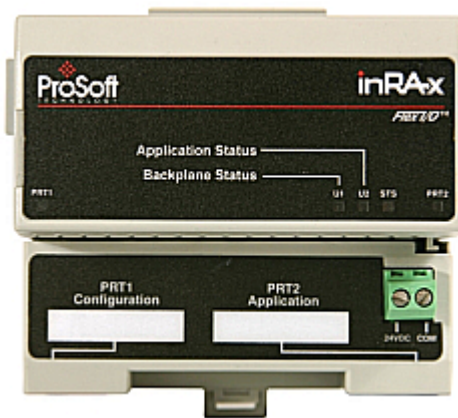




Where Automation Connects.



inRAX®
MVI94-DFCM

FLEX Platform

DF1 Communication Module

July 06, 2009

USER MANUAL

Important Installation Instructions

Power, Input and Output (I/O) wiring must be in accordance with Class I, Division 2 wiring methods, Article 501-4 (b) of the National Electrical Code, NFPA 70 for installation in the U.S., or as specified in Section 18-1J2 of the Canadian Electrical Code for installations in Canada, and in accordance with the authority having jurisdiction. The following warnings must be heeded:

- A** WARNING - EXPLOSION HAZARD - SUBSTITUTION OF COMPONENTS MAY IMPAIR SUITABILITY FOR CLASS I, DIV. 2;
- B** WARNING - EXPLOSION HAZARD - WHEN IN HAZARDOUS LOCATIONS, TURN OFF POWER BEFORE REPLACING OR WIRING MODULES, and
- C** WARNING - EXPLOSION HAZARD - DO NOT DISCONNECT EQUIPMENT UNLESS POWER HAS BEEN SWITCHED OFF OR THE AREA IS KNOWN TO BE NONHAZARDOUS.
- D** "THIS DEVICE SHALL BE POWERED BY CLASS 2 OUTPUTS ONLY.

MVI (Multi Vendor Interface) Modules

WARNING - EXPLOSION HAZARD - DO NOT DISCONNECT EQUIPMENT UNLESS POWER HAS BEEN SWITCHED OFF OR THE AREA IS KNOWN TO BE NON-HAZARDOUS.

AVERTISSEMENT - RISQUE D'EXPLOSION - AVANT DE DÉCONNECTER L'EQUIPMENT, COUPER LE COURANT OU S'ASSURER QUE L'EMPLACEMENT EST DÉSIGNÉ NON DANGEREUX.

CL I Div 2 GPs A, B, C, D

Temp Code T5

II 3 G

Ex nA IIC T5 X

0° C <= Ta <= 60° C

II - Equipment intended for above ground use (not for use in mines).

3 - Category 3 equipment, investigated for normal operation only.

G - Equipment protected against explosive gasses.

Warnings

North America Warnings

- A** Warning - Explosion Hazard - Substitution of components may impair suitability for Class I, Division 2.
- B** Warning - Explosion Hazard - When in Hazardous Locations, turn off power before replacing or rewiring modules.
Warning - Explosion Hazard - Do not disconnect equipment unless power has been switched off or the area is known to be nonhazardous.
- C** Suitable for use in Class I, division 2 Groups A, B, C and D Hazardous Locations or Non-Hazardous Locations.

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

cUL

C22.2 No. 213-1987



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

5201 Truxtun Ave., 3rd Floor
Bakersfield, CA 93309
+1 (661) 716-5100
+1 (661) 716-5101 (Fax)
www.prosoft-technology.com

Copyright © ProSoft Technology, Inc. 2009. All Rights Reserved.

MVI94-DFCM User Manual
July 06, 2009

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

ProSoft Technology® 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: 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 (0) 5.3436.87.20

Latin America: +1.281.298.9109

North America: +1.661.716.5100

Contents

Important Installation Instructions	2
MVI (Multi Vendor Interface) Modules	2
Warnings	2
Your Feedback Please.....	3
ProSoft Technology® Product Documentation	3
1 Quick Start	7
2 Hardware Installation	9
2.1 Verify Package Contents.....	9
2.2 Mounting the MVI 94 Flex I/O Base	10
2.3 Setting Jumpers	11
2.4 Installing the Module onto the Base	12
2.5 Installing the Serial Adapter Cables	13
2.6 Wiring the Power Connections.....	13
3 Configuration	15
3.1 Configuration File	15
3.2 [Module].....	17
3.3 [DF1 Port]	18
3.4 [Commands]	22
3.5 Uploading and Downloading the Configuration File.....	26
4 Diagnostics and Troubleshooting	33
4.1 The Configuration/Debug Menu	33
4.2 Required Hardware	34
4.3 Required Software.....	34
4.4 Using the Configuration/Debug Port	35
4.5 LED Status Indicators.....	48
5 Reference	49
5.1 Product Specifications.....	49
5.2 Cable Connections	51
5.3 Functional Overview.....	56
5.4 Error/Status Data.....	76
5.5 Error Codes	78
5.6 DF1 Command Set For ProSoft Technology Communication Modules	80
5.7 DF1 Command List Form.....	93
6 Support, Service & Warranty	95
6.1 How to Contact Us: Technical Support	95
6.2 Return Material Authorization (RMA) Policies and Conditions.....	96

6.3	LIMITED WARRANTY	97
Index		101

1 Quick Start

This section describes the procedure to be followed for installing and configuring the module for communication. These steps should be followed for successful implementation of a module in a user application.

- 1 Define the communication characteristics of the DF1 master or slave port.
- 2 If a master device will be simulated, define the command lists to be used on the DF1 master port.
- 3 Fill in the blank configuration form for application using the data sets defined in steps 1 and 2.
- 4 Edit the configuration text file `Master.Cfg` or `Slave.Cfg` to reflect the desired data from the configuration form and save the file under a different name. These text files are provided as a starting point to ease application development.
- 5 Connect the MVI94-DFCM module's Configuration/Debug Port to a computer containing HyperTerminal with a null-modem cable.
- 6 Connect the module to a 24 VDC power source.
- 7 Select the directory containing the correct configuration file on the computer.
- 8 Start the terminal emulation program on the computer.
- 9 Press the **[?]** key on the terminal to verify that the module is communicating with the computer and that the main menu mode is current.
- 10 Press the **[R]** key on the terminal emulator to select the receive option and confirm the request by pressing the **[Y]** key.
- 11 Press the **[ALT-F3]** key (DOS Version) or select the **SEND** option on the **TRANSFER** Menu (Windows Version) on the terminal emulator and enter the name of the configuration file to be loaded into the module. The configuration will be downloaded, and the module will restart using the new configuration.
- 12 Connect the module's application port to the DFCM network. If everything is configured correctly and the cable connections are correct, communications should be present on the port.
- 13 Monitor the communication statistics for the port to verify that everything is working correctly.
- 14 View the virtual Modbus database in the module using the terminal emulator.
- 15 Edit the ladder logic provided with the module for your system. The ladder logic is responsible for transferring the data between the module and processor.
- 16 Connect the module to the Flex I/O processor or ControlNet processor. If all is configured correctly, the data in the module should be visible in the processor.
- 17 Use the Configuration/Debug port to view the backplane transfer statistics.

2 Hardware Installation

In This Chapter

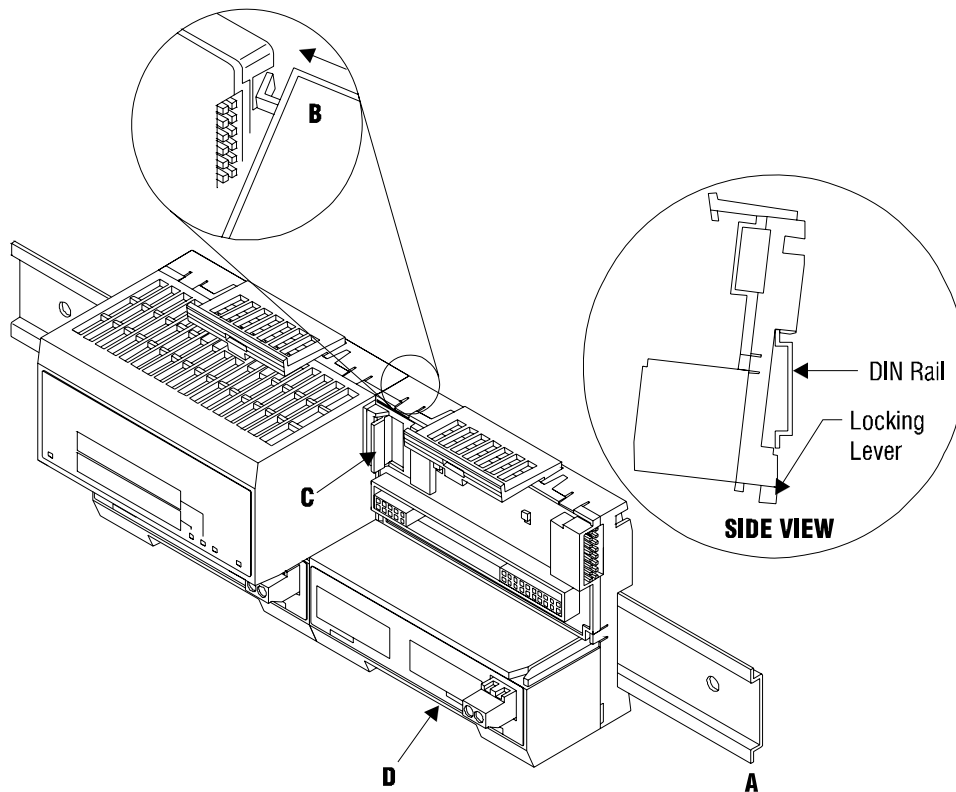
❖ Verify Package Contents	9
❖ Mounting the MVI 94 Flex I/O Base.....	10
❖ Setting Jumpers	11
❖ Installing the Module onto the Base	12
❖ Installing the Serial Adapter Cables	13
❖ Wiring the Power Connections	13

2.1 Verify Package Contents

Make sure that you verify the contents of the product before you discard the packing material. The following components should be included with the product:

- 1 A MVI94 Flex I/O Base
- 2 A MVI94 Module with 3 jumpers installed
- 3 One Serial Adapter Cable

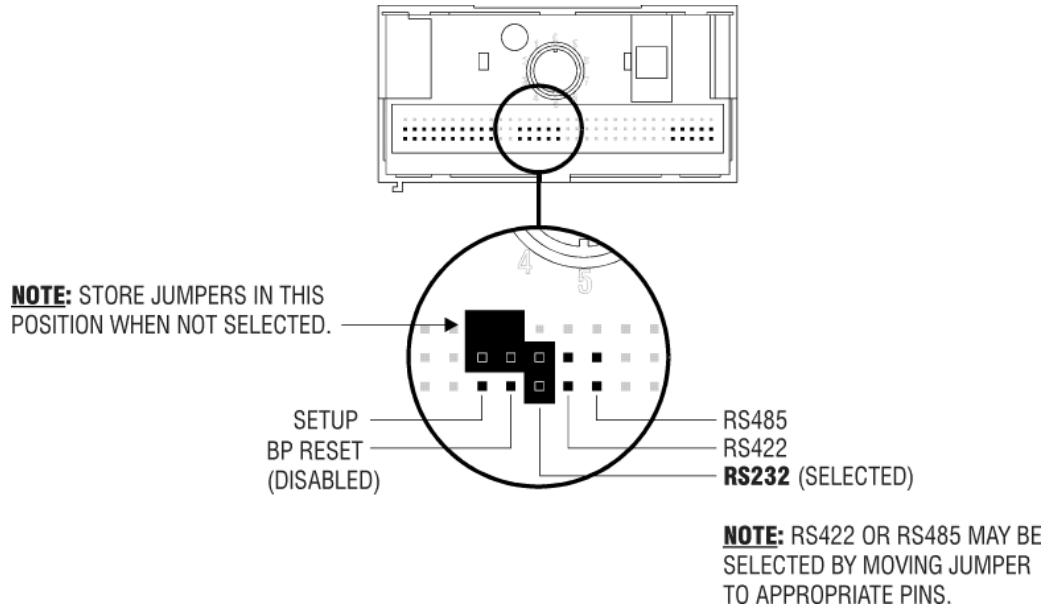
2.2 Mounting the MVI 94 Flex I/O Base



- 1 Remove the cover plug (if used) in the male connector of the unit to which you are connecting this Base.
- 2 Check to make sure that the 16 pins in the male connector on the adjacent device are straight and in line so that the mating female connector on this Base will mate correctly.
- 3 Make certain that the female flexbus connector **C** is **fully retracted** into the Base.
- 4 Position the Base on a 35 x 7.5mm DIN-rail **A** at a slight angle with the hook **B** on the left side of the Base hooked into the right side of the unit on the left.
- 5 Rotate the Base onto the DIN-rail with the top of the rail hooked under the lip on the rear of the Base. Use caution to make sure that the female flexbus connector does not strike any of the pins in the mating male connector.
- 6 Press the terminal base down onto the DIN-rail until flush. The locking tab **D** snaps into position and locks the terminal base to the DIN-rail.
- 7 If the Base does not lock in place, use a screwdriver or similar device to move the locking tab down, press the Base flush with the DIN-rail and release the locking tab to lock the base in place.
- 8 **Gently** push the female flexbus **C** connector into the adjacent base or adapter male connector to complete the flexbus connections.

2.3 Setting Jumpers

Before installing the MVI94 module onto its base, the module's configuration can be set using the jumpers on the bottom of the module as shown in this figure.



Port 2 RS-232/422/485: Select with jumper (shipped in 232).

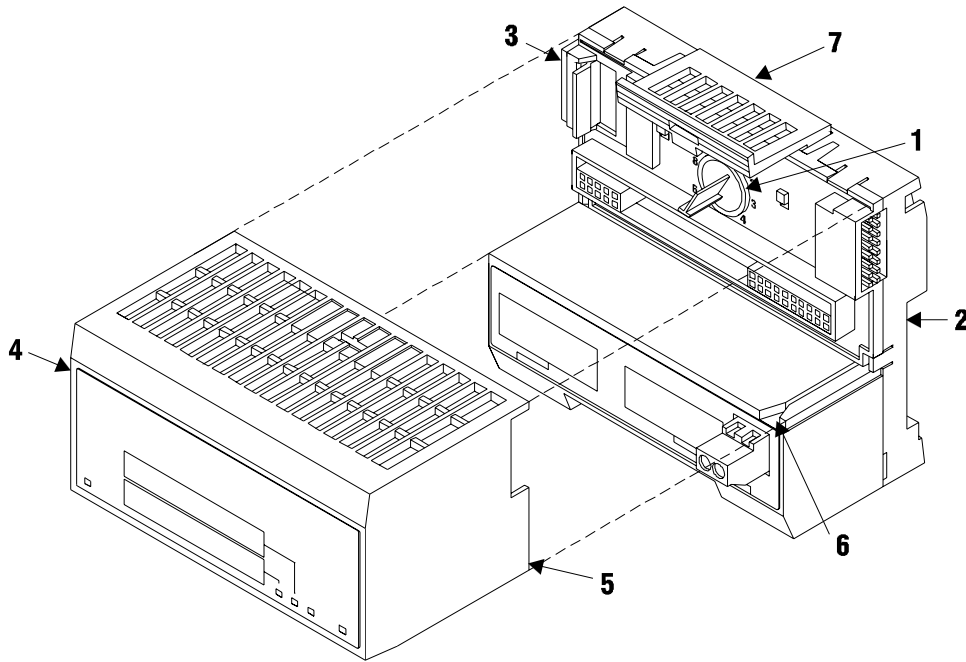
BP Reset: If the MVI94 module is to be reset when the Flex Bus is reset, install the BP RESET jumper in the Enabled position.

ATTENTION: Do not remove or replace a base unit when power is applied. Interruption of the flexbus can result in unintended operation or machine motion.

SETUP: To place the module in SETUP mode, install the jumper in the Selected position (DOS default). To prevent the module from being in Setup mode, leave the jumper in the disabled position.

2.4 Installing the Module onto the Base

- 1 Rotate the keyswitch 1 on the Base clockwise to position #1.
- 2 Make certain the flexbus connector 3 on the Base is pushed all the way to the left to connect with the neighboring base or adapter. **The Module cannot be installed unless the flexbus connector is fully extended.**
- 3 Make sure that the pins on the bottom of the Module are straight so they will align properly with the connector socket on the Base.
- 4 Position the Module with its alignment bar 5 aligned with the groove 6 on the Base.

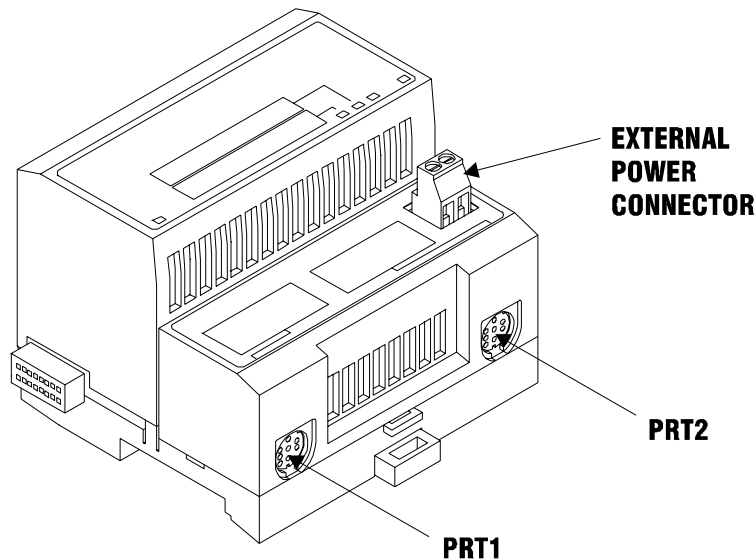


- 5 Press firmly and evenly to seat the Module in the Base. The Module is seated when the latch 7 on the Base is locked into the Module.

2.5 Installing the Serial Adapter Cables

Two identical serial adapter cables are supplied. Each cable has a locking-type 8 pin Mini-DIN plug on one end and a DB-9 male connector on the other end. The Mini-DIN connector on each cable is inserted into the Mini-DIN receptacles marked **PRT1** and **PRT2** on the Base.

To install the locking-type Mini-DIN connector, slide the spring-loaded sleeve back while inserting the plug into the receptacle on the Base, and then release the sleeve when fully seated. The locking mechanism prevents the cable from being removed during normal operation. To remove the cable, slide the sleeve back and remove the plug.



2.6 Wiring the Power Connections

External power is supplied to the Base on the 2 pin screw terminal block. The power supply can be either 24Vdc or 12Vdc, and should be located in close proximity of the base.

- Connect dc common to the **COM** terminal
- Connect +24V dc or +12V dc to the **24VDC** terminal

3 Configuration

In This Chapter

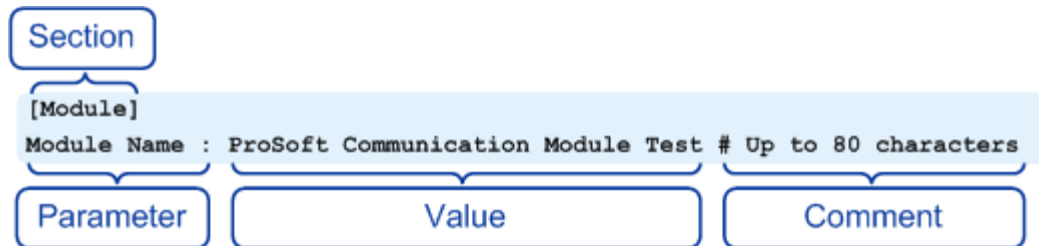
❖ Configuration File	15
❖ [Module].....	17
❖ [DF1 Port].....	18
❖ [Commands].....	22
❖ Uploading and Downloading the Configuration File.....	26

3.1 Configuration File

The MVI94-DFCM module stores its configuration in a text file called Master.Cfg or Slave.Cfg, located in the module's flash memory. When the module starts up, it reads the configuration file and uses the information to control how the DF1 Master/Slave protocol interacts with the module's application port(s).

The configuration file is arranged in *Sections*, with a heading in [] characters at the beginning of each section. Each *Section* contains a list of *Parameters* and *Values*, followed by an optional *Comment* that explains the parameter.

The following illustration shows an example of a *Section*, a *Parameter*, a *Value*, and a *Comment*.



The *Parameter* must be followed by a [:] (colon) character. The text following the [:] is a *Value*.

The module ignores "*comment*" text following the [#] character. Use comments to document your configuration settings.

You can get a sample configuration file for the module in the following places:

- Copy (page 26) the Master.Cfg or Slave.Cfg from the module's flash memory to your PC
- Copy the Master.Cfg or Slave.Cfg from the ProSoft Solutions CD-ROM supplied with the module
- Download the Master.Cfg or Slave.Cfg from the ProSoft Technology web site at www.prosoft-technology.com

3.1.1 Editing the Configuration File

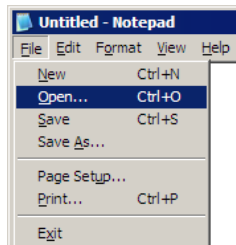
The Master.Cfg or Slave.Cfg file is a plain ASCII text file. Use a text editor such as Notepad.exe (included with Microsoft Windows) to open and edit the file.

To open the configuration file in Notepad

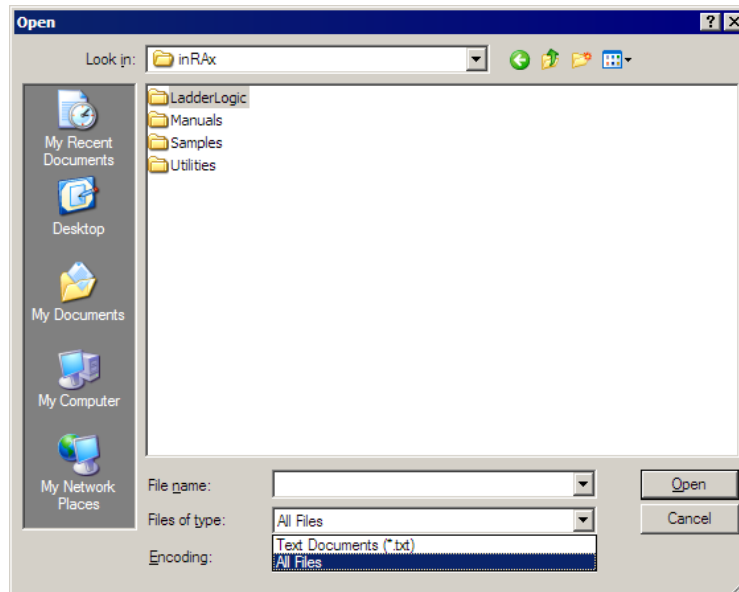
- 1 Click the Start button, and then choose Programs
- 2 Expand the Programs menu, and then choose Accessories.
- 3 On the Accessories menu, choose **Notepad**.



- 4 In Notepad, open the File menu, and then choose Open



- 5 In the Open dialog box, select "All Files" in the Files of Type: dropdown list.



Tip: Sample configuration files are stored under the LadderLogic folder on the ProSoft Solutions CD-ROM.

- 6 Navigate to the folder containing the configuration file, and then select the file to edit.
- 7 Click Open to open the file.
- 8 When you have finished editing, save the file and close Notepad.

Important: Changes to the configuration file will not take effect until you download the file to the module, and then reboot the module.

3.2 [Module]

The [MODULE] section configures the general module parameters for the user application.

3.2.1 *Error/Status Pointer:*

-1 to 3971

This value represents the relative starting position in the module's internal Modbus database where the Error/Status data will be stored. The table can be placed anywhere in the module's data space. If a value of -1 is set for the parameter, the data will not be placed in the database.

3.2.2 *Read Start Register:*

0 to 3995

This parameter specifies the starting register in the internal database to write over the backplane from the module (processor read block).

3.2.3 *Read Register Count:*

12 to 3996

This parameter specifies the number registers in the internal database to write over the backplane. This parameter computes the number of blocks to transfer from the module to the backplane. The number of blocks must be ≥ 2 for proper backplane data transfer.

3.2.4 *Write Start Register:*

0 to 3995

This parameter specifies the starting register in the internal database to fill with data read over the backplane to the module (processor write block).

3.2.5 *Write Register Count:*

12 to 3996

This parameter specifies the number registers in the internal database to be considered from the read operations over the backplane. This parameter computes the number of blocks to be transferred from the backplane to the module. The number of blocks must be ≥ 2 for proper backplane data transfer.

3.2.6 Backplane Fail Count

0 to 65535

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.

3.3 [DF1 Port]

The [DF1 PORT] section of the configuration file sets the DF1 port type and communication parameters, and defines the command list specifications for a master port.

3.3.1 Type

0 or 1

This parameter defines if the port is to simulate a DF1 master or slave device. A value of 0 is used for a master, and a value of 1 is used for a slave.

3.3.2 Local Station ID

0 to 254

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.

3.3.3 Protocol

0 or 1

This parameter specifies the DF1 protocol to apply on the port: 0=full duplex, 1=half-duplex

3.3.4 Termination Type

0 or 1

This parameter specifies the error checking for all DF1 messages. 0=BCC, 1=CRC

3.3.5 Baud Rate

This is the baud rate to be used on the port. Enter the baud rate as a value. For example, to select 19K baud, enter 19200.

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
19,200	19, 192 or 19200
28,800	28, 288 or 28800
57,600	57 or 576
115,200	115 or 1152

3.3.6 Parity

0 to 4

This is the Parity code to be used for the port. The coded values are as follows: 0=None, 1=Odd, 2=Even, 3=Mark and 4=Space.

3.3.7 Data Bits

5 to 8

This parameter sets the number of data bits for each word used by the protocol.

3.3.8 Stop Bits

1 or 2

This parameter sets the number of stop bits to be used with each data value sent.

3.3.9 Minimum Response Delay

0 to 65535

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.

3.3.10 RTS On

0 to 65535

This parameter sets the number of milliseconds to delay after RTS is asserted before the data will be transmitted.

3.3.11 RTS Off

0 to 65535

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.

3.3.12 Use CTS Line

Y or N

This parameter specifies if the CTS modem control line is to be used. If the parameter is set to N, the CTS line will not be monitored. If the parameter is set to Y, 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).

3.3.13 ENQ Delay:

0 to 65535

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.

3.3.14 Command Count

0 to 100

This parameter specifies the number of commands to be processed for the port.

3.3.15 Minimum Command Delay

0 to 65535

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.

3.3.16 Command Error Pointer

-1 to 3895

This parameter sets the address in the internal Modbus 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.

3.3.17 Response Timeout

0 to 65535

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.

3.3.18 Retry Count

0 to 10

This parameter specifies the number of times a command will be retried if it fails.

3.3.19 Error Delay Count:

0 to 65535

This parameter specifies the number of polls to skip 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.

3.3.20 Slave List Pointer

-1 to 3739

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 3739.

3.4 [Commands]

The [COMMANDS] section of the configuration file sets the DF1 master port command list. This list polls DF1 slave devices attached to a simulated master port. The module supports numerous commands. This permits the module to interface with a wide variety of DF1 protocol devices.

The command list is formatted differently than the other sections of the configuration file. Commands are present in a block between the labels **START** and **END**. These labels inform the program where the list resides. The module's program will parse all commands after the **START** label until it reaches the **END** label or until the command count entered for the port is reached. The following table describes the format of this section.

[Section]/Item	Value	Range	Description
[COMMANDS]			Start header for DF1 Master Command Data
START	N/A	N/A	Start of command list
			List of up to 100 DF1 commands
END	N/A	N/A	End of command list

Function codes used for each command are specific to the module and are defined to permit the module to interface with a wide variety of DF1 communication devices. Refer to DF1 Command Set (page 80) for the structure and content of each command.

3.4.1 Command List Overview

In order to interface the module with other nodes, you must construct a command list of up to 100 user-defined commands. The commands in the list specify the device to be addressed, the function to be performed (read or write), the data area in the device to interface with, and the registers in the internal database to be associated with the device data. The command list is processed from top (command #0) to bottom. A poll interval parameter is associated with each command to specify a minimum delay time in tenths of a second between the issuance of a command. If the user specifies a value of 10 for the parameter, the command is executed no more frequently than every (1) second.

Write commands have a special feature, as they can be set to execute only if the data in the write command changes. If the register data values in the command have not changed since the command was last issued, the command will not be executed. If the data in the command has changed since the command was last issued, the command is executed. Use of this feature can lighten the load on the network. In order to implement this feature; set the enable code for the command to a value of 2.

The module supports numerous commands. This permits the module to interface with a wide variety of devices.

The commands take the following parameters:

Enable

This field defines whether the command is to be executed and under what conditions. A **0** indicates that the command is disabled and will not be executed in the normal polling sequence. A **1** indicates that the command is executed each scan of the command list if the Poll Interval Time is set to zero. If the Poll Interval Time is set, the command is executed when the internal timer expires. A **2** indicates that the command only executes if the internal data associated with the command changes. This value is valid for write commands only.

Internal Address

This field specifies the database address in the module's internal database to be associated with the command. If the command is not a read function, the data received in the response message is placed at the specified location. If the command is a write function, data used in the command is sourced from the specified data area. Valid values are 0 to 3999.

Poll Interval

This parameter specifies the minimum interval to execute continuous commands (Enable code of 1). The parameter is entered in 1/10th of a second. Therefore, if a value of .50 is entered for a command, the command executes no more frequently than every 5 seconds. Valid values are 0 to 65535.

Count

This parameter specifies the number of registers or digital points to be associated with the command.

Swap Code

This parameter defines if the data received from the server is to be ordered differently than that received from the server device. This parameter is helpful when dealing with floating point or other multi-register values, as there is no standard method of storage of these data types in server devices. This parameter can be set to order the register data received in an order useful by other applications. The following defines the values and operations:

Swap Code	Description
0	None - No Change is made in the byte ordering (1234 = 1234)
1	Words - The words are swapped (1234=3412)
2	Words & Bytes - The words are swapped then the bytes in each word are swapped (1234=4321)
3	Bytes - The bytes in each word are swapped (1234=2143)

The words should be swapped only when using an even number of words.

Node Address

The address of the device being addressed by the command.

Function Code

These parameters specify the function to be executed by the command. Refer to DF1 Command Set (page 80) for more information on constructing commands for the module.

3.4.2 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

3.4.3 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

3.4.4 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

Function Code	Command	Function	Definition	PLC5	SLC500 & MicroLogix	Power-monitor II	ControlLogix
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
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.

3.5 Uploading and Downloading the Configuration File

ProSoft modules are shipped with a pre-loaded configuration file. In order to edit this file, you must transfer the file from the module to your PC. After editing, you must transfer the file back to the module.

This section describes these procedures.

Important: The illustrations of configuration/debug menus in this section are intended as a general guide, and may not exactly match the configuration/debug menus in your own module. For specific information about the configuration/debug menus in your module, refer to The Configuration/Debug Menu (page 33).

3.5.1 Required Hardware

You can connect directly from your computer's serial port to the serial port on the module to view configuration information, perform maintenance, and send (upload) or receive (download) configuration files.

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 UART hardware-based serial communications port available. USB-based virtual UART systems (USB to serial port adapters) often do not function reliably, especially during binary file transfers, such as when uploading/downloading configuration files or module firmware upgrades.
- A null modem serial cable.

3.5.2 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).

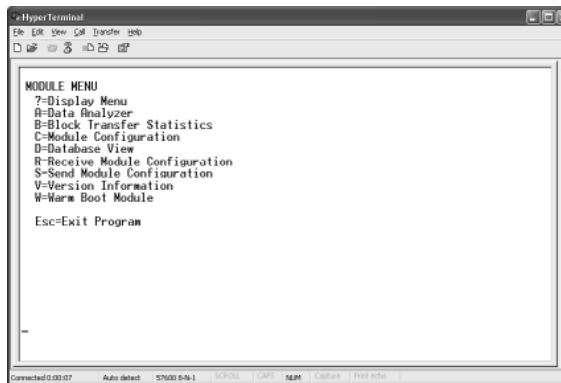
A simple communication program called HyperTerminal is pre-installed with recent versions of Microsoft Windows operating systems. If you are connecting from a machine running DOS, you must obtain and install a compatible communication program. The following table lists communication programs that have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

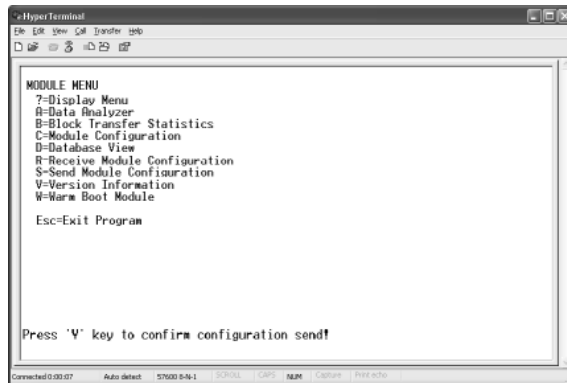
The module uses the Ymodem file transfer protocol to send (upload) and receive (download) configuration files from your module. If you use a communication program that is not on the list above, please be sure that it supports Ymodem file transfers.

3.5.3 Transferring the Configuration File to Your PC

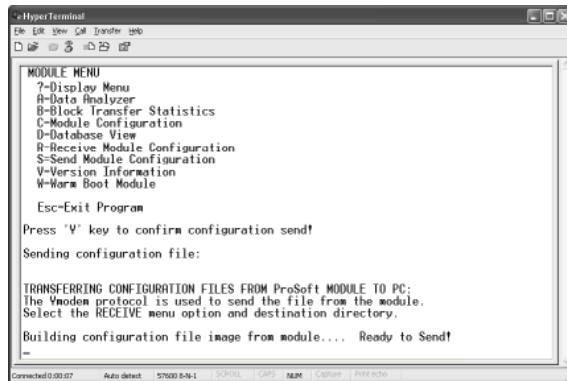
- 1 Connect your PC to the Configuration/Debug port of the module using a terminal program such as HyperTerminal. Press [?] to display the main menu.



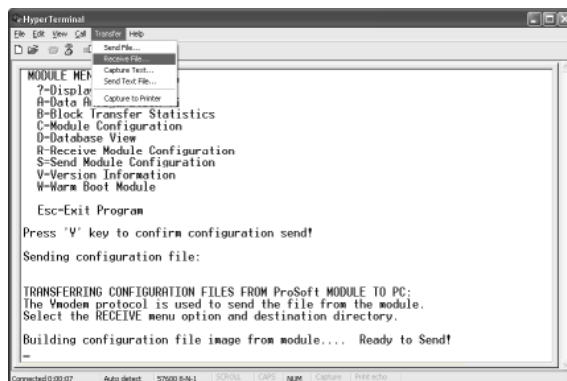
- 2 Press **[S]** (Send Module Configuration). The message "Press Y key to confirm configuration send!" is displayed at the bottom of the screen.



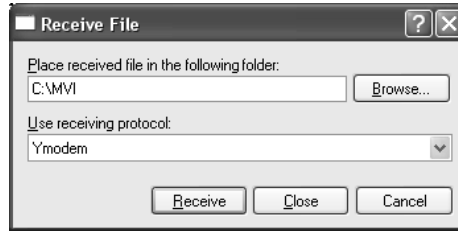
- 3 Press **[Y]**. The screen now indicates that the module is ready to send.



- 4 From the **Transfer** menu in HyperTerminal, select **Receive File**. This action opens the Receive File dialog box.

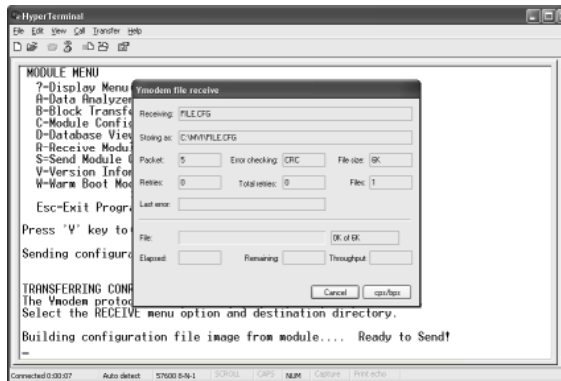


- 5 Use the Browse button to choose a folder on your computer to save the file.

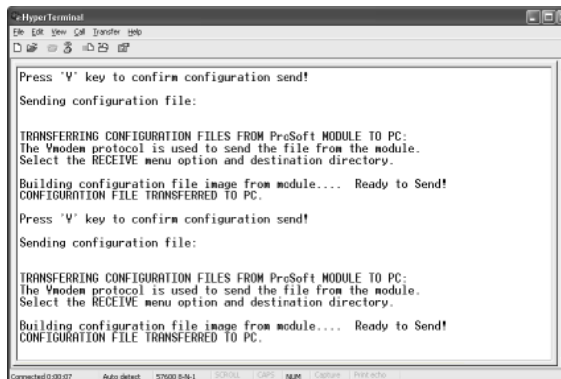


- **Note:** ProSoft Technology suggests that you upload the configuration file pre-loaded on your module. However, configuration files are also available on the ProSoft CD as well as the ProSoft Technology web site at www.prosoft-technology.com.

- 6 Select Ymodem as the receiving protocol.
- 7 Click the Receive button. This action opens the Ymodem File Receive dialog box, showing the progress of your file transfer.



When the configuration file has been transferred to your PC, the dialog box will indicate that the transfer is complete.



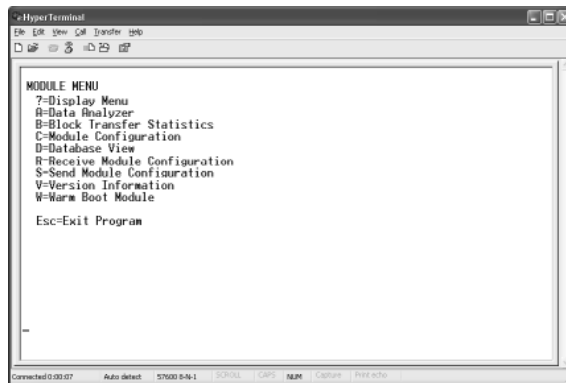
The configuration file is now on your PC at the location you specified.

- 8 You can now open and edit the file in a text editor such as Notepad. When you have finished editing the file, save it and close Notepad.

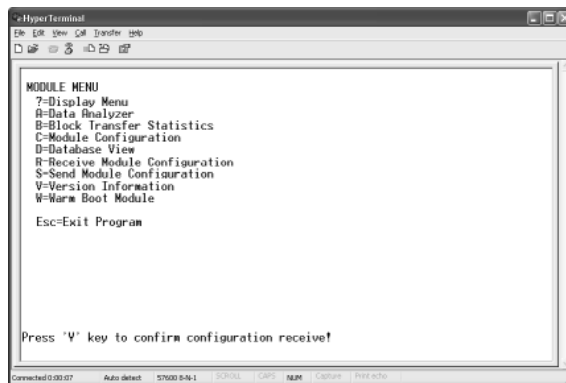
3.5.4 Transferring the Configuration File to the Module

Perform the following steps to transfer a configuration file from your PC to the module.

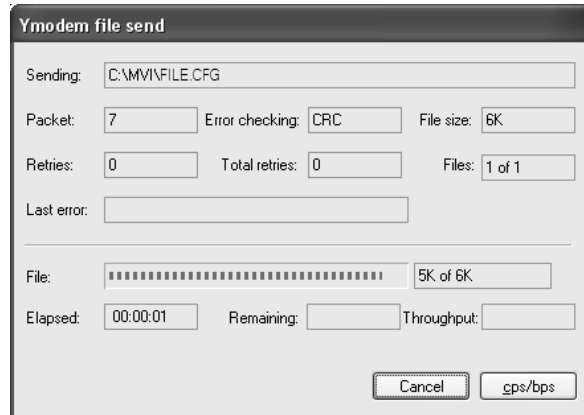
- 1 Connect your PC to the Configuration/Debug port of the module using a terminal program such as HyperTerminal. Press **[?]** to display the main menu.



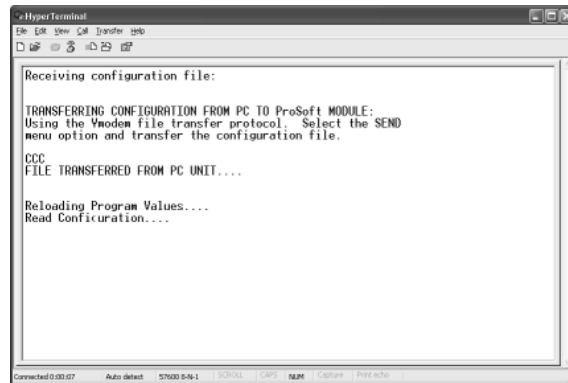
- 2 Press **[R]** (Receive Module Configuration). The message "Press Y key to confirm configuration receive!" is displayed at the bottom of the screen.



- 7 Click the Send button. This action opens the Ymodem File Send dialog box.



When the file transfer is complete, the module's configuration/debug screen indicates that the module has reloaded program values, and displays information about the module.



- 8 Your module now contains the new configuration.

4 Diagnostics and Troubleshooting

In This Chapter

❖ The Configuration/Debug Menu	33
❖ Required Hardware	34
❖ Required Software.....	34
❖ Using the Configuration/Debug Port.....	35
❖ LED Status Indicators.....	48

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 The Configuration/Debug Menu

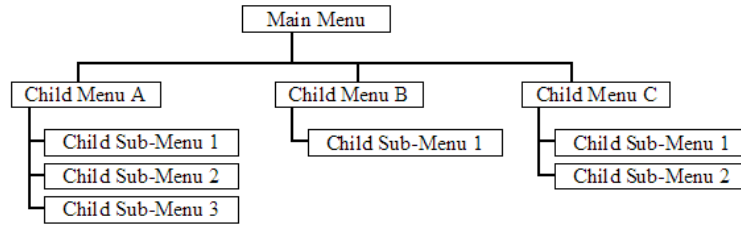
The Configuration and Debug menu for this module is arranged as a tree structure, with the Main Menu at the top of the tree, and one or more sub-menus for each menu command. The first menu you see when you connect to the module is the Main menu.

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.

4.1.1 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.

4.1.2 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.

4.2 Required Hardware

You can connect directly from your computer's serial port to the serial port on the module to view configuration information, perform maintenance, and send (upload) or receive (download) configuration files.

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 UART hardware-based serial communications port available. USB-based virtual UART systems (USB to serial port adapters) often do not function reliably, especially during binary file transfers, such as when uploading/downloading configuration files or module firmware upgrades.
- A null modem serial cable.

4.3 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

The module uses the Ymodem file transfer protocol to send (upload) and receive (download) configuration files from your module. If you use a communication program that is not on the list above, please be sure that it supports Ymodem file transfers.

4.4 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.
- 3 Verify that your communication software is using the correct settings for baud rate, parity and handshaking.
- 4 On computers with more than one serial port, verify that your communication program is connected to the same port that is connected to the module.

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

4.4.1 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
U=Version Information
W=Warm Boot Module
V=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 Master/Slave 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.

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

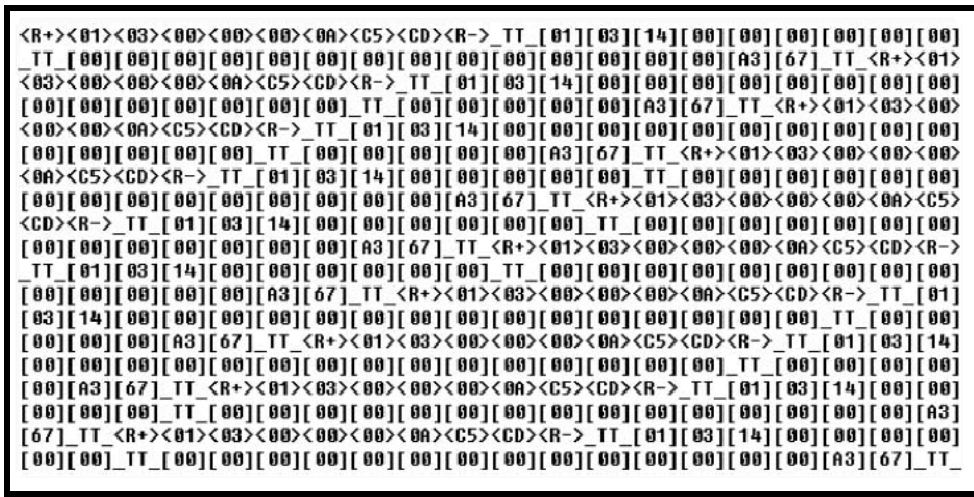
Press [H] to display the data on the current page in hexadecimal format.

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. The following illustration shows an example.



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.

4.4.3 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

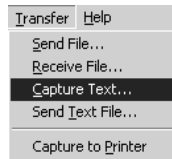
Port = 1, Format=HEX, Tick=10
```

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

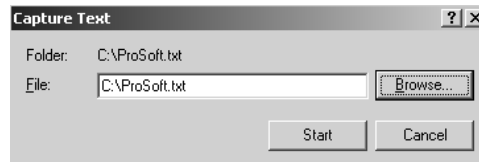
For most applications, HEX is the best format to view the data, and this does include ASCII based messages (because some characters will not display on HyperTerminal and by capturing the data in HEX, we can figure out what the corresponding ASCII characters are supposed to be).

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 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><0A><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__[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><0A><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><0A><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.

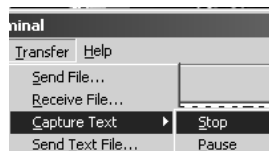
All characters shown in <> brackets are characters being sent out by the module.

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.

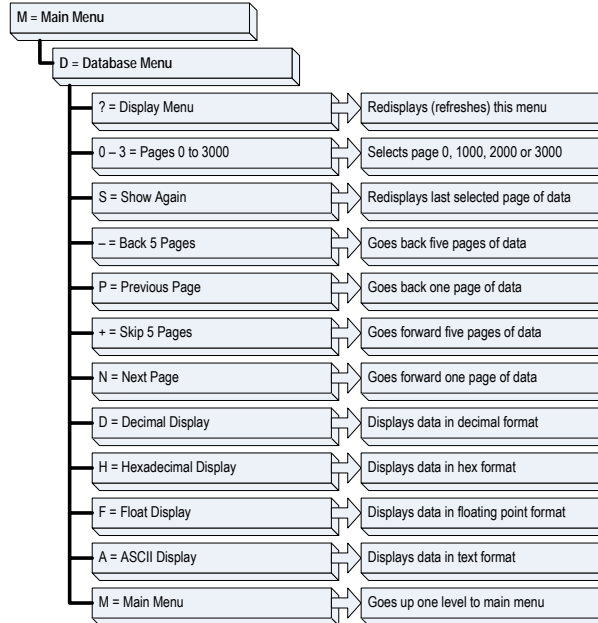
When you have captured the data you want to save, open the Transfer menu and choose Capture Text. On the secondary menu, choose Stop.



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.

4.4.4 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

Press **[S]** from the Database View menu to show 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
  
```

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 five pages back in the database to see the previous 100 registers of data.

Moving Forward Through 5 Pages of Registers

Press **[+]** from the Database View menu to skip five pages ahead in the database to see the next 100 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.

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

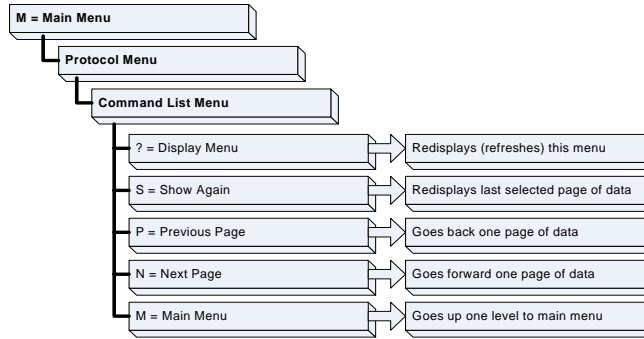
Press **[A]** to display the data on the current page in ASCII format. This is useful for regions of the database that contain ASCII data.

Returning to the Main Menu

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

4.4.5 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

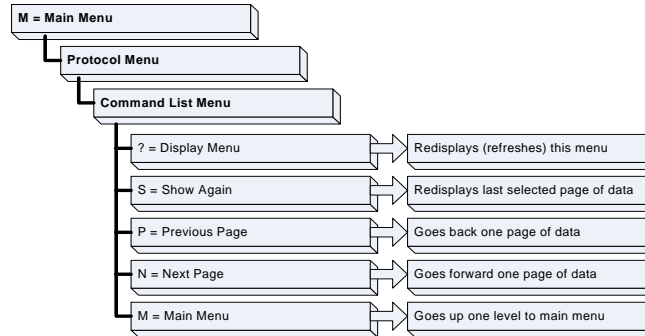
Press **[N]** to display the next page of commands.

Returning to the Main Menu

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

4.4.6 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.

4.5 LED Status Indicators

This section defines the indications provided on the MVI94-DFCM module through LEDs. A description of each LED is provided in the following topics.

4.5.1 PRT1

This LED indicates data transmit and receive activity on the configuration port. When the TXD or RXD pin is active on the port, the LED will be illuminated green. When the port is not active, the LED will be in the off state.

4.5.2 U1

This LED indicates backplane data transfer operation. When the module is successfully writing data to the FLEX I/O backplane, the LED will be in the off state. When the module is reading a new block of data from the FLEX I/O backplane, the LED will be in the on state (amber). During normal operation of the module, this LED should turn on and off at a very rapid rate. If the LED never turns on, check your ladder logic to verify that the data transfer is set up correctly.

4.5.3 U2

This LED indicates communication errors on the DF1 application port. The LED is off when no error exists on the port. If a communication error is recognized on the port, the LED will be illuminated. If the LED is turned on, check for errors in the command list to determine the error condition recognized by the module.

4.5.4 STS

This LED indicates the "health" of the module. When power is applied to the module, the LED will be illuminated. If the LED is green, the program is working correctly and the user configuration is being used. If the LED is red, the program is halted. Try restarting the module by cycling power. This should cause the module to return to its normal state.

4.5.5 PRT2

This LED indicates data transmit and receive activity on the DF1 master/slave port. When the TXD or RXD pin is active on the port, the LED will be illuminated green. When the port is not active, the LED will be in the off state.

5 Reference

In This Chapter

❖ Product Specifications	49
❖ Cable Connections	51
❖ Functional Overview	56
❖ Error/Status Data.....	76
❖ Error Codes	78
❖ DF1 Command Set For ProSoft Technology Communication Modules.	80
❖ DF1 Command List Form	93

5.1 Product Specifications

The MVI94 DF1 Master/Slave Communication Module is a FLEX backplane compatible module that allows Rockwell Automation FLEX compatible processors to interface easily 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.

The ProSoft Technology MVI94-DFCM module is the fastest and easiest way to add DF1 Half/Full Duplex Master and Slave capability to the FLEX platform. The MVI94-DFCM module is a single I/O slot, backplane compatible solution for the Allen-Bradley FLEX platform. This module has one powerful and fully configurable DF1 Half/Full Duplex Master/Slave port allowing the many SCADA and field devices supporting the DF1 protocol to be integrated into the FLEX platform.

The module has one serial port supporting the DF1 protocol, user-configurable to act as a master or as a slave. Data transfer between the module and the FLEX 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

Some of the general specifications include:

- Operation via simple ladder logic
- Complete setup and monitoring of module through Debug port and user configuration file
- Flex backplane interface via I/O access

5.1.2 FLEX I/O Interfaces

Specification	Description
Form Factor	Single Slot 1794 Backplane compatible Locate in any slot of Backplane
Backplane current load	20 mA @ 5 V
External power supply	12V to 24VDC 340 mA to 170 mA
Operating temperature	0 to 55°C (32 to 131°F)
Storage temperature	-40 to 85°C (-40 to 185°F)
Shock	30g operational 50g non-operational 5g from 10150 Hz
Relative humidity	5% to 95% (non-condensing)
LED indicators	Module status Backplane transfer status Application status Serial activity and error LED status
Configuration Serial port (PRT1)	Mini-DIN RS-232 Hardware handshaking
Application serial Port (PRT2)	Mini-DIN RS-232/422/485 jumper selectable 500V optical isolation from backplane
Dimensions (with Module installed in Base)	3.7H x 3.7W x 2.7D inches 94H x 94W x 69D mm

5.1.3 Functional Specifications

DF1 port

- Full and half duplex modes supported
- CRC and BCC error checking
- User-definable module memory usage, supporting the storage and transfer of up to 4000 registers to/from the control processor
- 125 word read and write command lengths supported
- Floating point data movement supported

DF1 Master Protocol Specifications

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

- Command List: Up to 100 commands, 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 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)

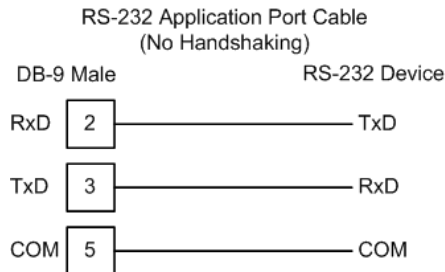
5.2 Cable Connections

The application ports on the MVI94-DFCM 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.

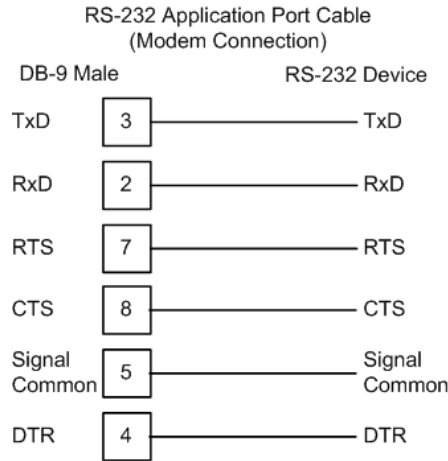
5.2.1 RS-232

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

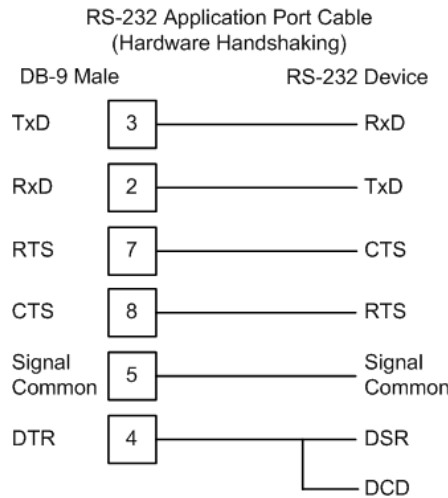
This type of connection is required between the module and a modem or other communication device.



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

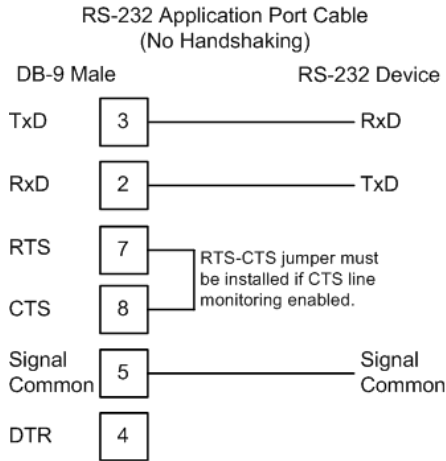
RS-232: Null Modem Connection (Hardware Handshaking)

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)

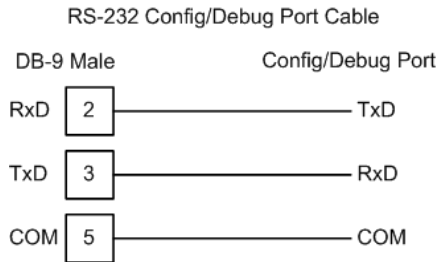
This type of connection can be used to connect the module to a computer or field device communication port.



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.

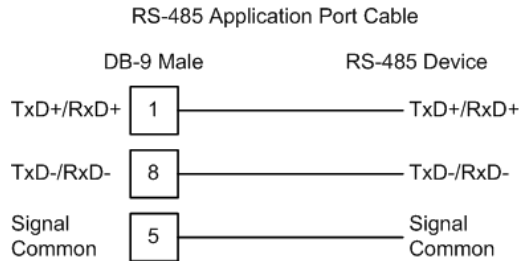
5.2.2 RS-232 Configuration/Debug Port

This port is physically a Mini-DIN connection. A Mini-DIN 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:



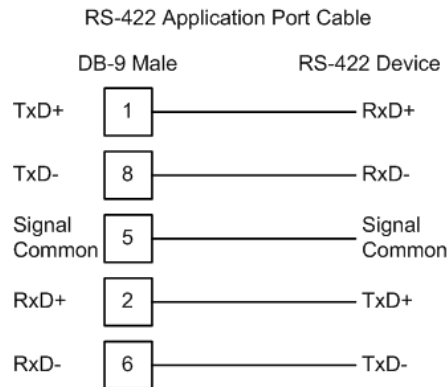
5.2.3 RS-485

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.

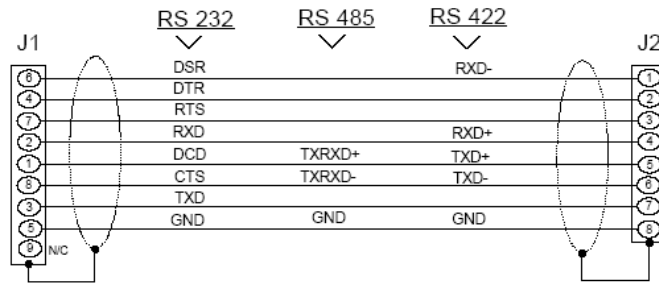
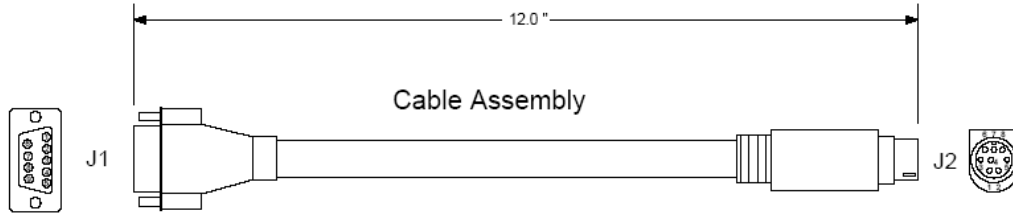
5.2.4 RS-422



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.

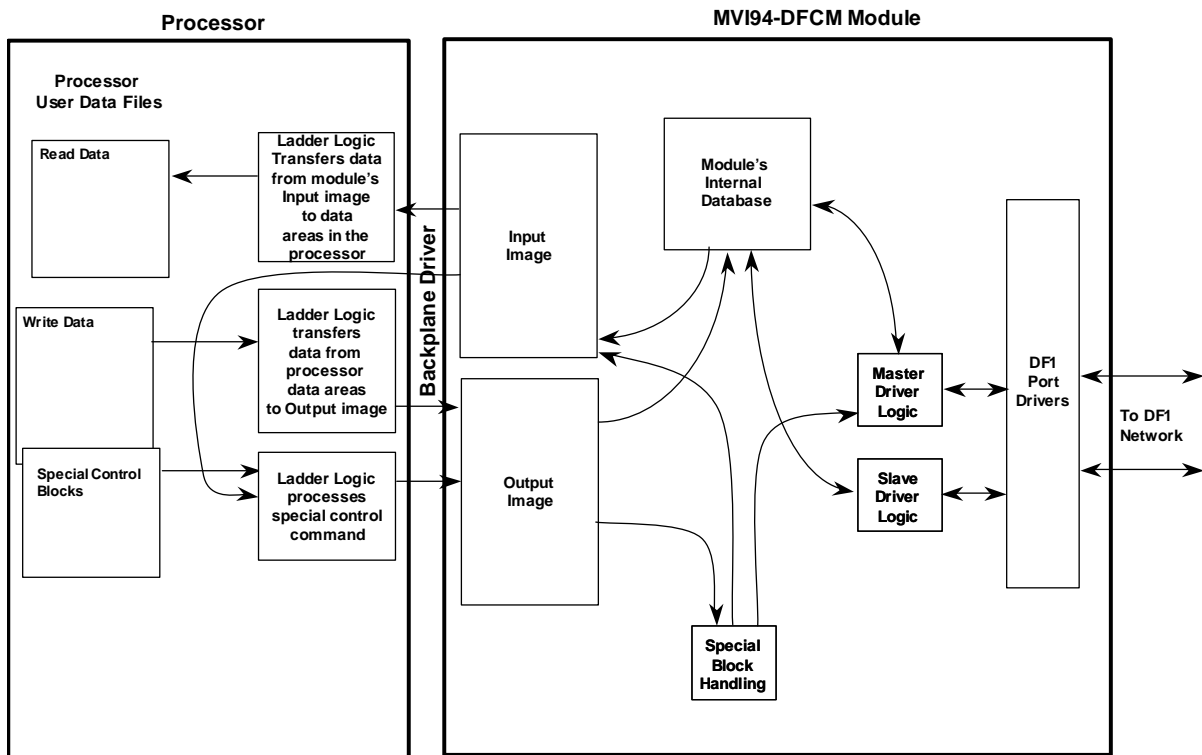
5.2.5 DB9 to Mini-DIN Adaptor (Cable 09)



Wiring Diagram

5.3 Functional Overview

The MVI94-DFCM communication module interfaces DF1 slave or master devices with the Flex I/O system. The module contains a virtual Modbus database that is defined by the user. This database is used for the request and command messages sent from the DF1 master port to DF1 slave devices. If the module is used as a virtual DF1 slave device, the database is used to source read requests and is the destination of remote master write requests. Data areas in the virtual Modbus database can be reserved for status and error information generated by the module under user control. The following illustration shows the data flow within the module and between the DF1 network and backplane.



The virtual Modbus database also interfaces with the Flex I/O system using the Flex I/O bus (backplane). Data is made available to the PLC or any processor on a ControlNet network using this backplane interface. Input and output image tables in the module present the data in the virtual Modbus database to the backplane.

If the module is used as a DF1 master device on the application port, it can continuously poll DF1 slave devices. Up to 100 user-defined commands can be defined for the port. Data read from the slave devices are placed in the virtual Modbus database. Any write requests for the slave devices are sourced with data from the virtual Modbus database.

The module can be configured to place slave devices that are not responding to commands at a lower priority. If the module recognizes a slave device has failed to respond to a message after the user defined retry count, it will mark the slave as "in communication failure" and set the error delay counter to the specified value. Each time the module encounters this slave in the command list, the counter will be decremented. When the value reaches zero, the slave will be placed in an active status. This facility can improve communication throughput on the network. These retries can be viewed through the Configuration/Debug port by selecting the "Communication Status" (1) option. Refer to Diagnostics and Troubleshooting (page 33).

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 previously discussed.

Commands can be activated in the module under processor control. This feature permits the processor to issue a command in the command list under program control. When a command is activated, it will be placed in the command queue for immediate execution. Normal command polling will begin after the command queue is completely processed.

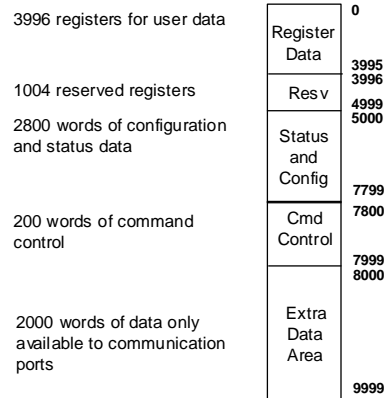
If the module is used as a DF1 slave device, it will respond to requests from a remote DF1 master device. All data in the module's database is available to the remote master device for read and write requests. The slave device emulated can operate in both full- and half-duplex mode.

The module provides a Configuration/Debug port for use with an external computer executing a terminal emulation program. The terminal emulation program provided with the module permits uploading and downloading of the configuration information required by the module. Additionally, the Configuration/Debug port provides a view into the virtual Modbus database, communication statistics and the configuration.

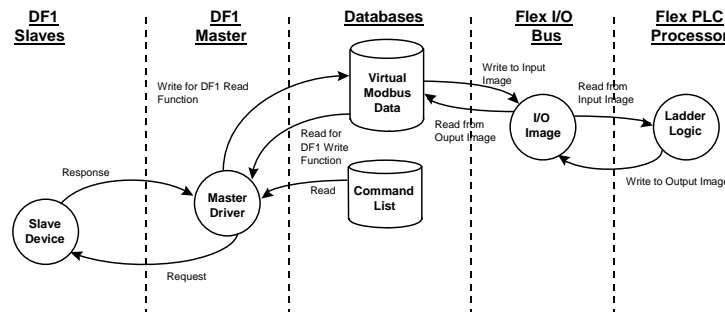
5.3.1 Virtual Modbus Database

Central to the functionality of the module is the virtual Modbus database. This database is used as the interface between remote DF1 devices and the Flex I/O bus. The content and structure of the user data area of the database is completely user defined. The following illustration shows the general format of the module's database.

Module's Internal Database Structure

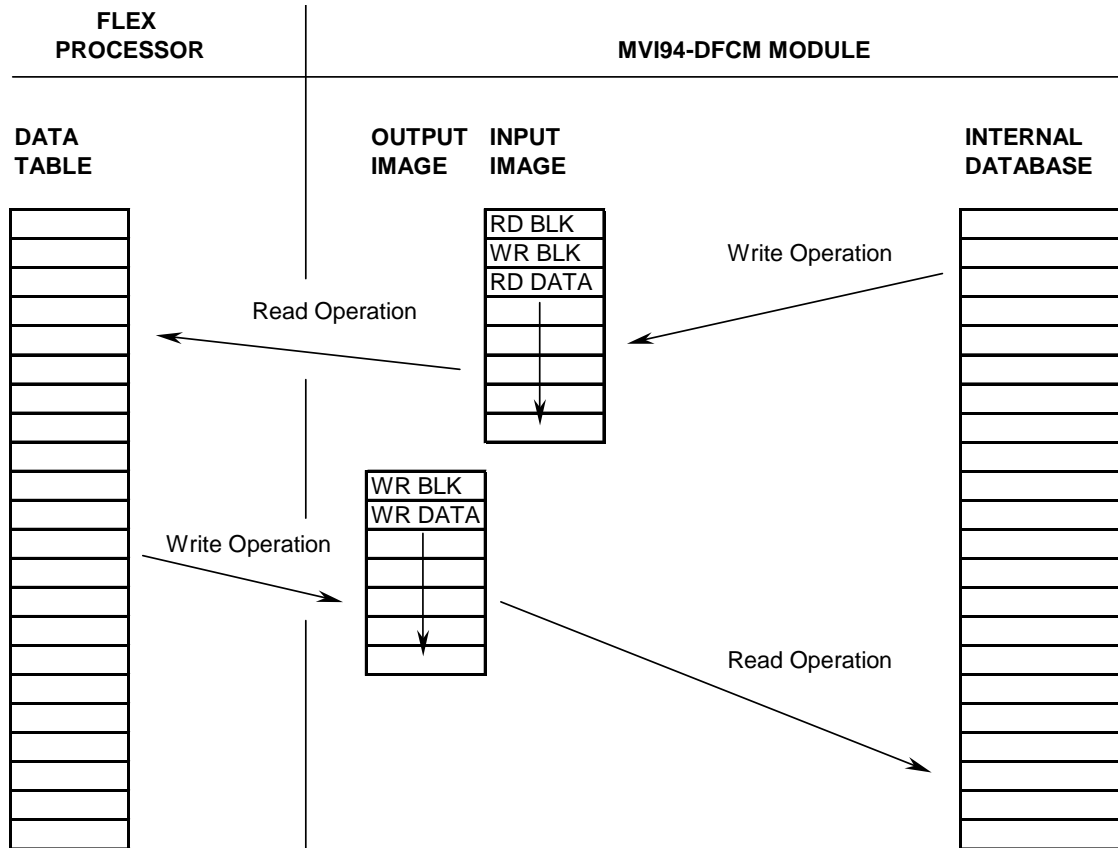


The Flex I/O bus reads data from and write data to the user data area using the backplane interface. The module interfaces data contained in remote DF1 slave devices to the virtual Modbus database using the DF1 application port. User commands are issued out the master port from a command list. These commands gather or control data in the slave devices. The following illustration shows the relationships discussed above for a simulated master device:



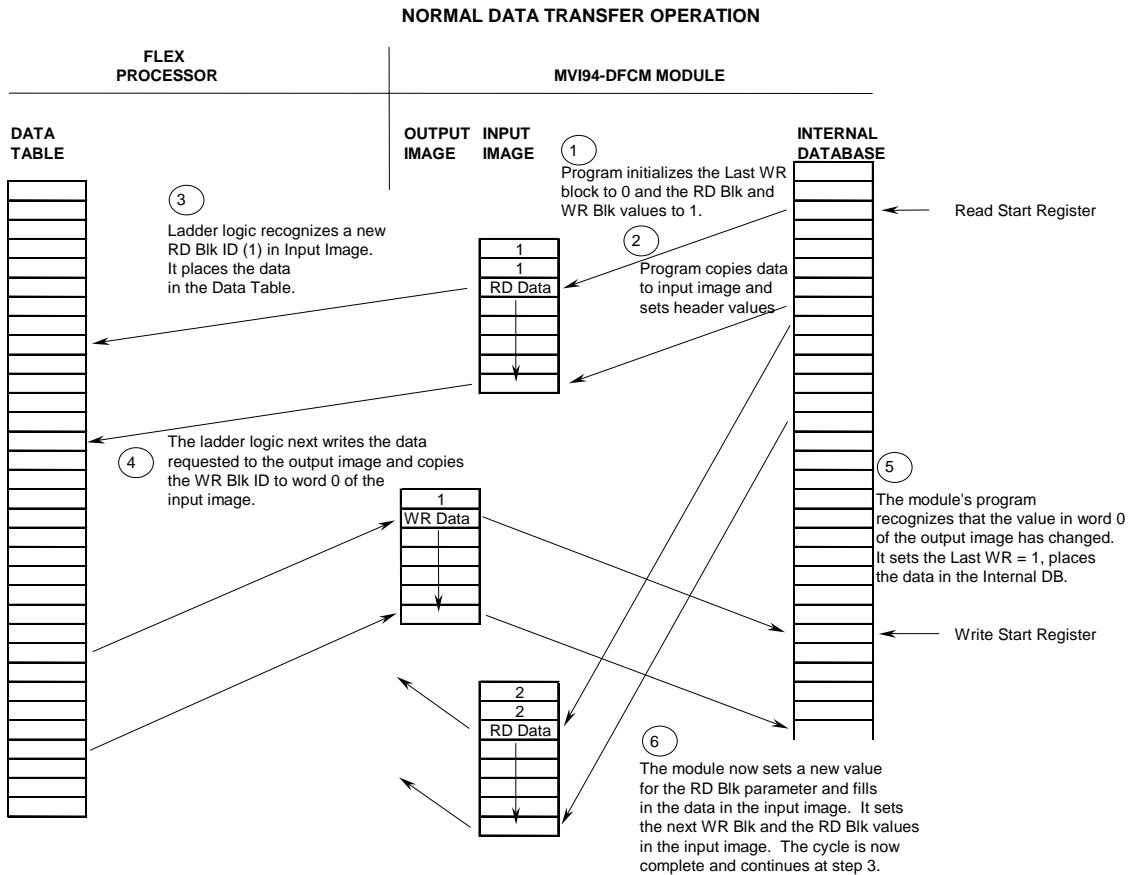
5.3.2 Data Transfer

Data is transferred over the backplane using the module's input and output images. The module is configured with an eight-word input image and a seven-word output image. The module and the Flex processor use these images to page data and commands. The input image is set (written) by the module and is read by the Flex processor. The output image is set (written) by the Flex processor and read by the module. The following illustration shows this relationship.



The module's program is responsible for setting the block identification code used to identify the data block written and the block identification code of the block it wants to read from the processor. User configuration information determines the read (Read Start Register) and write (Write Start Register) locations in the virtual Modbus database and the amount of data transferred (Read Register Count and Write Register Count). Each read and write operation transfers a six-word data area. The processor read operation contains a two-word header that defines the block identification code of the read data and the block identification code of the write block requested. These identification codes are in the range of 0 to 666. A value of zero indicates that the block contains no data and should be ignored. The first valid block identification code is one and refers to the first block of six words to be read or written.

The module and the processor constantly monitor input and output images. How does either one know when a new block of data is available? Recognizing a change in the header information of the image (word 0) solves the problem. For example, when the module recognizes a different value in the first word of the output image, new processor write data is available. When the processor recognizes a new value in the first word of the input image, new processor read data is available. This technique requires the storage of the previously processed data block identification code. The following illustration shows the normal sequence of events for data transfer:



The following steps outline this diagram:

- 1 During program initialization, the write and read block identification codes are set to one.
- 2 The last block write variable is set to zero.
- 3 The program copies the first six-word block of the virtual Modbus database starting at the user defined Read Start Register to the input image (words 2 to 7). It then sets the current write block code in word 1 of the input image. To "trigger" the write operation, the program places the current read block code into word 0 of the input image.

- The Flex processor recognizes a new value in word 0 of the input image (based on the last_read_block_code not equal to read_block_code) in its ladder logic. The ladder logic computes the offset into the file based on the following formula:

```
read_file_offset = (read_block_code - 1) * 6
```

- The new data contained in the input image (words 2 to 7) is copied to the offset in the processor's user data file. The last_read_block_code storage register in the processor is updated with the new read_block_code.

Note: If the data area transferred from the module exceeds the size of a single user file in the Flex processor, logic will be required to handle multiple files.

- Next the ladder logic examines the value of the write_block_code and computes the offset into the processor write data file as follows:

```
write_file_offset = (write_block_code - 1) * 6
```

- The required six-word, write data is copied to the module's output image (words 1 to 6). To "trigger" the transfer operation, the ladder logic moves the write_block_code into word 0 of the output image.
- The module's program recognizes the new write_block_code. It transfers the data to the correct offset in the virtual Modbus database using the following function:

```
Modbus_offset = Write_Start_Register + (write_block_code - 1) * 6
```

- The module sets the last_write_block_code to the value of write_block_code.
- The module now selects the next read and write blocks. The data for the processor read operation is placed in the input image and the write_block_code is set in word 1. The module "triggers" the transfer operation by setting the new read_block_code in word 0 of the input image. The sequence continues at step 3.

The previous discussion is for normal data transfer operation. The following table lists the block identification codes used by the module:

Block Range	Descriptions
1 to 666	User data blocks
3000 to 3001	Port 1 slave polling control
5000 to 5006	Port 1 command control
9998	Warm-boot control block
9999	Cold-boot control block

Data is transferred between the processor and the module using the block identification codes of 1 to 666. The other block codes control the module from the processor's ladder logic. They are implemented when the ladder logic needs to control the module. In order to use one of the blocks, the ladder logic inserts the data and code in the output image of the module. The data should be set before the code is placed in the block. This operation should be performed after the receipt of a new write block from the module. A discussion of each set of codes is given in the following topics.

Slave Polling Control Blocks (Blocks 3000 to 3001)

Slave status data sends status information of each slave device on a master port. This data is passed to the controller through the user database. The location of this data is set with the **Slave List Pointer parameter**. Slaves attached to the master port can have one of the following states:

Word	Description
0	The slave is inactive and not defined in the command list for the master port.
1	The slave is actively being polled or controlled by the master port and communications is successful.
2	The master port has failed to communicate with the slave device. Communications with the slave is suspended for a user defined period based on the scanning of the command list.
3	Communications with the slave has been disabled by the ladder logic. No communication will occur with the slave until this state is cleared by the ladder logic.

Slaves are defined to the system when the module initializes the master command list. Each slave defined will be set to a state of one in this initial step. If the master port fails to communicate with a slave device (retry count expired on a command), the master will set the state of the slave to a value of 2 in the status table. This suspends communication with the slave device for a user specified scan count (**Error Delay Counter** value in the module configuration for each port). Each time a command in the list is scanned that has the address of a suspended slave, the delay counter value will be decremented. When the value reaches zero, the slave state will be set to one. This will enable polling of the slave.

Word	Description
0	This word contains the block identification code of 3000 for Port 1 slave polling disable request.
1	This word contains the number of slaves listed in the block. The valid range of numbers is from 1 to 5.
2 to 6	These words contain the slave indexes to disable in the command list for the selected port. The number of slaves processed is set in word one of the block.

The module will respond with a block with the same identification code received and indicate the number of slaves acted on with the block. The format of this response block (read block in input image) is displayed in the following table.

Word	Description
0	This word contains the block code of 3000 requested by the processor.
1	Write Block ID
2	This word contains the number of slaves processed in the last request. This number should match the value passed in word one of the block request
3 to 7	Not used

Ladder logic can be written to override the value in the slave status table to enable the slave (state value of 1) by sending a special block. Slaves are enabled using block 3001 (write block in output image). This block contains the slave node addresses to enable. The following table describes the format for this block.

Word	Description
0	This word contains the block identification code of 3001 for Port 1 slave polling enable request.
1	This word contains the number of slaves listed in the block. The valid range of numbers is from 1 to 5.
2 to 6	These words contain the slave indexes to enable in the command list for the selected port. The number of slaves processed is set in word one of the block.

The module will respond with a block with the same identification code received and indicate the number of slaves acted on with the block. The format of this response block (read block in input image) is displayed in the following table.

Word	Description
0	This word contains the block code of 3001 requested by the processor.
1	Write Block ID
2	This word contains the number of slaves processed in the last request. This number should match the value passed in word one of the block request.
3 to 7	Not used

Commands (Blocks 5001 to 5006)

These control blocks are sent from the processor to the module to execute one or more commands in the module's command list out the DF1 master port. Commands selected for execution need not have the Enable Code set (1 or 2) but must be valid commands. The format for this block is shown in the following table.

Word	Description
0	Command queue block to 5006 for Port 1. identification code of 5000
1	This word contains the index in the command list for the first command to be entered into the command queue.
2	This word contains the index in the command list for the second command to be entered into the command queue.
3	This word contains the index in the command list for the third command to be entered into the command queue.
4	This word contains the index in the command list for the fourth command to be entered into the command queue.
5	This word contains the index in the command list for the fifth command to be entered into the command queue.
6	This word contains the index in the command list for the sixth command to be entered into the command queue.

When the module receives one of these blocks, it examines word 0 of the output image. This word defines the number of commands contained in the block. The command count is determined by subtracting 5000 from the word value. This permits the controller to set from one to six commands into the command queue. The indexes submitted in the block should be valid for the command list. After the module determines the number of commands to be considered, it inserts each command in the command queue. The response message sent from the module to the processor is as follows:

Word	Description
0	This word contains the block 5000 to 5006 requested by the processor.
1	Write Block ID
2	This word contains the number of commands in the block placed in the command queue.
3 to 7	Not used

The ladder logic can examine word 2 of the input image to determine the number of commands placed in the command queue. Any errors associated with the command can be viewed in the command list error table.

Warm Boot (Block 9998)

This block does not contain any data. When the processor places a value of 9998 in word 0 of the output image, the module will perform a warm-start. This involves clearing the configuration and all program status data. Finally, the program will load in the configuration information from the flash disk and begin running. There is no positive response to this message other than the status data being set to zero and the block polling starting over. The format of this block is shown in the following table.

Word	Description	Length
0	9998	1
1 to 6	Not used	6

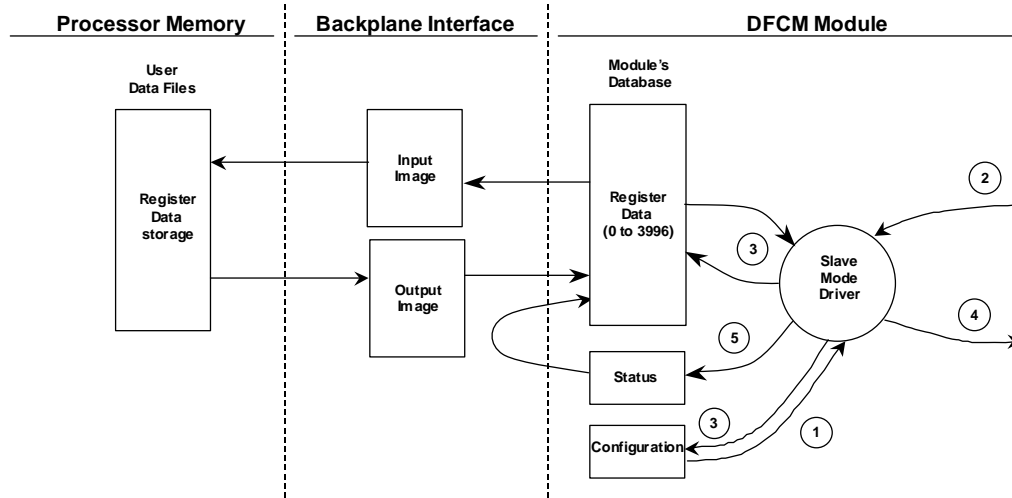
Cold Boot (Block 9999)

This block does not contain any data. When the processor places a value of 9999 in word 0 of the output image, the module will perform a hardware restart. This will cause the module to reboot and reload the program. There is no positive response to this message other than the status data being set to zero and the block polling starting over. The format of this block is shown in the following table.

Word	Description	Length
0	9999	1
1 to 6	Not used	6

5.3.3 Slave Driver

The Slave Driver Mode allows the MVI94-DFCM 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.

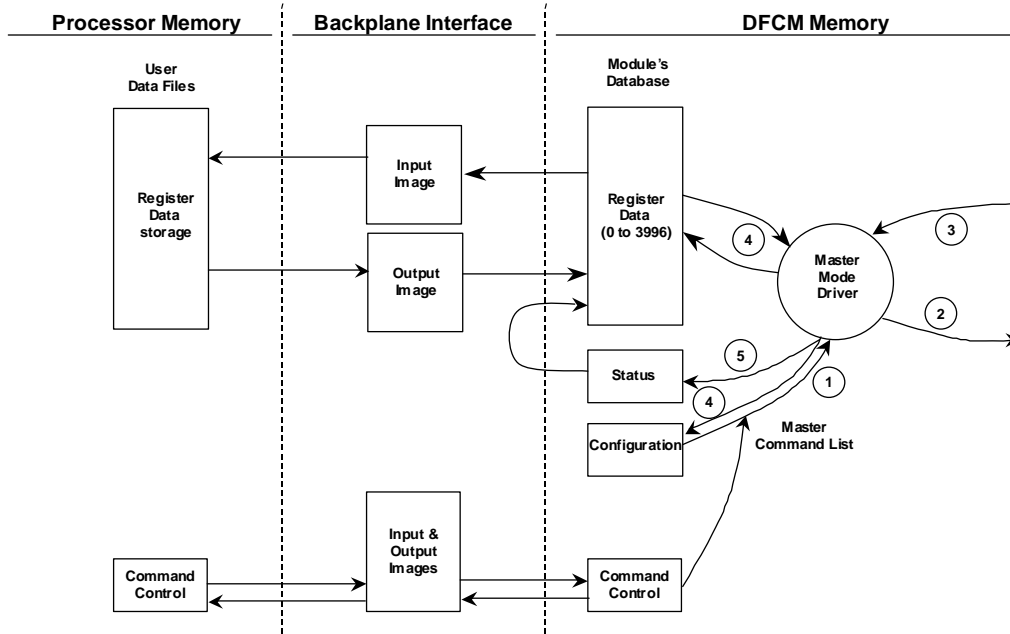


Step	Description
1	The DF1 slave port driver receives the configuration information from the Flash data area in the module. This information configures the serial port and define the slave node characteristics. The module simulates N-files to permit remote access of the database. Each file has a fixed 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.

The slave driver supports the following DF1 command set:

5.3.4 Master Driver

In the Master mode, the MVI94-DFCM 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 read from the module's Flash. 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 Flash area of the module. 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.
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 processor for each command in the Master Command List.

Refer to the **DF1 Command Set for ProSoft Technology Communication Modules** documentation for a complete discussion of the structure and content of each command. 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 are ignored by the module. 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.

5.3.5 MVI94-DFCM Block Formats

This section contains a listing of all block formats used by the module.

User Data Blocks

Read Data Block

Word	Description	Length
0	Read Block ID	1
1	Write Block ID	1
2 to 7	Read Block data	6

Write Data Block

Word	Description	Length
0	Write Block ID	1
1 to 6	Write Block data	6

MVI94-DFCM Special Blocks

Slave Polling Disable Request (Write Block)

Word	Description	Length
0	3000	1
1	Number of Slaves in Block	1
2 to 6	Slave indexes	5

Slave Polling Disable Response (Read Block)

Word	Description	Length
0	3000	1
1	Write Block ID	1
2	Number of slaves processed	1
3 to 7	Not used	5

Slave Polling Enable Request (Write Block)

Word	Description	Length
0	3001	1
1	Number of Slaves in Block	1
2 to 6	Slave indexes	5

Slave Polling Enable Response (Read Block)

Word	Description	Length
0	3001	1
1	Write Block ID	1
2	Number of slaves processed	1
3 to 7	Not used	5

Command Control Request (Write Block)

Word	Description	Length
0	5001 to 5006	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

Command Control Response (Read Block)

Word	Description	Length
0	5000 to 5006	1
1	Write Block ID	1
2	Number of commands added to command queue	1
3 to 7	Not used	5

Module Warm Boot Request (Write Block)

Word	Description	Length
0	9998	1
1 to 6	Not used	6

Module Cold Boot Request (Write Block)

Word	Description	Length
0	9999	1
1 to 6	Not used	6

5.3.6 MVI94-DFCM Database

This section contains a listing of the data contained in the modules database. The following table lists the general database format.

Register Range	File Start	File End	Content	Size
0 to 3995	N7:0	N26:195	User Data	3996
3996 to 4999	N26:196	N31:199	Reserved	1004
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	Reserved	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	Reserved	1200
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

The following topics show the content of the configuration, status, and command control portion of the database.

Backplane Setup

File	Register	Content	Description
N32:0	5000	Write Start Reg	This parameter specifies the starting register in the internal database to fill with data read over the backplane to the module (processor write block).
N32:1	5001	Write Reg Count	This parameter specifies the number registers in the internal database to be considered from the read operations over the backplane. This parameter computes the number of blocks to be transferred from the backplane to the module. The number of blocks must be ≥ 2 for proper backplane data transfer.
N32:2	5002	Read Start Reg	This parameter specifies the starting register in the internal database to write over the backplane from the module (processor read block).
N32:3	5003	Read Reg Count	This parameter specifies the number registers in the internal database to write over the backplane. This parameter computes the number of blocks to transfer from the module to the backplane. The number of blocks must be ≥ 2 for proper backplane data transfer.
N32:4	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.

File	Register	Content	Description
N32:5	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.
N32:6	5006	Spare	
N32:7	5007	Spare	
N32:8	5008	Spare	
N32:9	5009	Spare	

Port 1 Setup

File	Register	Content	Description																										
N32:10	5010	Enable	This parameter defines 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.																										
N32:11	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.																										
N32:12	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.																										
N32:13	5013	Protocol	0=full duplex, 1=half-duplex																										
N32:14	5014	Termination Type	This parameter specifies the error checking for all DF1 messages. 0=BCC, 1=CRC																										
N32:15	5015	Baud Rate	This is the baud rate to be used on the port. Enter the baud rate as a value. For example, to select 19K baud, enter 19200. <table border="1"> <thead> <tr> <th>Baud Rate</th> <th>Parameter Value</th> </tr> </thead> <tbody> <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>19,200</td><td>19, 192 or 19200</td></tr> <tr><td>28,800</td><td>28, 288 or 28800</td></tr> <tr><td>57,600</td><td>57 or 576</td></tr> <tr><td>115,200</td><td>115 or 1152</td></tr> </tbody> </table>	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	19,200	19, 192 or 19200	28,800	28, 288 or 28800	57,600	57 or 576	115,200	115 or 1152
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																												
19,200	19, 192 or 19200																												
28,800	28, 288 or 28800																												
57,600	57 or 576																												
115,200	115 or 1152																												
N32:16	5016	Parity	This is the Parity code to be used for the port. The coded values are as follows: 0=None, 1=Odd, 2=Even, 3=Mark and 4=Space.																										

File	Register	Content	Description
N32:17	5017	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.
N32:18	5018	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.
N32:19	5019	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.
N32:20	5020	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.
N32:21	5021	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.
N32:22	5022	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).
N32:23	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.
N32:24	5024	Command Count	This parameter specifies the number of commands to be processed for the port. Enter a value of 0 to 100.
N32:25	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.
N32:26	5026	Command Error Pointer	This parameter sets the address in the internal Modbus 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.

File	Register	Content	Description
N32:27	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.
N32:28	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.
N32:29	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.
N32:30	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.
N32:31	5031	Slave List Frequency	Not used in this version of the software
N32:32	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.
N32:33	5033	File Size	This parameter specifies 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.
N32:34	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.
N32:35 -	5035 -	Spare	
N32:69	5069	Spare	

Port 1 Commands

File	Register	Content	Description
N33:0 to N33:11	5200 to 5211	Command # 1	This set of registers contains the parameters for the first command in the master command list. Refer to the data object section of the documentation.
N33:12 to N33:23	5212 to 5223 -	Command # 2	Command #2 data set
N38:188 to N38:199	6388 to 6399	Command # 100	Command #100 data set
N39:0 to N44:199	6400 to 7599	Spare	

Miscellaneous Status

File	Register	Content	Description
N45:0	7600	Program Scan Count	This value is incremented each time a complete program cycle occurs in the module.
N45:1 N45:2	7601 7602	Product Code	These two registers contain the product code of "DFCM"
N45:3 N45:4	7603 7604	Product Version	These two registers contain the product version for the currently running software.
N45:5 N45:6	7605 7606	Operating System	These two registers contain the month and year values for the program operating system.
N45:7 N45:8	7607 7608	Run Number	These two registers contain the run number value for the currently running software.
N45:9	7609	Port 1 Command List Requests	This field contains the number of requests made from this port to slave devices on the network.
N45:10	7610	Port 1 Command List Response	This field contains the number of slave response messages received on the port.
N45:11	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.
N45:12	7612	Port 1 Requests	This field contains the total number of messages sent out the port.
N45:13	7613	Port 1 Responses	This field contains the total number of messages received on the port.
N45:14	7614	Port 1 Errors Sent	This field contains the total number of message errors sent out the port.
N45:15	7615	Port 1 Errors Received	This field contains the total number of message errors received on the port.
N45:16	7616	Read Block Count	This field contains the total number of read blocks transferred from the module to the processor.
N45:17	7617	Write Block Count	This field contains the total number of write blocks transferred from the processor to the module.

File	Register	Content	Description
N45:18	7618	Parse Block Count	This field contains the total number of blocks successfully parsed that were received from the processor.
N45:19	7619	Command Event Block Count	This field contains the total number of command event blocks received from the processor.
N45:20	7620	Command Block Count	This field contains the total number of command blocks received from the processor.
N45:21	7621	Error Block Count	This field contains the total number of block errors recognized by the module.
N45:22	7622	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.
N45:23	7623	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.

Command Control

File	Register	Content	Description
N46:0	7800	Command Code	Enter one of the valid control command codes in this register to control the module (9998 or 9999). Refer to the Reference chapter for more information.
N46:1 to N46:199	7801 to 7999	Command Data -	

5.3.7 MVI94-DFCM Command Control

Command Control data is received from other nodes on the network that can control the MVI94-DFCM module. Specific values are written to regions of this block to control the module. Currently, the module is programmed to handle the receipt of the following requests: warm boot and cold boot.

The remote node controls the module by writing one of the following values to register 7800 (address N46:0):

9998 Warm boot the module

9999 Cold boot the module

The control register is cleared (a value of 0) after the operation is executed.

5.4 Error/Status Data

The module error/status data areas are discussed in this section. The module contains three areas related to this data. The user defines the location of these data sets in the virtual Modbus database of the module. The error/status data contains module data, the command error list data set contains the errors associated with the command list and the slave status list contains the current communication status of each slave on the master port.

5.4.1 Error/Status Data

The error/status data table is located at the virtual Modbus address assigned by the user and at database address starting at 7600. If the address is set to -1, the data will not be placed in the user data area of the database. It will only be available through the Configuration/Debug Port and to nodes on the network. If a valid address value is assigned, the module will update the Modbus data area.

The data area will be initialized with zeros whenever the processor is started. This occurs during a cold-start (power-on), reset (reset push-button pressed) or a warm-boot operation (commanded or loading of new configuration).

The data area is a 24-word register block. The structure of the block is shown in the following table.

Offset	DB Reg	Content	Description
0	7600	Program Scan Count	This value is incremented each time a complete program cycle occurs in the module.
1 to 2	7601 to 7602	Product Code	These two registers contain the product code of "DFC9"
3 to 4	7603 to 7604	Product Version	These two registers contain the product version for the current running software.
5 to 6	7605 to 7606	Operating System	These two registers contain the month and year values for the program operating system.
7 to 8	7607 to 7608	Run Number	These two registers contain the run number value for the currently running softwa.
9	7609	Port 1 Command List Requests	This field contains the number of requests made from this port to slave devices on the network.
10	7610	Port 1 Command List Response	This field contains the number of slave response messages received on the port.
11	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.
12	7612	Port 1 Requests	This field contains the total number of messages sent out of the port.
13	7613	Port 1 Responses	This field contains the total number of messages received on the port.
14	7614	Port 1 Errors Sent	This field contains the total number of message errors sent out of the port.
15	7615	Port 1 Errors Received	This field contains the total number of message errors received on the port.

Offset	DB Reg	Content	Description
16	7616	Read Block Count	This field contains the total number of read blocks transferred from the module to the processor.
17	7617	Write Block Count	This field contains the total number of write blocks transferred from the processor to the module.
18	7618	Parse Block Count	Not used in this version of the module.
19	7619	Command Event Block Count	This field contains the total number of command event blocks received from the processor.
20	7620	Command Block Count	This field contains the total number of command blocks received from the processor.
21	7621	Error Block Count	This field contains the total number of block errors recognized by the module.
22	7622	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.
23	7623	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.

5.4.2 Command Error List

Each command in the command list has a reserved word value for a status/error code. This error data list can be read using the Configuration/Debug Port. Additionally, the data can be placed in the virtual Modbus database of the module. The configuration parameter "Command Error Pointer" defines the register address in the virtual Modbus database where the data will be placed.

The first word in the register location defined contains the status/error code for the first command in the port's command list. Each successive word in the command error list is associated with the next command in the list. Therefore, the size of the data area is dependent upon the number of commands defined.

Refer to the following Error Codes section to interpret the status/error codes present in the data area.

5.4.3 Slave Status List

The slave status list views the communication status of each slave device on a simulated master port. Slaves attached to the master port can have one of the following states:

Error Code	Description
0	The slave is inactive and not defined in the command list for the master port.
1	The slave is actively being polled or controlled by the master port and communication is successful.
2	The master port has failed to communicate with the slave device. Communication with the slave is suspended for a user defined period based on the scanning of the command list.

Slaves are defined to the system when the module initializes the master command list. Each slave defined will be set to a state of 1 in this initial step. If the master port fails to communicate with a slave device (retry count expired on a command), the master will set the state of the slave to a value of 2 in the status table. This suspends communication with the slave device for a user specified scan count (**Error Delay Count** value in the configuration). Each time a command in the list is scanned that has the address of a suspended slave, the delay counter value will be decremented. When the value reaches zero, the slave state will be set to one. This will enable polling of the slave. This data set can be located in the user data area of the module's database and therefore, transferred to the processor from the module. Additionally, the slave status list can be viewed using the module's configuration/debug port. The 'O' option on the main menu displays the status of all 256-slave units.

5.5 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:

5.5.1 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

5.5.2 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
-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)

5.5.3 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

Code (Int)	Code (Hex)	Description
-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

5.5.4 Module Specific Error (not DFCM Compliant)

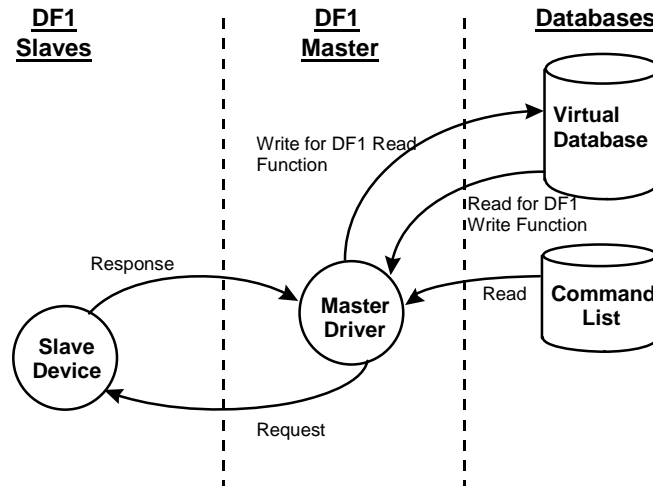
Code (Int)	Code (Hex)	Description
-1	0xFFFF	CTS modem control line not set before transmit
-2	0xFFFE	Timeout while transmitting message
-10	0xFFFF6	Timeout waiting for DLE-ACK after request
-11	0xFFFF5	Timeout waiting for response after request
-12	0xFFFF4	Reply data does not match requested byte count
-20	0xFFEC	DLE-NAK received after request
-21	0xFFEB	DLE-NAK sent after response

5.6 DF1 Command Set For ProSoft Technology Communication Modules

5.6.1 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.

5.6.2 Command Function Codes

This section describes DFCM 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.	

Column	Parameter	Description	Parameter
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.	
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.	

Column	Parameter	Description	Parameter
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	
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.	

Column	Parameter	Description	Parameter
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
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.	

Column	Parameter	Description	Parameter
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.	
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

Column	Parameter	Description	Parameter
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	

Column	Parameter	Description	Parameter
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.	

Column	Parameter	Description	Parameter
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.

5.6.3 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

Code	Description
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

5.6.4 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.

5.6.5 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.

5.6.6 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.

6 Support, Service & Warranty

In This Chapter

- ❖ How to Contact Us: Technical Support..... 95
- ❖ Return Material Authorization (RMA) Policies and Conditions..... 96
- ❖ LIMITED WARRANTY..... 97

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: www.prosoft-technology.com/support E-mail address: support@prosoft-technology.com
-----------------	--

Asia Pacific

+603.7724.2080, 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
Languages spoken include: French, English

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

+1.661.716.5100, 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
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 95). 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, and so on.

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:

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

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 warranted 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 or 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 97) 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

[

[Commands] • 22
[DF1 Port] • 18
[Module] • 17

A

All Product Returns: • 96
Allocation of Risks • 100
Analyzing Data for the first application port • 39
Analyzing Data for the second application port • 39

B

Backplane Fail Count • 18
Backplane Setup • 70
Basic Command Set Functions • 24
Baud Rate • 19
BT • 91

C

Cable Connections • 51
Cold Boot (Block 9999) • 64
Command Control • 75
Command Control Request (Write Block) • 69
Command Control Response (Read Block) • 69
Command Count • 20
Command Error List • 77
Command Error Pointer • 21
Command Function Codes • 81
Command List Overview • 22
Commands (Blocks 5001 to 5006) • 63
Configuration • 15
Configuration File • 15
Control • 90
Controlling Law and Severability • 100
ControlLogix Processor Specifics • 92
Count • 23

D

Data Analyzer • 39
Data Analyzer Tips • 41
Data Bits • 19
Data Transfer • 59
Database View Menu • 44
DB9 to Mini-DIN Adaptor (Cable 09) • 55
DF1 Command List Form • 93
DF1 Command Set For ProSoft Technology
Communication Modules • 22, 24, 80
Diagnostics and Troubleshooting • 33, 57
Disclaimer of all Other Warranties • 99
Disclaimer Regarding High Risk Activities • 98

Displaying the Current Page of Registers Again • 44
Displaying Timing Marks in the Data Analyzer • 39

E

Editing the Configuration File • 16
Enable • 23
ENQ Delay: • 20
Error Codes • 78
Error Delay Count: • 21
Error/Status Data • 76
Error/Status Pointer: • 17
Errors When EXT STS Is Present • 79
Exiting the Program • 39

F

FLEX I/O Interfaces • 50
Function Code • 24
Function Code #1 - Protected Write (Basic Command Set) • 81
Function Code #100 - Word Range Write (PLC-5 Command) (Binary Address) • 84
Function Code #101 - Word Range Read (PLC-5 Command) (Binary Address) • 84
Function Code #102 - Read-Modify-Write (PLC-5 Command) (Binary Address) • 85
Function Code #150 - Word Range Write (PLC-5 Command) (ASCII Address) • 86
Function Code #151 - Word Range Read (PLC-5 Command) (ASCII Address) • 86
Function Code #152 - Read-Modify-Write (PLC-5 Command) (ASCII Address) • 87
Function Code #2 - Unprotected Read (Basic Command Set) • 82
Function Code #3 - Protected Bit Write (Basic Command Set) • 82
Function Code #4 - Unprotected Bit Write (Basic Command Set) • 83
Function Code #5 - Unprotected Write (Basic Command Set) • 83
Function Code #501 - Protected Typed Logical Read (Two Address Fields) • 87
Function Code #502 - Protected Typed Logical Read (Three Address Fields) • 88
Function Code #509 - Protected Typed Logical Write (Two Address Fields) • 88
Function Code #510 - Protected Typed Logical Write (Three Address Fields) • 89
Function Code #511 - Protected Typed Logical Write with Mask (Three Address Fields) • 89
Functional Overview • 56
Functional Specifications • 50

G

General Specifications • 49

H

Hardware Installation • 9
How to Contact Us

Technical Support • 95, 96

I

Important Installation Instructions • 2
Installing the Module onto the Base • 12
Installing the Serial Adapter Cables • 13
Intellectual Property Indemnity • 99
Internal Address • 23
Introduction • 80

K

Keystrokes • 34

L

LED Status Indicators • 48
Limitation of Remedies ** • 100
LIMITED WARRANTY • 97
Local Station ID • 18
Local STS Error Codes • 78

M

Main Menu • 36
Master Command Error List Menu • 46
Master Command List Menu • 47
Master Driver • 66
MG • 91
MicroLogix Processor Specifics • 92
Minimum Command Delay • 20
Minimum Response Delay • 19
Miscellaneous Status • 74
Module Cold Boot Request (Write Block) • 69
Module Specific Error (not DFCM Compliant) • 80
Module Warm Boot Request (Write Block) • 69
Mounting the MVI 94 Flex I/O Base • 10
Moving Back Through 5 Pages of Registers • 45
Moving Forward Through 5 Pages of Registers • 45
MVI (Multi Vendor Interface) Modules • 2
MVI94-DFCM Block Formats • 68
MVI94-DFCM Command Control • 75
MVI94-DFCM Database • 70
MVI94-DFCM Special Blocks • 68

N

Navigation • 33
No Other Warranties • 100
Node Address • 24

O

Opening the Command Error List Menu • 37
Opening the Command List Menu • 37
Opening the Data Analyzer Menu • 36
Opening the Database Menu • 37

P

Parity • 19
PD* • 90
Pinouts • 13, 51, 55

PLC-5 Command Set Functions • 24
PLC-5 Processor Specifics • 90
PLC-5 Sub-Element Codes • 90
Poll Interval • 23
Port 1 Commands • 74
Port 1 Setup • 71
Procedures for Return of Units Out of Warranty: • 96
Procedures for Return of Units Under Warranty: • 96
Product Specifications • 49
ProSoft Technology® Product Documentation • 3
Protocol • 18
PRT1 • 48
PRT2 • 48

Q

Quick Start • 7

R

Read Data Block • 68
Read Register Count: • 17
Read Start Register: • 17
Redisplaying the Current Page • 46, 47
Reference • 49
Remote STS Error Codes • 79
Removing Timing Marks in the Data Analyzer • 40
Required Hardware • 26, 34
Required Software • 27, 34
Response Timeout • 21
Retry Count • 21
Return Material Authorization (RMA) Policies and Conditions • 96
Returning to the Main Menu • 41, 45, 46, 47
RS-232 • 51
 Modem Connection • 52
 Null Modem Connection (Hardware Handshaking) • 52
 Null Modem Connection (No Hardware Handshaking) • 53
RS-232 Configuration/Debug Port • 53
RS-422 • 54
RS-485 • 54
RS-485 and RS-422 Tip • 54
RTS Off • 20
RTS On • 20

S

Setting Jumpers • 11
Slave Driver • 65
Slave List Pointer • 21
Slave Polling Control Blocks (Blocks 3000 to 3001) • 62
Slave Polling Disable Request (Write Block) • 68
Slave Polling Disable Response (Read Block) • 68
Slave Polling Enable Request (Write Block) • 69
Slave Polling Enable Response (Read Block) • 69
Slave Status List • 77
SLC File Types • 91, 92
SLC Processor Specifics • 91
SLC-500 Command Set Functions • 24

Starting the Data Analyzer • 40
Stop Bits • 19
Stopping the Data Analyzer • 41
STS • 48
Support, Service & Warranty • 95
Swap Code • 23

Write Register Count: • 17
Write Start Register: • 17

Y

Your Feedback Please • 3

T

Termination Type • 18
The Configuration/Debug Menu • 26, 33
Time Limit for Bringing Suit • 100
Timer / Counter • 90
Transferring Module Configuration to the Processor •
38
Transferring the Configuration File to the Module • 30
Transferring the Configuration File to Your PC • 27
Type • 18

U

U1 • 48
U2 • 48
Uploading and Downloading the Configuration File •
15, 26
Use CTS Line • 20
User Data Blocks • 68
Using the Configuration/Debug Port • 35

V

Verify Package Contents • 9
Viewing Block Transfer Statistics • 36
Viewing Communication Status • 38
Viewing Data in ASCII (Text) Format • 40, 45
Viewing Data in Decimal Format • 45
Viewing Data in Floating Point Format • 45
Viewing Data in Hexadecimal Format • 40, 45
Viewing Module Configuration • 37
Viewing Port Configuration • 38
Viewing Register Pages • 44
Viewing the Next 100 Registers of Data • 45
Viewing the Next 20 Commands • 46
Viewing the Next 50 Commands • 47
Viewing the Next Page of Commands • 46, 47
Viewing the Previous 100 Registers of Data • 45
Viewing the Previous 20 Commands • 46
Viewing the Previous 50 Commands • 47
Viewing the Previous Page of Commands • 46, 47
Viewing the Slave Status List (Port 1 and 2) • 37
Viewing Version Information • 37
Virtual Modbus Database • 58

W

Warm Boot (Block 9998) • 64
Warm Booting the Module • 38
Warnings • 2
What Is Covered By This Warranty • 97, 99
What Is Not Covered By This Warranty • 98
Wiring the Power Connections • 13
Write Data Block • 68