2 Errors Sent	This field contains the total number of message errors sent out the port.
7622	Port 2 Errors Received	This field contains the total number of message errors received on the port.

Register	Content	Description
7623	Read Block Count	This field contains the total number of read blocks transferred from the module to the processor.
7624	Write Block Count	This field contains the total number of write blocks transferred from the processor to the module.
7625	Parse Block Count	This field contains the total number of blocks successfully parsed that were received from the processor.
7626	Command Event Block Count	This field contains the total number of command event blocks received from the processor.
7627	Command Block Count	This field contains the total number of command blocks received from the processor.
7628	Error Block Count	This field contains the total number of block errors recognized by the module.
7629	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.
7630	Port 1 Last Error/Index	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.
7631	Port 2 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.
7632	Port 2 Last Error/Index	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.

2.2.7 Command Control

Register	Content	Description
7800	Command Code	Enter one of the valid control command codes in this register to control the module (9997, 9998 or 9999). Refer to Command Control Blocks (page 34, page 71, page 70, page 71, page 71).
7801 to 7999	Command Data	Reserved for future use

2.2.8 Port 1 Override File Maps

Register	Content	Description
8000 to 8003	File Map #1	This set of registers contains the first override file map for the slave port.
8004 to 8007	File Map #2	This set of registers contains the second override file map for the slave port.
-		
8196 to 8199	Command # 50	This set of registers contains the last override file map for the slave port.

2.2.9 Port 2 Override File Maps

Register	Content	Description
8200 to 8203	File Map #1	This set of registers contains the first override file map for the slave port.
8204 to 8207	File Map #2	This set of registers contains the second override file map for the slave port.
-		
8396 to 8399	Command # 50	This set of registers contains the last override file map for the slave port.

3 Ladder Logic

In This Chapter

Ladder logic is required for application of the MVI56-DFCMR 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.

3.1 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.

3.1.1 Data Transfer Parameters (DFCMModule)

Name	Data Type	Description
WriteStartReg	INT	Start reg to transfer from PLC to module
WriteRegCnt	INT	Number of registers to write from PLC
ReadStartReg	INT	Start reg to transfer from module to PLC
ReadRegCnt	INT	Number of registers to transfer from module
BPFail	INT	Determines module operation if BP fails 0=continue,>0=number of retries before comm shutdown
ErrStatPtr	INT	Internal DB start register for status data (-1=Ignore)

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/40) + if(MOD(RegCnt,40), 1,0)

If the register count is evenly divisible by 40, the number of blocks is easy to compute and the ladder is much simpler to write. If the number is not evenly divisible by 40, special handling of the last block of data must be developed, as it must transfer less than 40 words. It is recommended that the count values always be set to values evenly divisible by 40.

The BPFail parameter determines if the module should continue communicating on the DF1 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 with the network. For example, if you enter a value of 10 for the parameter, the module will stop all DF1 communications if 10 successive backplane errors are recognized. When a successful transfer is recognized, the module will resume communications on the network.

The ErrStatPtr parameter defines the location in the module's database where the error/status data will be stored. If the value is set to -1, the data will not be stored in the user data area. A value between 0 and 4939 will cause the module's program to store the data at the specified location.

3.1.2 DF1 Master Commands (DFCMCmd)

This object defines the parameters for each command in the master command list. The **DFCMModuleDef** object contains an array of these objects that define the complete list for each port. The definition of each parameter required for each command is given below:

Name	Data Type	Description
Enable	INT	0=Disable,1=Continuous,2=Event Command
IntAddress	INT	Module's internal address associated with the command
PollInt	INT	Minimum number of seconds between issuance of command (0-65535 Sec)
Count	INT	Number of registers associated with the command
Swap	INT	Swap code used with command
Node	INT	Address of device to interface with on the DF1 network
Func	INT	DF1 function code as defined for MVI56-DFCM module
Parameter_1	INT	First parameter for function
Parameter_2	INT	Second parameter for function
Parameter_3	INT	Third parameter for function
Parameter_4	INT	Fourth parameter for function

Enable: This parameter defines if the command will be executed or will be disregarded. The following values are valid: 0=Disables the command and it will not execute. 1=The command will be considered for execution each scan of the command list and will be controlled by the **PollInt** parameter. And 2=The command will only execute if the data associated with the command has changed since the command was last issued. This option is only available for write commands.

IntAddress: This parameter specifies the starting internal register address to be associated with the command. Valid entry for this parameter is 0 to 4999.

PollInt: This parameter defines the minimum number of seconds to wait between the execution of continuous commands (Enable=1). This poll interval command can be used to lighten the communications load on a busy network. Valid entry for this parameter is 0 to 65535.

Count: This parameter defines the number of registers to be considered by the command. Valid entry for this parameter is 1 to 100.

Swap: This parameter specifies if the data used in the command must be altered when reading data from a node on the network. Values that can be assigned are as follows: 0=no swapping of data, 1=swap word values, 2=swap word and byte values and 3=swap byte values. This option is used when interfacing the module with ASCII and floating-point data on other devices.

Node: This parameter assigns the DF1 slave node address for the module to reach with the command on the network. This parameter can be assigned values from 0 to 255.

Func: This parameter specifies the function to be performed by the command. Valid entries are those defined in the **DF1 Command Set for ProSoft Technology. Communication Modules** found at the end of this manual.

Parameter_1 to Parameter_4: These are the parameters required for the selected function. Each command has its own unique set of one or more parameters. Refer to the DF1 Command Set document for a complete list of command parameters.

3.1.3 DF1 Port Parameters (DFCMPort)

Name	Data Type	Description
Enabled	INT	0=Port Disabled,1=Port Enabled
Type	INT	0=Master, 1=Slave
StationID	INT	0-254 local station id for device emulated on this port
Protocol	INT	0=Full-duplex, 1=half-duplex
TerminationType	INT	0=BCC, 1=CRC
Baudrate	INT	Baudrate for port (110 to 115.2K)
Parity	INT	0=None, 1=Odd, 2=Even, 3=Mark, 4=Space
DataBits	INT	5 to 8 data bits
StopBits	INT	1 or 2 stop bits
MinResp	INT	0-65535 mSec minimum time before response to request

Name	Data Type	Description
RTSOn	INT	0-65535 mSec delay before data
RTSOff	INT	0-65535 mSec delay after data
UseCTS	INT	0=No, 1=Yes to use CTS modem line
ENQDelay	INT	0-65535 delay after ACK before sending ENQ request (half-duplex)
CmdCount	INT	Command list count (Master)
MinCmdDelay	INT	0-65535 mSec minimum time between each command (Master)
CmdErrPtr	INT	Internal DB location to place command error list (Master)
RespTO	INT	0-65535 mSec response timeout for command (Master)
Retry_Count	INT	Retry count for failed request (Master)
ErrorDelayCntr	INT	0-65535 Command cycle count if error (Master)
SlaveListPtr	INT	-1=No slave list in database, 0-4743=Reg start location for data (Master)
SlaveListFreq	INT	0=No slave list in database, 1-65535=cycle count for update (Master)
FirstFile	INT	0-100 First file to emulate in database (Slave)
FileSize	INT	0-1000 size of each file to emulate (Slave)
FileOffset	INT	0-4999 register offset into database where file emulation starts (Slave)
DataFileMapCnt	INT	0-50 Data File Mapping Count to Re-Direct data Reads and Writes (Slave)

This object defines the parameters for the operation of each of the DF1 ports on the module. Refer to Configuration Data Definition (page 27) for the definition of each parameter.

3.1.4 File Map Override

This object defines specific data files that will override the default map file list. When the module is configured as a DF1 slave, there may be instances where you cannot use the default addressing shown in section 2.3 of this manual.

The File Map Override allows you to define memory (DBReg), specific file number (FileNo), and element number (Element). The length will determine how many registers will follow.

For example, if you must have registers 105 to 108 of module memory set up as addresses N45:32 to N45:35, you would set the following parameters:

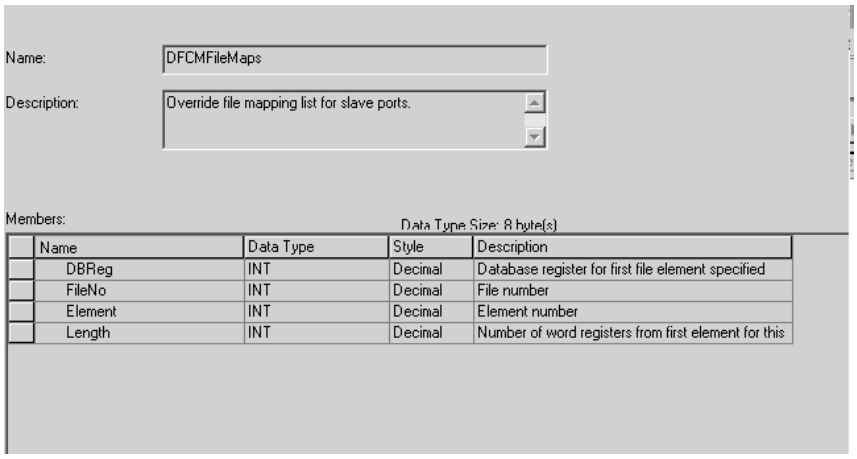
DBReg = 105

FileNo = 45

Element = 32

Length = 4

Note: You must place a value in the DFCM1.Port[x].MapCnt in order for this parameter to be used. If the DFCM1.Port[x].MapCnt is set to a value of 0, default addressing will be used.



3.2 Module Data Object (DFCMModuleDef)

All status and variable data related to the MVI56-DFCMR 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 object has the following structure.

Name	Data Type	Description
ModDef	DFCMModule	Backplane operation characteristics
Port	DFCMPort[2]	MB settings for each port
P1Cmd	DFCMCmd[100]	Master Command List for Port 1
P2Cmd	DFCMCmd[100]	Master Command List for Port 2
InStat	DFCMInStat	Status information in each read block
ReadData	INT[600]	Data read from module
WriteData	INT[600]	Data to write to module
BP	DFCMBackplane	Data to handle backplane logic
P1Slaves	INT[256]	Port 1 slave status values
P2Slaves	INT[256]	Port 2 slave status values
P1DataFileMap	DFCMDataFileMap[50]	Port 1 Data File Mappings
P2DataFileMap	DFCMDataFileMap[50]	Port 2 Data File Mappings

This object contains objects that define variables to be used with the module and status data related to the module. Each of these object types is discussed in the following topics of the document.

3.3 Status Object (DFCMInStat)

This object views the status of the module. The **DFCMInStat** 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".

Name	Data Type	Description
PassCnt	INT	Program cycle counter
Product	INT[2]	Product Name
Rev	INT[2]	Revision Level Number
OP	INT[2]	Operating Level Number
Run	INT[2]	Run Number
PrtErrs	DFCMPortErrors[2]	Port error statistics
CmdReq	INT	Total number of command list requests sent
CmdResp	INT	Total number of command list responses received
CmdErr	INT	Total number of command list errors
Requests	INT	Total number of requests for port
Responses	INT	Total number of responses for port
ErrSent	INT	Total number of errors sent
ErrRec	INT	Total number of errors received
BlkErrs	DFCMBlkStat	Block transfer statistics
Read	INT	Total number of read block transfers
Write	INT	Total number of write block transfers
Parse	INT	Total number of blocks parsed
Event	INT	Total number of event blocks received
Cmd	INT	Total number of command blocks received
Err	INT	Total number of block transfer errors
P1CErr	INT	Port 1 current error/index
P1LErr	INT	Port 1 last error/index
P2CErr	INT	Port 2 current error/index
P2LErr	INT	Port 2 current error/index

Refer to Status Data Definition (page 85) for a complete listing of the data stored in this object.

3.4 User Data Objects

These objects hold data to be transferred between the processor and the MVI56-DFCMR module. The user data is the read and write data transferred between the processor and the module as "pages" of data up to 40 words long.

	ReadData	INT[600]	Decimal	Data read from module
	WriteData	INT[600]	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 **DFCMModule** object. For ease of use, this array should be dimensioned as an even increment of 40 words. This data is paged up to 40 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 **DFCMModule** object. For ease of use, this array should be dimensioned as even increments of 40 words. This data is paged up to 40 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.

3.5 Slave Polling Control and Status

Two arrays are allocated in the module's primary object to hold the polling status of each slave on the master ports. This status data can be used to determine which slaves are currently active on the port, are in communication error or have their polling suspended and disabled. Ladder logic in the processor can be written to monitor and control the status of each slave on a master port. The objects used are displayed below:

P1Slaves	INT[256]	Decimal	Port 1 slave status values
P2Slaves	INT[256]	Decimal	Port 2 slave status values

Using special blocks, the processor can request the current data for the slaves. Through the use of other blocks, the processor can enable or disable the polling of selected slaves.

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-DFCMR module returns a 29-word Status Data block that can be used to determine the module's operating status. 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.

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:

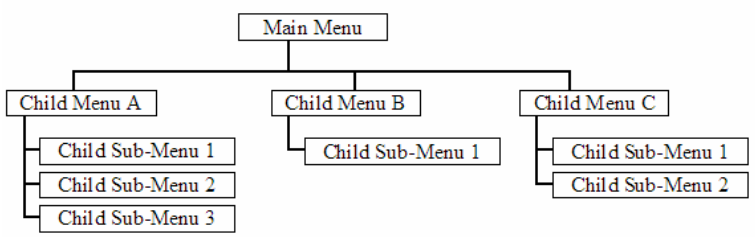
1. Verify that the null modem cable is connected properly between your computer's serial port and the module. A regular serial cable will not work.
2. Verify that RSLinx is not controlling the COM port. Refer to Disabling the RSLinx Driver for the Com Port on the PC (page 79).
3. Verify that your communication 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.

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][/]**.

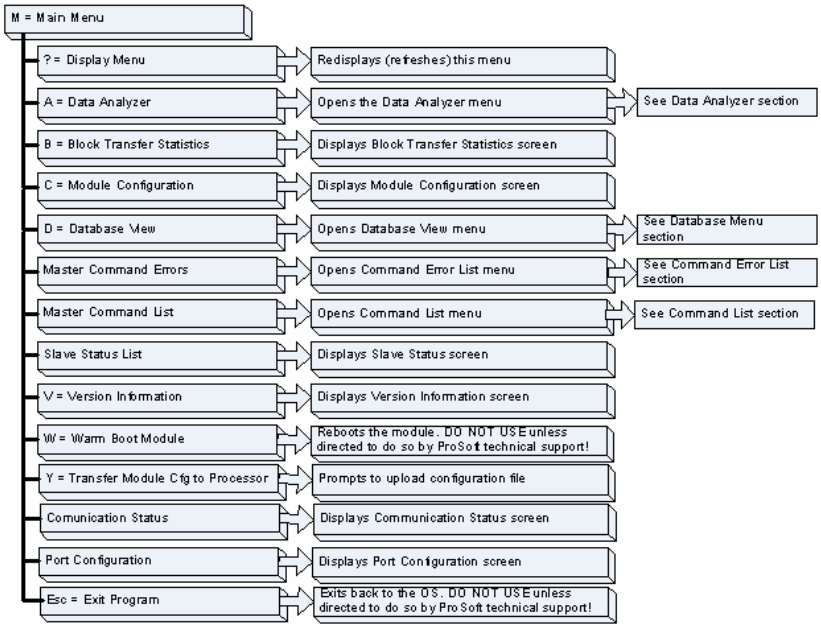
Also, take care to distinguish capital letter **[I]** from lower case letter **[l]** (L) and number **[1]**; likewise for capital letter **[O]** and number **[0]**. Although these characters look nearly the same on the screen, they perform different actions on the module.

4.1.5 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:

```
DF1 MASTER/SLAVE COMMUNICATION MODULE
  ?=Display Menu
  A=Data Analyzer
  B=Block Transfer Statistics
  C=Module Configuration
  D=Database View
  Master Command Errors : E=Port 1   F=Port 2
  Master Command List   : I=Port 1   J=Port 2
  Slave Status List     : O=Port 1   P=Port 2
  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.

Opening the Command Error List Menu

Press **[I]** to open the Command Error List. This list consists of multiple pages of command list error/status data. Press **[?]** to view a list of commands available on this menu.

Opening the Command List Menu

Press **[L]** to open the Command List menu. Use this command to view the configured command list for the module.

Viewing the Slave Status List (Port 1 and 2)

Press **[O]** (port 1) or **[P]** (port 2) to view the 256 slave status values associated with the ports. The slave status values are defined as follows:

0 = slave is not used,

1 = slave being actively polled,

2 = slave suspended and

3 = slave disabled.

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 Communication Status

Press **[1]** to view the communication status and statistics of the DF1 Network for the module's node address. This command is useful for troubleshooting purposes.

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.

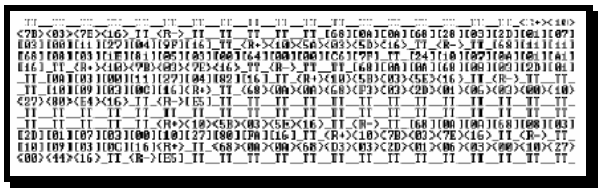
4.1.6 Data Analyzer

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

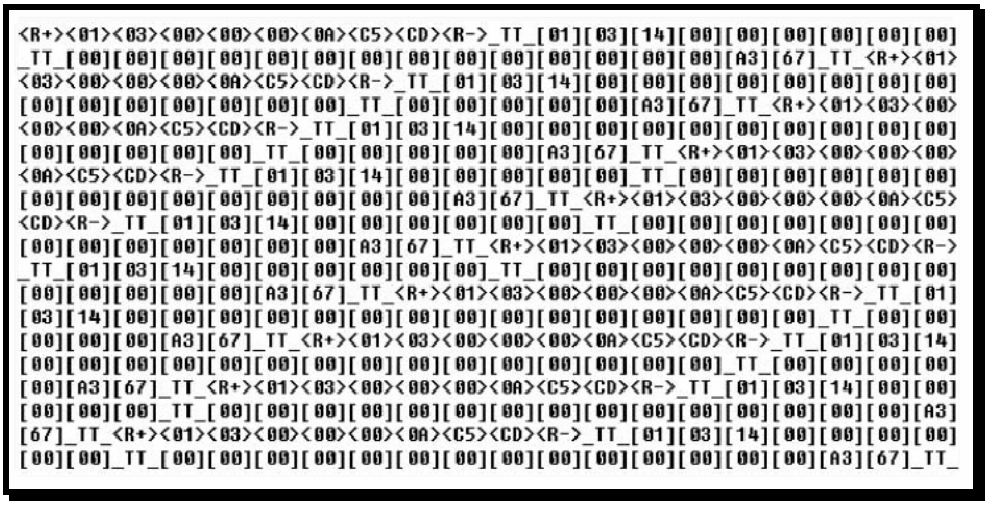
Press **[H]** to display the data on the current page in hexadecimal format.

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.

Character	Definition
<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:

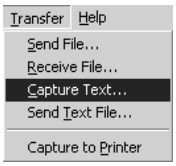
```
DATA ANALYZER VIEW MENU
 ?=Display Menu
 1=Select Port 1
 2=Select Port 2
 5=1 mSec Ticks
 6=5 mSec Ticks
 7=10 mSec Ticks
 8=50 mSec Ticks
 9=100 mSec Ticks
 0=No mSec Ticks
 H=Hex Format
 A=ASCII Format
 B=Start
 S=Stop
 M=Main Menu

 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.

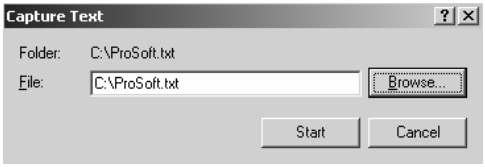
For most applications, HEX is the best format to view the data, and this does include ASCII based messages (because some characters will not display on HyperTerminal and by capturing the data in HEX, we can figure out what the corresponding ASCII characters are supposed to be).

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.

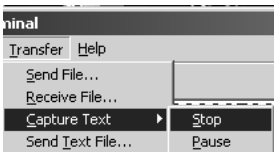
All characters shown in <> brackets are characters being sent out by the module.

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.

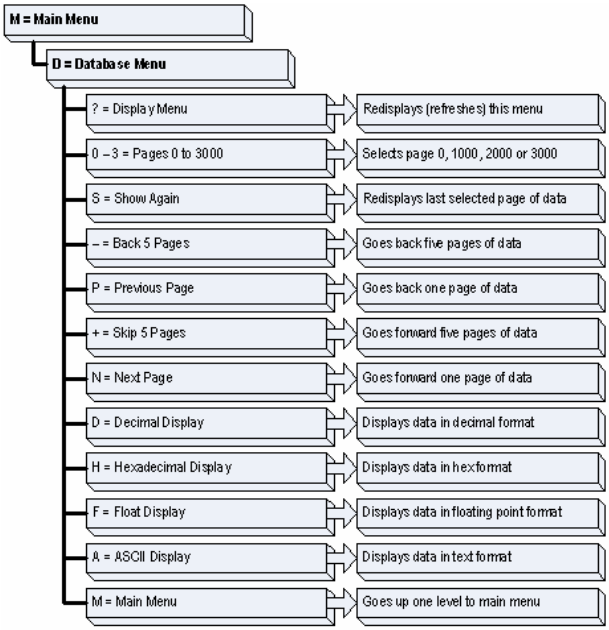
When you have captured the data you want to save, open the Transfer menu and choose Capture Text. On the secondary menu, choose Stop.



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

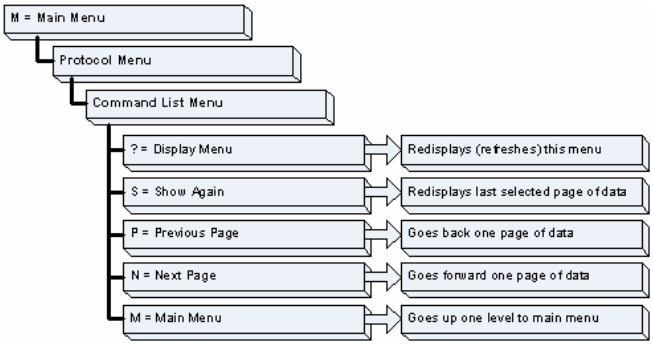
Press **[A]** to display the data on the current page in ASCII format. This is useful for regions of the database that contain ASCII data.

Returning to the Main Menu

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

4.1.9 Master Command Error List Menu

Use this menu to view the command error list for the module. Press **[?]** to view a list of commands available on this menu.



Redisplaying the Current Page

Press **[S]** to display the current page of data.

Viewing the Previous 20 Commands

Press **[-]** to display data for the previous 20 commands.

Viewing the Previous Page of Commands

Press **[P]** to display the previous page of commands.

Viewing the Next 20 Commands

Press **[+]** to display data for the next 20 commands.

Viewing the Next Page of Commands

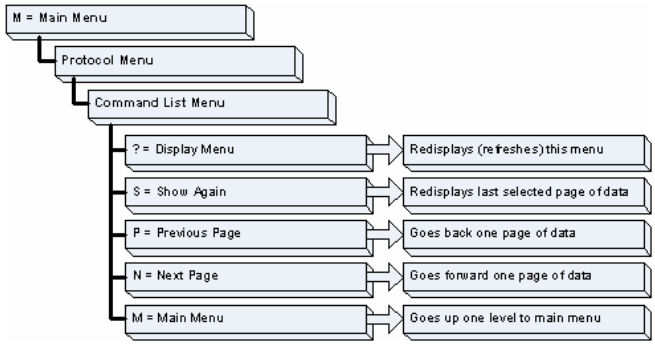
Press **[N]** to display the next page of commands.

Returning to the Main Menu

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

4.1.10 Master Command List Menu

Use this menu to view the command list for the module. Press **[?]** to view a list of commands available on this menu.



Redisplaying the Current Page

Press **[S]** to display the current page of data.

Viewing the Previous 50 Commands

Press **[-]** to view the previous 50 commands.

Viewing the Previous Page of Commands

Press **[P]** to display the previous page of commands.

Viewing the Next 50 Commands

Press **[+]** to view the next 50 commands from the master command list.

Viewing the Next Page of Commands

Press **[N]** to display the next page of commands.

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:

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 DF1 network on Port 1.
		Off	No data is being transferred on the port.
P2	Green	On	Data is being transferred between the module and the DF1 network on Port 2.
		Off	No data is being transferred on the port.
APP	Amber	On	The MVI56-DFCMR is working normally.
		Off	The MVI56-DFCMR module program has recognized a communication error on one of its 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 will be red and the APP and BP ACT LEDs are on.

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.

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 shown in the following table.

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 40	Spare	36
41	-2 or -3	1

The bits in each configuration word are shown below. The module configuration error word has the following definition:

Bit	Description	Value
0	Read block start value is greater than the database size.	0x0001
1	Read block start value is less than zero.	0x0002
2	Read block count value is less than zero.	0x0004
3	Read block count + start is greater than the database size.	0x0008
4	Write block start value is greater that the database size.	0x0010
5	Write block start value is less than zero.	0x0020
6	Write block count value is less than zero.	0x0040
7	Write block count + start is greater than the database size.	0x0080
8		0x0100
9		0x0200
10		0x0400
11		0x0800
12		0x1000
13		0x2000
14		0x4000
15		0x8000

The port configuration error words have the following definitions:

Bit	Description	Value
0	Type code is not valid. Enter a value from 0 (master) to 1 (slave).	0x0001
1	Protocol parameter is not valid.	0x0002
2	Termination type parameter is not valid.	0x0004
3	Baud rate parameter is not valid.	0x0008
4	Parity parameter is not valid.	0x0010
5	Data bits parameter is not valid.	0x0020
6	Stop bits parameter is not valid.	0x0040
7	Command count parameter is not valid.	0x0080
8	Retry count parameter is not valid.	0x0100
9	Spare	0x0200
10	Spare	0x0400
11	Spare	0x0800
12	Spare	0x1000
13	Spare	0x2000
14	Spare	0x4000
15	Spare	0x8000

Correct any invalid data in the configuration for proper module operation. When the configuration contains a valid parameter set, all the bits in the configuration words will be clear. This does not indicate that the configuration is valid for the user application. Make sure each parameter is set correctly for the specific application.

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 DF1 Master/Slave Communication Module with Reduced Data Block is a ControlLogix backplane compatible module that allows ControlLogix I/O compatible processors to easily interface with DF1 protocol compatible devices and hosts. Devices commonly supporting the protocol include Rockwell Automation PLCs and power monitoring equipment, as well as several other third party devices in the marketplace.

This module uses a small I/O data area for data transfer between the module and the ControlLogix processor, making it ideal for ControlNet or Ethernet applications with the module in a remote rack.

The MVI56 DF1 Communication Module for Remote Chassis allows ControlLogix I/O compatible processors to interface easily with other DF1 protocol compatible devices.

The module has two serial ports supporting the DF1 protocol, with each port user-configurable to act as a master or as a slave. Data transfer between the module and the ControlLogix processor is asynchronous to the DF1 network, with the module's internal database being used to exchange data between the processor and the DF1 network.

5.1.1 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

5.1.2 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, 485 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

5.1.3 Functional Specifications

This module uses a small I/O data area for data transfer between the module and the ControlLogix processor.

DF1 ports

- Full and half duplex modes supported
- CRC and BCC error checking
- Full hardware handshaking control provides radio, modem and multi-drop support
- User-definable module memory usage, supporting the storage and transfer of up to 5000 registers to/from the control processor
- Up to 125 word read and write command lengths supported
- Floating point data movement supported

DF1 Master Protocol Specifications

The ports on the DF1 module can be individually configured as Master ports. When configured in master mode, the DFCM module is capable of reading and writing data to remote DF1 devices, enabling the processor to act as a SCADA sub-master.

- Command List: Up to 100 commands per Master port, each fully-configurable for function, slave address, register to/from addressing and word/byte count
- Status Data: Error codes available on an individual command basis. In addition, a slave status list is maintained per active master port
- Polling of Command List: User-configurable polling of commands, including disabled, continuous, and on change of data (write only)

DF1 Slave Protocol Specifications

The module accepts DF1 commands from an attached DF1 master unit. When in slave mode, the module can accept DF1 commands from a master to read/write data stored in the module's internal registers. This data can be derived from other DF1 slave devices on the network through a master port or from the processor and is easily transferred to the processor's data registers.

Tested Hardware Connections

Several hardware connections have been tested by ProSoft Technology or have been customer field tested. The following physical connections have been tested successfully:

- RA Panel view (Full Duplex point-point, DFCM as slave)
- RA Processors (Full/Half duplex, DFCM as either master or slave)
- RA Power Monitors (485 Half-Duplex DFCM as Master)

5.2 Functional Overview

This section provides an overview of how the MVI56-DFCMR module transfers data using the DFCMR 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 discussion explains several concepts that are important for understanding the operation of the MVI56-DFCMR module.

Module Power Up

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

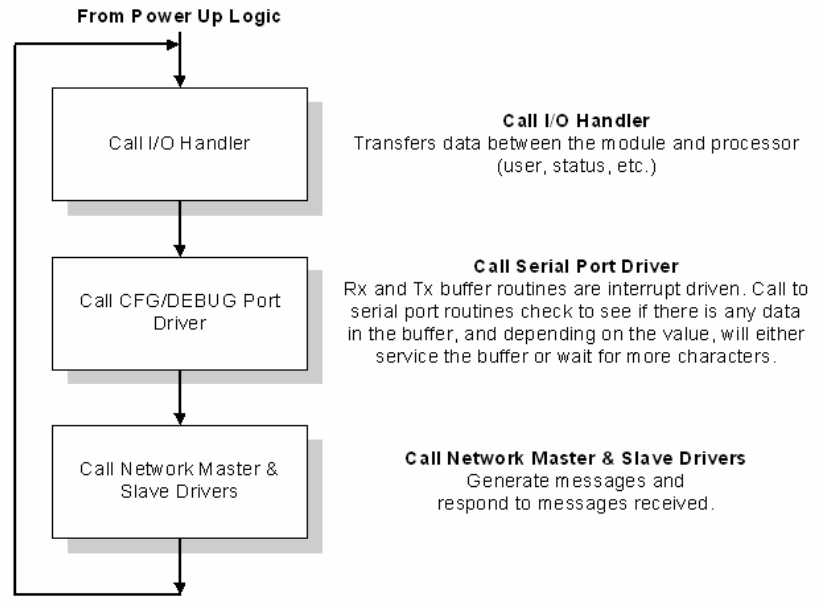
1. Initialize hardware components
   - Initialize ControlLogix backplane driver
   - Test and Clear all RAM
   - Initialize the serial communication ports
2. Wait for Module Configuration from ControlLogix processor

3 Initialize Module Register space
4 Enable Slave Driver on selected ports
5 Enable Master 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 following functions:



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 DF1 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.

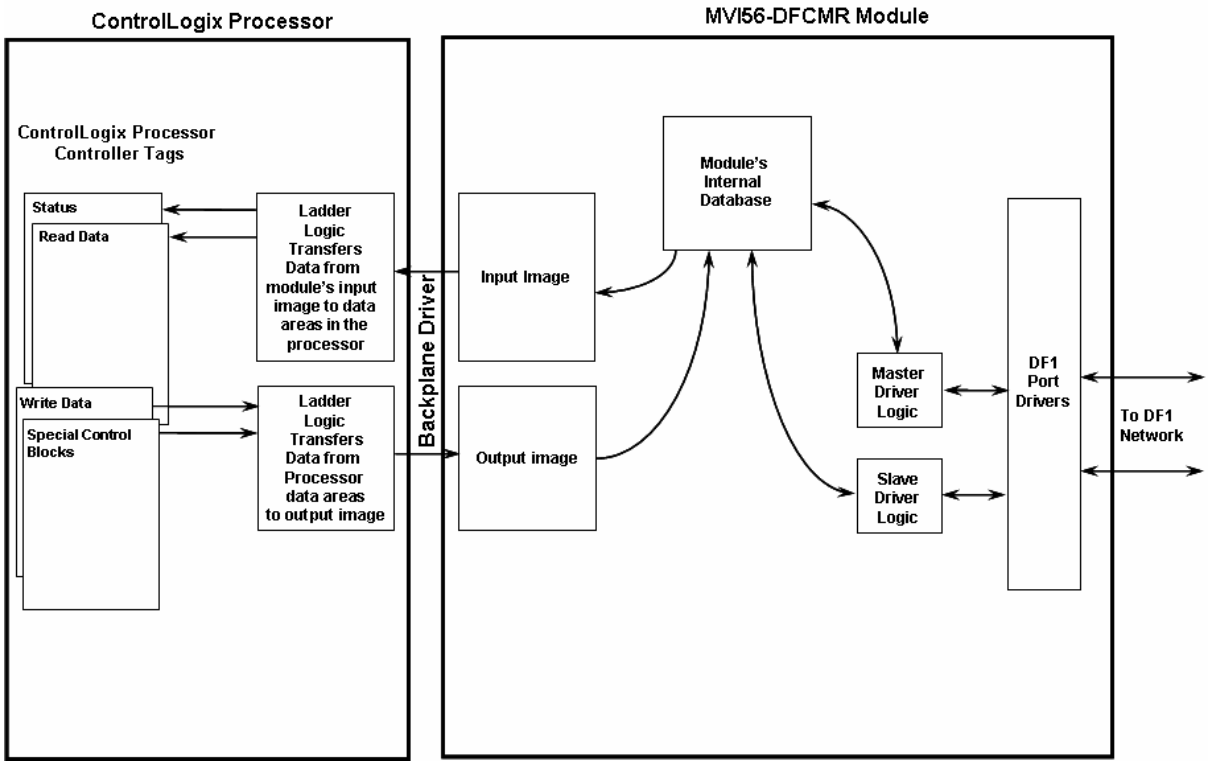
Backplane Data Transfer

The MVI56-DFCMR 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 and MSG instructions. 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 for the input and output images. Typical updates are in the range of 1 to 10 milliseconds. Execution time for the MSG instructions for data transfer are dependant on the amount of unscheduled time in the Network Update Time (NUT) and how frequently the instruction is executed.

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 42 words. This smaller data area permits fast throughput of data between the module and the processor as well as easier setup of the module in RSNETWORX.

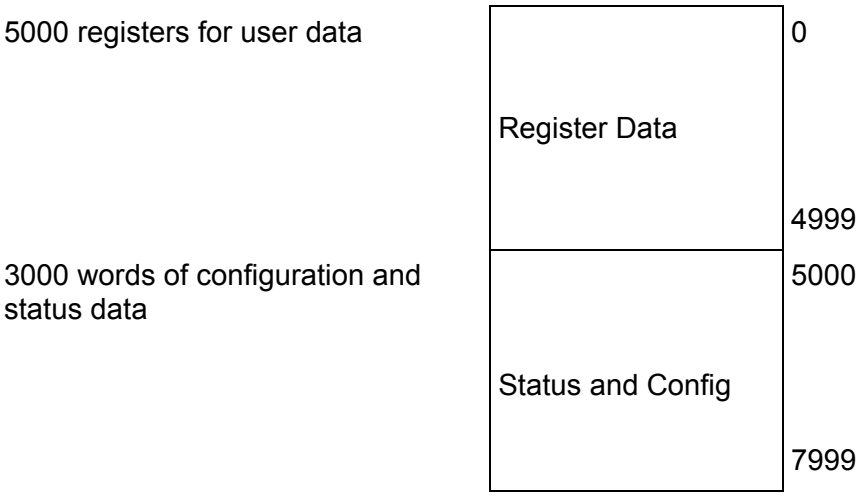
The processor inserts data in the module's output image to transfer to the module. The module's program extracts the data and places it in the module's internal database. The output image for the module is also set to 42 words.

The following illustration shows the data transfer method used to move data between the ControlLogix processor, the MVI56-DFCMR module, and the DF1 network.



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. The following illustration shows the layout of the database.

Module's Internal Database Structure



Data contained in this database is paged through the input and output images by coordination of the ControlLogix ladder logic and the MVI56-DFCMR module's program. Up to 42 words of data can be transferred from the module to the processor at a time. Up to 42 words of data can be transferred from the processor to the module. The read and write block identification codes in each data block determine the function to be performed or the content of the data block. The module uses the following block numbers:

MVI56-DFCMR Block Assignments

Block Range	Descriptions
-1	Status data block
0	Status data block
1 to 250	Read or write data
1000	Event Port 1
2000	Event Port 2
3000 to 3001	Port 1 slave polling control
3002 to 3009	Port 1 slave status
3100 to 3101	Port 2 slave polling control
3102 to 3109	Port 2 slave status
5001 to 5006	Port 1 command control
5101 to 5106	Port 2 command control
6000 to 6033	Port 1 commands (sent to module)
6100 to 6133	Port 2 commands (sent to module)
7000 to 7004	Port 1 Override File Maps (Sent to module)
7100 to 7104	Port 2 Override File Maps (Sent to module)
9000	Backplane Configuration data (sent to module)
9001	DFCM Port 1 Configuration (sent to module)
9002	DFCM Port 2 Configuration (sent to module)
9998	Warm-boot control block
9999	Cold-boot control block

Each image has a defined structure depending on the data content and the function of the data transfer.

Normal Data Transfer

Normal data transfer includes the transferring of data received by, or to be transmitted to, the DF1 drivers and the status data. This data is transferred through read (input image) and write (output image) blocks. Refer to **Module Set Up** 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	Write Block ID (1 to 250)	1
1 to 40	Read Data	40
41	Read Block ID (1 to 250)	1

The Block Identification Code (word 41) is used to signal to the ControlLogix processor that a new block is ready for processing and informs the processor of the contents of the block. If the value of the code is set to 1, the block contains the first 40 words of data contained in the database of the module.

The block also contains the block identification code the module expects to receive from the processor (word 0 in the block). Under normal data transfer conditions, the ladder logic should use the code to build the appropriate block for the module in the output image.

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 to 250)	1
1 to 40	Write Data	40
41	Spare	1

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 40 words.

Command Control Blocks

Command control blocks are special blocks used to control the module or request special data from the module. The current version of the software supports five command control blocks: event command control, command control, write configuration, warm boot and cold boot.

Event Command

Event command control blocks send DF1 commands directly from the ladder logic to one of the master ports. The format for these blocks are shown below:

Event Request (Write Block)

Offset	Description	Length
0	1000 or 2000	1
1	Internal DB Address	1
2	Point Count	1
3	Swap Code	1
4	Node Address	1
5	Function Code	1
6	Parameter #1	1
7	Parameter #2	1
8	Parameter #3	1
9	Parameter #4	1
10 to 41	Spare	32

The block number defines the DF1 port to be considered. Block 1000 commands are directed to Port 1, and Block 2000 commands are directed to Port 2. The parameters passed with the block construct the command. The Point Count parameter defines the number of registers for the command. The Swap Code changes the word or byte order. The Node Address parameter defines the device on the DF1 network to consider. The Function Code parameter is one of those defined in the ProSoft DF1 Command Set section. The parameter fields in the block should be completed as required by the selected function code. Each command has its own set of parameters. When the block is received, the module will process it and place the command in the command queue. The module will respond to each event command block with a read block with the following format:

Event Response (Read Block)

Offset	Description	Length
0	Write Block ID	1
1	0=Fail, 1=Success	1
2 to 40	Spare	39
41	1000 or 2000	1

Word 2 of the block can be used by the ladder logic to determine if the command was added to the command queue in the module. The command will only fail if the command queue for the port is full (100 commands for each queue) or the command requested is invalid.

Command Control Block

Command control blocks place commands in the command list into the command queue. Each port has a command queue of up to 100 commands. This module services commands in the queue before the master command list. This gives high priority to commands in the queue. Commands placed in the queue through this mechanism must be defined in the master command list. Under normal command list execution, the module will only execute commands with the Enable parameter set to one or two. If the value is set to zero, the command is skipped. Commands may be placed in the command list with an Enable parameter set to zero. These commands can then be executed using the command control blocks. One to six commands can be placed in the command queue with a single request. The format of the block is shown in the following table.

Command Control Request (Write Block)

Offset	Description	Length
0	5001 to 5006 or 5101 to 5106	1
1	Command index	1
2	Command index	1
3	Command index	1
4	Command index	1
5	Command index	1
6	Command index	1
7 to 41	Spare	35

Blocks in the range of 5001 to 5006 are used for Port 1, and blocks in the range of 5101 to 5106 are used for Port 2. The last digit in the block code defines the number of commands to process in the block. For example, a block code of 5003 contains 3 command indexes that are to be used with Port 1. The Command index parameters in the block have a range of 0 to 99 and correspond to the master command list entries.

The module responds to a command control block with a block containing the number of commands added to the command queue for the port. The format of the block is shown in the following table.

Command Control Response (Read Block)

Offset	Description	Length
0	Write Block ID	1
1	Number of commands added to command queue	1
2 to 40	Spare	39
41	5001 to 5006 or 5101 to 5106	1

Module 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. Refer to Configuration Data (page 27) for information on configuration data requests.

The blocks transferred from the module are as follows.

Module Configuration (Read Block)

Offset	Description	Length
0	9000 to 9002	1
1	Module Configuration Errors	1
2	Port 1 Configuration Errors	1
3	Port 2 Configuration Errors	1
4 to 40	Spare	37
41	-2 or -3	1

Module Configuration (Write Block)

Offset	Description	Length
0	9000	1
1 to 6	Backplane Set Up	6
7 to 41	Spare	35

Module Configuration (Write Block)

Offset	Description	Length
0	9001	1
1 to 26	Port 1 Configuration	26
27 to 41	Spare	15

Module Configuration (Write Block)

Offset	Description	Length
0	9002	1
1 to 26	Port 2 Configuration	26
27 to 41	Spare	15

Override File Map Configuration

The user has the ability to override the default file map configuration in order to define specific files when using the module as a DF1 slave device.

Override File Map Configuration (50/port)

Offset	Description	Length
0	7000 to 7004 or 7100 to 7104	1
1 to 4	Map #1 Definition	4
5 to 8	Map #2 Definition	4

Offset	Description	Length
9 to 12	Map #3 Definition	4
13 to 16	Map #4 Definition	4
17 to 20	Map #5 Definition	4
21 to 24	Map #6 Definition	4
25 to 28	Map #7 Definition	4
29 to 32	Map #8 Definition	4
33 to 36	Map #9 Definition	4
37 to 40	Map #10 Definition	4
41	Spare	1

Map Definition Locations

Each file map definition has the following structure.

Index	Description
1	DBReg: Database register for first file element specified.
2	FileNo: File Number
3	Element: Element Number
4	Length: Number of word registers

Warm Boot Block

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 41	Spare	41

In this version of the module, the warm and cold boot processes perform the same operation as many of the variables that must be initialized are fixed when the module first boots and cannot be changed after the application starts.

Cold Boot Block

Block 9999 performs a cold-boot operation on the module. The format of the block constructed by the processor is as follows:

Offset	Description	Length
0	9999	1
1 to 41	Spare	41

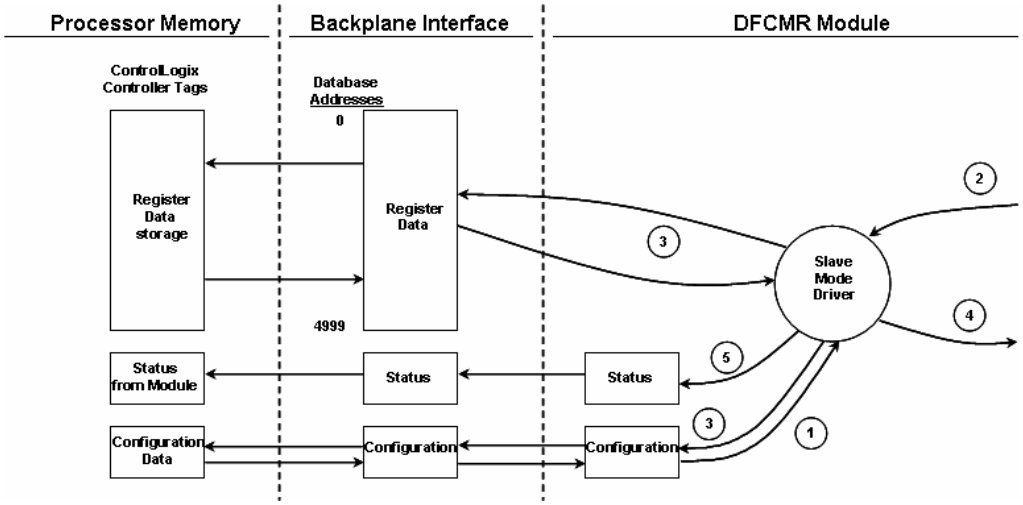
In this version of the module, the warm and cold boot processes perform the same operation as many of the variables that must be initialized are fixed when the module first boots and cannot be changed after the application starts.

5.2.2 Data Flow between MVI56-DFCMR Module and ControlLogix Processor

The following topics describe the flow of data between the two pieces of hardware (ControlLogix processor and MVI56-DFCMR module) and other nodes on the DF1 network under the module's different operating modes. Each port on the module is configured to emulate a DF1 master device or a DF1 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-DFCMR module to respond to data read and write commands issued by a master on the DF1 network. The following flow chart and associated table describe the flow of data into and out of the module.



Step	Description
1	The DF1 slave port driver receives the configuration information from the ControlLogix processor. This information configures the serial port and define the slave node characteristics. The module simulates N-files to permit remote access of the database. Each file has a fixed length of 200-word registers.
2	A Host device, such as the Rockwell Automation PLC or an HMI application 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 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 into the database and a response message is built.
4	After the data processing has been completed in Step 3, 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.

Review the **Module Set Up** section for a complete list of the parameters that must be defined for a slave port. The slave driver supports the following DF1 command set:

Basic Command Set Functions

Function Code	Command	Function	Definition	PLC5	SLC500 & MicroLogix	Power-monitor II	ControlLogix
1	0x00	N/A	Protected Write	X			X
2	0x01	N/A	Unprotected Read	X	X		X
3	0x02	N/A	Protected Bit Write	X			X
4	0x05	N/A	Unprotected Bit Write	X			X
5	0x08	N/A	Unprotected Write	X	X		X

PLC-5 Command Set Functions

Function Code	Command	Function	Definition	PLC5	SLC500 & MicroLogix	Power-monitor II	ControlLogix
100	0x0F	0x00	Word Range Write (Binary Address)	X			X
101	0x0F	0x01	Word Range Read (Binary Address)	X			X
102	0x0F	0x26	Read-Modify-Write (Binary Address)	X			X
150	0x0F	0x00	Word Range Write (ASCII Address)	X			X
151	0x0F	0x01	Word Range Read (ASCII Address)	X			X
152	0x0F	0x26	Read-Modify-Write (ASCII Address)	X			X

SLC-500 Command Set Functions

Function Code	Command	Function	Definition	PLC5	SLC500 & MicroLogix	Power-monitor II	ControlLogix
501	0x0F	0xA1	Protected Typed Logical Read With Two Address Fields		X		X
502	0x0F	0XA2	Protected Typed Logical Read With Three Address Fields		X	X	X
509	0x0F	0XA9	Protected Typed Logical Write With Two Address Fields		X		X
510	0x0F	0XAA	Protected Typed Logical Write With Three Address Fields		X	X	X
511	0x0F	0XAB	Protected Typed Logical Write With Mask (Three Address Fields)		X		X

Each command list record has the same general format. The first part of the record contains the information relating to the communication module and the second part contains information required to interface to the DF1 slave device.

The PLC-5 and SLC-500 command set require the use of files. These files are emulated in the module. The module defines these files each as containing 200-word registers that overlay the internal database. The following table shows the relationship of the files to the user data area of the internal database:

File	→	Database Register
N7:0	→	0
N8:0	→	200
N9:0	→	400
N10:0	→	600
N11:0	→	800
N12:0	→	1000
N13:0	→	1200
N14:0	→	1400
N15:0	→	1600
N16:0	→	1800
N17:0	→	2000
N18:0	→	2200
N18:0	→	2400
N20:0	→	2600
N21:0	→	2800
N22:0	→	3000
N23:0	→	3200
N24:0	→	3400
N25:0	→	3600
N26:0	→	3800
N27:0	→	4000
N28:0	→	4200
N29:0	→	4400
N30:0	→	4600
N31:0	→	4800
N32:0	→	5000

Note: The way these files are emulated depends of the *First File* and *File Size* parameters. The previous example shows using the *First File* parameter set to 7 and the *File Size* parameter set to 200.

In order to retrieve data from the modules database register 200, the remote master would issue a command using the address N8:0. In order to interface with database base register 405, the remote master would use the address N9:5. The following table outlines the complete file emulation for the module:

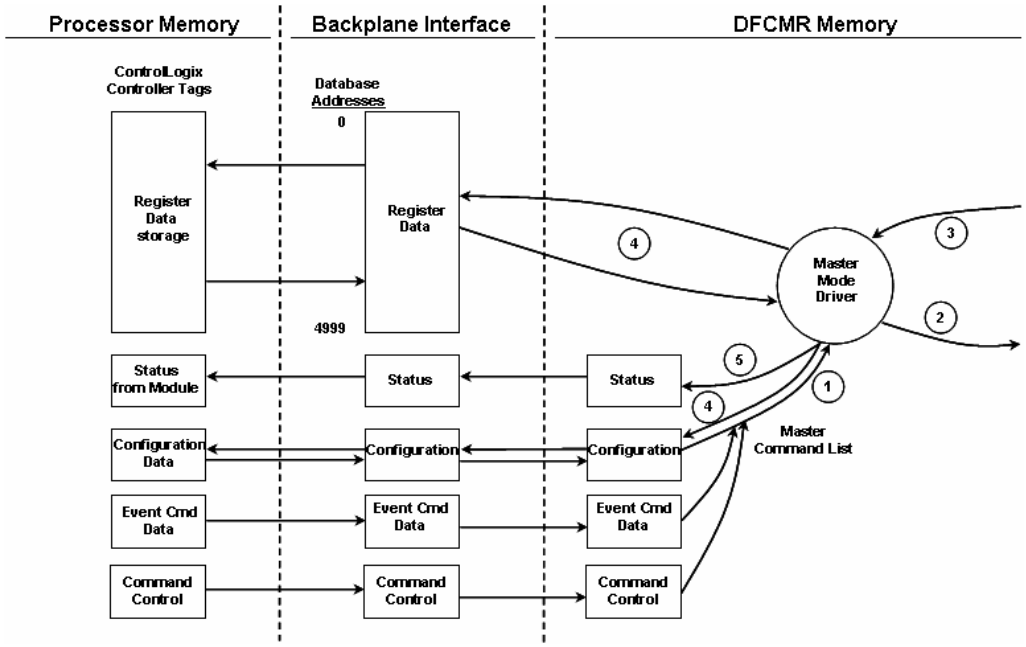
Register Range	File Start	File End	Content	Size
0 to 4999	N7:0	N31:199	User Data	5000
5000 to 5009	N32:0	N32:9	Backplane Configuration	10
5010 to 5039	N32:10	N32:39	Port 1 Setup	30
5040 to 5069	N32:40	N32:69	Port 2 Setup	30
5070 to 5199	N32:70	N32:199	Reserved	130
5200 to 6399	N33:0	N38:199	Port 1 Commands	1200
6400 to 7599	N39:0	N44:199	Port 2 Commands	1200

Register Range	File Start	File End	Content	Size
7600 to 7700	N45:0	N45:199	Misc. Status Data	200
7800 to 7999	N46:0	N46:199	Command Control	200
8000 to 9999	N47:0	N56:199	Reserved	2000

All the data in the module is available to a remote host. This permits the host device to remotely configure the module and view the status data.

5.2.3 Master Driver

In Master Mode, the MVI56-DFCMR module is responsible for issuing read or write commands to slave devices on the DF1 network. These commands are user configured in the module via the Master Command List received from the ControlLogix processor or issued directly from the ControlLogix processor (event command control). Command status is returned to the processor for each individual command in the command list status block. The location of this status block in the module's internal database is user defined. The following flow chart and associated table describe the flow of data into and out of the module.



Step	Description
1	The Master driver obtains configuration data from the ControlLogix processor. The configuration data obtained includes the number of commands and the Master Command List. These values are used by the Master driver to determine the type of commands to be issued to the other nodes on the DF1 network.
2	After configuration, the Master driver begins transmitting read and/or write commands to the other nodes on the network. If writing data to another node, the data for the write command is obtained from the module's internal database to build the command.
3	Presuming successful processing by the node specified in the command, a response message is received into the Master driver for processing.

Step	Description
4	Data received from the node on the network is passed into the module's internal database, assuming a read command.
5	Status is returned to the ControlLogix processor for each command in the Master Command List.

Refer to the **Module Set Up** section for a complete description of the parameters required to define the virtual DF1 master port. Care must be taken in constructing each command in the list for predictable operation of the module. If two commands write to the same internal database address of the module, the results will not be as desired. All commands containing invalid data will be ignored by the module. The module does not support the PLC-5 ASCII Address commands (150 to 152) as outlined in the DF1 Command Set documentation. The following table describes the functions supported by the module and the format of each command:

	Module Information Data ←					→ Device Information Data					
Col #	1	2	3	4	5	6	7	8	9	10	11
Function Code	Enable Code	Internal Address	Poll Interval Time	Count	Swap Code	Node Address	Function Code	Function Parameters			
FC 1	Code	Register	Seconds	Count	Code	Node	1	Word Address			
FC 2	Code	Register	Seconds	Count	Code	Node	2	Word Address			
FC 3	Code	Register	Seconds	Count	0	Node	3	Word Address			
FC 4	Code	Register	Seconds	Count	0	Node	4	Word Address			
FC 5	Code	Register	Seconds	Count	Code	Node	5	Word Address			
FC 100	Code	Register	Seconds	Count	Code	Node	100	File Number	Element	Sub-Element	
FC 101	Code	Register	Seconds	Count	Code	Node	101	File Number	Element	Sub-Element	
FC 102	Code	Register	Seconds	Count	0	Node	102	File Number	Element	Sub-Element	
FC 501	Code	Register	Seconds	Count	Code	Node	501	File Type	File Number	Element	
FC 502	Code	Register	Seconds	Count	Code	Node	502	File Type	File Number	Element	Sub-Element
FC 509	Code	Register	Seconds	Count	Code	Node	509	File Type	File Number	Element	
FC 510	Code	Register	Seconds	Count	Code	Node	510	File Type	File Number	Element	Sub-Element
FC 511	Code	Register	Seconds	Count	0	Node	511	File Type	File Number	Element	Sub-Element

Node Address = Destination Address for Message

If the DF1 master port is configured to support the DF1 half-duplex protocol, the master port can be used to route messages between slaves. Peer-to-peer communication is accomplished by the master constantly polling all the slaves on the network and relaying the messages received. The slaves must contain ladder logic with MSG commands to generate and accept messages. This routing can be used in conjunction with the normal command processing discussed above. If the slave node to be polled is not included in the command list, a special command is required in the command list. Enter a 999 in the **Enable Code** and the slave's node address in the **Node Address** fields of the command. No other parameters are required for a device poll message. This command will force the master port to issue an enquiry request to the slave device without first issuing a command. Any messages held in the slave's message queue will be sent to the master and the master will route the messages that do not contain the master's station code

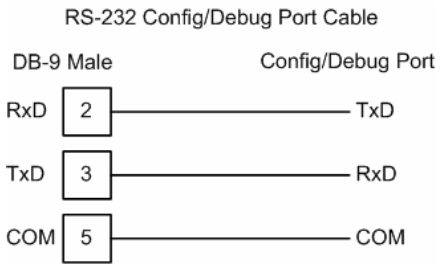
5.3 Cable Connections

The application ports on the MVI56-DFCMR 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

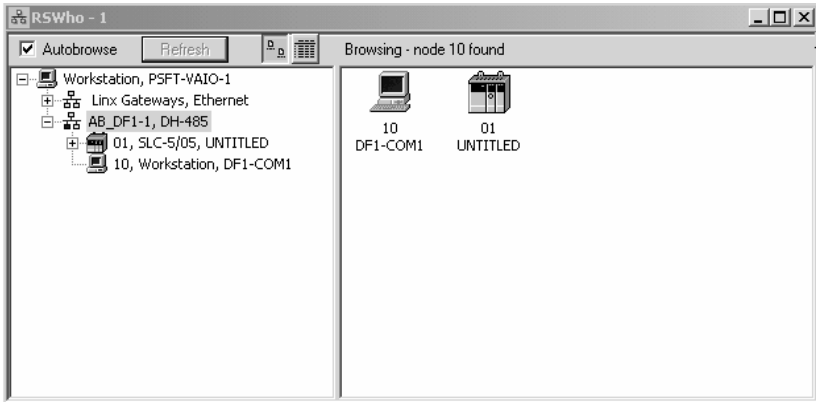
This port is physically an RJ45 connection. An RJ45 to DB-9 adapter cable is included with the module. This port permits a PC based terminal emulation program to view configuration and status data in the module and to control the module. The cable for communications on this port is shown in the following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



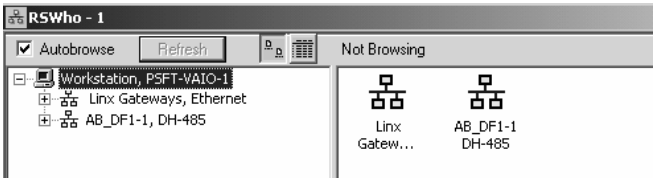
Disabling the RSLinx Driver for the Com Port on the PC

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:



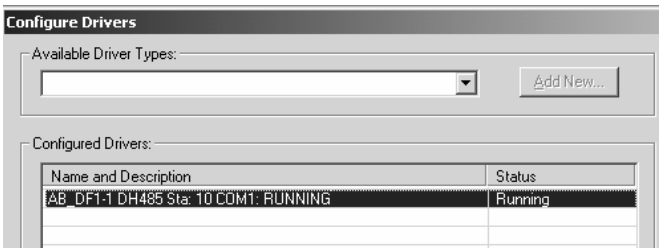
3 Notice how the DF1 driver is opened, and the driver is looking for a processor on node 1. If the network is being browsed, then you will not be able to stop this driver. To stop the driver your RSWho screen should look like this:



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

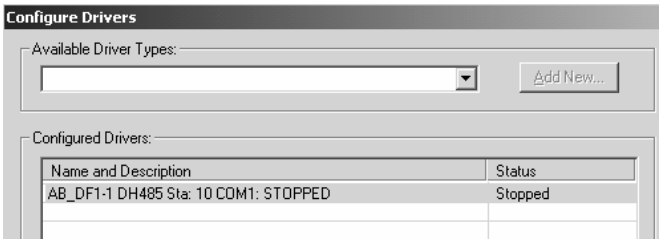
4 When you have verified that the driver is not being browsed, go to

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:

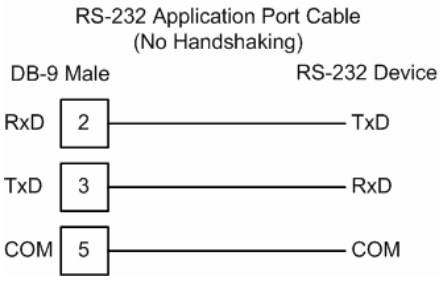


6 Upon seeing this, you may now use that com port to connect to the debug port of the module.

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.

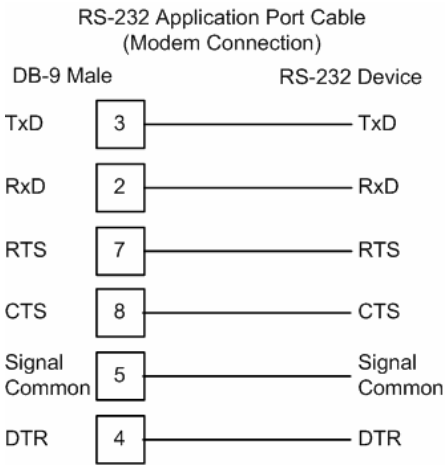
5.3.2 RS-232

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

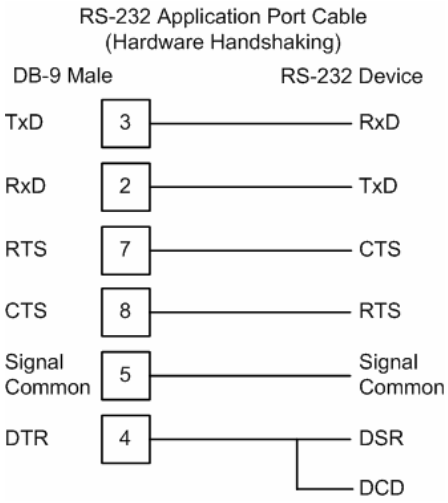
This type of connection is required between the module and a modem or other communication device.



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

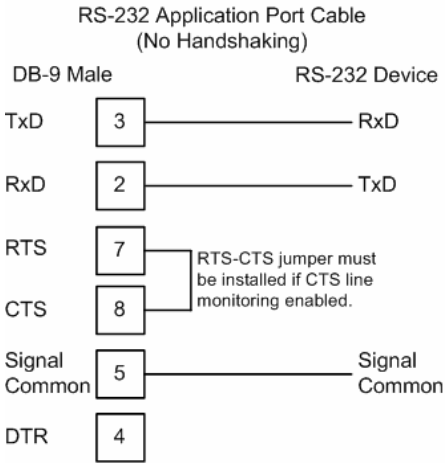
RS-232: Null Modem Connection (Hardware Handshaking)

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



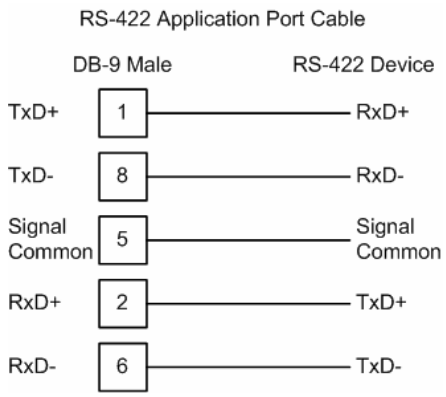
RS-232: Null Modem Connection (No Hardware Handshaking)

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



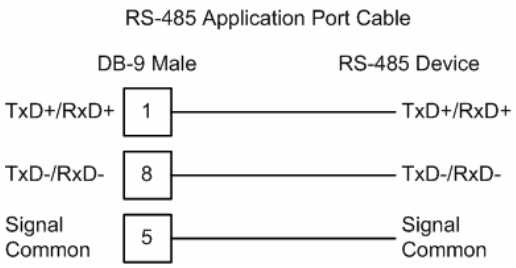
Note: If the port is configured with the "Use CTS Line" set to 'Y', then a jumper is required between the RTS and the CTS line on the module connection.

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:

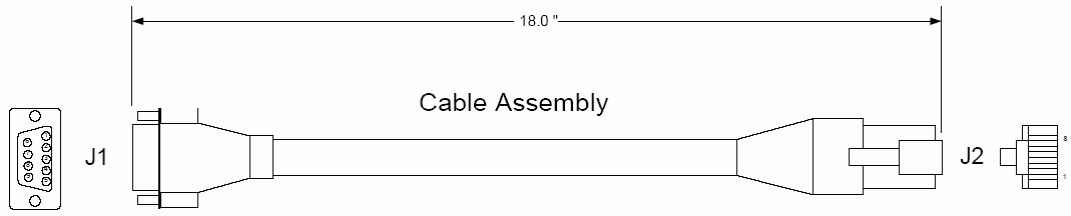


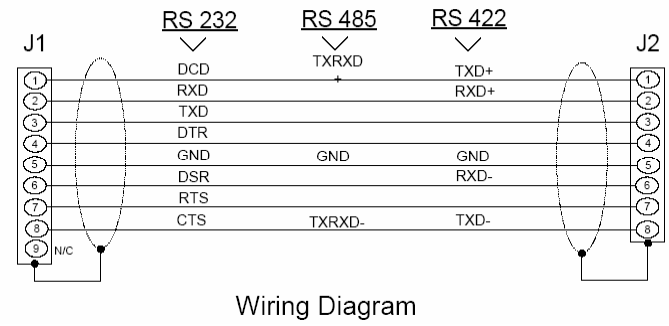
Note: Terminating resistors are generally not required on the RS-485 network, unless you are experiencing communication problems that can be attributed to signal echoes or reflections. In this case, install a 120 ohm terminating resistor on the RS-485 line.

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-DFCMR Database Definition

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

Register Range	Content	Size
0 to 4999	User Data	5000
5000 to 5009	Backplane Configuration	10
5010 to 5039	Port 1 Setup	30
5040 to 5069	Port 2 Setup	30
5070 to 5199	Reserved	130
5200 to 6399	Port 1 Commands	1200
6400 to 7599	Port 2 Commands	1200
7600 to 7700	Misc. Status Data	200
7800 to 7999	Command Control	200
8000 to 9999	Reserved	2000

The user data area holds data collected from other nodes on the network (master read commands) or data received from the processor (write blocks). Additionally, this data area is used as a data source for the processor (read blocks) or other nodes on the network (write commands).

Detailed definition of the miscellaneous status data area can be found in Status Data Definition (page 85)

Definition of the configuration data areas can be found in the data definition section of this document and in Configuration Data (page 27).

5.5 MVI56-DFCMR Status Data Definition

5.5.1 Status Data Block (Block IDs 0 and -1)

Offset	Content	Description
0	Write Blk ID	Write block ID to receive next
1	Program Scan Count	This value is incremented each time a complete program cycle occurs in the module.
2 to 3	Product Code	These two registers contain the product code of "DFCM"
4 to 5	Product Version	These two registers contain the product version for the current running software.
6 to 7	Operating System	These two registers contain the month and year values for the program operating system.
8 to 9	Run Number	These two registers contain the run number value for the currently running software.
10	Port 1 Command List Requests	This field contains the number of requests made from this port to slave devices on the network.
11	Port 1 Command List Response	This field contains the number of slave response messages received on the port.
12	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.
13	Port 1 Requests	This field contains the total number of messages sent out of the port.

Offset	Content	Description
14	Port 1 Responses	This field contains the total number of messages received on the port.
15	Port 1 Errors Sent	This field contains the total number of message errors sent out of the port.
16	Port 1 Errors Received	This field contains the total number of message errors received on the port.
17	Port 2 Command List Requests	This field contains the number of requests made from this port to slave devices on the network.
18	Port 2 Command List Response	This field contains the number of slave response messages received on the port.
19	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.
20	Port 2 Requests	This field contains the total number of messages sent out the port.
21	Port 2 Responses	This field contains the total number of messages received on the port.
22	Port 2 Errors Sent	This field contains the total number of message errors sent out of the port.
23	Port 2 Errors Received	This field contains the total number of message errors received on the port.
24	Read Block Count	This field contains the total number of read blocks transferred from the module to the processor.
25	Write Block Count	This field contains the total number of write blocks transferred from the processor to the module.
26	Parse Block Count	This field contains the total number of blocks successfully parsed that were received from the processor.
27	Command Event Block Count	This field contains the total number of command event blocks received from the processor.
28	Command Block Count	This field contains the total number of command blocks received from the processor.
29	Error Block Count	This field contains the total number of block errors recognized by the module.
30	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.
31	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.
32	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.
33	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.
34 to 40	Spare	Reserved for future use.
41	Read Blk ID	Read block ID 0 or -1 to indicate this block contains status data.

5.6 Error Codes

The module error codes are listed in this section. Error codes returned from the command list process are stored in the command list error memory region. A word is allocated for each command in the memory area. The error codes are formatted in the word as follows: The least-significant byte of the word contains the extended status code and the most-significant byte contains the status code.

Use the error codes returned for each command in the list to determine the success or failure of the command. If the command fails, use the error code to determine the cause of failure.

Note: The Module Specific error codes (not DF1 compliant) are returned from within the module and never returned from an attached DF1 slave device. These are error codes that are part of the DF1 protocol or are extended codes unique to this module. The standard DF1 error codes can be found in the DF1 Protocol and Command Set Reference Manual (Publication 1770-6.5.16) from Rockwell Automation. The most common errors for the DF1 protocol are shown in the following tables:

5.6.1 Local STS Error Codes

Code (Int)	Code (Hex)	Description
0	0x0000	Success, no error
256	0x0100	DST node is out of buffer space
512	0x0200	Cannot guarantee delivery (Link Layer)
768	0x0300	Duplicate token holder detected
1024	0x0400	Local port is disconnected
1280	0x0500	Application layer timed out waiting for response
1536	0x0600	Duplicate node detected
1792	0x0700	Station is offline
2048	0x0800	Hardware fault

5.6.2 Remote STS Error Codes

Code (Int)	Code (Hex)	Description
0	0x0000	Success, no error
4096	0x1000	Illegal command or format
8192	0x2000	Host has a problem and will not communicate
12288	0x3000	Remote node host is missing, disconnected or shut down
16384	0x4000	Host could not complete function due to hardware fault
20480	0x5000	Addressing problem or memory protect rungs
24576	0x6000	Function not allowed due to command protection selection
26872	0x7000	Processor is in Program mode
-32768	0x8000	Compatibility mode file missing or communication zone problem
-28672	0x9000	Remote node cannot buffer command
-24576	0xA000	Wait ACK (1775-KA buffer full)
-20480	0xB000	Remote node problem due to download

Code (Int)	Code (Hex)	Description
-16384	0xC000	Wait ACK (1775-KA buffer full)
-12288	0xD000	Not used
-8192	0xE000	Not used
	0xF0nn	Error code in the EXT STS byte (nn contains EXT error code)

5.6.3 Errors When EXT STS Is Present

Code (Int)	Code (Hex)	Description
-4096	0xF000	Not used
-4095	0xF001	A field has an illegal value
-4094	0xF002	Less levels specified in address than minimum for any address
-4093	0xF003	More levels specified in address than system supports
-4092	0xF004	Symbol not found
-4091	0xF005	Symbol is of improper format
-4090	0xF006	Address does not point to something usable
-4089	0xF007	File is wrong size
-4088	0xF008	Cannot complete request
-4087	0xF009	Data or file is too large
-4086	0xF00A	Transaction size plus word address is too large
-4085	0xF00B	Access denied, improper privilege
-4084	0xF00C	Condition cannot be generated - resource is not available
-4083	0xF00D	Condition already exists - resource is already available
-4082	0xF00E	Command cannot be executed
-4081	0xF00F	Histogram overflow
-4080	0xF010	No access
-4079	0xF011	Illegal data type
-4078	0xF012	Invalid parameter or invalid data
-4077	0xF013	Address reference exists to deleted area
-4076	0xF014	Command execution failure for unknown reason
-4075	0xF015	Data conversion error
-4074	0xF016	Scanner not able to communicate with 1771 rack adapter
-4073	0xF017	Type mismatch
-4072	0xF018	1171 module response was not valid
-4071	0xF019	Duplicate label
-4070	0xF01A	File is open; another node owns it
-4069	0xF01B	Another node is the program owner
-4068	0xF01C	Reserved
-4067	0xF01D	Reserved
-4066	0xF01E	Data table element protection violation
-4065	0xF01F	Temporary internal problem

5.6.4 Module Specific Error (not DFNT Compliant)

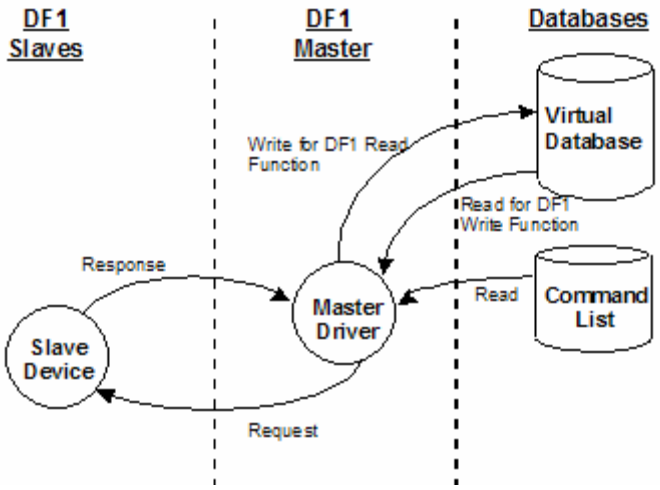
Code (Int)	Code (Hex)	Description
-1	0xFFFF	CTS modem control line not set before transmit
-2	0xFFFE	Timeout while transmitting message
-10	0xFFF6	Timeout waiting for DLE-ACK after request
-11	0xFFF5	Timeout waiting for response after request
-12	0xFFF4	Reply data does not match requested byte count
-20	0xFFEC	DLE-NAK received after request
-21	0xFFEB	DLE-NAK sent after response

5.7 DF1 Command Set For ProSoft Technology Communication Modules

5.7.1 Introduction

This document contains a complete description of the command set required to communicate with DF1 protocol devices using a ProSoft communication module. ProSoft communication modules that contain a virtual DF1 master device use this command set to control and monitor data in DF1 protocol devices. These include Rockwell Automation PLC, SLC, MicroLogix and ControlLogix controllers and field devices supporting the DF1 protocol. ProSoft supports the DF1 protocol on both the serial and network interface. The network interface requires the use of the port service address 0xAF12 as specified in the ControlNet Specification. Rockwell Automation supports this feature in the ControlLogix 5550, PLC5 xx/E and SLC 5/05 processors.

The ProSoft modules contain a virtual database that is defined by the user. This database is used as the source for write commands and the destination for read commands issued on the virtual DF1 master devices. The module interfaces data contained in remote DF1 slave devices to the virtual database using the DF1 master. User commands are issued out of the DF1 master from a command list. These commands gather or control data in the DF1 slave devices. The following illustration shows the relationships discussed above:



Each command issued from the DF1 master contains a field that indicates the location in the virtual database to be associated with the command. Care must be taken when designing a system to be sure the read and write data regions for the database do not overlap for a single device. The read area of one device can overlap the write section of another device to transfer the data from one slave device to another.

5.7.2 Command Function Codes

This section describes DFCMR commands to be configured by the user.

	Module Information Data			←	→	Device Information Data				
1	2	3	4	5	6	7	8	9	10	11
Enable Code	Internal Address	Poll Interval Time	Count	Swap Code	Node Address	Function Code	Function Parameters			

Function Code #1 - Protected Write (Basic Command Set)

Column	Parameter	Description	Parameter
1	Enable/Type Word	0=Disabled, 1=Continuous and 2=Conditional.	
2	Virtual Database Address	This parameter defines the database address of the first data point to be associated with the command.	
3	Poll Interval	Minimum time in tenths of a second to wait before polling with this command.	
4	Count	Number of data word values to be considered by the function.	
5	Swap Type Code	Swap type code for command: 0=None, 1=Swap words, 2=Swap words & bytes and 3=swap bytes in each word.	
6	Node Address	Address of unit to reach on the data highway.	
7	Function Code = 1	Protected Write Function	
8	Word Address	Word address where to start the write operation.	P1
9 to 11	Not Used	These fields are not used by the command. Values entered in these columns will be ignored.	P2 to P4

This function writes one or more words of data into a limited area of the slave device. This function should work on the following devices: 1774-PLC, PLC-2, PLC-3, PLC-5 and PLC-5/250.

Function Code #2 - Unprotected Read (Basic Command Set)

Column	Parameter	Description	Parameter
1	Enable/Type Word	0=Disabled and 1=Continuous.	
2	Virtual Database Address	This parameter defines the database address of the first data point to be associated with the command.	
3	Poll Interval	Minimum time in tenths of a second to wait before polling with this command.	
4	Count	Number of data word values to be considered by the function.	
5	Swap Type Code	Swap type code for command: 0=None, 1=Swap words, 2=Swap words & bytes and 3=swap bytes in each word.	

Column	Parameter	Description	Parameter
6	Node Address	Address of unit to reach on the data highway.	
7	Function Code = 2	Unprotected Read Function	
8	Word Address	Word address where to start the read operation.	P1
9 to 11	Not Used	These fields are not used by the command. Values entered in these columns will be ignored.	P2 to P4

This function reads one or more words of data from the PLC memory. This function should work on the following devices: 1774-PLC, PLC-2, PLC-3, PLC-5, SLC 500, SLC 5/03, SLC 5/04 and MicroLogix 1000.

Function Code #3 - Protected Bit Write (Basic Command Set)

Column	Parameter	Description	Parameter
1	Enable/Type Word	0=Disabled, 1=Continuous and 2=Conditional.	
	Virtual Database Address	This parameter defines the database address for the data to be associated with the command. The address defined represents a register address and not a bit address. This function will update one or more words of data as defined by the count parameter.	
3	Poll Interval	Minimum time in tenths of a second to wait before polling with this command.	
4	Count	Number of data word values to be considered by the function.	
5	Swap Type Code	Swap type code for command: Always zero (0).	
6	Node Address	Address of unit to reach on the data highway.	
7	Function Code = 3	Protected Bit Write Function	
8	Word Address	Word address where to start the write operation.	P1
9 to 11	Not Used	These fields are not used by the command. Values entered in these columns will be ignored.	P2 to P4

This function sets or resets individual bits within a limited area of the PLC data table. This function should work on the following devices: 1774-PLC, PLC-2, PLC-3, PLC-5 and PLC-5/250.

Function Code #4 - Unprotected Bit Write (Basic Command Set)

Column	Parameter	Description	Parameter
1	Enable/Type Word	0=Disabled, 1=Continuous and 2=Conditional.	
2	Virtual Database Address	This parameter defines the database address for the data to be associated with the command. The address defined represents a register address and not a bit address. This function will update one or more words of data as defined by the count parameter.	
3	Poll Interval	Minimum time in tenths of a second to wait before polling with this command.	
4	Count	Number of data word values to be considered by the function.	
5	Swap Type Code	Swap type code for command: Always zero (0).	
6	Node Address	Address of unit to reach on the data highway.	
7	Function Code = 4	Unprotected Bit Write Function	

Column	Parameter	Description	Parameter
8	Word Address	Word address where to start the write operation.	P1
9 to 11	Not Used	These fields are not used by the command. Values entered in these columns will be ignored.	P2 to P4

This function sets or resets individual bits within a limited area of the PLC data table. This function should work on the following devices: 1774-PLC, PLC-2, PLC-3 and PLC-5.

Function Code #5 - Unprotected Write (Basic Command Set)

Column	Parameter	Description	Parameter
1	Enable/Type Word	0=Disabled, 1=Continuous and 2=Conditional.	
2	Virtual Database Address	This parameter defines the database address of the first data point to be associated with the command.	
3	Poll Interval	Minimum time in tenths of a second to wait before polling with this command.	
4	Count	Number of data word values to be considered by the function.	
5	Swap Type Code	Swap type code for command: 0=None, 1=Swap words, 2=Swap words & bytes and 3=swap bytes in each word.	
6	Node Address	Address of unit to reach on the data highway.	
7	Function Code = 5	Unprotected Write Function	
8	Word Address	Word address where to start the write operation.	P1
9 to 11	Not Used	These fields are not used by the command. Values entered in these columns will be ignored.	P2 to P4

This function writes one or more words of data to the PLC memory. This function should work on the following devices: 1774-PLC, PLC-2, PLC-3, PLC-5, SLC 500, SLC 5/03, SLC 5/04 and MicroLogix 1000.

Function Code #100 - Word Range Write (PLC-5 Command) (Binary Address)

Column	Parameter	Description	Parameter
1	Enable/Type Word	0=Disabled, 1=Continuous and 2=Conditional.	
2	Virtual Database Address	This parameter defines the database address of the first data point to be associated with the command.	
3	Poll Interval	Minimum time in tenths of a second to wait before polling with this command.	
4	Count	Number of data word values to be considered by the function.	
5	Swap Type Code	Swap type code for command: 0=None, 1=Swap words, 2=Swap words & bytes and 3=swap bytes in each word.	
6	Node Address	Address of unit to reach on the data highway.	
7	Function Code = 100	Word Range Write Command.	
8	File Number	PLC-5 file number to be associated with the command. If a value of -1 is entered for the parameter, the field will not be used in the command, and the default file will be used.	P1

Column	Parameter	Description	Parameter
9	Element Number	The parameter defines the element in the file where write operation will start. If a value of -1 is entered for the parameter, the field will not be used in the command, and the default element will be used.	P2
10	Sub-Element Number	This parameter defines the sub-element to be used with the command. Refer to the AB documentation for a list of valid sub- element codes. If the value is set to -1, the default sub-element number will be used.	P3
11	Not Used	This field is not used by the command. Values entered in this column will be ignored.	P4

This function writes one or more words of data to a PLC data table. This function should work on the following devices: PLC-5.

Function Code #101 - Word Range Read (PLC-5 Command) (Binary Address)

Column	Parameter	Description	Parameter
1	Enable/Type Word	0=Disabled and 1=Continuous.	
2	Virtual Database Address	This parameter defines the database address of the first data point to be associated with the command.	
3	Poll Interval	Minimum time in tenths of a second to wait before polling with this command.	
4	Count	Number of data word values to be considered by the function.	
5	Swap Type Code	Swap type code for command: 0=None, 1=Swap words, 2=Swap words & bytes and 3=swap bytes in each word.	
6	Node Address	Address of unit to reach on the data highway.	
7	Function Code = 101	Word Range Write Command.	
8	File Number	PLC-5 file number to be associated with the command. If a value of -1 is entered for the parameter, the field will not be used in the command, and the default file will be used.	P1
9	Element Number	The parameter defines the element in the file where write operation will start. If a value of -1 is entered for the parameter, the field will not be used in the command, and the default element will be used.	P2
10	Sub-Element Number	This parameter defines the sub-element to be used with the command. Refer to the AB documentation for a list of valid sub- element codes. If the value is set to -1, the default sub-element number will be used.	P3
11	Not Used	This field is not used by the command. Values entered in this column will be ignored.	P4

This function reads one or more words of data from a PLC data table. This function should work on the following devices: PLC-5.

Function Code #102 - Read-Modify-Write (PLC-5 Command) (Binary Address)

Column	Parameter	Description	Parameter
1	Enable/Type Word	0=Disabled, 1=Continuous and 2=Conditional.	
2	Virtual Database Address	This parameter defines the database address for the data to be associated with the command.	

Column	Parameter	Description	Parameter
3	Poll Interval	Minimum number of seconds to wait before polling with this command.	
4	Count	Number of data word values to be considered by the function.	
5	Swap Type Code	Swap type code for command: Always zero (0).	
6	Node Address	Address of unit to reach on the data highway.	
7	Function Code = 102	Read-Modify-Write Command.	
8	File Number	PLC-5 file number to be associated with the command. If a value of -1 is entered for the parameter, the field will not be used in the command, and the default file will be used.	P1
9	Element Number	The parameter defines the element in the file where write operation will start. If a value of -1 is entered for the parameter, the field will not be used in the command, and the default element will be used.	P2
10	Sub-Element Number	This parameter defines the sub-element to be used with the command. Refer to the AB documentation for a list of valid sub-element codes. If the value is set to -1, the default sub-element number will be used.	P3
11	Not Used	This field is not used by the command. Values entered in this column will be ignored.	P4

This function writes one or more words of data to a PLC data table. This function should work on the following devices: PLC-5. The command constructed contains an AND mask and an OR mask. Values in the AND mask have the following definitions: 0=Reset and 1=Leave the Same. Values in the OR mask have the following definitions: 0=Leave the Same and 1=Set. The module is responsible for setting the mask values to correctly construct the message from the virtual database values.

Function Code #150 - Word Range Write (PLC-5 Command) (ASCII Address)

Column	Parameter	Description	Parameter
1	Enable/Type Word	0=Disabled, 1=Continuous and 2=Conditional.	
2	Virtual Database Address	This parameter defines the database address of the first data point to be associated with the command.	
3	Poll Interval	Minimum time in tenths of a second to wait before polling with this command.	
4	Count	Number of data word values to be considered by the function.	
5	Swap Type Code	Swap type code for command: 0=None, 1=Swap words, 2=Swap words & bytes and 3=swap bytes in each word.	
6	Node Address	Address of unit to reach on the data highway.	
7	Function Code = 150	Word Range Write Command.	
8	File String	PLC-5 address as specified as an ASCII string. For example, N10:300.	P1
9 to 11	Not Used	These fields are not used by the command. Values entered in these columns will be ignored.	P2 to P4

This function writes one or more words of data to a PLC data table. This function should work on the following devices: PLC-5.

Function Code #151 - Word Range Read (PLC-5 Command) (ASCII Address)

Column	Parameter	Description	Parameter
1	Enable/Type Word	0=Disabled and 1=Continuous.	
2	Virtual Database Address	This parameter defines the database address of the first data point to be associated with the command.	
3	Poll Interval	Minimum time in tenths of a second to wait before polling with this command.	
4	Count	Number of data word values to be considered by the function.	
5	Swap Type Code	Swap type code for command: 0=None, 1=Swap words, 2=Swap words & bytes and 3=swap bytes in each word.	
6	Node Address	Address of unit to reach on the data highway.	
7	Function Code = 151	Word Range Read Command.	
8	File String	PLC-5 address as specified as an ASCII string. For example, N10:300.	P1
9 to 11	Not Used	These fields are not used by the command. Values entered in these columns will be ignored.	P2 to P4

This function reads one or more words of data from a PLC data table. This function should work on the following devices: PLC-5.

Function Code #152 - Read-Modify-Write (PLC-5 Command) (ASCII Address)

Column	Parameter	Description	Parameter
1	Enable/Type Word	0=Disabled, 1=Continuous and 2=Conditional.	
2	Virtual Database Address	This parameter defines the database address for the data to be associated with the command. The first database register is used as the AND mask for the command, and the second is used for the OR mask. Values in the AND mask have the following definitions: 0=Reset and 1=Leave the Same. Values in the OR mask have the following definitions: 0=Leave the Same and 1=Set.	
3	Poll Interval	Minimum time in tenths of a second to wait before polling with this command.	
4	Count	Number of data word values to be considered by the function.	
5	Swap Type Code	Swap type code for command: Always zero (0).	
6	Node Address	Address of unit to reach on the data highway.	
7	Function Code = 152	Read-Modify-Write Command.	
8	File String	PLC-5 address as specified as an ASCII string. For example, N10:300.	P1
9 to 11	Not Used	These fields are not used by the command. Values entered in these columns will be ignored.	P2 to P4

This function writes one or more words of data to a PLC data table. This function should work on the following devices: PLC-5. The command constructed contains an AND mask and an OR mask. Values in the AND mask have the following definitions: 0=Reset and 1=Leave the Same. Values in the OR mask have the following definitions: 0=Leave the Same and 1=Set. The module is responsible for setting the mask values to correctly construct the message from the virtual database values.

Function Code #501 - Protected Typed Logical Read (Two Address Fields)

Column	Parameter	Description	Parameter
1	Enable/Type Word	0=Disabled and 1=Continuous.	
2	Virtual Database Address	This parameter defines the database address of the first data point to be associated with the command.	
3	Poll Interval	Minimum number of seconds to wait before polling with this command.	
4	Count	Number of data word values to be considered by the function.	
5	Swap Type Code	Swap type code for command: 0=None, 1=Swap words, 2=Swap words & bytes and 3=swap bytes in each word.	
6	Node Address	Address of unit to reach on the data highway.	
7	Function Code = 501	Logical Read Command	
8	File Type	SLC file type letter as used in file name string. Valid values for the system are N, S, F, A, ….	P1
9	File Number	SLC file number to be associated with the command.	P2
10	Element Number	The parameter defines the element in the file where write operation will start.	P3
11	Not Used	This field is not used by the command. Values entered in this column will be ignored.	P4

This function reads one or more words of data from a PLC data table.

Function Code #502 - Protected Typed Logical Read (Three Address Fields)

Column	Parameter	Description	Parameter
1	Enable/Type Word	0=Disabled and 1=Continuous.	
2	Virtual Database Address	This parameter defines the database address of the first data point to be associated with the command.	
3	Poll Interval	Minimum number of seconds to wait before polling with this command.	
4	Count	Number of data word values to be considered by the function.	
5	Swap Type Code	Swap type code for command: 0=None, 1=Swap words, 2=Swap words & bytes and 3=swap bytes in each word.	
6	Node Address	Address of unit to reach on the data highway.	
7	Function Code = 502	Logical Read Command	
8	File Type	SLC file type letter as used in file name string. Valid values for the system are N, S, F, A, ….	P1
9	File Number	SLC file number to be associated with the command.	P2
10	Element Number	The parameter defines the element in the file where write operation will start.	P3
11	Sub-Element Number	This parameter defines the sub-element to be used with the command. Refer to the AB documentation for a list of valid sub-element codes.	P4

This function reads one or more words of data from a PLC data table. This function should work on the following devices: SLC 500, SLC 5/03 and SLC 5/04.

Function Code #509 - Protected Typed Logical Write (Two Address Fields)

Column	Parameter	Description	Parameter
1	Enable/Type Word	0=Disabled, 1=Continuous and 2=Conditional.	
2	Virtual Database Address	This parameter defines the database address of the first data point to be associated with the command.	
3	Poll Interval	Minimum time in tenths of a second to wait before polling with this command.	
4	Count	Number of data word values to be considered by the function.	
5	Swap Type Code	Swap type code for command: 0=None, 1=Swap words, 2=Swap words & bytes and 3=swap bytes in each word.	
6	Node Address	Address of unit to reach on the data highway.	
7	Function Code = 509	Logical Write Command	
8	File Type	SLC file type letter as used in file name string. Valid values for the system are N, S, F, A, ….	P1
9	File Number	SLC file number to be associated with the command.	P2
10	Element Number	The parameter defines the element in the file where write operation will start.	P3
11	Not Used	This field is not used by the command. Values entered in this column will be ignored.	P4

This function writes one or more words of data to a PLC data table.

Function Code #510 - Protected Typed Logical Write (Three Address Fields)

Column	Parameter	Description	Parameter
1	Enable/Type Word	0=Disabled, 1=Continuous and 2=Conditional.	
2	Virtual Database Address	This parameter defines the database address of the first data point to be associated with the command.	
3	Poll Interval	Minimum time in tenths of a second to wait before polling with this command.	
4	Count	Number of data word values to be considered by the function.	
5	Swap Type Code	Swap type code for command: 0=None, 1=Swap words, 2=Swap words & bytes and 3=swap bytes in each word.	
6	Node Address	Address of unit to reach on the data highway.	
7	Function Code = 510	Logical Write Command	
8	File Type	SLC file type letter as used in file name string. Valid values for the system are N, S, F, A, ….	P1
9	File Number	SLC file number to be associated with the command.	P2
10	Element Number	The parameter defines the element in the file where write operation will start.	P3
11	Sub-Element Number	This parameter defines the sub-element to be used with the command. Refer to the AB documentation for a list of valid sub-element codes.	P4

This function writes one or more words of data to a PLC data table. This function should work on the following devices: SLC 500, SLC 5/03 and SLC 5/04.

Function Code #511 - Protected Typed Logical Write with Mask (Three Address Fields)

Column	Parameter	Description	Parameter
1	Enable/Type Word	0=Disabled, 1=Continuous and 2=Conditional.	
2	Virtual Database Address	This parameter defines the database address of the data to be associated with the command. The first word of data contains the bit mask and the second word contains the data.	
3	Poll Interval	Minimum time in tenths of a second to wait before polling with this command.	
4	Count	Number of data word values to be considered by the function.	
5	Swap Type Code	Swap type code for command: Always zero (0).	
6	Node Address	Address of unit to reach on the data highway.	
7	Function Code = 511	Logical Write with mask	
8	File Type	SLC file type letter as used in file name string. Valid values for the system are N, S, F, A, ….	P1
9	File Number	SLC file number to be associated with the command.	P2
10	Element Number	The parameter defines the element in the file where write operation will start.	P3
11	Sub-Element Number	This parameter defines the sub-element to be used with the command. Refer to the AB documentation for a list of valid sub-element codes.	P4

This function writes one or more words of data from a PLC data table controlling individual bits in the table. The bit mask used for the command is 0xFFFF. This provides direct manipulation of the data in the device with the internal data of the module. The function requires that all data associated with the command use the same mask.

5.7.3 PLC-5 Processor Specifics

This section contains information specific to the PLC-5 processor with relation to the DF1 command set. The commands specific to the PLC-5 processor contain a sub-element code field. This field selects a sub-element field in a complex data table. For example, to obtain the current accumulated value for a counter or timer, the sub-element field should be set to 2. The tables below show the sub-element codes for PLC-5 complex data tables.

PLC-5 Sub-Element Codes

Timer / Counter

Code	Description
0	Control
1	Preset
2	Accumulated

Control

Code	Description
0	Control
1	Length
2	Position

PD*

Code	Description
0	Control
2	SP
4	Kp
6	Ki
8	Kd
26	PV

*All PD values are floating point values, so they are two words long.

BT

Code	Description
0	Control
1	RLEN
2	DLEN
3	Data file #
4	Element #
5	Rack/Grp/Slot

MG

Code	Description
0	Control
1	Error
2	RLEN
3	DLEN

5.7.4 SLC Processor Specifics

This section contains information specific to the SLC processor based family when used with the DF1 command set. The SLC processor commands support a file type field entered as a single character to denote the data table to interface with in the command. The following table defines the relationship of the file types accepted by the module and the SLC file types:

SLC File Types

File Type	File Type Command Code	Description
S	83	Status
B	66	Bit
T	84	Timer
C	67	Counter
R	82	Control
N	78	Integer
F	70	Floating-point
Z	90	String
A	65	ASCII

The File Type Command Code is the ASCII character code value of the File Type letter. This is the value to enter into the "File Type" parameter of the DF1 Command configurations in the data tables in the ladder logic.

Additionally, the SLC specific functions (502, 510 and 511) support a sub-element field. This field selects a sub-element field in a complex data table. For example, to obtain the current accumulated value for a counter or timer, the sub-element field should be set to 2.

5.7.5 MicroLogix Processor Specifics

This section contains information specific to the MicroLogix processor based family when used with the DF1 command set. The MicroLogix processor commands support a file type field entered as a single character to denote the data table to interface with in the command. This field is the same as that used for a SLC processor. The following table defines the relationship of the file types accepted by the module and the SLC file types:

SLC File Types

File Type	File Type Command Code	Description
S	83	Status
B	66	Bit
T	84	Timer
C	67	Counter
R	82	Control
N	78	Integer
F	70	Floating-point
Z	90	String
A	65	ASCII

The File Type Command Code is the ASCII character code value of the File Type letter. This is the value to enter into the "File Type" parameter of the DF1 Command configurations in the data tables in the ladder logic.

Additionally, the SLC specific functions (502, 510 and 511) support a sub-element field. This field selects a sub-element field in a complex data table. For example, to obtain the current accumulated value for a counter or timer, the sub-element field should be set to 2.

5.7.6 ControlLogix Processor Specifics

This section contains information specific to the ControlLogix processor when used with the DF1 command set. The current implementation of the DF1 command set does not use functions that can directly interface with the ControlLogix Tag Database. In order to interface with this database, the table-mapping feature provided by RSLogix 5000 must be used. The software permits the assignment of ControlLogix Tag Arrays to virtual PLC 5 data tables. The ProSoft module using the PLC 5 command set defined in this document can then reach this controller data.

5.8 DF1 Command List Form

	Module Information Data ←					→ Device Information Data					
Column #	1	2	3	4	5	6	7	8	9	10	11
Function Code	Enable Code	Internal Address	Poll Interval Time	Count	Swap Code	Node Address	Function Code	Function Parameters			

6 Support, Service & Warranty

In This Chapter

ProSoft Technology, Inc. (ProSoft) is committed to providing the most efficient and effective support possible. Before calling, please gather the following information to assist in expediting this process:

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

If the issue is hardware related, we will also need information regarding:

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

6.1 How to Contact Us: Technical Support

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

Asia Pacific

+603.7724.2080, support.asia@prosoft-technology.com (mailto:support.asia@prosoft-technology.com)

Languages spoken include: Chinese, English

Europe (location in Toulouse, France)

+33 (0) 5.34.36.87.20, support.EMEA@prosoft-technology.com (mailto:support.emea@prosoft-technology.com)

Languages spoken include: French, English

North America/Latin America (excluding Brasil) (location in California)

+1.661.716.5100, support@prosoft-technology.com (mailto:support@prosoft-technology.com)

Languages spoken include: English, Spanish

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.*

Brasil (location in Sao Paulo)

+55-11-5084-5178 , eduardo@prosoft-technology.com (mailto:eduardo@prosoft-technology.com)

Languages spoken include: Portuguese, English

6.2 Return Material Authorization (RMA) Policies and Conditions

The following RMA Policies and Conditions (collectively, "RMA Policies") apply to any returned Product. These RMA Policies are subject to change by ProSoft without notice. For warranty information, see "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:

a) 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.
b) 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 (page 103). A Technical Support Engineer will request that you perform several tests in an attempt to isolate the problem. If after completing these tests, the Product is found to be the source of the problem, we will issue an RMA.
c) 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 and receipt date. 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 using a shipment method other than that specified by ProSoft or shipped without an RMA number will be returned to the Customer, freight collect. Contact ProSoft Technical Support for further information.
d) A 10% restocking fee applies to all warranty credit returns whereby a Customer has an application change, ordered too many, does not need, etc.

6.2.2 Procedures for Return of Units Under Warranty:

A Technical Support Engineer must approve the return of Product under ProSoft's Warranty:

a) A replacement module will be shipped and invoiced. A purchase order will be required.
b) Credit for a product under warranty will be issued upon receipt of authorized product by ProSoft at designated location referenced on the Return Material Authorization.

6.2.3 Procedures for Return of Units Out of Warranty:

a) Customer sends unit in for evaluation
b) If no defect is found, Customer will be charged the equivalent of $100 USD, plus freight charges, duties and taxes as applicable. A new purchase order will be required.
c) If unit is repaired, charge to Customer will be 30% of current list price (USD) plus freight charges, duties and taxes as applicable. A new purchase order will be required or authorization to use the purchase order submitted for evaluation fee.

The following is a list of non-repairable units:

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

6.2.4 Purchasing Warranty Extension:

a) ProSoft's standard warranty period is three (3) years from the date of shipment as detailed in "Limited Warranty (page 106)". The Warranty Period may be extended at the time of equipment purchase 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

6.3 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.

6.3.1 What Is Covered By This Warranty

a) *Warranty On New Products*: ProSoft warrants, to the original purchaser, 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 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 39 months. 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.
b) *Warranty On Services*: Materials and labor performed by ProSoft to repair a verified malfunction or defect are warranteed in 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.

6.3.2 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) This Warranty 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, "C" or any variant of "C" programming languages) 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; or (viii) disasters such as fire, flood, earthquake, wind and lightning.
c) 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 guide included with your original product purchase 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.

6.3.3 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 and 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.

6.3.4 Intellectual Property Indemnity

Buyer shall indemnify and hold harmless ProSoft and its employees from and against all liabilities, losses, claims, costs and expenses (including attorney's fees and expenses) related to any claim, investigation, litigation or proceeding (whether or not ProSoft is a party) which arises or is alleged to arise from Buyer's acts or omissions under these Terms or in any way with respect to the Products. Without limiting the foregoing, Buyer (at its own expense) shall indemnify and hold harmless ProSoft and defend or settle any action brought against such Companies to the extent based on a claim that any Product made to Buyer specifications infringed intellectual property rights of another party. ProSoft makes no warranty that the product is or will be delivered free of any person's claiming of patent, trademark, or similar infringement. The Buyer assumes all risks (including the risk of suit) that the product or any use of the product will infringe existing or subsequently issued patents, trademarks, or copyrights.

a) Any documentation included with Product purchased from ProSoft is protected by copyright and may not be duplicated 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.
f) **Additional Restrictions Relating To Software And Other Intellectual Property**

   In addition to compliance 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.

6.3.5 Disclaimer of all Other Warranties

The Warranty set forth in What Is Covered By This Warranty (page 106) 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.

*6.3.6 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 or its Dealer will not be responsible for included, 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.

6.3.7 Time Limit for Bringing Suit

Any action for breach of warranty must be commenced within 39 months following shipment of the Product.

6.3.8 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.

6.3.9 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.

6.3.10 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

U

V

W

Y