ProSoft TECHNOLOGY

Where Automation Connects.

inRAx®

MVI71-DFNT

EtherNet/IP Client/Server
Communication Module

February 3, 2011

USER MANUAL

Your Feedback Please

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

How to Contact Us

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

MVI71-DFNT User Manual

January 18, 2011

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-ROM, 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.

North America: +1.661.716.5100

Asia Pacific: +603.7724.2080

Europe, Middle East, Africa: +33 (0) 5.3436.87.20

Latin America: +1.281.298.9109

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
C WARNING - EXPLOSION HAZARD - DO NOT DISCONNECT EQUIPMENT UNLESS POWER HAS BEEN SWITCHED OFF OR THE AREA IS KNOWN TO BE NON-HAZARDOUS.
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'ÉQUIPEMENT, COUPER LE COURANT OU S'ASSURER QUE L'EMPLACEMENT EST DÉSIGNÉ NON DANGEREUX.

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.

ATEX Warnings and Conditions of Safe Usage:

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

A Warning - Explosion Hazard - When in hazardous locations, turn off power before replacing or wiring modules.
B Warning - Explosion Hazard - Do not disconnect equipment unless power has been switched off or the area is known to be non-hazardous.
C These products are intended to be mounted in an IP54 enclosure. The devices shall provide external means to prevent the rated voltage being exceeded by transient disturbances of more than 40%. This device must be used only with ATEX certified backplanes.
D DO NOT OPEN WHEN ENERGIZED.

Warning: This module is not hot-swappable! Always remove power from the rack before inserting or removing this module, or damage may result to the module, the processor, or other connected devices.

Battery Life Advisory

The MVI46, MVI56, MVI56E, MVI69, and MVI71 modules use a rechargeable Lithium Vanadium Pentoxide battery to backup the real-time clock and CMOS. The battery should last for the life of the module. The module must be powered for approximately twenty hours before the battery becomes fully charged. After it is fully charged, the battery provides backup power for the CMOS setup and the real-time clock for approximately 21 days. When the battery is fully discharged, the module will revert to the default BIOS and clock settings.

Note: The battery is not user replaceable.

Markings

Electrical Ratings

- Backplane Current Load: 800 mA @ 5 Vdc
- Operating Temperature: 0°C to 60°C (32°F to 140°F)
- Storage Temperature: -40°C to 85°C (-40°F to 185°F)
- Shock: 30 g operational, 50 g non-operational; Vibration: 5 g from 10 Hz to 150 Hz
- Relative Humidity 5% to 95% (with no condensation)
- All phase conductor sizes must be at least 1.3 mm(squared) and all earth ground conductors must be at least 4mm(squared).

Label Markings

Agency Approvals and Certifications

Agency	Applicable Standards
ANSI / ISA	ISA 12.12.01 Class I Division 2, GPs A, B, C, D
CSA/cUL	C22.2 No. 213-1987
CSA CB Certified	IEC61010
ATEX	EN60079-0 Category 3, Zone 2 EN60079-15

243333

Contents

Guide to the MVI71-DFNT User Manual

Function		Section to Read	Details
Introduction (Must Do)	→	Start Here (page 11)	This section introduces the customer to the module. Included are: package contents, system requirements, hardware installation, and basic configuration.
Diagnostic and Troubleshooting	→	Diagnostics and Troubleshooting (page 45)	This section describes Diagnostic and Troubleshooting procedures.
Reference Product Specifications Functional Overview	→	Reference (page 65) Product Specifications (page 66) Functional Overview (page 69)	These sections contain general references associated with this product, Specifications, and the Functional Overview.
Support, Service, and Warranty Index	→	Support, Service and Warranty (page 165) Index	This section contains Support, Service and Warranty information. Index of chapters.

1 Start Here

In This Chapter

To get the most benefit from this User Manual, you should have the following skills:

- **Rockwell Automation® RSLogix™ software:** launch the program, configure ladder logic, and transfer the ladder logic to the processor
- **Microsoft Windows:** install and launch programs, execute menu commands, navigate dialog boxes, and enter data
- **Hardware installation and wiring:** install the module, and safely connect Ethernet/IP and PLC devices to a power source and to the MVI71-DFNT module's application port(s)

1.1 System Requirements

The MVI71-DFNT module requires the following minimum hardware and software components:

- Rockwell Automation PLC processor, with compatible power supply and one free slot in the rack, for the MVI71-DFNT module. The module requires 800mA of available power.
- The PLC Processor must provide for at least 64 words of BTR/BTW area, otherwise the module may not function correctly.
- Rockwell Automation RSLogix 5 programming software.
- Rockwell Automation RSLinx communication software
- Pentium® 100 MHz minimum. Pentium III 700 MHz (or better) recommended
- Supported operating systems:
  - Microsoft Windows XP
  - Microsoft Windows 2000
  - Microsoft Windows NT v4.0 with Service Pack 3 or greater
  - Microsoft Windows ME
  - Microsoft Windows 98
- 64 Mbytes of RAM minimum, 256 Mbytes of RAM recommended
- 100 Mbytes of free hard disk space (or more based on application requirements)
- 256-color VGA graphics adapter, 800 x 600 minimum resolution (True Color 1024 × 768 recommended)
- CD-ROM drive
- HyperTerminal or other terminal emulator program capable of file transfers using Zmodem protocol.

1.2 Package Contents

The following components are included with your MVI71-DFNT module, and are all required for installation and configuration.

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

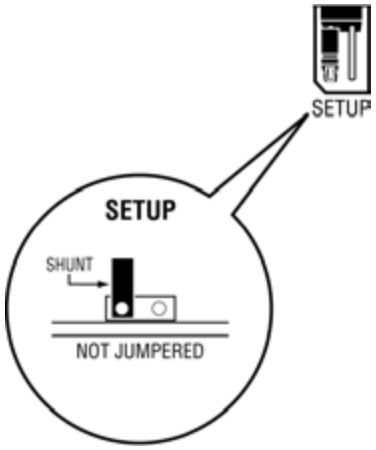
Qty.	Part Name	Part Number	Part Description
1	MVI71-DFNT Module	MVI71-DFNT	EtherNet/IP Client/Server Communication Module
1	Cable	Cable #15, RS232 Null Modem	For RS232 Connection to the CFG Port
3	Cable	Cable #14, RJ45 to DB9 Male Adapter cable	For DB9 Connection to Module's Port
2	Adapter	1454-9F	Two Adapters, DB9 Female to Screw Terminal. For RS422 or RS485 Connections to Port 1 and 2 of the Module
1	ProSoft Solutions CD		Contains sample programs, utilities and documentation for the MVI71-DFNT module.

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

1.3 Setting Jumpers

The Setup Jumper acts as "write protection" for the module's flash memory. In "write protected" mode, the Setup pins are not connected, and the module's firmware cannot be overwritten. Do not jumper the Setup pins together unless you are directed to do so by ProSoft Technical Support.

The following illustration shows the MVI71-DFNT jumper configuration.



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

1.4 Install the Module in the Rack

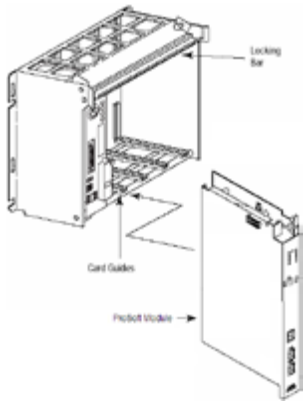
If you have not already installed and configured your PLC processor and power supply, please do so before installing the MVI71-DFNT module. Refer to your Rockwell Automation product documentation for installation instructions.

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

After you have checked the placement of the jumpers, insert MVI71-DFNT into the PLC™ chassis. Use the same technique recommended by Rockwell Automation to remove and install PLC modules.

Warning: This module is not hot-swappable! Always remove power from the rack before inserting or removing this module, or damage may result to the module, the processor, or other connected devices.

1. Turn power OFF.
2. Align the module with the top and bottom guides, and slide it into the rack until the module is firmly against the backplane connector.

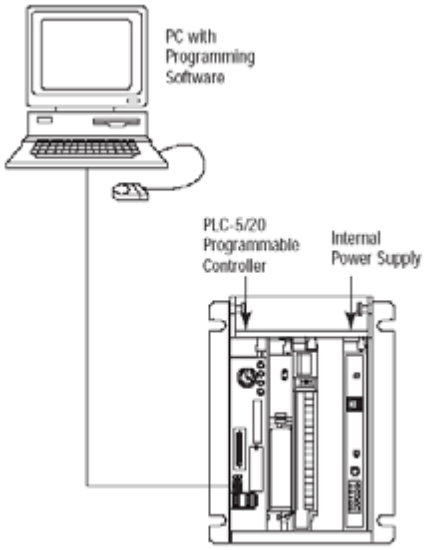


3. With a firm but steady push, snap the module into place.
4. Check that the holding clips on the top and bottom of the module are securely in the locking holes of the rack.
5. Make a note of the slot location. You will need to identify the slot in which the module is installed in order for the sample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

Note: If you insert the module improperly, the system may stop working, or may behave unpredictably.

1.5 Connect your PC to the Processor

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



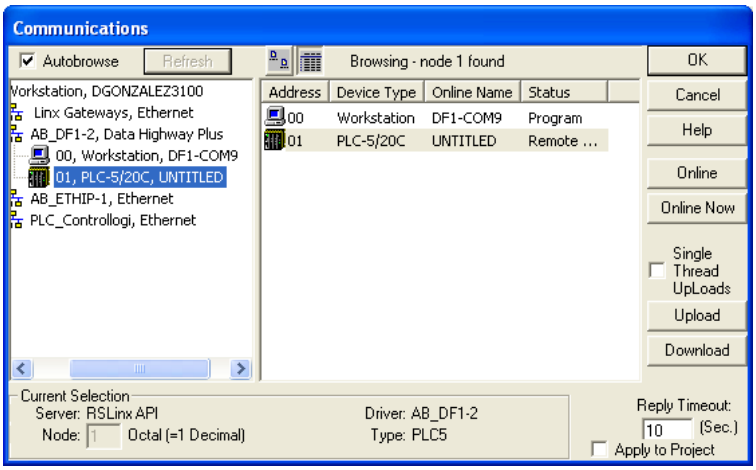
2 Connect the straight connector end of the cable to the serial port on your computer.

1.6 Download the Sample Program to the Processor

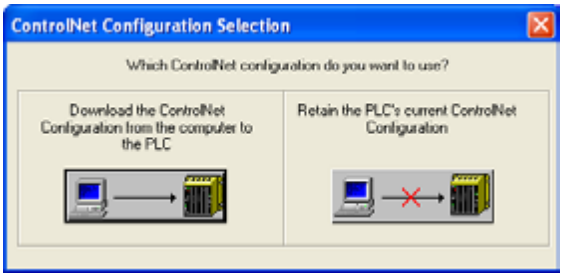
To download the sample program from RSLogix 5 to the PLC processor

Note: The key switch on the front of the PLC processor must be in the REM position.

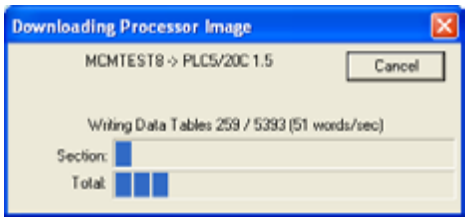
1 If you are not already online to the processor, open the Communications menu, and then choose Download. RSLogix will establish communication with the processor.



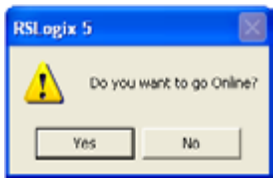
2 Click the Download button to transfer the sample program to the processor.
3 When prompted, choose Computer to PLC



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



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

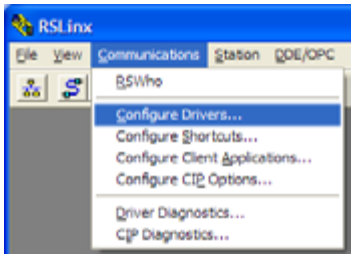


Note: If you receive an error message during these steps, refer to your RSLogix documentation to interpret and correct the error.

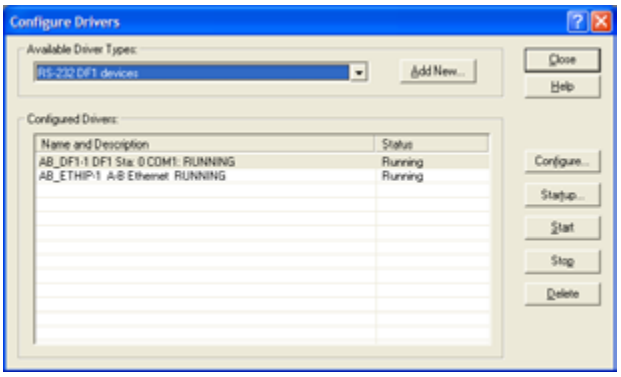
1.6.1 Configuring the RSLinx Driver for the PC COM Port

If RSLogix is unable to establish communication with the processor, follow these steps.

1 Open *RSLinx*.
2 Open the **COMMUNICATIONS** menu, and choose **CONFIGURE DRIVERS**.



This action opens the *Configure Drivers* dialog box.



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

3 Click to select the driver, and then click **CONFIGURE**. This action opens the *Configure RS-232 DF1 Devices* dialog box.

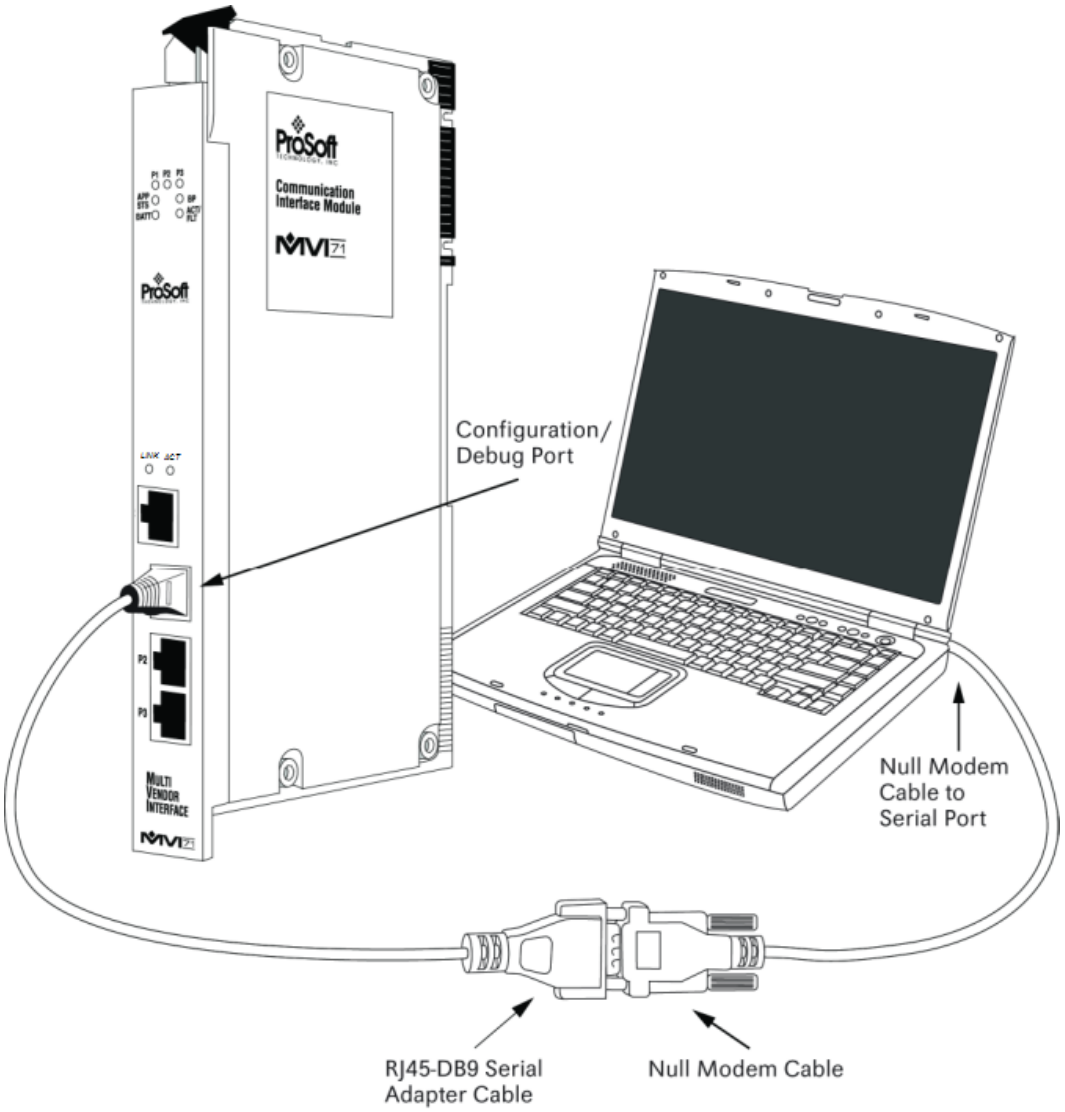
4 Click the **AUTO-CONFIGURE** button. RSLinx will attempt to configure your serial port to work with the selected driver.
5 When you see the message *Auto Configuration Successful*, click the **OK** button to dismiss the dialog box.

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

1.7 Connect your PC to the Module

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

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



2 Installing and Configuring the Module

In This Chapter

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

1. Modify the module's configuration files to meet the needs of your application, and copy the updated configuration to the module. Example configuration files are provided on the CD-ROM. Refer to the Modifying the Example Configuration File section, later in this chapter, for more information on the configuration files.
2. Modify the example ladder logic to meet the needs of your application, and copy the ladder logic to the processor. Example ladder logic files are provided on the CD-ROM.

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

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

Set up of the MVI71-DFNT module requires software configuration using the RSLogix 5 program and the DFNT.CFG file on the Compact Flash Disk in the module. The easiest method to implement the module is to start with the appropriate example provided with the module MVI71DFNT_BT.RSP for the block transfer interface and the default configuration file. If you are installing this module in an existing application, you can copy the elements required from the example ladder logic to your application.

It is now time to edit the DFNT.CFG file to set up the module for the specific application. Refer to the **Configuration File** section of this document. Download this configuration to the module along with the associated ladder logic.

Enter the ladder logic to handle the blocks transferred between the module and the PLC. Download the program to the PLC and test the program with the module.

The module is now set up and ready for your application. Insert the module in the rack (with the power turned off) and attach the serial communication cables. Download the new application to the controller and place the processor in run mode. Download the new DFNT.CFG file to the module using a terminal emulation program. If all the configuration parameters are set correctly and the module is attached to a network, the module's Application LED (APP LED) should remain off and the backplane activity LED (BP ACT) should blink very rapidly. Refer to Diagnostics and Troubleshooting if you encounter errors. Attach a computer or terminal to Port 0 on the module and look at the status of the module using the Configuration/Debug Menu in the module.

2.1 Module Configuration

This section contains the setup procedure, data, and ladder logic for successful application of the MVI71-DFNT module. Each step in the setup procedure is defined in order to simplify the use of the module.

2.1.1 Obtain the Sample Configuration Files

The ProSoft Solutions CD is organized in folders by module name. In the folder for the module you are using, you will find sample configuration files and other information.

1. Use Windows Explorer to locate the sample configuration files for your MVI71 module on the MVI71 CD.
2. When you have located the correct configuration files, use the **COPY** and **PASTE** commands to move the files to a location on your PC's hard drive. We recommend C:\temp.
3. Files copied from a CD-ROM are read-only. You must make the files writable. Navigate to the directory where you copied the files, then select the files and click the right mouse button to open a shortcut menu. On the shortcut menu, select **PROPERTIES,** and clear (uncheck) the **READ ONLY** check box.
4. Next, open the configuration files in a text editor such as Notepad, which comes with Windows. To start Notepad, click the **START** button, and then choose **PROGRAMS / ACCESSORIES / NOTEPAD.**
5. When Notepad starts, open the **FILE** menu, and then choose **OPEN.** Navigate to the folder where you copied the configuration file on your PC and select the file. Click **OPEN.** The configuration file will open in Notepad, ready for editing.

Note: We do not recommend opening the configuration file in a word processor such as Microsoft Word, because the file may be saved in a format that cannot be read by the module.

2.2 Configuration File

In order for the module to operate, a configuration file (DFNT.CFG) is required. This configuration file contains information to set the data transfer characteristics between the module and the processor, to configure the module's client and command list, and to configure the pass-through features. Each parameter in the file must be set carefully in order for the application to be implemented successfully.

The configuration file is separated into sections with topic header names enclosed in the **[ ]** characters. The configuration file consists of the following sections:

[Section]	Description
[Module]	General module configuration information
[DFNT Client 0]	Configuration for the DFNT client
[DFNT Client 0 Commands]	Command list for the DFNT client
[DF1 Pass-Through Server Port 1]	Parameters for the pass-through port of the second port on the module
[DF1 Pass-Through Port]	Parameters for the DF1 port emulated on the third port of the module

After each section header, the file contains a set of parameters. Unique labels are used under each section to specify a parameter. Each label in the file must be entered exactly as shown in the file for the parameter to be identified by the program. If the module is not considering a parameter, look at the label for the data item. Each parameter's value is separated from the label with the "**:**" character. This character is used by the program to delimit the position in the data record where to start reading data. All data for a parameter must be placed after the "**:**" character. For numeric parameter values any text located after the value will not be used. There must be at least one space character between the end of the parameter value and the following text.

Any record that begins with the "**#**" character is considered to be a comment record. These records can be placed anywhere in the file as long as the "**#" character** is found in the first column of the line. These lines are ignored in the file and can be used to provide documentation within the configuration file. Liberal use of comments within the file can ease the use and interpretation of the data in the file.

The client command list section is formatted differently than the other sections. These sections contain lists of parameters to be used. Each list begins with the label **START** and when the **END** label is reached. When entering the records into the list, make certain that the first character in each line is left blank.

The **[DFNT CLIENT 0 COMMANDS]** section defines the Ethernet/IP commands to be issued from the module to server devices on the network. These commands can be used for data collection and/or control of devices on the TCP/IP network.

2.2.1 [Module]

This section provides the module with a unique name, identifies the method of failure for the communications for the module if the processor is not in run, and describes how to initialize the module upon startup.

Module Name

0 to **80** characters

This parameter assigns a name to the module that can be viewed using the configuration/debug port. Use this parameter to identify the module and the configuration file.

Read Register Start

0 to **3999**

This parameter specifies the starting register address of a block of data registers to transfer from the module to the processor.

Read Register Count

0 to **3960**

This parameter specifies the number of registers to be transferred from the module to the processor.

Write Register Start

0 to **3999**

This parameter specifies the starting register address of a module register block where data transferred from the processor will be stored.

Write Register Count

0 to **3960**

This parameter specifies the number of registers to transfer from the processor to the module.

Failure Flag Count

0 through **65535**

This parameter specifies the number of successive transfer errors that must occur before halting communication on the application port(s). If the parameter is set to **0**, the application port(s) 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.

Initializing Output Data

YES or **NO**

This parameter determines if the output data for the module should be initialized with values from the processor. If the value is set to **NO** (0), the output data will be initialized to 0. If the value is set to **YES** (1), the data will be initialized with data from the processor. Use of this option requires associated ladder logic to pass the data from the processor to the module.

DFNT Server File Size

100 or **1000**

Sets the maximum file size (100 or 1000) for the servers

2.2.2 [DF1 Pass-Through Server Port 1]

This section is used to define the DF1 pass-through server on Port 1 (the second port)

```
[DF1 Pass-Through Server Port 1]
Enabled                :   Yes  #Y=Use server, N=Do not use server
Service Port Number    : 15000  #TCP service port for this server
Busy Timeout           :   500  #Time to wait for not Busy (100-65535
milliseconds)
Baud Rate              : 19200  #Baud rate for port 110-19200
Parity                 :     N  #N=None,O=Odd,E=Even,M=Mark,S=Space
Data Bits              :     8  #5, 6, 7 or 8
Stop Bits              :     1  #1 or 2
```

Enabled

Yes or No

This parameter determines if the server will be utilized by the module. If a value of "Yes" is entered, the server will be used. Any other value will disable the server.

Service Port Number

1 to 65535

This parameter sets the TCP/IP service port for this server. Each server can have its own unique service port or can share the same number with other servers.

Busy Timeout

100 to 65535 milliseconds

This parameter sets the number of milliseconds the server will wait for the serial pass-through port to become available. Valid data range for this parameter is 100 to 65535.

Baud Rate - 110 to 19200

110 to 19200

This parameter specifies the baud rate for the primary port on the module. Baud rate entries are 110, 150, 300, 600, 1200, 2400, 4800, 9600 or 19200

Parity

None, Odd, Even

Parity is a simple error checking algorithm used in serial communication. This parameter specifies the type of parity checking to use.

All devices communicating through this port must use the same parity setting.

Data Bits

5, 6, 7 or 8

This parameter sets the number of data bits for each word used by the protocol. All devices communicating through this port must use the same number of data bits.

Stop Bits

1 or **2**

Stop bits signal the end of a character in the data stream. For most applications, use one stop bit. For slower devices that require more time to re-synchronize, use two stop bits.

All devices communicating through this port must use the same number of stop bits.

2.2.3 [DF1 Pass-Through Port]

This section is used to define the configuration for the DF1 pass-through port on Port 2 (the third port)

```
[DF1 Pass-Through Port]
Enabled                :    Y   #Y=Use port, N=Do not use port
Local Station ID       :    1   #DF1 node address
Protocol               :    H   #F=Full-Duplex, H=Half-Duplex
Termination Type       :  CRC   #B=BCC, C=CRC
Baud Rate              : 38400  #Baud rate for port 1200-38400
Parity                 : None   #N=None,O=Odd,E=Even,M=Mark,S=Space
Data Bits              :    8   #5, 6, 7 or 8
Stop Bits              :    1   #1 or 2
RTS On                 :    0   #0-65536 mSec before message
RTS Off                :    0   #0-65536 mSec after message
Use CTS Line           :    N   #Use CTS modem control line (Y/N)

Retry Count            :    3   #Response failure retry count
Request Timeout        :  500   #Request message timeout (0-10000 mSec)
Busy Timeout           :  500   #Port Busy timeout (0-10000 mSec)
```

Enabled

Yes or No

This parameter determines if the server will be utilized by the module. If a value of "Yes" is entered, the server will be used. Any other value will disable the server.

Local Station ID

0 to 254

This parameter specifies the local station ID for all DF1 messages sent to this port. A value of 255 is not permitted as this is the broadcast address. The application will only accept messages with this node address.

Protocol

F (Full duplex) or H (Half duplex)

The value selected should match that set for the PLC processor.

Termination Type

BCC or CRC

This parameter specifies the error checking for all DF1 messages. Enter BCC or CRC.

Baud Rate - 1200 to 38400

1200 to 38400

This is the baud rate to be used for the port. Enter the baud rate as a value. Baud rate entries are 1200, 2400, 4800, 9600, 19200, 28800 or 38400.

Parity

None, Odd, Even, Mark, Space

Parity is a simple error checking algorithm used in serial communication. This parameter specifies the type of parity checking to use.

All devices communicating through this port must use the same parity setting.

Data Bits

5, 6, 7 or 8

This parameter sets the number of data bits for each word used by the protocol. All devices communicating through this port must use the same number of data bits.

Stop Bits

1 or **2**

Stop bits signal the end of a character in the data stream. For most applications, use one stop bit. For slower devices that require more time to re-synchronize, use two stop bits.

All devices communicating through this port must use the same number of stop bits.

RTS On

0 to **65535** milliseconds

This parameter sets the number of milliseconds to delay after *Ready To Send* (RTS) is asserted before data will be transmitted.

RTS Off

0 to **65535** milliseconds

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.

Use CTS Line

YES or **NO**

This parameter specifies if the Clear To Send (CTS) modem control line is to be used or not. If the parameter is set to **NO**, the CTS line will not be monitored. If the parameter is set to **YES**, 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). This procedure is commonly referred to as *hardware handshaking*.

Retry Count

0 to **10**

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

Request Timeout

0 to 10000 milliseconds

This parameter specifies the number of milliseconds to wait for a complete request message. The timer is started after the DLE-STX character sequence is received for the full-duplex protocol or the DLE-SOH sequence for the half-duplex protocol. If the timer expires, the current request message will be aborted.

Busy Timeout

0 to 10000 milliseconds

This parameter specifies the number of milliseconds to wait for the pass-through port to become available.

ACK Timeout

0 to 10000 milliseconds

This parameter specifies the number of milliseconds to wait for a DLE-ACK character sequence after a response is issued.

2.2.4 [DFNT Client 0]

This section is used to define the configuration for the master device simulated on network port

```
[DFNT Client 0]
Minimum Command Delay  :   30   #Minimum number of msec's between commands
Response Timeout       : 1000   #Response message timeout (0-5000 mSec)
Retry Count            :    3   #Response failure retry count
```

Minimum Command Delay

0 to **65535** milliseconds

This parameter specifies the number of milliseconds to wait between the initial issuances of a command. This parameter can be used to delay all commands sent to servers 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.

Response Timeout

0 to **65535** milliseconds

This is the time in milliseconds that a Client will wait before re-transmitting a command if no response is received from the addressed server. The value to use depends on the type of communication network used, and the expected response time of the slowest device on the network.

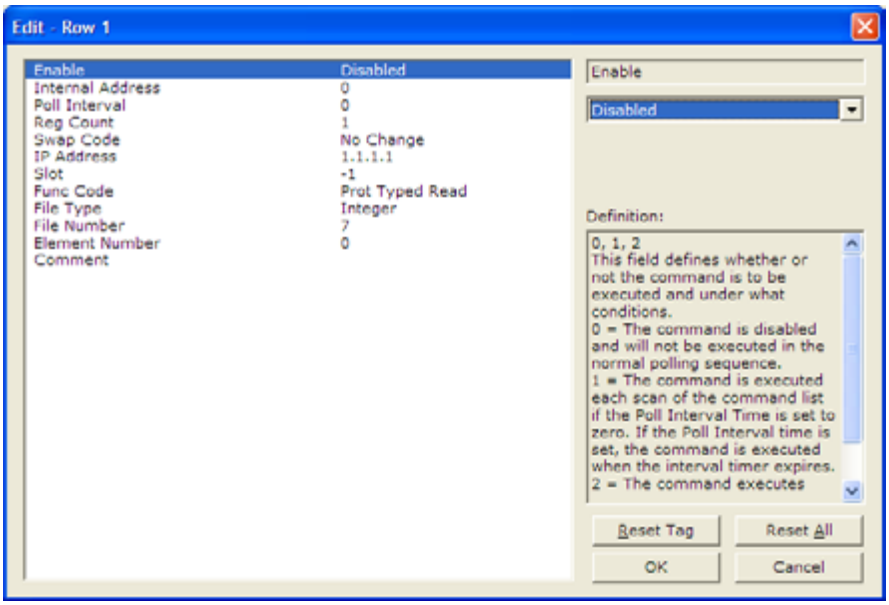
Retry Count

0 to **10**

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

2.2.5 [DFNT Client x Commands]

This section defines the EtherNet/IP commands to be issued from the module to server devices on the network. These commands can be used for data collection and/or control of devices on the TCP/IP network.



Command List

In order to interface the virtual database with DF1 slave devices, you must construct a command list. The commands in the list specify the DF1 slave device to be utilized, the function to be performed (read or write), the data area in the device to interface with and the position in the virtual database to be associated with the device data. There is a separate command list for each DF1 master device emulated. The list is processed from top (command #0) to bottom. A poll interval parameter is associated with each command to specify a minimum delay time between the issuance of a command. If the user specifies a value of 10 for the parameter, the command will be executed no more frequently than every 10 seconds for the serial implementation and 1 second for the network implementation.

Write commands have a special feature, as they can be set to execute only if the data in the write command changes. If the data in the command has 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 will be executed. Use of this feature can lighten the load on the DF1 network. In order to implement this feature; set the enable code for the command to a value of 2.

If the module is configured for the serial DF1 half-duplex protocol, the module can act as a master device routing messages between attached slave devices. This peer-to-peer communication is defined in the DF1 protocol specification. The master polls each DF1 slave device until no more data is available from the device. Response messages from the slaves that have a destination address that do not match the module are routed with a request message header back out onto the network. This facility offers communication between the slave devices for control and data monitoring. This feature is not available if the module is configured for DF1 full-duplex mode (point-to-point).

The module supports numerous commands. This permits the module to interface with a wide variety of DF1 protocol devices. This includes PLC2, PLC5, SLC-500 series, MicroLogix and ControlLogix processors. Additionally, other devices supplied by Rockwell Automation that use the DF1 protocol are supported.

The format of each command in the list depends on the function being executed. To simplify command construction, the module uses its own set of function codes to associate a command with a DF1 command/function type. The tables below list the functions supported by the module:

Basic Command Set Functions

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

PLC-5 Command Set Functions

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

SLC-500 Command Set Functions

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

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

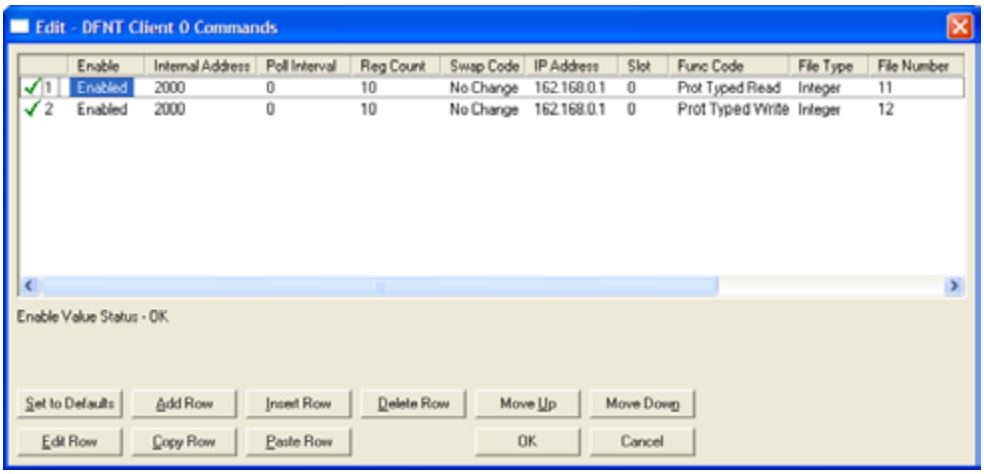
Command Entry Formats

The format of each command in the list depends on the function being executed. Refer to Command Function Codes (page 34) for a complete discussion of the commands supported by the module and of the structure and content of each command.

The following table shows the structure of the configuration data necessary for each of the supported commands.

Module Information Data							**Device Information Data**					
Column #	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
Function Code	**Enable Code**	**Internal Address**	**Poll Interval Time**	**Count**	**Swap Code**	**IP Address**	**Slot Number**	**Function Code**	**Function Parameters**			
FC 1	Code	Register	Seconds	Count	Code	Node	Slot	1	Word Address			
FC 2	Code	Register	Seconds	Count	Code	Node		2	Word Address			
FC 3	Code	Register	Seconds	Count	Code	Node		3	Word Address			
FC 4	Code	Register	Seconds	Count	Code	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	Code	Node		102	File Number	Element	Sub-Element	
FC 150	Code	Register	Seconds	Count	Code	Node		150	File String			
FC 151	Code	Register	Seconds	Count	Code	Node		151	File String			
FC 152	Code	Register	Seconds	Count	Code	Node		152	File String			
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	Code	Node		511	File Type	File Number	Element	Sub-Element

The first part of the record is the Module Information, which relates to the module. The second part contains information required to interface to the Server device. An example of a command list section of the configuration file is shown in the following illustration.




```
[DFNT Client 0 Commands]
#
# The file contains examples for a ControlLogix processor with the N7 file
# configured. This example uses SLC and PLC5 commands.
#
# LOCATION      :
# DATE          : 04/05/2000
# CONFIGURED BY: RAR
# MODIFIED      :
#
#   1     2      3     4    5  6                  7    8    9   10   11   12
#         DB    Poll       Swap                      Func File File  Elm  Sub
#Enab  Addr   Delay Count Code  Node IP Address Slot Code Type    #    #  Elm
START
#     1  2000      0    10    0  192.168.0.100      0  501    N   11    0
#     1  2000      0    10    0  192.168.0.100      0  509    N   12    0
#
#         DB    Poll       Swap                      Func File  Elm  Sub
#Enab  Addr   Delay Count Code  Node IP Address Slot Code    #    #  Elm
END
```

The following table describes each parameter

<table>
<tr><th>Parameter</th><th>Range</th><th>Description</th></tr>
<tr><td>Enable</td><td>0, 1, 2</td><td>This field defines whether the command is to be executed and under what conditions.
<table>
<tr><th>Value</th><th>Description</th></tr>
<tr><td>0</td><td>The command is disabled and will not be executed in the normal polling sequence.</td></tr>
<tr><td>1</td><td>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 interval timer expires.</td></tr>
<tr><td>2</td><td>The command executes only if the internal data associated with the command changes. This value is valid for write commands only.</td></tr>
</table></td></tr>
<tr><td>Internal Address</td><td>0 to 3999</td><td>This field specifies the database address in the module's internal database to be associated with the command. If the command is a read function, the data received in the response message is placed at the specified location. If the command is write function, data used in the command is sourced from the specified data area.</td></tr>
<tr><td>Poll Delay</td><td>0 to 1000</td><td>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 100 is entered for a command, the command executes no more frequently than every 10 seconds.</td></tr>
<tr><td>Count</td><td>Command dependent.</td><td>This parameter specifies the number of registers or digital points to be associated with the command. See Command Function Codes (page 34) for information</td></tr>
<tr><td>Swap Code</td><td>0, 1, 2, 3</td><td>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 table defines the values and their associated operations:
<table>
<tr><th>Swap Code</th><th>Description</th></tr>
<tr><td>0</td><td>None - No Change is made in the byte ordering (1234 = 1234)</td></tr>
<tr><td>1</td><td>Words - The words are swapped (1234=3412)</td></tr>
<tr><td>2</td><td>Words & Bytes - The words are swapped then the bytes in each word are swapped (1234=4321)</td></tr>
<tr><td>3</td><td>Bytes - The bytes in each word are swapped (1234=2143)</td></tr>
</table>
The words should be swapped only when using an even number of words.</td></tr>
<tr><td>Node IP Address</td><td>xxx.xxx.xxx.xxx</td><td>The IP address of the device being addressed by the command.</td></tr>
<tr><td>Slot</td><td></td><td>When addressing a ControlLogix processor, the slot number corresponds to the slot in the rack containing the controller being addressed. In the ControlLogix platform, the controller can be placed in any slot and the rack may contain multiple processors. This parameter uniquely selects a controller in the rack.
Use a value of -1 when interfacing to an SLC 5/05 or a
PLC5. These devices do not have a slot number.</td></tr>
</table>

<table>
<tr><th>Parameter</th><th>Range</th><th>Description</th></tr>
<tr><td>Function Code</td><td>See Command Function Codes (page 34)</td><td>These parameters specify the function to be executed by the command. The Reference chapter in this manual describes the meaning of these values for each of the available supported commands. Following is a complete list of the command supported by the Client driver.
Function Code Listing
Basic Command Set
<table>
<tr><td>1</td><td>Protected Write</td></tr>
<tr><td>2</td><td>Unprotected Read</td></tr>
<tr><td>3</td><td>Protected Bit Write</td></tr>
<tr><td>4</td><td>Unprotected Bit Write</td></tr>
<tr><td>5</td><td>Unprotected Write</td></tr>
</table>
PLC-5 Command Set (0x0F)
<table>
<tr><td>100</td><td>Word Range Write (Binary Address)</td></tr>
<tr><td>101</td><td>Word Range Read (Binary Address)</td></tr>
<tr><td>102</td><td>Read-Modify-Write (Binary Address)</td></tr>
<tr><td>150</td><td>Word Range Write (ASCII Address)</td></tr>
<tr><td>151</td><td>Word Range Read (ASCII Address)</td></tr>
<tr><td>152</td><td>Read-Modify-Write (ASCII Address)</td></tr>
</table>
SLC Command Set (0x0F)
<table>
<tr><td>501</td><td>Prot Typed Read with 2 addr fields</td></tr>
<tr><td>502</td><td>Prot Typed Read with 3 addr fields</td></tr>
<tr><td>509</td><td>Prot Typed Write with 2 addr fields</td></tr>
<tr><td>510</td><td>Prot Typed Write with 3 addr fields</td></tr>
<tr><td>511</td><td>Prot Type Write with Mask 3 addr field</td></tr>
</table></td></tr>
<tr><td>Function Parameters</td><td>See Command Function Codes (page 34)</td><td>The number of auxiliary parameters required depends on the function code selected for the command.</td></tr>
</table>

2.3 IP Address

In addition to the DFNT.CFG, the MVI71-DFNT module requires a second configuration file that identifies its Ethernet configuration. Without this configuration file, the module will not communicate properly on the network.

This file contains the Ethernet address information to be used by the module and may be transferred to and from the module from the **Network** command available on the debug port of the module. Please consult your network administrator for the correct settings for your network before placing this or any other Ethernet TCP/IP device upon your network.

Important: If the field "my_ip" does not exist, or if the wattcp.cfg file is corrupted or does not exist, the module will not function.

To set the Module's IP Address

1. Locate the sample configuration files for your module on the ProSoft Solutions CD.
2. Copy the configuration files and ladder to a location on your PC's hard drive. We recommend C:\temp.
3. After you move the files, right-click on each of the files, choose Properties, and clear the READ ONLY check box.
4. Start Notepad.exe, or any other editor that can save plain text files.
5. Open the file WATTCP.CFG. The following example shows the contents of a typical WATTCP.CFG file.

```
# ProSoft Technology
# Default private class 3 address
my_ip=192.168.0.100

# Default class 3 network mask
netmask=255.255.255.0

# The gateway I wish to use
gateway=192.168.0.1,192.168.0.0,255.255.255.0
```

6. Edit the file, using the IP addresses supplied by your network administrator.

Important: The module does not support DHCP (Dynamic Host Configuration Protocol) for obtaining an IP address from a server. This module must have its own static IP address that does not duplicate the IP address of any other device on the Ethernet network.

7. Save the file as WATTCP.CFG. You must now transfer the file to the module. Refer to Transferring WATTCP.CFG to the module (page 40, page 61) for the correct procedure.

2.4 Uploading and Downloading the Configuration File

ProSoft modules are shipped with a pre-loaded configuration file. In order to edit this file, you may transfer the file from the module to your PC or locate and load the file from the distribution CD-ROM supplied with the module. After editing, you must transfer the file back to the module for your changes to take effect.

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 51).

2.4.1 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). 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, ProSoft Configuration Builder (PCB)
Windows NT/2000/XP/Vista/7	HyperTerminal, ProSoft Configuration Builder (PCB)

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

2.5 Installing ProSoft Configuration Builder Software

You must install the *ProSoft Configuration Builder (PCB)* software to configure the module. You can always get the newest version of *ProSoft Configuration Builder* from the ProSoft Technology website.

Installing ProSoft Configuration Builder from the ProSoft website

8 Open your web browser and navigate to *http://www.prosoft-technology.com/pcb*
9 Click the **DOWNLOAD HERE** link to download the latest version of *ProSoft Configuration Builder*.
10 Choose **SAVE** or **SAVE FILE** when prompted.
11 Save the file to your *Windows Desktop*, so that you can find it easily when you have finished downloading.
12 When the download is complete, locate and open the file, and then follow the instructions on your screen to install the program.

If you do not have access to the Internet, you can install *ProSoft Configuration Builder* from the *ProSoft Solutions Product CD-ROM*, included in the package with your module.

Installing ProSoft Configuration Builder from the Product CD-ROM

13 Insert the *ProSoft Solutions Product CD-ROM* into the CD-ROM drive of your PC. Wait for the startup screen to appear.
14 On the startup screen, click **PRODUCT DOCUMENTATION**. This action opens a *Windows Explorer* file tree window.
15 Click to open the **UTILITIES** folder. This folder contains all of the applications and files you will need to set up and configure your module.
16 Double-click the **SETUP CONFIGURATION TOOL** folder, double-click the **PCB_*.EXE** file and follow the instructions on your screen to install the software on your PC. The information represented by the **"*"** character in the file name is the *PCB* version number and, therefore, subject to change as new versions of *PCB* are released.

Note: Many of the configuration and maintenance procedures use files and other utilities on the CD-ROM. You may wish to copy the files from the *Utilities* folder on the CD-ROM to a convenient location on your hard drive.

2.6 Module Data

All data related to the MVI71-DFNT module is stored in a user defined data files. It is the responsibility of the ladder logic programmer to construct all the data files required by the program and to write the ladder logic required to interface to these files.

3 Ladder Logic

Ladder logic is required for application of the MVI71-DFNT module. Tasks that must be handled by the ladder logic are module data transfer, special block handling, and status data receipt. Additionally, a power-up handler may be needed to handle the initialization of the module's data and to clear any processor fault conditions.

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

4 Diagnostics and Troubleshooting

In This Chapter

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

- LED status indicators on the front of the module provide general information on the module's status.
- Status data contained in the module can be viewed through the Configuration/Debug port, using the troubleshooting and diagnostic capabilities of *ProSoft Configuration Builder (PCB)*.
- Status data values can be transferred from the module to processor memory and can be monitored there manually or by customer-created logic.

4.1 LED Status Indicators

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

LED	Color	Status	Indication
CFG	Green	On	Data is being transferred between the module and a remote terminal using the Configuration/Debug port.
		Off	No data is being transferred on the Configuration/Debug port.
P1	Green	On	Data is being transferred between the module and the processors Channel 0 port.
		Off	No data is being transferred on this port
P2	Green	On	Data is being transferred on this port and the remote device connected to the port
		Off	No data is being transferred on this port
APP	Amber	Off	The MVI71-DFNT is working normally.
		On	The MVI71-DFNT module program has recognized an error.
BP ACT	Amber	On	The LED is on when the module is performing a write operation on the backplane.
		Off	The LED is off when the module is performing a read operation on the backplane. Under normal operation, the LED should blink rapidly on and off.
OK	Red/ Green	Off	The card is not receiving any power and is not securely plugged into the rack.
		Green	The module is operating normally.
		Red	The program has detected an error or is being configured. If the LED remains red for over 10 seconds, the program has probably halted. Remove the card from the rack and re-insert the card to restart the module's program.
BAT	Red	Off	The battery voltage is OK and functioning.
		On	The battery voltage is low or battery is not present. Allow battery to charge by keeping module plugged into rack for 24 hours. If BAT LED still does not go off, contact ProSoft Technology, as this is not a user serviceable item.

If a configuration error is found for the client, the client configuration error word will have a value other than zero. The configuration error word bits have the following definitions:

Bit	Description	Value
0		0x0001
1		0x0002
2		0x0004
3		0x0008
4	Invalid retry count (0 to 10)	0x0010
5		0x0020
6		0x0040
7		0x0080
8		0x0100
9		0x0200
10		0x0400
11		0x0800
12		0x1000
13		0x2000
14		0x4000
15		0x8000

If a configuration error is present for the pass-through server, the configuration error word contains a value other than zero. The configuration error word bits have the following definitions:

Bit	Description	Value
0	Invalid enable code	0x0001
1	Invalid busytimeout setting (< 100 milliseconds)	0x0002
2		0x0004
3		0x0008
4		0x0010
5	Invalid baud rate	0x0020
6	Invalid parity	0x0040
7	Invalid data bits	0x0080
8	Invalid stop bits	0x0100
9		0x0200
10		0x0400
11		0x0800
12		0x1000
13		0x2000
14		0x4000
15		0x8000

If a configuration error is present for the pass-through port, the configuration error word contains a value other than zero. The configuration error word bits have the following definitions:

Bit	Description	Value
0	Invalid enable code	0x0001
1	Invalid local station ID	0x0002
2	Invalid protocol or termination type	0x0004
3	Invalid baud rate	0x0008
4	Invalid parity	0x0010
5	Invalid data bits	0x0020
6	Invalid stop bits	0x0040
7		0x0080
8	Invalid Use CTS Line selection	0x0100
9	Invalid retry count	0x0200
10		0x0400
11		0x0800
12		0x1000
13		0x2000
14		0x4000
15		0x8000

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

Refer to the Configuration/Debug menu for configuration error words.

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

4.1.1 Ethernet LED Indicators

LED	State	Description
Data	OFF	No activity on the Ethernet port.
	GREEN Flash	The Ethernet port is actively transmitting or receiving data.
Link	OFF	No physical network connection is detected. No Ethernet communication is possible. Check wiring and cables.
	GREEN Solid	Physical network connection detected. This LED must be ON solid for Ethernet communication to be possible.

4.1.2 Clearing a Fault Condition

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

To clear the condition, follow these steps:

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

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

4.1.3 Troubleshooting

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

Processor Errors

Problem description	Steps to take
Processor fault	Verify that the module is plugged into the slot that has been configured for the module in the I/O Configuration of RSLogix. Verify that the slot location in the rack has been configured correctly in the ladder logic.
Processor I/O LED flashes	This indicates a problem with backplane communications. A problem could exist between the processor and any installed I/O module, not just the MVI71-DFNT. Verify that all modules in the rack are correctly configured in the ladder logic.

Module Errors

Problem description	Steps to take
BP ACT LED (not present on MVI56E modules) remains OFF or blinks slowly MVI56E modules with scrolling LED display: *<Backplane Status>* condition reads ERR	This indicates that backplane transfer operations are failing. Connect to the module's Configuration/Debug port to check this. To establish backplane communications, verify the following items: ▪ The processor is in RUN or REM RUN mode. ▪ The backplane driver is loaded in the module. ▪ The module is configured for read and write data block transfer. ▪ The ladder logic handles all read and write block situations. ▪ The module is properly configured in the processor I/O configuration and ladder logic.
OK LED remains RED	The program has halted or a critical error has occurred. Connect to the Configuration/Debug port to see if the module is running. If the program has halted, turn off power to the rack, remove the card from the rack and re-insert it, and then restore power to the rack.

4.2 The Configuration/Debug Menu

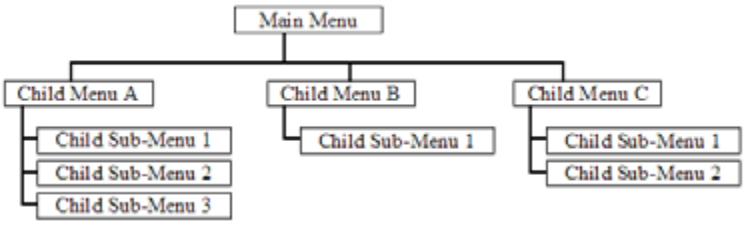
The Configuration and Debug menu for this module is arranged as a tree structure, with the Main Menu at the top of the tree, and one or more sub-menus for each menu command. The first menu you see when you connect to the module is the Main menu.

Because this is a text-based menu system, you enter commands by typing the command letter from your computer keyboard in *Prosoft Configuration Builder (PCB)*. 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 the *Prosoft Configuration Builder (PCB)* application.

4.2.1 Navigation

All of the submenus for this module contain commands to redisplay the menu or return to the previous menu. You can always return from a submenu 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 the menus available for this module, and briefly discusses the commands available to you.

Keystrokes

The keyboard commands on these menus are usually not case sensitive. You can enter most commands in lowercase or uppercase 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** and */*.

Also, take care to distinguish the different uses for uppercase letter "eye" (**I**), lowercase letter "el" (**L**), and the number one (**1**). Likewise, uppercase letter "oh" (**O**) and the number zero (**0**) are not interchangeable. Although these characters look alike on the screen, they perform different actions on the module and may not be used interchangeably.

4.2.2 Using the Configuration/Debug Port

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

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

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

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

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

1. Verify that the null modem cable is connected properly between your computer's serial port and the module. A regular serial cable will not work.
2. Verify that RSLinx is not controlling the COM port. Refer to Disabling the RSLinx Driver for the Com Port on the PC (page 84).
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.

4.2.3 Main Menu

Features available through the use of the configuration/debug port on the MVI71-DFNT module are all accessed using single keystrokes on your computer. There is a single main menu and several sub-menus presented on the port. To view the current selections available, press the [?] key on your computer. If you are at the main menu, the following menu appears:

If this menu is not shown, press the [M] key to display the main menu. All facilities offered by the configuration/debugger are shown on the main menu. Each option is discussed in the following topics.

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 View Menu

Press **[D]** to open the *Database View* menu.

Use this menu command to view the current contents of the module's database. For more information about this submenu, see Database View Menu (page 57).

Opening the Client Command Error List Menu

Press **[E]** to open the Client 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 Client Command List Menu

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

Transferring the Configuration File from the PC to the Module

On the Diagnostics Menu this is referred to as *Receive Module Configuration*.

Press **[R]** to receive (download) the configuration file from your PC to the module and store the file on the module's Compact Flash Card (Personality Module) or Flash RAM.

Press **[Y]** to confirm the file transfer, and then follow the instructions on the terminal screen to complete the file transfer process.

After the file has been successfully downloaded, the module will restart the program and load the new configuration information. Review the new configuration using menu commands **[6]** and **[0]** to verify that the module is configured correctly.

Transferring the Configuration File from The Module to the PC

On the Diagnostics Menu this is referred to as *Send Module Configuration*.

Press **[S]** to send (upload) the configuration file from the module to your PC.

Press **[Y]** to confirm the file transfer, and then follow the instructions on the terminal screen to complete the file transfer process.

After the file has been successfully uploaded, you can open and edit the file to change the module's configuration.

Resetting Diagnostic Data

Press **[U]** to reset the status counters for the Client and/or server(s) in the module.

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

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 reboot.

Viewing Client Communication Status

Press **[1]** to view client communication status. Use this command to view the statistics of the DFNT client commands sent by the MVI71-DFNT. The following illustration shows an example of the information on this screen.

```
CLIENT 0 STATUS:
  Retries : 0       Cur Cmd : 0       TSN : 22499
  State   : 7       Timeout : 995546
  Cfg Err : 0x0000  Cur Err : 0       Last Err: -33
  Number of Command Requests: 22499   Number of Cmd Responses  : 22499
  Number of Command Errors  : 56      Number of Requests       : 22501
  Number of Responses       : 22501   Number of Errors Received: 0
  Number of Errors Sent     : 0
```

Viewing Server Status Data

Press **[2], [3], [4]** or **[5]** to view status data for the DFNT servers. The following illustration shows an example of the status screen for the selected servers:

```
DFNT SERVER STATUS
 SERVER 0: Port : 0000      IP      : 0.0.0.0
           State: 0         Connect: 0         Open    : 1
           Estab: 0         Read    : 0         Message: 0
           Write: 0         Close   : 0         Timeout: 0
 SERVER 1: Port : 0000      IP      : 0.0.0.0
           State: 0         Connect: 0         Open    : 1
           Estab: 0         Read    : 0         Message: 0
           Write: 0         Close   : 0         Timeout: 0
 SERVER 2: Port : 0000      IP      : 0.0.0.0
           State: 0         Connect: 0         Open    : 1
           Estab: 0         Read    : 0         Message: 0
           Write: 0         Close   : 0         Timeout: 0
 SERVER 3: Port : 0000      IP      : 0.0.0.0
           State: 0         Connect: 0         Open    : 1
           Estab: 0         Read    : 0         Message: 0
           Write: 0         Close   : 0         Timeout: 0
 SERVER 4: Port : 0A29      IP      : 192.168.0.57
           State: 1         Connect: 1         Open    : 1
           Estab: 1         Read    : 245       Message: 245
           Write: 244       Close   : 0         Timeout: 0
```

2 = Display servers 0 to 4

3 = Display servers 5 to 9

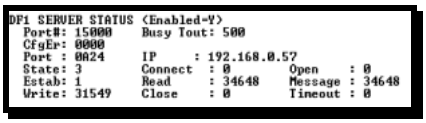
4 = Display servers 10 to 14

5 = Display servers 15 to 19

Note: Some implementations of the DFNT protocol support fewer DFNT servers (for example, 5201-DFNT-BACNET, which supports only five servers, rather than the twenty shown in this illustration)

Viewing DF1 Server Status Data

Press **[7]** to view communication status for the DF1 pass through server.



Viewing Client Configuration

Press **[8]** to display the configuration information for the client.

Viewing DF1 Pass-Through Server Configuration

Press **[9]** to view configuration information for the DF1 pass-through server.

Viewing DF1 Pass Through Port Configuration

Press **[0]** (zero) to view configuration information for the DF1 pass-through port.

Opening the Network Menu

Press **[@]** to open the *Network* menu.

The *Network* menu allows you to send, receive and view the WATTCP.CFG file that contains the IP, gateway and other network specification information. For more information about this submenu, see Network Menu (page 61).

Exiting the Program

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

4.2.4 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 database. Press **[?]** to view a list of commands available on this menu.

```
DB Menu Selected

DATABASE VIEW MENU
 ?=Display Menu
 0-9=Display 0-9000
 S=Show Again
 -=Back 5 Pages
 P=Previous Page
 +=Skip 5 Pages
 N=Next Page
 D=Decimal Display
 H=Hexadecimal Display
 F=Float Display
 A=ASCII Display
 M=Main 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
     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 100 registers of data starting 500 registers before the currently displayed page.

Moving Forward (Skipping) Through 5 Pages of Registers

Press **[+]** from the *Database View* menu to skip five pages ahead in the database to see the 100 registers of data starting 500 registers after the currently displayed page.

Viewing the Previous Page of Registers

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

Viewing the Next Page of Registers

Press **[N]** from the *Database View* menu to display the next page of data.

Viewing Data in Decimal Format

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

Viewing Data in Hexadecimal Format

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

Viewing Data in Floating-Point Format

Press **[F]** from the *Database View* menu 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

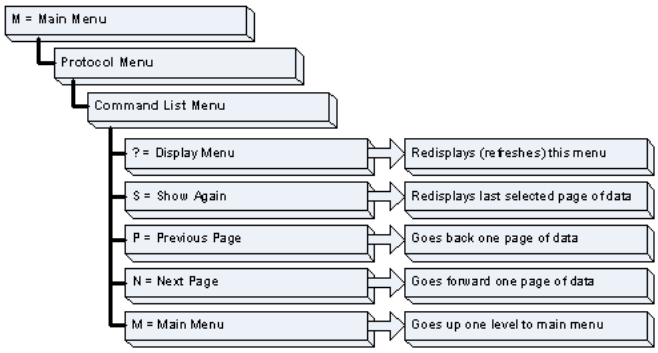
Press **[A]** from the *Database View* menu to display the data on the current page in ASCII format. This is useful for regions of the database that contain ASCII data.

Returning to the Main Menu

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

4.2.5 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.

Moving Back Through 5 Pages of Commands

Press **[-]** to display data for last 5 page commands.

Viewing the Previous Page of Commands

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

Moving Forward (Skipping) Through 5 Pages of Commands

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

Viewing the Next Page of Commands

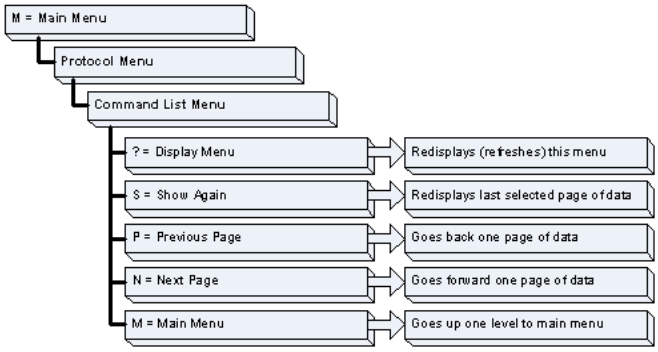
Press **[N]** to display the next page of commands.

Returning to the Main Menu

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

4.2.6 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

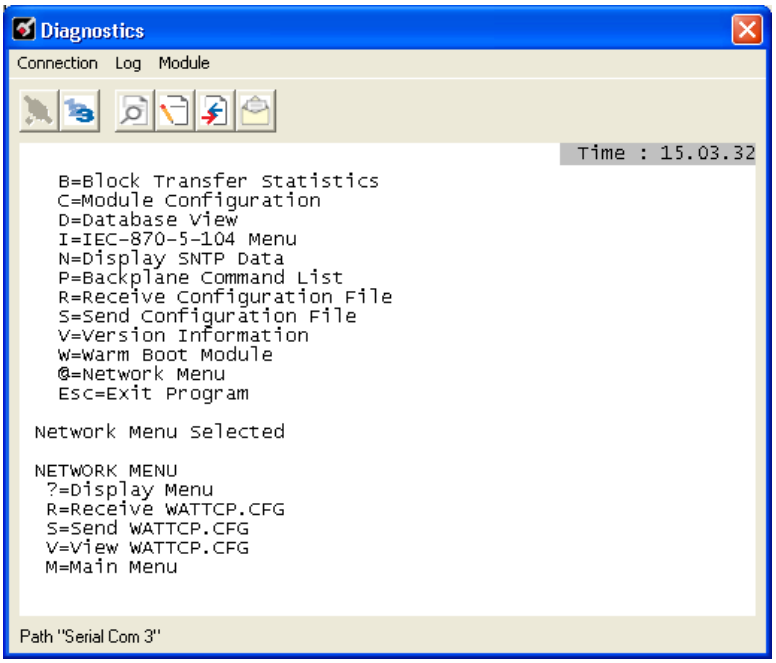
Press **[N]** to display the next page of commands.

Returning to the Main Menu

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

4.2.7 Network Menu

From the *main* menu press **[@]** to display the *Network* menu screen. The *Network* menu allows you to send, receive, and view the WATTCP.CFG file that contains the IP and module addresses, and other network information.



Transferring WATTCP.CFG to the Module

Press **[R]** to transfer a new WATTCP.CFG file from the PC to the module. Use this command to change the network configuration for the module (for example, the module's IP address).

Press **[Y]** to confirm the file transfer, and then follow the instructions on the terminal screen to complete the file transfer process.

Transferring WATTCP.CFG to the PC

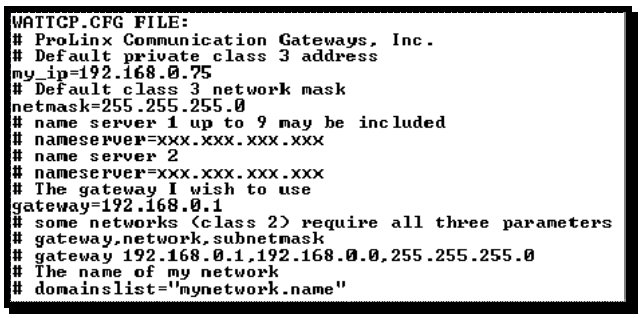
Press **[S]** to transfer the WATTCP.CFG file from the module to your PC.

Press **[Y]** to confirm the file transfer, and then follow the instructions on the terminal screen to complete the file transfer process.

After the file has been successfully transferred, you can open and edit the file to change the module's network configuration.

Viewing the WATTCP.CFG File on the module

Press **[V]** to view the module's WATTCP.CFG file. Use this command to confirm the module's current network settings.



Returning to the Main Menu

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

4.3 Reading Status Data from the Module

The MVI71-DFNT module returns two status data blocks that can be used to determine the module's operating status. This data is requested by the ladder logic and returned in the module's M1 file. This data can also be viewed using the Configuration/Debug port with *Prosoft Configuration Builder (PCB)*. The Configuration/Debug port provides the following functionality:

- Full view of the module's configuration data
- View of the module's status data
- Complete display of the module's internal database (registers 0 to 3999)
- Version Information
- Control over the module (warm boot, cold boot)
- Facility to upload and download the module's configuration file

5 Reference

In This Chapter

5.1 Product Specifications

5.1.1 EtherNet/IP (Explicit Messaging) Compatible Devices

List of Rockwell Automation material that support EPIC:

- PLC5/E rev C/N, D/E, E/D
- SLC5/05 series A, OS503 frn4
- 1785-ENET Series A, rev D
- Interchange V6.2
- MicroLogix 1100/1400/ANY via 1761-NET-ENI
- CompactLogix 1768-L43/L45 via 1768-ENBT
- CompactLogix 1769-L32E/L35E/ANY via 1761-NET-ENI
- CompactLogix L23E
- RSLinx Gateway V1.7+
- ControlLogix 1756-ENET/ENBT/EN2T

5.1.2 General Specifications

- Single Slot - 1771 backplane compatible
- The module is recognized as an Input/Output module and has access to processor memory for data transfer between processor and module
- Ladder Logic is used for data transfer between module and processor. Sample ladder file included.
- Configuration data obtained from configuration text file downloaded to module. Sample configuration file included.

5.1.3 Hardware Specifications

Specification	Description
Form Factor	Single Slot 1771 chassis compatible BTR/BTW data transfer Local or remote rack
Backplane current load	800 mA @ 5 V
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 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 (CFG)	DB-9M PC compatible RS-232 Hardware handshaking
Ethernet Port (Ethernet modules)	RJ45 Connector Link and activity LED indicators Electrical Isolation 1500 V rms at 50 Hz to 60 Hz for 60 s, applied as specified in section 5.3.2 of IEC 60950: 1991 Ethernet Broadcast Storm Resiliency = less than or equal to 5000 [ARP] frames-per-second and less than or equal to 5 minutes duration

5.1.4 Functional Specifications

- PLC processor in-chassis EtherNet/IP bridge to optimize data throughput while communicating with HMI or SCADA and other control platforms without losing valuable process control bandwidth
- Support of up to 4000 PLC registers user data files
- User-definable module memory usage
- 10/100 Base-T Ethernet compatible interface
- Functions as a server or a client
- Configurable parameters for the client include:
  - A minimum response delay of 0 to 65535 milliseconds
  - A response timeout of 1 to 65535 milliseconds
  - A retry count of 0 to 20
- Status data available in ladder logic

Support for PLC processor programming over Ethernet using a TCP/IP service and a serial port on the module connected to channel 0 of the processor. The module's third port emulates Channel 0 of the processor to pass through messages from the port to the processor

Server Specifications

- Supports EtherNet/IP explicit, connected, and unconnected class messaging
- Twenty independent server connections permit remote clients to interact with all data contained in the module
- Data can be derived from other clients on the network, through the client on the module, or from the PLC processor

Client Specifications

- Actively issues connected, explicit messages to other nodes on the network
- Supports 100 user-defined commands from a single client
- Allows command control from ladder logic
- Pass-through services
- Permits remote programming of the PLC processor on the Ethernet network via a pass-through
- TCP/IP service and a serial communication port (pass-through port) on the module
- The third port on the module can emulate the Channel 0 port on the PLC, This allows a DF1 master device attached to the emulated (Channel 0) port to monitor and control data in the PLC

5.2 Functional Overview

5.2.1 General Concepts

The following discussion explains several concepts that are important for understanding module operation.

Module Power Up

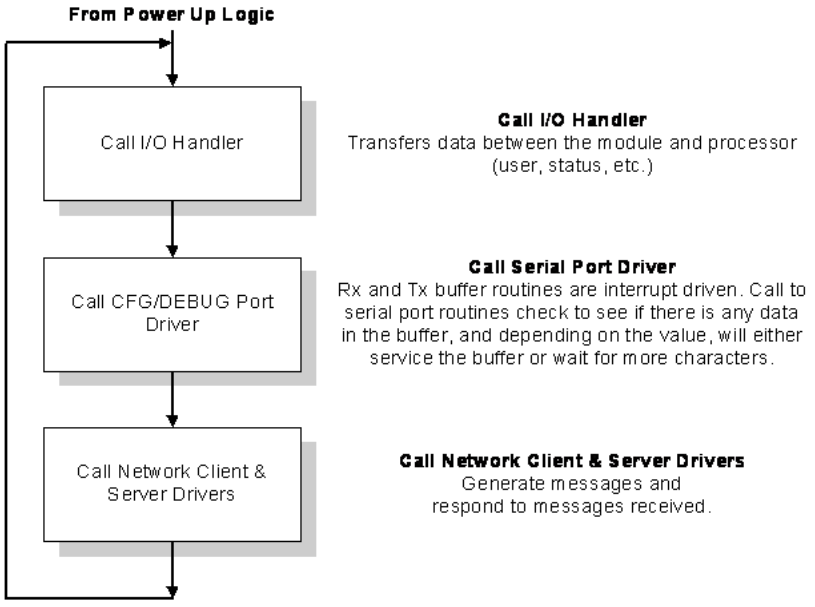
On power up the module begins performing the following logical functions:

1. Initialize hardware components
2. Initialize PLC backplane driver
   - Test and clear all RAM
   - Initialize the serial communication ports
   - Read configuration for module from DFNT.CFG file on Compact Flash Disk
3. Initialize Module Register space
4. Enable Server Drivers
5. Enable Client Driver
6. Initialize all serial communication ports

After the module receives the configuration, the module begins communicating with other nodes on the network, depending on the configuration.

Main Logic Loop

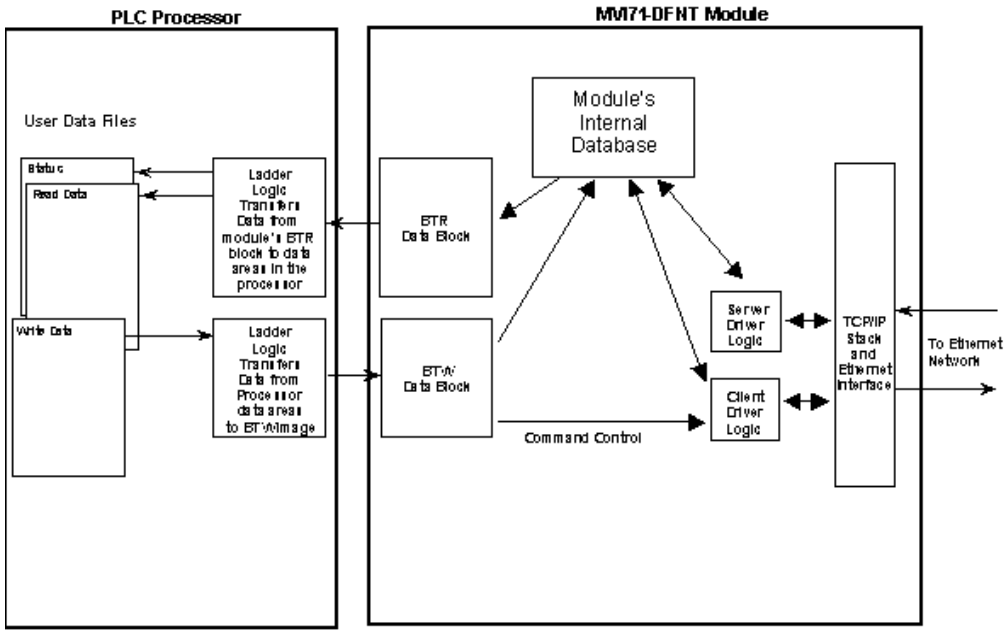
Upon completing the power-up configuration process, the module enters an infinite loop that performs the functions shown in the following diagram.



Block Transfer Backplane Data Transfer

The MVI71-DFNT module communicates directly over the PLC backplane. Data is paged between the module and the PLC processor across the backplane using BTR and BTW operations. Data is transferred from the module to the processor using the BTR blocks, and data is transferred from the processor to the module using BTW blocks.

The following illustration shows the data transfer method used to move data between the PLC processor, the MVI71-DFNT module, and the Ethernet Network.



As shown in the diagram, all data transferred between the module and the processor over the backplane is through the BTR and BTW blocks. Ladder logic must be written in the PLC processor to interface the block data with the module's internal database. All data used by the module is stored in its internal database. The following illustration shows the layout of the database:

Module's Internal Database Structure

4000 registers for user data		0
	Register Data	
		3999

5.2.2 Normal Data Transfer

Normal data transfer includes the transferring of data between the MVI71-DFNT database and the PLC data files. These data are transferred through read (BTR) and write (BTW) blocks. Refer to the Module Configuration section for a description of the ladder logic required to perform the data movement within the PLC processor. The structure and function of each block is discussed in the following topics.

Read Block

These blocks of data transfer information from the module to the PLC processor. When data is received on one of the servers, a data block is built. The structure of this block type is shown in the following table.

Word Offset	Description
0	Read Block ID
1	Next Write Block ID
2 to 63	Read Data

Write Block

These blocks of data transfer information from the PLC processor to the module. The structure of the BTW block is shown in the following table.

Word Offset	Description
0	Write Block ID (copied from the previous Read block)
1 to 63	Write Data

The following shows the valid block IDs for normal transfer:

Block ID	Definition
-1 and 0	Null blocks that do not contain any data.
1 to 67	Data read and write blocks to transfer data for the module's database between the processor and the module.

How Data is Transferred

In order to understand how the data is transferred between the processor and the module, you must understand the Read Data and Write Data area concept in the module's database. The module's database can be partially, or totally divided into Read Data Areas and Write Data Areas.

These areas are defined by the user when the configuration file is being edited. The following parameters define the Read and Write data areas:

Read Register Start = 0

Read Register Count = 120

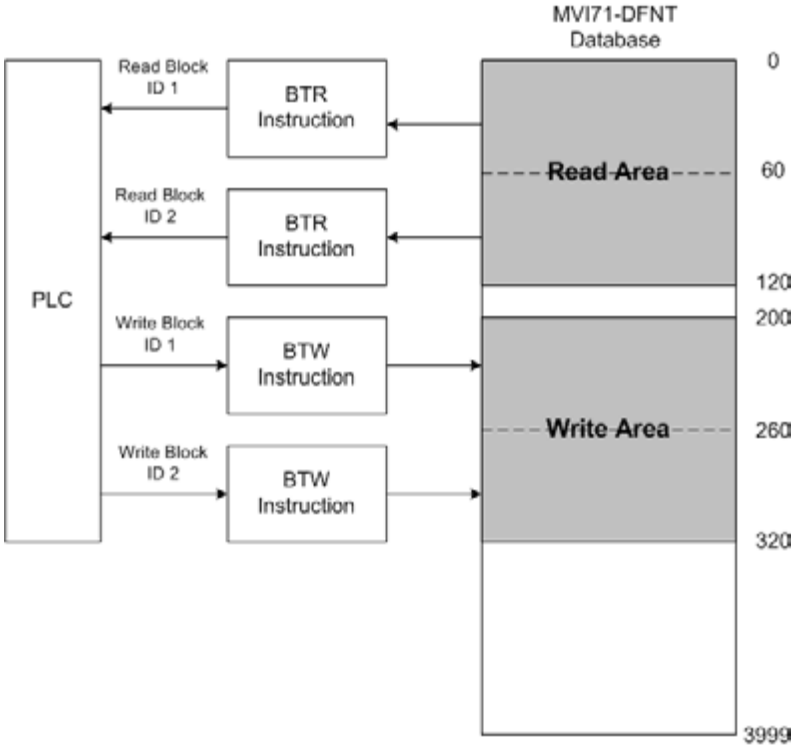
Write Register Start = 200

Write Register Count = 120

Each area is broken down into blocks of 60 words. Therefore, the Read Register Count and Write Register Count parameters should be multiples of 60.

The Read Data Area will be transferred from the module to the PLC processor. The Write Data Area will be transferred from the PLC processor to the module.

The following example shows the resulting data flow:



5.2.3 Module Control Blocks

Specific write block IDs are reserved for module control operations. These blocks request that the module perform specific tasks. The following write blocks are valid for module control.

Block ID	Definition
250 and 251	Status data request and response blocks
1000 to 1066	Blocks used to initialize the module's database with values in the processor on startup.
2000	Request and respond with command list error data for a set of commands.
3000	Set the enable code for a set of commands to 0 to disable polling.
3001	Set the enable code for a set of commands to 1 to enable polling.
3002	Set the enable code for a set of commands to 2 to enable conditional polling.
9998	Request block to warm boot the module
9999	Request block to cold boot the module

Status Data Request Block (250 to 254)

In order to read the module's general error and status data to the PLC, it must make a special request using the status data request block. The following tables lists the values recognized by the module:

Offset	Description
250	Module and pass-through port end server status
251	Status for servers 0 to 4
252	Status for servers 5 to 9
253	Status for servers 10 to 14
254	Status for servers 15 to 19

MVI71-DFNT Status Data Definition contains a complete listing of the data returned for the status blocks.

Initialize Output Data Blocks (1000 to 1066)

When the module performs a restart operation, it requests output data from the processor to initialize the module's read data area. Use the **Initialize Output Data** parameter in the configuration file to bring the module to a known state after a restart operation. The structure of the block used to request the data is shown in the following table.

Block Request

Offset	Description	Length
0	1000 to 1066	1
1	1000 to 1066	1
2 to 63	Not used	62

The command control value of 1000 is moved as the Block Transfer Write ID to indicate that the module is requesting initialization of the Read Data area. Ladder logic in the processor must recognize this command and place the correct information in the database. The format of the returned write block is shown in the following table.

Block Response

Offset	Description	Length
0	1000 to 1066	1
1 to 60	Data to place in database	60
61 to 63	Not used	3

For example, for a Read Data Area of 2 blocks (120 words), blocks 1000 and 1001 would be used.

Command Error List Request Block (2000)

This command control request (control code of 2000) requests the command list error data set. The error codes returned in the block are DFNT error codes noted in the Reference chapter. The format of the request block from the ladder logic has the following format:

Block Request

Offset	Description	Length
0	2000	1
1	Command start index	1
2	Command count	1
3 to 63	Not used	61

After the module processes the block, it supplies the following values in the control register area:

Block Response

Offset	Description	Length
0	2000	1
1	Block write ID	1
2	Command start index	1
3	Number of errors in list	1
4 to 63	Command error list returned	60

Command Control Blocks (3000 to 3002)

Blocks 3000 to 3002 alter the command type field for a set of commands in the client command lists. Block 3000 disables commands by setting the enable type field to value of 0. Block 3001 enables commands by setting the enable type field to a value of 1. The commands will be issued at the time interval no more frequent than set in the poll interval parameter for the command. Block 3002 sets the enable type field to a value of 2. This operation should only be used for write functions as the command is only executed when the data referenced by the command changes. The general format for the blocks is as follows:

Block Request

Offset	Description	Length
0	3000 to 3002	1
1	Command count	1
2 to 63	List of command indices on which to perform the operation.	62

After the module processes the block, it supplies the following values in the control register area:

Block Response

Offset	Description	Length
0	3000 to 3002	1
1	Block write ID	1
2	Number of commands processed	1
3 to 63	Not used	61

Warm Boot Block (9998)

This block is sent from the PLC processor to the module when the module is required to perform a warm-boot (software reset) operation. This block is commonly sent to the module any time configuration data modifications are made in the controller tags data area. This causes the module to read the new configuration information and to restart. The following table describes the format of the control block.

Block Request

Offset	Description	Length
0	9998	1
1 to 63	Not used	63

Cold Boot Block (9999)

This block is sent from the PLC processor to the module when the module is required to perform the cold boot (hardware reset) operation. This block is sent to the module when a hardware problem is detected by the ladder logic that requires a hardware reset. The following table describes the format of the control block.

Block Request

Offset	Description	Length
0	9999	1
1 to 63	Not used	63

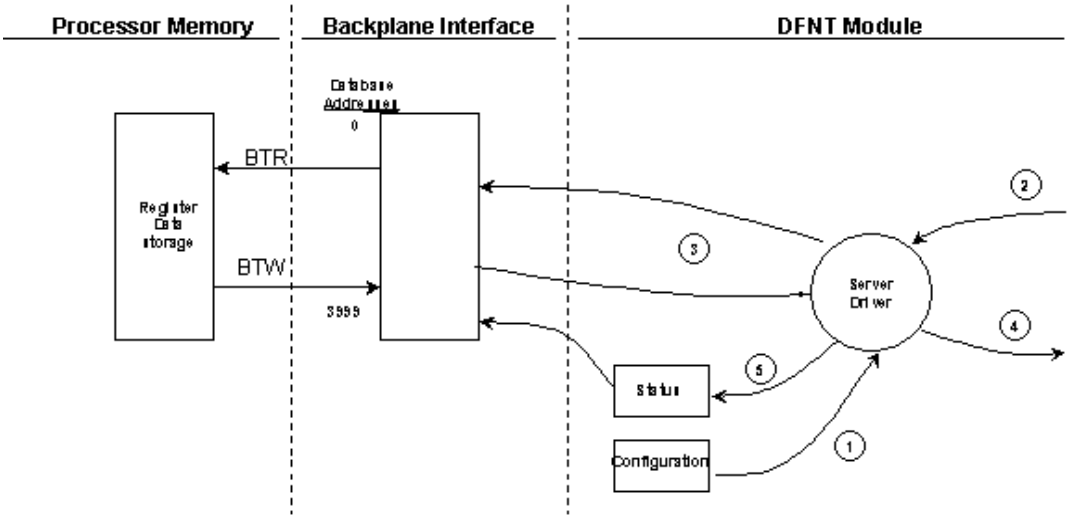
5.2.4 Data Flow between MVI71-DFNT Module and PLC Processor

The following discussion outlines the flow of data between the two pieces of hardware (PLC processor and MVI71-DFNT module) and other nodes on the TCP/IP network under the module's different operating modes. The module contains both servers and a Client.

The following topics discuss the operation of the server and Client drivers.

Server Driver

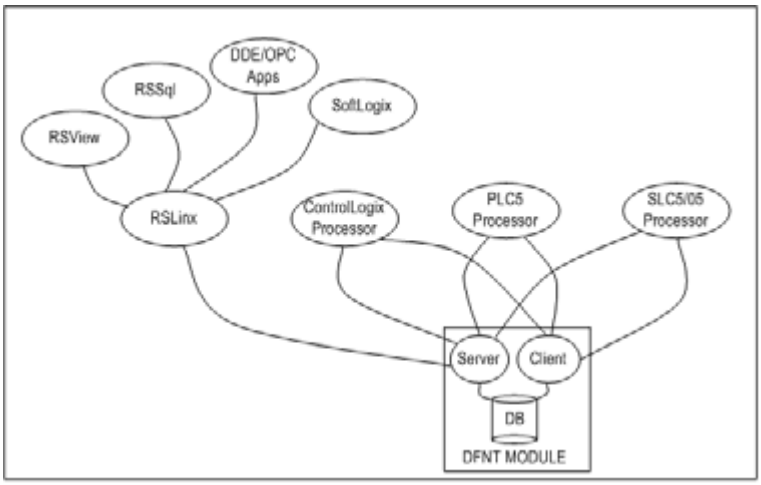
The Server Driver allows the MVI71-DFNT module to respond to data read and write commands issued by clients on the Ethernet/IP network using explicit messaging. The following flow chart and associated table describe the flow of data into and out of the module.



Step	Description
1	The server driver receives the configuration information from the configuration file on the Compact Flash Disk, and the module initializes the servers.
2	A Host device, such as a ControlLogix processor, RSLinx or an HMI application issues a read or write command to the module. The server 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	Status data for the servers is passed to the processor under ladder logic control using the command control data area in the M1 file.

The DFNT module supports server functionality using the reserved ControlNet service port 0xAF12. Services supported in the module permit client applications (that is, RSView, ControlLogix processors and RSLinx) to read from and write to the module's database. This document discusses the requirements for attaching to the module using several client applications.

The following illustration shows the relationship of the DFNT module's functionality to devices on an Ethernet network:



Server functionality places all data transfer operations outside the module. There is no configuration required in the module other than setting up the network and database parameters in the configuration file. Ladder logic in attached processors use MSG instructions to perform read and write operations on the module's internal database.

When RSLinx links a user application to the module, the module's server functionality must be used. RSLinx exists on an Ethernet network only as a client application. It cannot act as a server. User applications can use the DDE/OPC capabilities built into RSLinx to interface with the data in the DFNT module. RSView can link directly to the module using drivers supplied by RSLinx.

The internal database of the DFNT module is used as the source (read requests) and destination (write requests) for requests from remote clients. Access to the database depends on the MSG command type executed to interface with the database. The following table defines the relationship of the module's internal database to the addresses required in the MSG instructions:

MSG Instruction Type

Database Address	PLC2	PLC5 or SLC	ControlLogix	
			PCCC	CIP Integer
0	0	N10:0	N10:0	Int_data[0]
999	999	N19:99	N19:99	Int_data[999]
1000	1000	N20:0	N20:0	Int_data[1000]
1999	1999	N29:99	N29:99	Int_data[1999]
2000	2000	N30:0	N30:0	Int_data[2000]
2999	2999	N39:99	N39:99	Int_data[2999]
3000	3000	N40:0	N40:0	Int_data[3000]
3999	4000	N49:99	N49:99	Int_data[3999]

MSG Instruction Type

Database Address	CIP Boolean	ControlLogix			
		CIP Bit Array	CIP Byte	CIP Double Int	CIP Real
0	BoolData[0]	BitAData[0]	SIntData[0]	DIntData[0]	RealData[0]
999	BoolData[15984]		SIntData[1998]		
1000	BoolData[16000]	BitAData[500]	SIntData[2000]	DIntData[500]	RealData[500]
1999	BoolData[31984]		SIntData[3998]		
2000	BoolData[32000]	BitAData[1000]	SIntData[4000]	DIntData[1000]	RealData[1000]
2999	BoolData[47984]		SIntData[5998]		
3000	BoolData[48000]	BitAData[1500]	SIntData[6000]	DIntData[1500]	RealData[1500]
3999	BoolData[63999]		SIntData[9998]		

When using PLC5 or SLC commands, access to the database is through simulated "N" files. For example, to access database element 3012, use the file address of N40:12. When using CIP Data Table Read or Write commands, use the various data[ ] tag arrays described in the following table. For example, use int_data[3012] to access database register 3012 as an integer value.

Data Type	Tag Name	Length of Each Element in CIP message	Array Range for 4000 Element Database
BOOL	BOOLData[ ]	1	0 to 63999
Bit Array	BITAData[ ]	4	0 to 1999
SINT	SINTData[ ]	1	0 to 7999
INT	INT_Data[ ]	2	0 to 3999
DINT	DINTData[ ]	4	0 to 1999
REAL	REALData[ ]	4	0 to 1999

Before attempting to use the module on a network, verify that the DFNT module is correctly configured and connected to the network. A network program such as PING can be utilized to make certain the module can be seen on the network. Use ProSoft Configuration Builder to verify correct operation, and to transfer configuration files to and from the module.

The following table shows the supported commands when the module acts as a slave (server):

Basic Command Set Functions

Command	Function	Definition	Supported in Slave
0x00	N/A	Protected Write	X
0x01	N/A	Unprotected Read	X
0x02	N/A	Protected Bit Write	X
0x05	N/A	Unprotected Bit Write	X
0x08	N/A	Unprotected Write	X

PLC-5 Command Set Functions

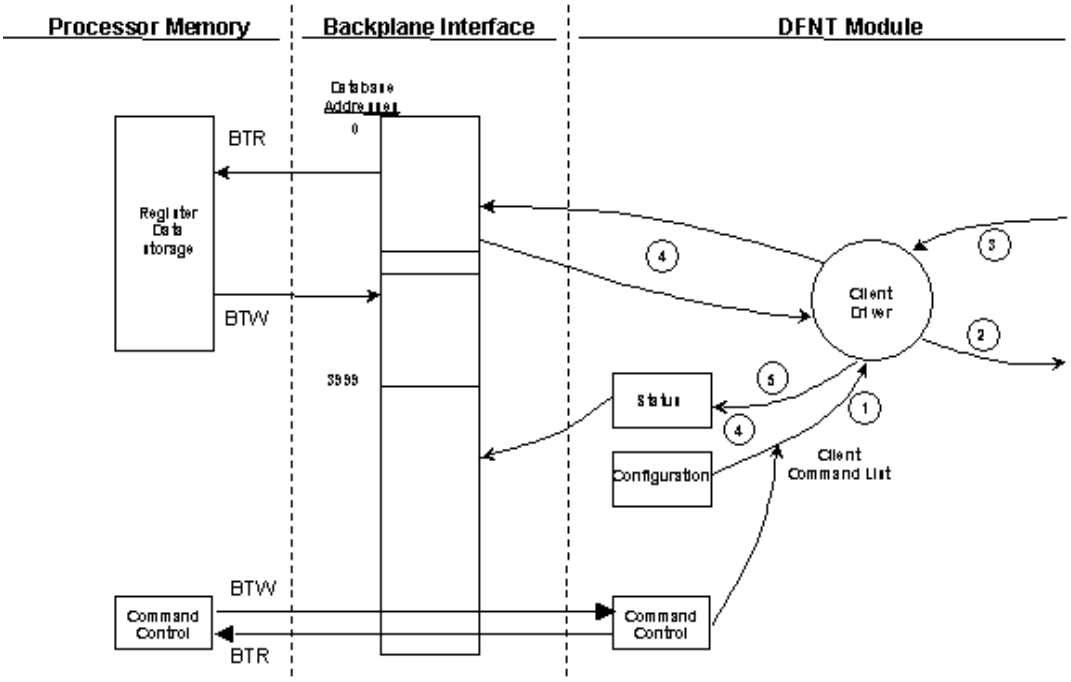
Command	Function	Definition	Supported in Slave
0x0F	0x00	Word Range Write (Binary Address)	X
0x0F	0x01	Word Range Read (Binary Address)	X
0x0F		Typed Range Read (Binary Address)	X
0x0F		Typed Range Write (Binary Address)	X
0x0F	0x26	Read-Modify-Write (Binary Address)	
0x0F	0x00	Word Range Write (ASCII Address)	X
0x0F	0x01	Word Range Read (ASCII Address)	X
0x0F	0x26	Read-Modify-Write (ASCII Address)	

SLC-500 Command Set Functions

Command	Function	Definition	Supported in Slave
0x0F	0xA1	Protected Typed Logical Read With Two Address Fields	X
0x0F	0XA2	Protected Typed Logical Read With Three Address Fields	X
0x0F	0XA9	Protected Typed Logical Write With Two Address Fields	X
0x0F	0XAA	Protected Typed Logical Write With Three Address Fields	X
0x0F	0XAB	Protected Typed Logical Write With Mask (Three Address Fields)	

Client Driver

In the client driver, the MVI71-DFNT module is responsible for issuing read or write commands to servers on the Ethernet/IP network using explicit, connected messaging. These commands are user configured in the module via the Client Command List received from the module's configuration file (DFNT.CFG). Command status is returned to the processor for each individual command in the command list status block in the command control data area. Ladder logic is responsible for acquiring this data from the module. The following flow chart and associated table show the flow of data into and out of the module.



Step	Description
1	The client driver obtains configuration data from the DFNT.CFG file when the module restarts. The configuration data obtained includes the timeout parameters and the Command List. These values are used by the driver to determine the type of commands to be issued to the other nodes on the Ethernet/IP (see Module Configuration).
2	After configuration, the client 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 client 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 data is returned to the PLC processor for the client and a Command List error table can be established in the module's internal database. This data is requested using the command control data area and is a responsibility of the ladder logic.

The Module Setup section provides a complete description of the parameters required to define the client.

Client Command List

In order for the client to function, the module's Client Command List must be defined. This list contains up to 100 individual entries, with each entry containing the information required to construct a valid command. This includes the following:

- Command enable mode ((0) disabled, (1) continuous or (2) conditional)
- IP address of the remote server
- Slot number for processor when interfacing with a ControlLogix processor
- Command Type - Read or Write command
- Database Address - Determines where data will be placed and/or obtained
- Address information to access data in remote unit
- Count - Select the number of words to be transferred
- Poll Delay - (1/10th seconds)

5.3 Cable Connections

The MVI71-DFNT module has the following functional communication connections installed:

- One Ethernet port (RJ45 connector)
- One RS-232 Configuration/Debug port (RJ45 connector)

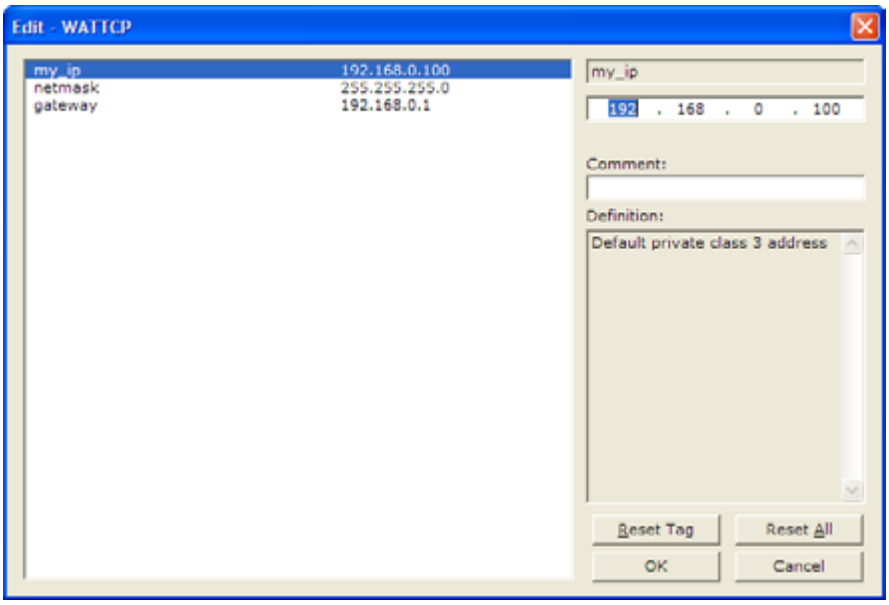
5.3.1 Ethernet Connection

The MVI71-DFNT module has an RJ45 port located on the front of the module, labeled *Ethernet*, for use with the TCP/IP network. The module is connected to the Ethernet network using an Ethernet cable between the module's Ethernet port and an Ethernet switch or hub.

Note: Depending on hardware configuration, you may see more than one RJ45 port on the module. The Ethernet port is labeled *Ethernet*.
Warning: The MVI71-DFNT module is NOT compatible with Power Over Ethernet (IEEE802.3af / IEEE802.3at) networks. Do NOT connect the module to Ethernet devices, hubs, switches or networks that supply AC or DC power over the Ethernet cable. Failure to observe this precaution may result in damage to hardware, or injury to personnel.
Important: The module requires a static (fixed) IP address that is not shared with any other device on the Ethernet network. Obtain a list of suitable IP addresses from your network administrator BEFORE configuring the Ethernet port on this module.

Ethernet Port Configuration - wattcp.cfg

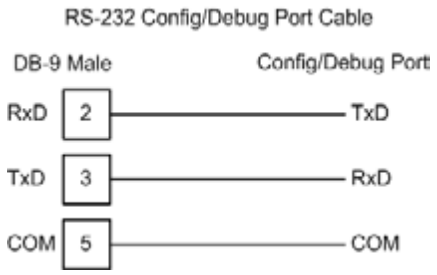
The wattcp.cfg file must be set up properly in order to use a TCP/IP network connection. You can view the current network configuration in *ProSoft Configuration Builder (PCB)*, as shown:



You may also view the network configuration using a PC serial port connection and an ASCII terminal program (like Windows HyperTerminal) by selecting **[@]** (Network Menu) and **[V]** (View) options when connected to the Debug port. For more information on serial port access, see the chapter on Diagnostics and Troubleshooting (page 45).

5.3.2 RS-232 Configuration/Debug Port

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



Disabling the RSLinx Driver for the Com Port on the PC

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

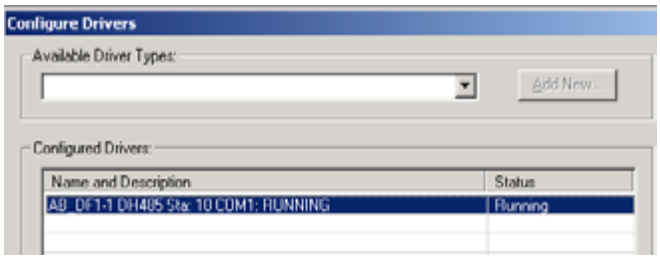
1. Open *RSLinx* and go to **COMMUNICATIONS** > **RSWHO**.
2. Make sure that you are not actively browsing using the driver that you wish to stop. The following shows an actively browsed network.

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

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

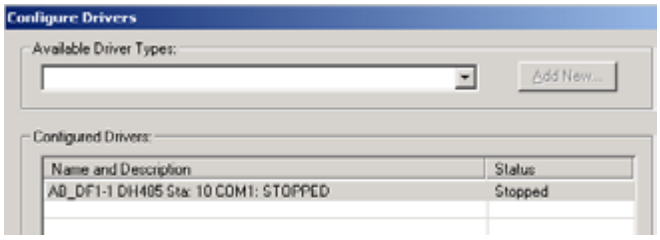
4 When you have verified that the driver is not being browsed, go to **COMMUNICATIONS** > **CONFIGURE DRIVERS**.

You may see something like this:



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

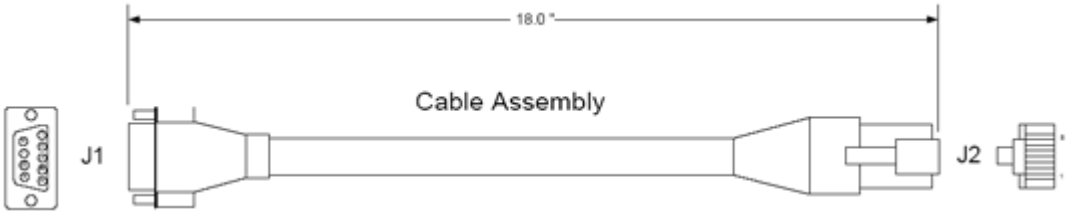
5 After you have stopped the driver you will see the following.

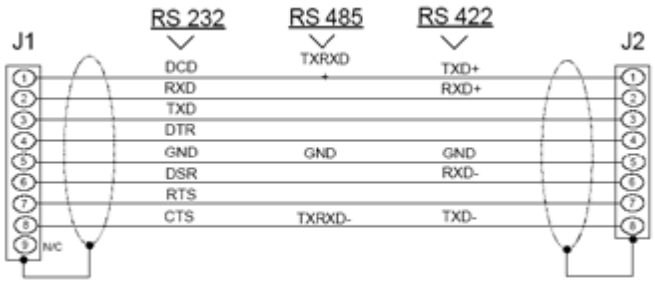


6 You may now use the com port to connect to the debug port of the module.

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

5.3.3 DB9 to RJ45 Adaptor (Cable 14)

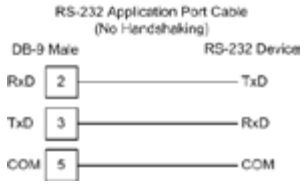


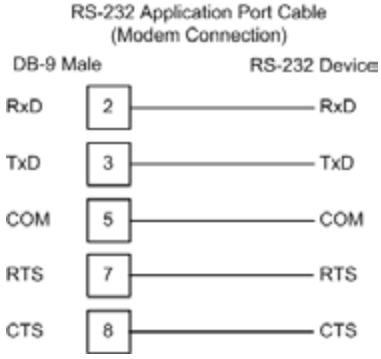


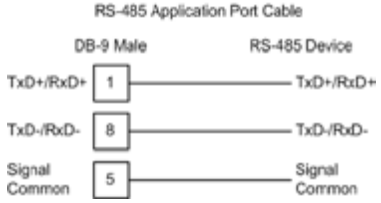
Wiring Diagram

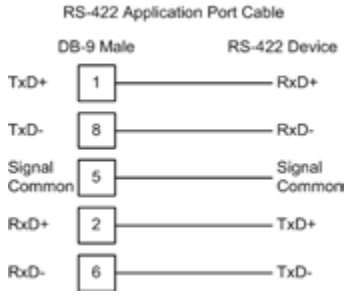
5.4 Pass-Through Ports

Two pass-through ports are provided on the module. Port 2 can be connected to the processor's Channel 0 port and Port 3 can be connected to a remote DF1 master device. The cable configuration used on the ports depends on the RS-interface selected for the port using the jumpers located on the MVI circuit board. The following are port pin-outs for several configurations of the ports:









5.5 MVI71-DFNT Status Data Definition

This section contains a description of the members present in the status data blocks returned to the PLC processor under ladder logic control. The five blocks, 250, 251, 252, 253, and 254, are requested and returned in the module's M1 file in the command control data area under ladder logic control.

The data set returned to the processor in the M1 file for a 250 request has the following definition:

5.5.1 BTR Response Block (250)

Offset	Content	Description
0	Block read ID	This word contains the value of 250 to indicate this specific status block
1	Block Write ID	This word contains the next write block to receive from the processor.
2	Program Scan Count	This value is incremented each time a complete program cycle occurs in the module.
3 to 4	Product	Product Name (ASCII)
5 to 6	Rev	Revision (ASCII)
7 to 8	Op Sys	Operating System (ASCII)
9 to 10	Run	Production Run Number (ASCII)

Reserved

Offset	Content	Description
11	Not Used	Reserved
12	Not Used	Reserved
13	Not Used	Reserved
14	Not Used	Reserved
15	Not Used	Reserved
16	Not Used	Reserved
17	Not Used	Reserved
18	Not Used	Reserved
19	Not Used	Reserved
20	Not Used	Reserved

Client 0 Status

Offset	Content	Description
21	Client Cmd Request	This value is incremented each time a command request is issued.
22	Client Cmd Response	This value is incremented each time a command response is received.
23	Client Cmd Error	This value is incremented each time an error message is received from a remote unit or a local error is generated for a command.
24	Client Request Count	This value is incremented each time a request message is issued.
25	Client Response Count	This value is incremented each time a response message is received.
26	Client Error Sent Count	This value is incremented each time an error is sent from the client.
27	Client Error Received Count	This value is incremented each time an error is received from a remote unit.
28	Client Cfg Error Word	This word contains a bit map that defines configuration errors in the configuration file for the client.
29	Client Current Error Code	This value corresponds to the current error code for the client.
30	Client Last Error Code	This value corresponds to the last error code recorded for the client.

Block Status

Offset	Content	Description
31	Read Block Count	This field contains the total number of read blocks transferred from the module to the processor.
32	Write Block Count	This field contains the total number of write blocks transferred from the processor to the module.
33	Parse Block Count	This field contains the total number of blocks successfully parsed that were received from the processor.
34	Command Event Block Count	This field contains the total number of command event blocks received from the processor.
35	Command Block Count	This field contains the total number of command blocks received from the processor.
36	Error Block Count	This field contains the total number of block errors recognized by the module.

Pass-Through Server Status

Offset	Content	Description
37	Socket State	
38	Connection State	
39	Open Count	Total number of times the server has performed an open operation.
40	Establish Count	Total number of times a connection has been established on the server.
41	Close Count	Total number of times the server has performed a close operation.
42	Read	Total number of packets received by the server.
43	Message	Total number of message receive by the server.
44	Write	Total number of packets sent from the server to the client.
45	Timeout	Total number of times the server as reached a connection timeout condition.
46	Host Port	Service port on client connected to the server.
47 to 48	Host IP Address	IP address of the client connected to the server.
49	PTP Cfg Error Word	Pass-through port configuration error word.
50	PTS Cfg Error Word	Pass-through server configuration error word.

5.5.2 BTR Response Block (251)

Offset	Content	Description
0	Block read ID	This word contains the value of 251 to indicate this specific status block
1	Block Write ID	This word contains the next write block to receive from the processor.

Server 0 Status

Offset	Content	Description
2	Socket State	
3	Connection State	
4	Open Count	Total number of times the server has performed an open operation.
5	Establish Count	Total number of times a connection has been established on the server.
6	Close Count	Total number of times the server has performed a close operation.
7	Read	Total number of packets received by the server.
8	Message	Total number of message receive by the server.
9	Write	Total number of packets sent from the server to the client.
10	Timeout	Total number of times the server as reached a connection timeout condition.
11	Host Port	Service port on client connected to the server.
12 to 13	Host IP Address	IP address of the client connected to the server.

Server 1 Status

Offset	Content	Description
14	Socket State	
15	Connection State	
16	Open Count	Total number of times the server has performed an open operation.
17	Establish Count	Total number of times a connection has been established on the server.
18	Close Count	Total number of times the server has performed a close operation.
19	Read	Total number of packets received by the server.
20	Message	Total number of message receive by the server.
21	Write	Total number of packets sent from the server to the client.
22	Timeout	Total number of times the server as reached a connection timeout condition.
23	Host Port	Service port on client connected to the server.
24 to 25	Host IP Address	IP address of the client connected to the server.

Server 2 Status

Offset	Content	Description
26	Socket State	
27	Connection State	
28	Open Count	Total number of times the server has performed an open operation.
29	Establish Count	Total number of times a connection has been established on the server.
30	Close Count	Total number of times the server has performed a close operation.
31	Read	Total number of packets received by the server.
32	Message	Total number of message receive by the server.
33	Write	Total number of packets sent from the server to the client.
34	Timeout	Total number of times the server as reached a connection timeout condition.
35	Host Port	Service port on client connected to the server.
36 to 37	Host IP Address	IP address of the client connected to the server.

Server 3 Status

Offset	Content	Description
38	Socket State	
39	Connection State	
40	Open Count	Total number of times the server has performed an open operation.
41	Establish Count	Total number of times a connection has been established on the server.
42	Close Count	Total number of times the server has performed a close operation.
43	Read	Total number of packets received by the server.
44	Message	Total number of message receive by the server.
45	Write	Total number of packets sent from the server to the client.
46	Timeout	Total number of times the server as reached a connection timeout condition.
47	Host Port	Service port on client connected to the server.
48 to 49	Host IP Address	IP address of the client connected to the server.

Server 4 Status

Offset	Content	Description
50	Socket State	
51	Connection State	
52	Open Count	Total number of times the server has performed an open operation.
53	Establish Count	Total number of times a connection has been established on the server.
54	Close Count	Total number of times the server has performed a close operation.
55	Read	Total number of packets received by the server.
56	Message	Total number of message receive by the server.
57	Write	Total number of packets sent from the server to the client.
58	Timeout	Total number of times the server as reached a connection timeout condition.
59	Host Port	Service port on client connected to the server.
60 to 61	Host IP Address	IP address of the client connected to the server.

5.5.3 BTR Response Block (252)

Offset	Content	Description
0	Block read ID	This word contains the value of 251 to indicate this specific status block
1	Block Write ID	This word contains the next write block to receive from the processor.

Server 5 Status

Offset	Content	Description
2	Socket State	
3	Connection State	
4	Open Count	Total number of times the server has performed an open operation.
5	Establish Count	Total number of times a connection has been established on the server.
6	Close Count	Total number of times the server has performed a close operation.
7	Read	Total number of packets received by the server.
8	Message	Total number of message receive by the server.
9	Write	Total number of packets sent from the server to the client.
10	Timeout	Total number of times the server as reached a connection timeout condition.
11	Host Port	Service port on client connected to the server.
12 to 13	Host IP Address	IP address of the client connected to the server.

Server 6 Status

Offset	Content	Description
14	Socket State	
15	Connection State	
16	Open Count	Total number of times the server has performed an open operation.
17	Establish Count	Total number of times a connection has been established on the server.
18	Close Count	Total number of times the server has performed a close operation.
19	Read	Total number of packets received by the server.
20	Message	Total number of message receive by the server.
21	Write	Total number of packets sent from the server to the client.
22	Timeout	Total number of times the server as reached a connection timeout condition.
23	Host Port	Service port on client connected to the server.
24 to 25	Host IP Address	IP address of the client connected to the server.

Server 7 Status

Offset	Content	Description
26	Socket State	
27	Connection State	
28	Open Count	Total number of times the server has performed an open operation.
29	Establish Count	Total number of times a connection has been established on the server.
30	Close Count	Total number of times the server has performed a close operation.
31	Read	Total number of packets received by the server.
32	Message	Total number of message receive by the server.
33	Write	Total number of packets sent from the server to the client.
34	Timeout	Total number of times the server as reached a connection timeout condition.
35	Host Port	Service port on client connected to the server.
36 to 37	Host IP Address	IP address of the client connected to the server.

Server 8 Status

Offset	Content	Description
38	Socket State	
39	Connection State	
40	Open Count	Total number of times the server has performed an open operation.
41	Establish Count	Total number of times a connection has been established on the server.
42	Close Count	Total number of times the server has performed a close operation.
43	Read	Total number of packets received by the server.
44	Message	Total number of message receive by the server.
45	Write	Total number of packets sent from the server to the client.
46	Timeout	Total number of times the server as reached a connection timeout condition.
47	Host Port	Service port on client connected to the server.
48 to 49	Host IP Address	IP address of the client connected to the server.

Server 9 Status

Offset	Content	Description
50	Socket State	
51	Connection State	
52	Open Count	Total number of times the server has performed an open operation.
53	Establish Count	Total number of times a connection has been established on the server.
54	Close Count	Total number of times the server has performed a close operation.
55	Read	Total number of packets received by the server.
56	Message	Total number of message receive by the server.
57	Write	Total number of packets sent from the server to the client.
58	Timeout	Total number of times the server as reached a connection timeout condition.
59	Host Port	Service port on client connected to the server.
60 to 61	Host IP Address	IP address of the client connected to the server.

5.5.4 BTR Response Block (253)

Offset	Content	Description
0	Block read ID	This word contains the value of 251 to indicate this specific status block
1	Block Write ID	This word contains the next write block to receive from the processor.

Server 10 Status

Offset	Content	Description
2	Socket State	
3	Connection State	
4	Open Count	Total number of times the server has performed an open operation.
5	Establish Count	Total number of times a connection has been established on the server.
6	Close Count	Total number of times the server has performed a close operation.
7	Read	Total number of packets received by the server.
8	Message	Total number of message receive by the server.
9	Write	Total number of packets sent from the server to the client.
10	Timeout	Total number of times the server as reached a connection timeout condition.
11	Host Port	Service port on client connected to the server.
12 to 13	Host IP Address	IP address of the client connected to the server.

Server 11 Status

Offset	Content	Description
14	Socket State	
15	Connection State	
16	Open Count	Total number of times the server has performed an open operation.
17	Establish Count	Total number of times a connection has been established on the server.
18	Close Count	Total number of times the server has performed a close operation.
19	Read	Total number of packets received by the server.
20	Message	Total number of message receive by the server.
21	Write	Total number of packets sent from the server to the client.
22	Timeout	Total number of times the server as reached a connection timeout condition.
23	Host Port	Service port on client connected to the server.
24 to 25	Host IP Address	IP address of the client connected to the server.

Server 12 Status

Offset	Content	Description
26	Socket State	
27	Connection State	
28	Open Count	Total number of times the server has performed an open operation.
29	Establish Count	Total number of times a connection has been established on the server.
30	Close Count	Total number of times the server has performed a close operation.
31	Read	Total number of packets received by the server.
32	Message	Total number of message receive by the server.
33	Write	Total number of packets sent from the server to the client.
34	Timeout	Total number of times the server as reached a connection timeout condition.
35	Host Port	Service port on client connected to the server.
36 to 37	Host IP Address	IP address of the client connected to the server.

Server 13 Status

Offset	Content	Description
38	Socket State	
39	Connection State	
40	Open Count	Total number of times the server has performed an open operation.
41	Establish Count	Total number of times a connection has been established on the server.
42	Close Count	Total number of times the server has performed a close operation.
43	Read	Total number of packets received by the server.
44	Message	Total number of message receive by the server.
45	Write	Total number of packets sent from the server to the client.
46	Timeout	Total number of times the server as reached a connection timeout condition.
47	Host Port	Service port on client connected to the server.
48 to 49	Host IP Address	IP address of the client connected to the server.

Server 14 Status

Offset	Content	Description
50	Socket State	
51	Connection State	
52	Open Count	Total number of times the server has performed an open operation.
53	Establish Count	Total number of times a connection has been established on the server.
54	Close Count	Total number of times the server has performed a close operation.
55	Read	Total number of packets received by the server.
56	Message	Total number of message receive by the server.
57	Write	Total number of packets sent from the server to the client.
58	Timeout	Total number of times the server as reached a connection timeout condition.
59	Host Port	Service port on client connected to the server.
60 to 61	Host IP Address	IP address of the client connected to the server.

5.5.5 BTR Response Block (254)

Offset	Content	Description
0	Block read ID	This word contains the value of 251 to indicate this specific status block
1	Block Write ID	This word contains the next write block to receive from the processor.

Server 15 Status

Offset	Content	Description
2	Socket State	
3	Connection State	
4	Open Count	Total number of times the server has performed an open operation.
5	Establish Count	Total number of times a connection has been established on the server.
6	Close Count	Total number of times the server has performed a close operation.
7	Read	Total number of packets received by the server.
8	Message	Total number of message receive by the server.
9	Write	Total number of packets sent from the server to the client.
10	Timeout	Total number of times the server as reached a connection timeout condition.
11	Host Port	Service port on client connected to the server.
12 to 13	Host IP Address	IP address of the client connected to the server.

Server 16 Status

Offset	Content	Description
14	Socket State	
15	Connection State	
16	Open Count	Total number of times the server has performed an open operation.
17	Establish Count	Total number of times a connection has been established on the server.
18	Close Count	Total number of times the server has performed a close operation.
19	Read	Total number of packets received by the server.
20	Message	Total number of message receive by the server.
21	Write	Total number of packets sent from the server to the client.
22	Timeout	Total number of times the server as reached a connection timeout condition.
23	Host Port	Service port on client connected to the server.
24 to 25	Host IP Address	IP address of the client connected to the server.

Server 17 Status

Offset	Content	Description
26	Socket State	
27	Connection State	
28	Open Count	Total number of times the server has performed an open operation.
29	Establish Count	Total number of times a connection has been established on the server.
30	Close Count	Total number of times the server has performed a close operation.
31	Read	Total number of packets received by the server.
32	Message	Total number of message receive by the server.
33	Write	Total number of packets sent from the server to the client.
34	Timeout	Total number of times the server as reached a connection timeout condition.
35	Host Port	Service port on client connected to the server.
36 to 37	Host IP Address	IP address of the client connected to the server.

Server 18 Status

Offset	Content	Description
38	Socket State	
39	Connection State	
40	Open Count	Total number of times the server has performed an open operation.
41	Establish Count	Total number of times a connection has been established on the server.
42	Close Count	Total number of times the server has performed a close operation.
43	Read	Total number of packets received by the server.
44	Message	Total number of message receive by the server.
45	Write	Total number of packets sent from the server to the client.
46	Timeout	Total number of times the server as reached a connection timeout condition.
47	Host Port	Service port on client connected to the server.
48 to 49	Host IP Address	IP address of the client connected to the server.

Server 19 Status

Offset	Content	Description
50	Socket State	
51	Connection State	
52	Open Count	Total number of times the server has performed an open operation.
53	Establish Count	Total number of times a connection has been established on the server.
54	Close Count	Total number of times the server has performed a close operation.
55	Read	Total number of packets received by the server.
56	Message	Total number of message receive by the server.
57	Write	Total number of packets sent from the server to the client.
58	Timeout	Total number of times the server as reached a connection timeout condition.
59	Host Port	Service port on client connected to the server.
60 to 61	Host IP Address	IP address of the client connected to the server.

5.5.6 Client Configuration Error Word

The format of the client configuration error word is as follows:

Bit	Description	Value
0		0x0001
1		0x0002
2		0x0004
3		0x0008
4	Invalid retry count (0 to 10)	0x0010
5		0x0020
6		0x0040
7		0x0080
8		0x0100
9		0x0200
10		0x0400
11		0x0800
12		0x1000
13		0x2000
14		0x4000
15		0x8000

5.5.7 Pass-Through Port Configuration Error Word

The format of the pass-through port configuration error word is as follows:

Bit	Description	Value
0	Invalid enable code	0x0001
1	Invalid local station ID	0x0002
2	Invalid protocol or termination type	0x0004
3	Invalid baud rate	0x0008
4	Invalid parity	0x0010
5	Invalid data bits	0x0020
6	Invalid stop bits	0x0040
7		0x0080
8	Invalid Use CTS Line selection	0x0100
9	Invalid retry count	0x0200
10		0x0400
11		0x0800
12		0x1000
13		0x2000
14		0x4000
15		0x8000

5.5.8 Pass-Through Server Configuration Error Word

The format of the pass-through server configuration error word is as follows:

Bit	Description	Value
0	Invalid enable code	0x0001
1	Invalid busytimeout setting (< 100 milliseconds)	0x0002
2		0x0004
3		0x0008
4		0x0010
5	Invalid baud rate	0x0020
6	Invalid parity	0x0040
7	Invalid data bits	0x0080
8	Invalid stop bits	0x0100
9		0x0200
10		0x0400
11		0x0800
12		0x1000
13		0x2000
14		0x4000
15		0x8000

5.5.9 Pass-Through Server State Parameter

The values for the pass-through server state parameter have the following definition:

State	Definition
-1	Listen function called to set up server
0	Waiting for connection to be established on server
1	Waiting and processing data received
2	Waiting for pass-through port to be free
3	Pass-through server owns port and processes all data received and written.
100	Closing server on program termination
1000	Initiate a close on the server
1001	Waiting for server to successfully close

The connection state for the pass-through server is not used at this time.

5.5.10 Socket State Parameter

The following table defines the values represented in the Socket State parameter:

State	Definition
-1	Listen function called to set up server
0	Waiting for connection to be established on server
1	Handle first request and build response for first part of multi-request message.
2	Handle more requests in packet received.
100	Closing server on program termination
1000	Initiate a close on the server
1001	Waiting for server to successfully close

5.5.11 Connection State Parameter

The following table defines the values represented in the Connection State parameter:

State	Definition
0	No session exists on socket
1	Session established but no connections
2	Session established and connected

5.6 Error Codes

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

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

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

5.6.1 Local STS Error Codes

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

5.6.2 Remote STS Error Codes

Code (Int)	Code (Hex)	Description
0	0x0000	Success, no error
4096	0x1000	Illegal command or format
8192	0x2000	Host has a problem and will not communicate
12288	0x3000	Remote node host is missing, disconnected or shut down
16384	0x4000	Host could not complete function due to hardware fault
20480	0x5000	Addressing problem or memory protect rungs
24576	0x6000	Function not allowed due to command protection selection
26872	0x7000	Processor is in Program mode
-32768	0x8000	Compatibility mode file missing or communication zone problem
-28672	0x9000	Remote node cannot buffer command
-24576	0xA000	Wait ACK (1775-KA buffer full)
-20480	0xB000	Remote node problem due to download
-16384	0xC000	Wait ACK (1775-KA buffer full)
-12288	0xD000	Not used
-8192	0xE000	Not used
	0xF0nn	Error code in the EXT STS byte (nn contains EXT error code)

5.6.3 Errors When EXT STS Is Present

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

5.6.4 Module Specific Error (not DFNT Compliant)

Code (Int)	Code (Hex)	Description
-1	0xFFFF	CTS modem control line not set before transmit
-2	0xFFFE	Timeout while transmitting message
-10	0xFFF6	Timeout waiting for DLE-ACK after request
-11	0xFFF5	Timeout waiting for response after request
-12	0xFFF4	Reply data does not match requested byte count
-20	0xFFEC	DLE-NAK received after request
-21	0xFFEB	DLE-NAK sent after response
-200	0xFF38	DLE-NAK received after request

5.7 TCP/IP Interface Errors

5.7.1 Timeout Errors

Error (Int)	Error (Hex)	Description
-33	0xFFDF	Failed to connect to target
-34	0xFFDE	Failed to register session with target (timeout)
-35	0xFFDD	Failed forward open response timeout
-36	0xFFDC	PCCC command response timeout
-37	0xFFDB	No TCP/IP connection error
-47	0xFFD1	ARP could not resolve MAC from IP (bad IP address, not part of a network, invalid parameter to ARP routine).
-48	0xFFD0	Error during ARP operation: the response to the ARP request did not arrive to the module after a 5 second timeout.

Note: When the client gets error -47 or -48, it uses the adjustable ARP Timeout parameter in the configuration file to set an amount of time to wait before trying again to connect to this non-existent server. This feature allows the client to continue sending commands and polling other existing servers, while waiting for the non-existent server to appear on the network.

5.7.2 Register Session Response Errors

Error (Int)	Error (Hex)	Description
-49	0xFFCF	Invalid response length
-50	0xFFCE	Command field invalid
-51	0xFFCD	Invalid length field parameter
-52	0xFFCC	Status error reported
-53	0xFFCB	Context field not matched
-54	0xFFCA	Invalid version

5.7.3 Forward Open Response Errors

Error (Int)	Error (Hex)	Description
-65	0xFFBF	Message Length received not valid
-66	0xFFBE	Command code returned not valid
-67	0xFFBD	Session handle field invalid
-68	0xFFBC	Status error reported
-69	0xFFBB	Context field not matched
-70	0xFFBA	CPF item count not correct
-71	0xFFB9	CPF address field error
-72	0xFFB8	CPF packet tag invalid
-73	0xFFB7	CPF bad command code
-74	0xFFB6	CPF invalid IOI
-75	0xFFB5	CPF status error reported

5.7.4 PCCC Response Errors

Error (Int)	Error (Hex)	Description
-81	0xFFAF	Message Length received not valid
-82	0xFFAE	Command code returned not valid
-83	0xFFAD	Session handle field invalid
-84	0xFFAC	Status error reported
-85	0xFFAB	Context field not matched
-86	0xFFAA	CPF item count not correct
-87	0xFFA9	CPF address field error
-88	0xFFA8	CPF packet tag invalid
-89	0xFFA7	CPF bad command code
-90	0xFFA6	CPF invalid IOI
-91	0xFFA5	CPF status error reported
-92	0xFFA4	
-93	0xFFA3	TSN in PCCC message not matched
-94	0xFFA2	CPF not correct message number
-95	0xFFA1	CPF incorrect connection ID value returned
-96	0xFFA0	Incorrect session handle returned

5.8 Configuration Data

This section contains a listing of the parameters and their definitions for the MVI71-DFNT module configuration.

[Section]/Item	Value	Range	Description
[MODULE]			Configuration header for general module information
Module Name:		Up to 80 chars	Name of the module for use on reports. Use this parameter to identify your module in your system.
Read Register Start:		0 to 3999	This parameter specifies the starting register in the module where data will be transferred from the module to the processor. Valid range for this parameter is 0 to 3999.
Read Register Count:		0 to 3960	This parameter specifies the number of registers to be transferred from the module to the processor. Valid entry for this parameter is 0 to 3960.
Write Register Start:		0 to 3999	This parameter specifies the starting register in the module where the data transferred from the processor will be placed. Valid range for this parameter is 0 to 3999.
Write Register Count:		0 to 3960	This parameter specifies the number of registers to transfer from the processor to the module. Valid entry for this parameter is 0 to 3960.
Failure Flag 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.
Initialize Output Data:		0 or 1	This parameter determines if the output data for the module should be initialized with values from the processor. If the value is set to 0, the output data will be initialized to 0. If the value is set to 1, the data will be initialized with data from the processor. Use of this option requires associated ladder logic to pass the data from the processor to the module.
DFNT Server File Size		100 or 1000	Sets the maximum file size (100 or 1000) for the servers

[Section]/Item	Value	Range	Description
[DF1 Pass-Through Server Port 1]			Start header for the pass-through server
Enabled:		Y or N	This parameter determines if the pass-through server will be utilized.
Server Port Number:		1 to 65535	Service port number to be associated with this server. The number assigned must match that used by the client software to establish the connection.
Busy Timeout:		100 to 65535 milliseconds	This parameter sets the number of milliseconds the server will wait for the pass-through port to become available. Valid data range for this parameter is 100 to 65535.

[Section]/Item	Value	Range	Description
Baud Rate:		1200 to 19200	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.
Parity:		N, O, E	This is the Parity code to be used for the port. The coded values are as follows: N=None, O=Odd, E=Even.
Data Bits:		5 to 8	This parameter sets the number of data bits for each word used by the protocol.
Stop Bits:		1 or 2	This parameter sets the number of stop bits for each data value sent.

[Section]/Item	Value	Range	Description
[DF1 Pass-Through Port]			Start header for the pass-through port
Enabled:		Y or N	This parameter specifies if the pass-through port will be utilized. This port should only be used if the DF1 pass-through server is enabled.
Local Station ID:		0 to 254	This parameter specifies the local station ID for all DF1 messages sent to this port. A value of 255 is not permitted as this is the broadcast address. The application will only accept messages with this node address.
Protocol:		F or H	F=full duplex, H=half-duplex. The value selected should match that set for the PLC processor.
Termination Type:		B or C	This parameter sets the termination message type for the DF1 protocol. The value selected should match that used in the PLC. Valid values are B for BCC and C for CRC.
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.
Parity:		N, O, E, M or S	This is the Parity code to be used for the port. The coded values are as follows: N=None, O=Odd, E=Even, M=Mark and S=Space.
Data Bits:		5 to 8	This parameter sets the number of data bits for each word used by the protocol.
Stop Bits:		1 or 2	This parameter sets the number of stop bits for each data value sent.
RTS On:		0 to 65535	This parameter sets the number of milliseconds to delay after RTS is asserted before the data will be transmitted.
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.
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).

[Section]/Item	Value	Range	Description
Retry Count:		0 to 10	This parameter specifies the number of attempts for each response message. If a message fails, it will be retried up to the count specified.
Request Timeout:		0 to 65535	This parameter specifies the number of milliseconds to wait for a complete request message. The timer is started after the DLE-STX character sequence is received for the full-duplex protocol or the DLE-SOH sequence for the half-duplex protocol. If the timer expires, the current request message will be aborted.
Busy Timeout:		0 to 65535	This parameter specifies the number of milliseconds to wait for the pass-through port to become available.
ACK Timeout:		0 to 65535	This parameter specifies the number of milliseconds to wait for a DLE-ACK character sequence after a response is issued.

[Section]/Item	Value	Range	Description
[DFNT Client 0]			Start header for Client 0
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.
Response Timeout:		0 to 65535	This parameter represents the message response timeout period in 1-ms increments. This is the time that a client 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.
Retry Count:		0 to 10	This parameter specifies the number of times a command will be retried if it fails.

5.9 DFNT Command Entry Form

The following form can be used to design the application's command list:

Module Information Data						Device Information Data						
Column #	1	2	3	4	5	6	7	8	9	10	11	12
Function Code	Enable Code	Internal Address	Poll Interval Time	Count	Swap Code	IP Address	Slot Number	Function Code	Function Parameters			

IP Address = IP address of processor to reach
Slot Number = -1 for PLC5 & SLC, processor slot number of ControlLogix

5.10 Command Function Codes

In the following discussion, the Column values are described for the serial DFCM Command list (See Node Address parameter #6). In the DFNT module, the Node address has been replaced with two parameters; the IP Address and the Slot Number, causing the Function Code and Function Parameters to occupy positions 8 to 12 instead of 7 to 11. Aside from this difference, all other information is correct.

DFNT Configuration Form

Module Information Data						Device Information Data					
1	2	3	4	5	6	7	8	9	10	11	12
Enable Code	Internal Address	Poll Interval Time	Count	Swap Code	IP Address	Slot Number	Function Code	Function Parameters			

DFNT Command Structure

Module Information Data							Device Information Data					
Column #	1	2	3	4	5	6	7	8	9	10	11	12
Function Code	Enable Code	Internal Address	Poll Interval Time	Count	Swap Code	IP Address	Slot Number	Function Code	Function Parameters			

5.11 General Command Structure

	DF1 Master Port Command Structure (File for each Master Port)	Description
Communication Module Information	Enable/Type Word	0=Disabled, 1=Continuous, 2=Conditional and 999=Poll. The conditional type only applies to the write functions.
	Virtual Database Address	This parameter defines the virtual database register to be associated with the command.
	Poll Interval	Minimum time in tenths of a second to wait before polling with this command.
	Count	Number of data values or registers to be considered by the function.
	Swap Type Code	Swap type code for command: 0=None, 1=Swap words, 2=Swap words & bytes and 3=swap bytes in each word.
Device Information	Node Address	Node address of unit to be reached on the data highway.
	Function Code	This parameter defines the module function code to be associated with the command. Each function code requires a set of parameters to construct the DF1 message.
	Parameters	Up to four parameter fields can follow the function code field to define the element or data register to be considered by the function.

Note that the Node Address field in the serial implementation has been expanded to two fields: IP Address and Slot Number in the DFNT Ethernet implementation. This is required for network support. The IP Address specifies the IP address of the device to reach on the network. The Slot Number has specific meaning determined by the processor. For ControlLogix processors, the Slot Number is the location in the ControlLogix rack of the processor. For a four-slot rack, this parameter would have a valid range of 0 to 3. For the PLC5 and SLC family of processors, the Slot Number parameter is always set to -1. These processors do not have a slot number in the path field as the Ethernet interface is resident on the processor.

The following tables define the parameters required for each function.

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

Column	Parameter	Description	Parameter
1	Enable/Type Word	0=Disabled, 1=Continuous and 2=Conditional.	
2	Virtual Database Address	This parameter defines the database address of the first data point to be associated with the command.	
3	Poll Interval	Minimum time in tenths of a second to wait before polling with this command.	
4	Count	Number of data word values to be considered by the function.	
5	Swap Type Code	Swap type code for command: 0=None, 1=Swap words, 2=Swap words & bytes and 3=swap bytes in each word.	
6	Node Address	Address of unit to reach on the data highway.	
7	Slot Number	Processor slot number in Control/CompactLogix rack. Use -1 for PLC5 & SLC processors.	
8	Function Code = 1	Protected Write Function	
9	Word Address	Word address where to start the write operation.	P1
10 to 12	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.

5.11.2 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	Slot Number	Processor slot number in Control/CompactLogix rack. Use -1 for PLC5 & SLC processors.	
8	Function Code = 2	Unprotected Read Function	
9	Word Address	Word address where to start the read operation.	P1
10 to 12	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.

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

Column	Parameter	Description	Parameter
1	Enable/Type Word	0=Disabled, 1=Continuous and 2=Conditional.	
	Virtual Database Address	This parameter defines the database address for the data to be associated with the command. The address defined represents a register address and not a bit address. This function will update one or more words of data as defined by the count parameter.	
3	Poll Interval	Minimum time in tenths of a second to wait before polling with this command.	
4	Count	Number of data word values to be considered by the function.	
5	Swap Type Code	Swap type code for command: Always zero (0).	
6	Node Address	Address of unit to reach on the data highway.	
7	Slot Number	Processor slot number in Control/CompactLogix rack. Use -1 for PLC5 & SLC processors.	
8	Function Code = 3	Protected Bit Write Function	
9	Word Address	Word address where to start the write operation.	P1
10 to 12	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.

5.11.4 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	Slot Number	Processor slot number in Control/CompactLogix rack. Use -1 for PLC5 & SLC processors.	
8	Function Code = 4	Unprotected Bit Write Function	
9	Word Address	Word address where to start the write operation.	P1
10 to 12	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.

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

Column	Parameter	Description	Parameter
1	Enable/Type Word	0=Disabled, 1=Continuous and 2=Conditional.	
2	Virtual Database Address	This parameter defines the database address of the first data point to be associated with the command.	
3	Poll Interval	Minimum time in tenths of a second to wait before polling with this command.	
4	Count	Number of data word values to be considered by the function.	
5	Swap Type Code	Swap type code for command: 0=None, 1=Swap words, 2=Swap words & bytes and 3=swap bytes in each word.	
6	Node Address	Address of unit to reach on the data highway.	
7	Slot Number	Processor slot number in Control/CompactLogix rack. Use -1 for PLC5 & SLC processors.	
8	Function Code = 5	Unprotected Write Function	
9	Word Address	Word address where to start the write operation.	P1
10 to 12	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.

5.11.6 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	Slot Number	Processor slot number in Control/CompactLogix rack. Use -1 for PLC5 & SLC processors.	
8	Function Code = 100	Word Range Write Command.	
9	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
10	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
11	Sub-Element Number	This parameter defines the sub-element for 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
12	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.

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

Column	Parameter	Description	Parameter
1	Enable/Type Word	0=Disabled and 1=Continuous.	
2	Virtual Database Address	This parameter defines the database address of the first data point to be associated with the command.	
3	Poll Interval	Minimum time in tenths of a second to wait before polling with this command.	
4	Count	Number of data word values to be considered by the function.	
5	Swap Type Code	Swap type code for command: 0=None, 1=Swap words, 2=Swap words & bytes and 3=swap bytes in each word.	
6	Node Address	Address of unit to reach on the data highway.	
7	Slot Number	Processor slot number in Control/CompactLogix rack. Use -1 for PLC5 & SLC processors.	
8	Function Code = 101	Word Range Write Command.	
9	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
10	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
11	Sub-Element Number	This parameter defines the sub-element for 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
12	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.

5.11.8 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	Slot Number	Processor slot number in Control/CompactLogix rack. Use -1 for PLC5 & SLC processors.	
8	Function Code = 102	Read-Modify-Write Command.	
9	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
10	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
11	Sub-Element Number	This parameter defines the sub-element for 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
12	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.

5.11.9 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	Slot Number	Processor slot number in Control/CompactLogix rack. Use -1 for PLC5 & SLC processors.	
8	Function Code = 150	Word Range Write Command.	
9	File String	PLC-5 address as specified as an ASCII string. For example, N10:300.	P1
10 to 12	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.

5.11.10 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	Slot Number	Processor slot number in Control/CompactLogix rack. Use -1 for PLC5 & SLC processors.	
8	Function Code = 151	Word Range Read Command.	
9	File String	PLC-5 address as specified as an ASCII string. For example, N10:300.	P1
10 to 12	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.

5.11.11 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	Slot Number	Processor slot number in Control/CompactLogix rack. Use -1 for PLC5 & SLC processors.	
8	Function Code = 152	Read-Modify-Write Command.	
9	File String	PLC-5 address as specified as an ASCII string. For example, N10:300.	P1
10 to 12	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.

5.11.12 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	Slot Number	Processor slot number in Control/CompactLogix rack. Use -1 for PLC5 & SLC processors.	
8	Function Code = 501	Logical Read Command	
9	File Type	SLC file type letter as used in file name string. Valid values for the system are N, S, F, A, ….	P1
10	File Number	SLC file number to be associated with the command.	P2
11	Element Number	The parameter defines the element in the file where write operation will start.	P3
12	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.

5.11.13 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	Slot Number	Processor slot number in Control/CompactLogix rack. Use -1 for PLC5 & SLC processors.	
8	Function Code = 502	Logical Read Command	
9	File Type	SLC file type letter as used in file name string. Valid values for the system are N, S, F, A, ….	P1
10	File Number	SLC file number to be associated with the command.	P2
11	Element Number	The parameter defines the element in the file where write operation will start.	P3
12	Sub-Element Number	This parameter defines the sub-element for 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.

5.11.14 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	Slot Number	Processor slot number in Control/CompactLogix rack. Use -1 for PLC5 & SLC processors.	
8	Function Code = 509	Logical Write Command	
9	File Type	SLC file type letter as used in file name string. Valid values for the system are N, S, F, A, ….	P1
10	File Number	SLC file number to be associated with the command.	P2
11	Element Number	The parameter defines the element in the file where write operation will start.	P3
12	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.

5.11.15 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	Slot Number	Processor slot number in Control/CompactLogix rack. Use -1 for PLC5 & SLC processors.	
8	Function Code = 510	Logical Write Command	
9	File Type	SLC file type letter as used in file name string. Valid values for the system are N, S, F, A, ….	P1
10	File Number	SLC file number to be associated with the command.	P2
11	Element Number	The parameter defines the element in the file where write operation will start.	P3
12	Sub-Element Number	This parameter defines the sub-element for 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.

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

Column	Parameter	Description	Parameter
1	Enable/Type Word	0=Disabled, 1=Continuous and 2=Conditional.	
2	Virtual Database Address	This parameter defines the database address of the data to be associated with the command. The first word of data contains the bit mask and the second word contains the data.	
3	Poll Interval	Minimum time in tenths of a second to wait before polling with this command.	
4	Count	Number of data word values to be considered by the function.	
5	Swap Type Code	Swap type code for command: Always zero (0).	
6	Node Address	Address of unit to reach on the data highway.	
7	Slot Number	Processor slot number in Control/CompactLogix rack. Use -1 for PLC5 & SLC processors.	
8	Function Code = 511	Logical Write with mask	
9	File Type	SLC file type letter as used in file name string. Valid values for the system are N, S, F, A, ….	P1
10	File Number	SLC file number to be associated with the command.	P2
11	Element Number	The parameter defines the element in the file where write operation will start.	P3
12	Sub-Element Number	This parameter defines the sub-element for 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.

5.12 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.

5.12.1 PLC-5 Sub-Element Codes

Timer / Counter

Code	Description
0	Control
1	Preset
2	Accumulated

Control

Code	Description
0	Control
1	Length
2	Position

*PD**

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

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

BT

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

MG

Code	Description
0	Control
1	Error
2	RLEN
3	DLEN

5.13 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:

5.13.1 SLC File Types

File Type	Description
S	Status
B	Bit
T	Timer
C	Counter
R	Control
N	Integer
F	Floating-point
Z	String
A	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.

5.14 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:

5.14.1 SLC File Types

File Type	Description
S	Status
B	Bit
T	Timer
C	Counter
R	Control
N	Integer
F	Floating-point
Z	String
A	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.

5.15 ControlLogix Processor Specifics

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

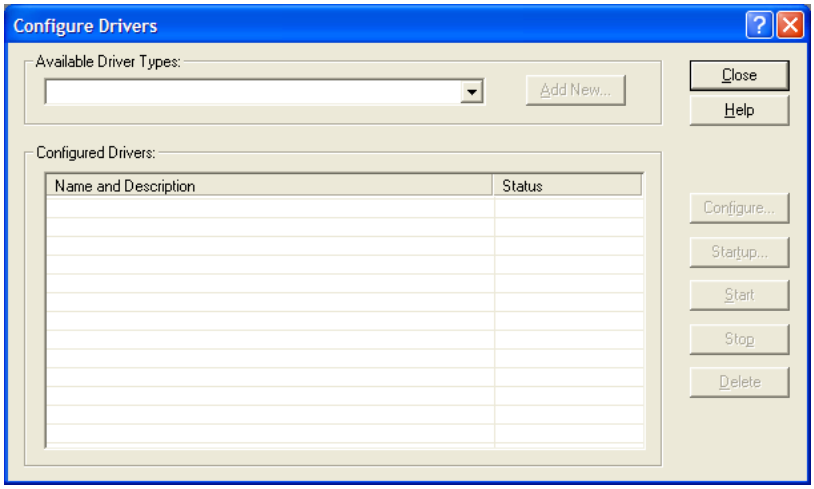
5.16 Server Driver

The Server Driver allows the MVI71-DFNT module to respond to data read and write commands issued by clients on the Ethernet/IP network using explicit messaging.

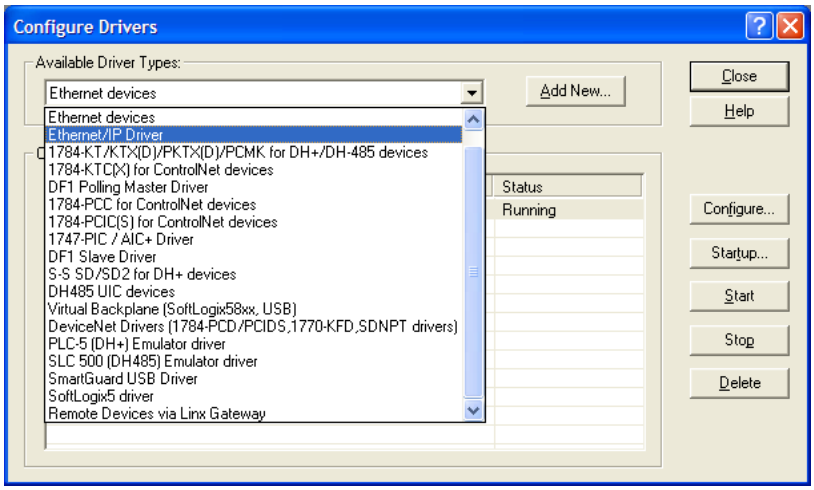
5.16.1 RSLinx Software

RSLinx is used by many personal computer-based applications to interface with Rockwell Automation products. For example, RSView requires the use of RSLinx for communication to remote nodes on a network. The following procedure is recommended for accessing the DFNT database using RSLinx version 2.54, or higher.

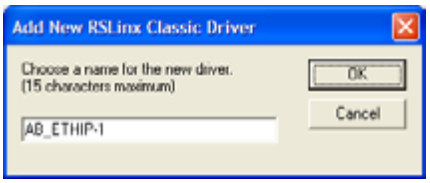
1 Start RSLinx software.
2 Open the **COMMUNICATIONS** menu, and then select **CONFIGURE DRIVERS**. This action opens the **CONFIGURE DRIVERS** dialog box.



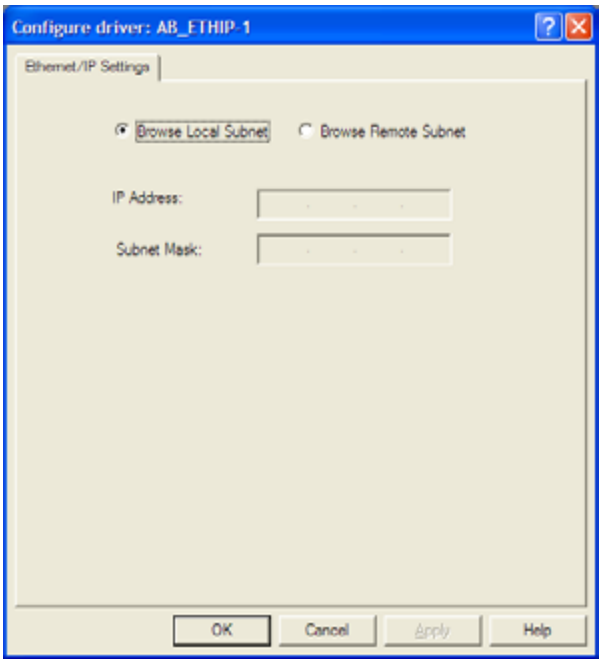
3 Click the arrow to the right of the Available Driver Types field, and then select **ETHERNET/IP DRIVER** from the dropdown list.



4 Click the **ADD/NEW** button, and then click **OK** in the **ADD NEW RSLINX DRIVER** dialog box.

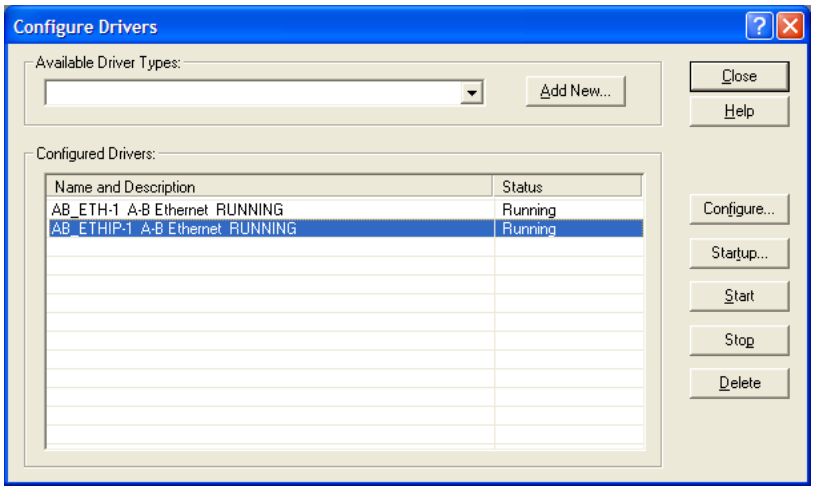


5 In the **CONFIGURE DRIVER** dialog box, make sure the Browse Local Subnet item is selected.



6 Make sure the **BROWSE LOCAL SUBNET** item is selected. RSLinx software will browse your local subnet and retrieve the IP address.

7 Click **OK** to save your settings and dismiss the **CONFIGURE DRIVER** dialog box. The new driver will appear in the list of configured drivers.

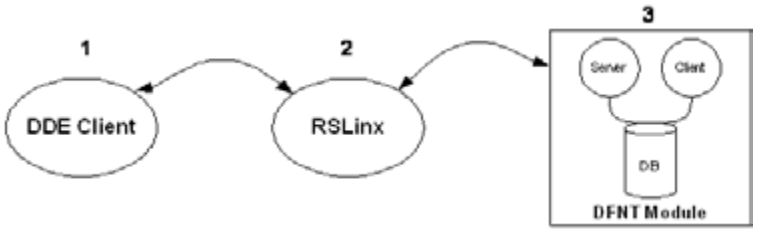


8 Close RSLinx software.

DDE Connection

This section discusses setting up a DDE connection to the module in order to transfer data between the module and a DDE compliant application.

Each DDE connection requires three basic elements: Application, Topic and Item. These three properties of a DDE link define the program providing the connection, the topic to connect to and the item in the topic that you wish to interface. All three are required for a connection. For this discussion, the Application will always be RSLinx. The Topic is defined in the RSLinx OEM release software and the Item is defined in the DDE client application where the data is required. The link between the DDE server and the DDE client can be established after these parameters are defined. The following illustration shows the relationship of these elements and the facilities used in the DDE link.



1 The DDE Client application specifies the DDE link by specifying the APPLICATION, TOPIC, and ITEM elements. For example, in Excel, enter =RSLINX|CIP_Sample! 'ReadData[0]' into a cell.

2 RSLinx is the DDE Server: APPLICATION = RSLINX.

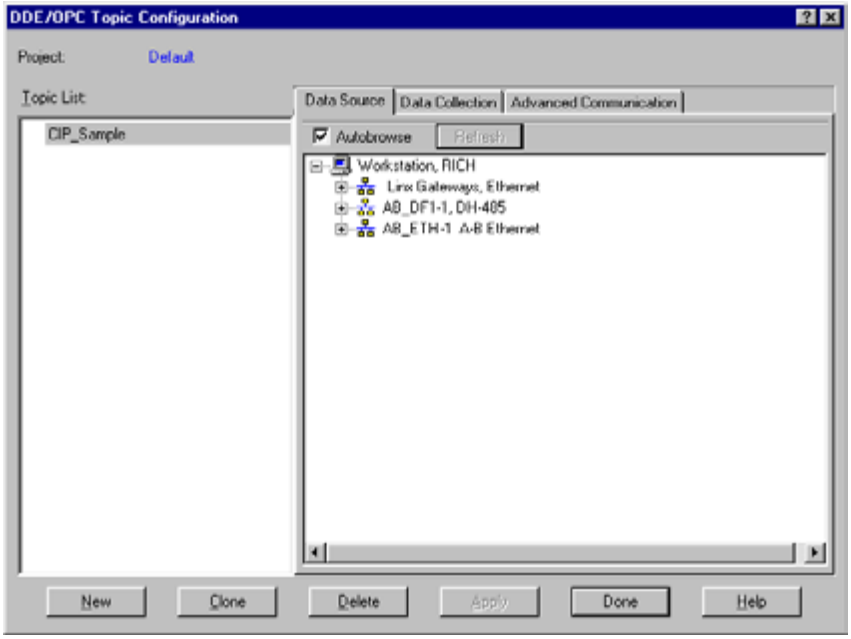
And

RSLinx defines the TOPIC.

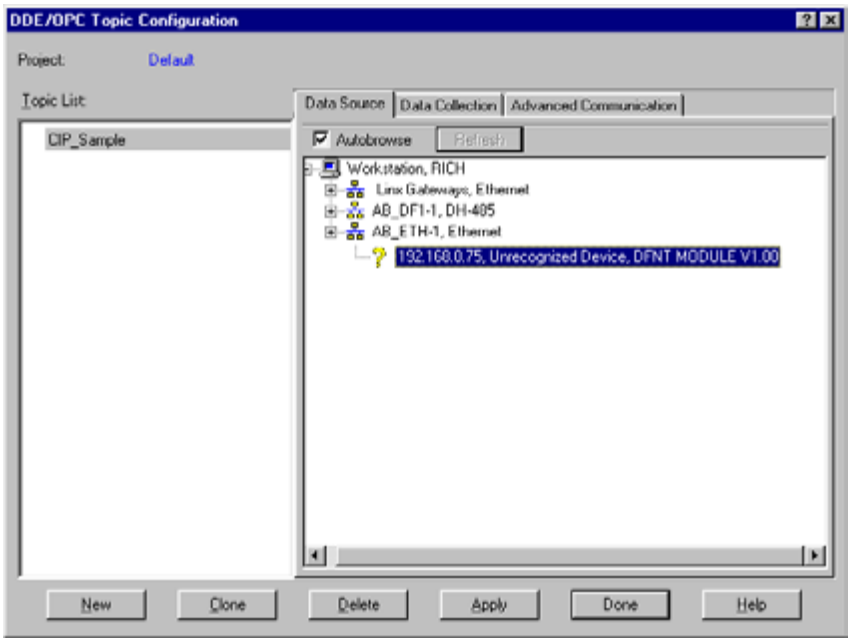
This is the communication link to the DFNT module. In this example, a DDE TOPIC is called "CIP_Sample" for the communication link to the DFNT module.

3 The DFNT module is used to serve data to RSLinx using the EtherNet/IP driver with explicit messaging. Database is accessed using tag names (e.g. ReadData[0]).

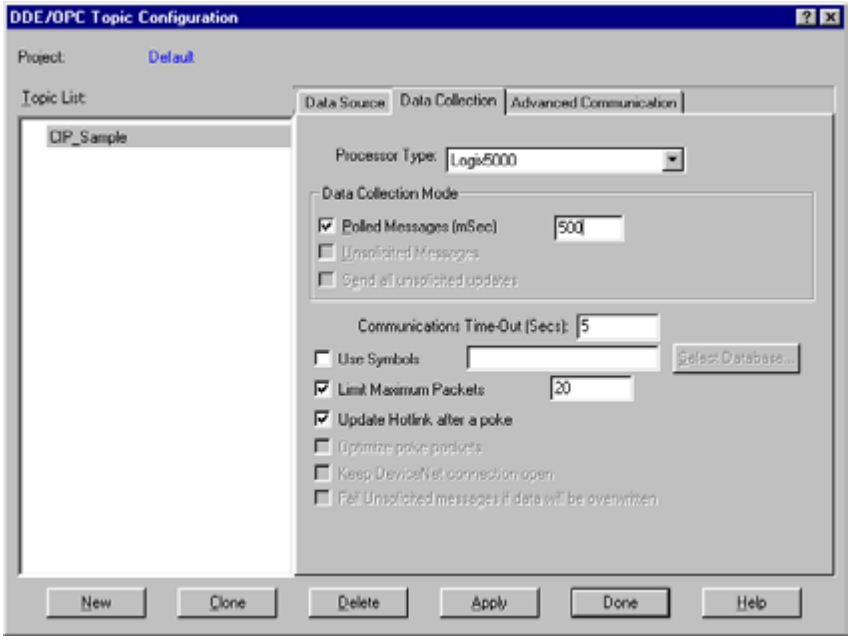
As discussed in the previous section, RSLinx must be used to define the Topic element for the DDE link definition. This is accomplished using the following procedure. It is assumed that the module can be seen in the RSLinx software. Refer to the RSLinx section of this manual to set up this connection. To define a new Topic, select the **TOPIC CONFIGURATION** option on the **DDE/OPC** menu from the RSLinx Main Menu. This causes the following dialog box to appear:



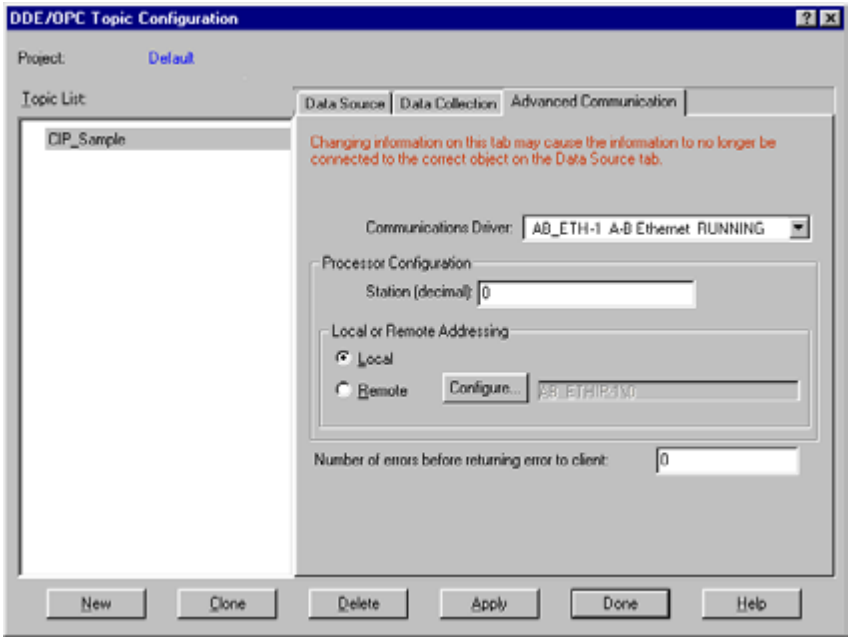
Click the **NEW** button to add a new Topic to the **TOPIC LIST**. This displays a default name. Edit the Topic name for the link to be formed. This name should reflect the unit or location to which the connection is being made. Do not press the Enter key. Instead, double-click the mouse on the DFNT device you want to connect to the entered Topic name. The dialog should now appear as follows:



Now select the **DATA COLLECTION** tab on the dialog box. Fill in the form to define the characteristics of the DDE link. The following screen shows an example:



You must set the **PROCESSOR TYPE** to Logix5000. Refer to the RSLinx on-line help for a discussion of each of the parameters on the form. Next select the **ADVANCED COMMUNICATION** tab on the dialog box. The following is displayed after selecting the tab.



The **COMMUNICATION DRIVER** should be set to the ETHIP driver. Make sure the **LOCAL** option is selected in the **LOCAL OR REMOTE ADDRESSING** section of the dialog box. You should not have to alter any data on this tab, as RSLinx knows the communication path. Now click the **APPLY** button to implement the options and to establish the topic.

You are now ready to use the DDE link in a DDE client application. The following example shows how to define a DDE link in an Excel Spreadsheet. It is also possible to define a DDE link in a Visual Basic program. Any other Windows DDE client application could be used including SoftLogix and RSSql.

Defining a DDE link in Excel

This is the simplest DDE link to define and should be used to make sure the Topic is defined correctly before using more advanced applications. Before attempting to make the link, verify that RSLinx is running and that the DFNT module is seen. To make a DDE link in Excel, enter the application, topic and item elements as a formula into a cell. The format for the formula is as follows:

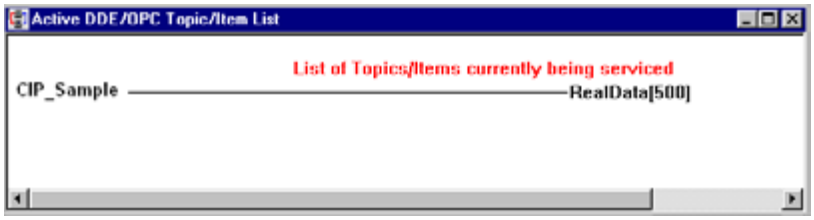
```
=APPLICATION|TOPIC!ITEM
```

The "**|**" character (piping symbol) separates the application and topic fields and the "**!**" (exclamation symbol) separates the topic and item fields. For our example topic of ProLinx1, the entry into the cell is:

```
=RSLinx|CIP_Sample!'RealData[500]'
```

This causes the current value at the database double-word offset 500 (starting at word address 1000) in the DFNT module to be displayed in the cell as a floating-point value. This value updates at the frequency defined in the Topic configuration in RSLinx. You can now place any database point in the DFNT module using the same procedure in your work sheet. Note that the tag array name is used for the item property and must be enclosed within the quote marks. This is because the tag array name item reference looks like an Excel work sheet reference. If you do not include the quotes, a formula error occurs. You can select any of the defined tag array names defined in the module on your spreadsheet.

When the items are used for the topic, RSLinx displays the following after selecting the Active Topic/Items command on the DDE/OPC menu option:

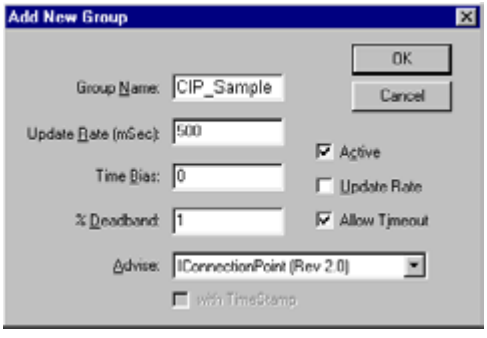


Any DDE compliant program can be used in the same manner. For maximum utility Visual Basic applications can be used to interface with module's database using DDE connectivity.

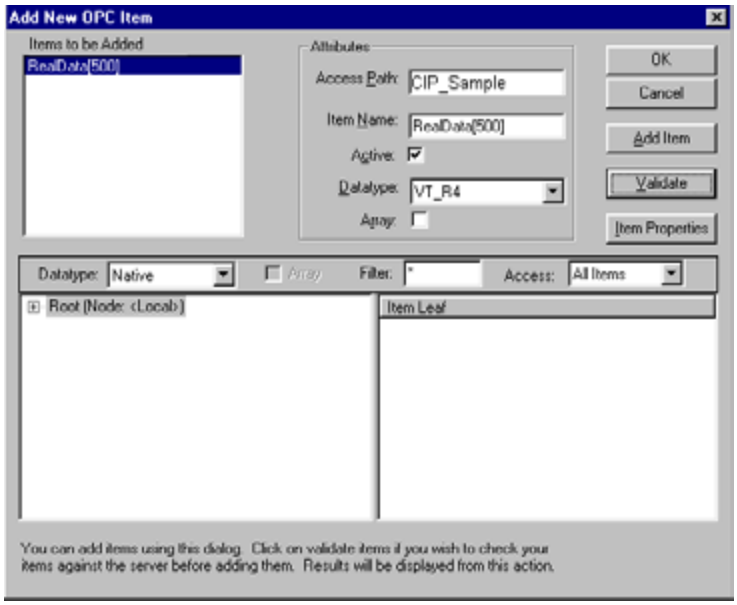
OPC Connection

This section discusses setting up an OPC connection to the module in order to transfer data between the module and an OPC compliant client. Follow the instructions for setting up the DDE connection outlined in the previous section. This will define the connection required by the OPC server. RSLinx will now be configured to interface with an OPC client application.

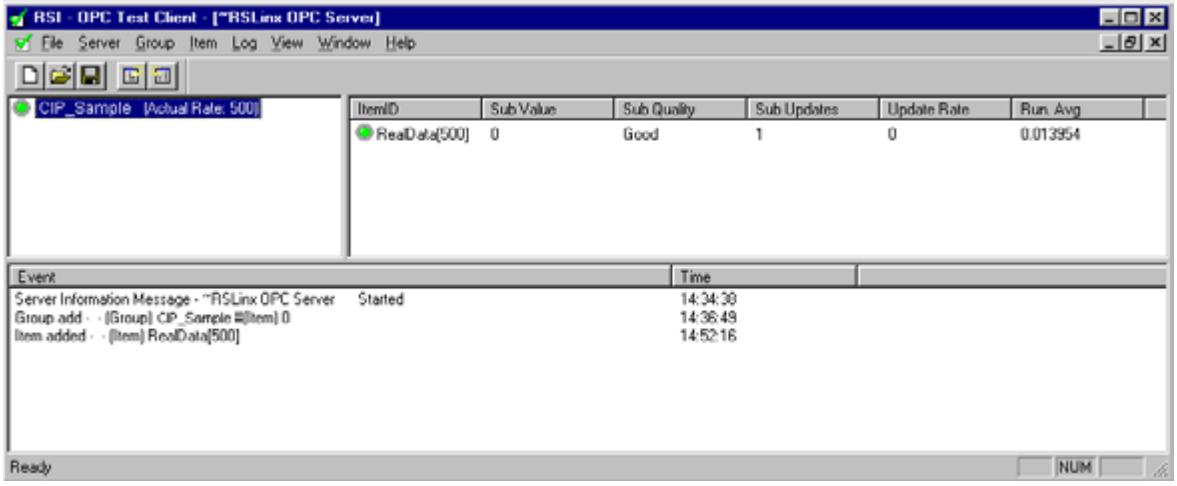
The example used in this section uses the OPC test client (opctest.exe) available from Rockwell Automation. First start the application and select the Connect… option from the Server Menu. In the dialog box shown, select the RSLinx OPC server. Next add a group using the Group menu option and fill in the Group Name using any name that is meaningful for the points to be monitored and controlled. The following dialog displays an example:



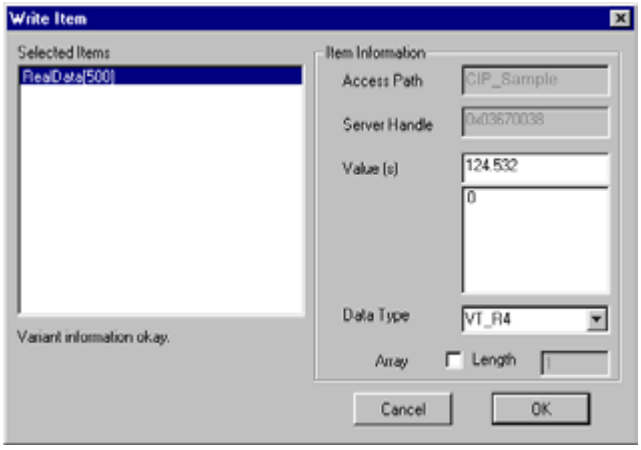
Next, add items to the client. The following shows an example dialog used to add a floating-point data item:



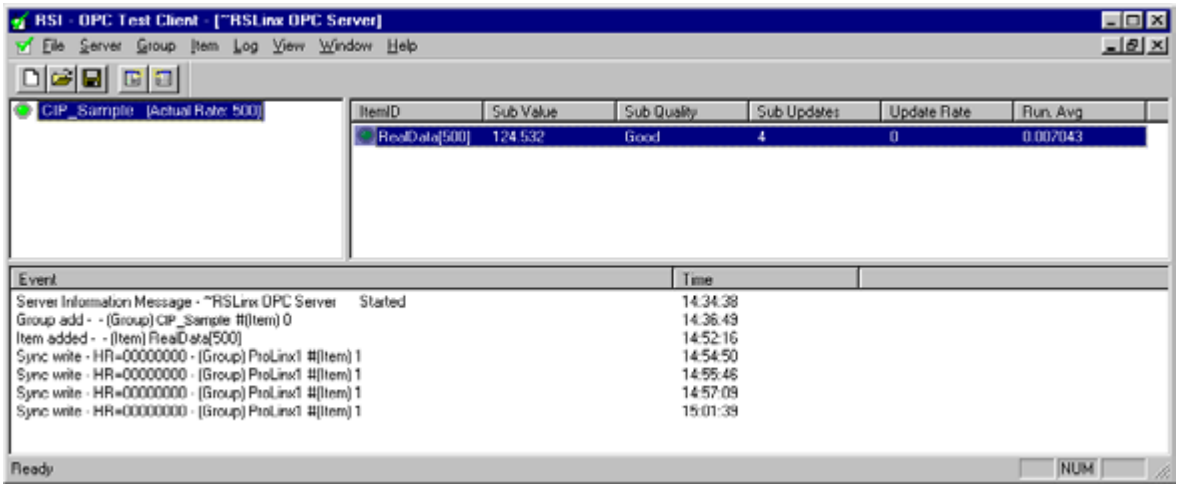
It is important to set the Access Path value to the Topic name assigned in RSLinx and to set the Item Name to a valid controller tag in the DFNT module. The Datatype parameter must be set to match that of the controller tag. In the example shown, the VT_R4 data type is selected for the floating-point tag. The Validate button can be used to verify that the point is valid in the OPC server. After configuring the new data item, click OK to add the point. The following illustration shows the new item and event windows.



You have now connected an OPC client to data in the DFNT module. In order to change the value for the item, select the Sync Write option from the Item menu and enter a new value as shown in the following dialog box:



Click **OK** to transfer the new value to the module. The following illustration shows the updated item data window.



5.16.2 ControlLogix (CLX) Processor

In order to exchange data between a ControlLogix processor and the module, the MSG instruction is used. There are two basic methods of data transfer supported by the module when using the MSG instruction: Encapsulated PCCC messages and CIP Data Table messages. Either method can be used, and the selection is left to the application developer.

Encapsulated PCCC Messages

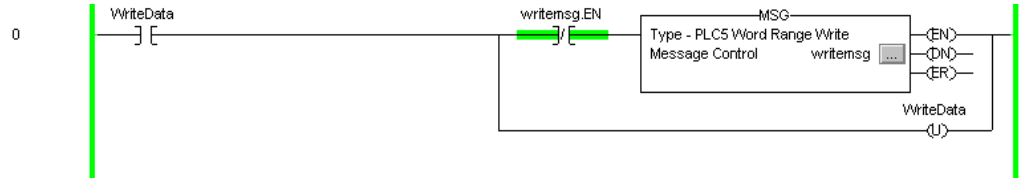
PLC5 and SLC5/05 processors containing an Ethernet interface use the encapsulated PCCC message method. The module simulates these devices and accepts both read and write commands. The following topics describe the support for the read and write operations.

Encapsulated PCCC Write Message

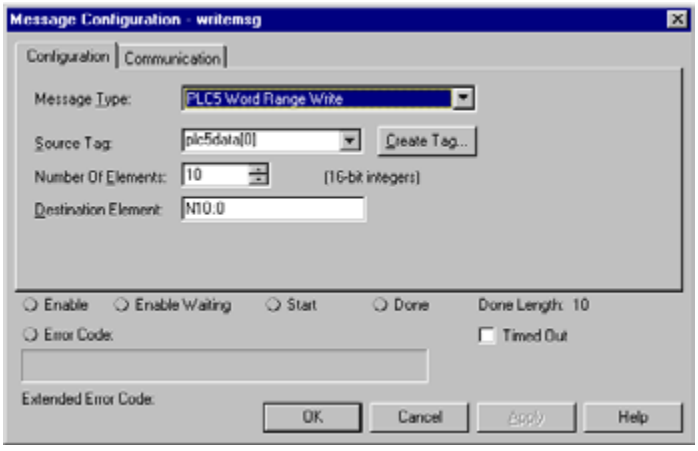
Write commands transfer data from the ControlLogix processor to the module. The following encapsulated PCCC commands are supported from a ControlLogix Processor:

- PLC2 Unprotected Write
- PLC5 Typed Write
- PLC5 Word Range Write
- PLC Typed Write

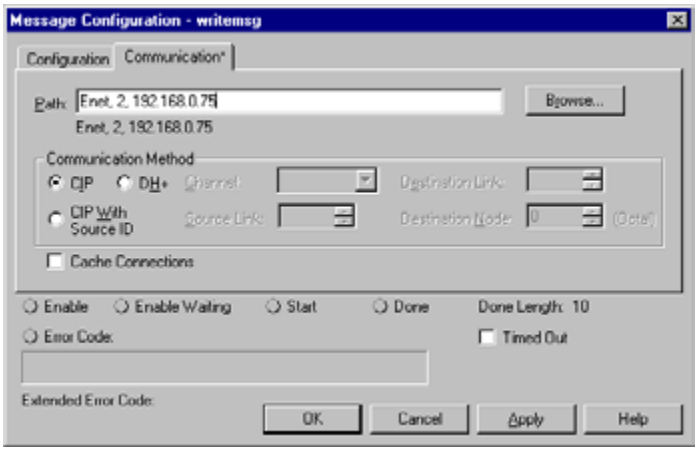
An example rung used to execute a write command is shown in the following diagram:



The **MESSAGE CONFIGURATION** dialog box must be completed to define the data set to be transferred from the processor to the module. An example of the dialog box follows:



Complete the dialog box for the data area to be transferred. For PLC5 and SLC messages, the **DESTINATION ELEMENT** should be an element in a data file (such as, N10:0). For the PLC2 Unprotected Write message, the **DESTINATION ELEMENT** is the address in the module's internal database and cannot be set to a value less than ten. This is not a limitation of the module but of the RSLogix software. For a PLC2 unprotected write or read function, the database address should be entered in octal format. Additionally, the **COMMUNICATION** information must also be configured. The following is an example of the dialog box.



Verify that the **CIP** radio-button is selected as the **COMMUNICATION METHOD**. The **PATH** specifies the message route from the ControlLogix processor to the DFNT module. Path elements are separated by commas. In the example path shown, the first element is "Enet", which is the user-defined name given to the 1756-ENET module in the chassis (you could substitute the slot number of the ENET module for the name), the second element, "2", represents the Ethernet port on the 1756-ENET module, and the last element of the path, "192.168.0.75", is the IP address of the DFNT module, the target for the message.

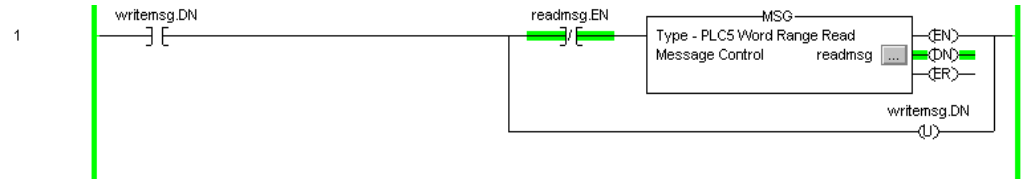
More complex paths are possible if routing to other networks using multiple 1756-ENET modules and racks. Refer to the Rockwell Automation Support Knowledgebase for more information on Ethernet routing and path definitions.

Encapsulated PCCC Read Message

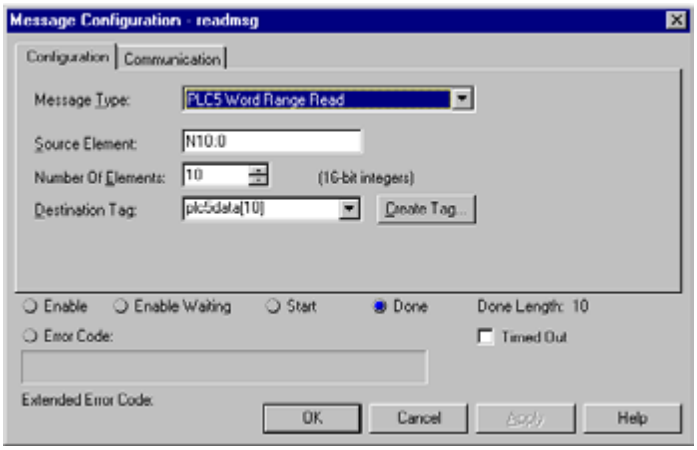
Read commands transfer data from the module to a ControlLogix processor. The following encapsulated PCCC commands are supported from a ControlLogix Processor:

- PLC2 Unprotected Read
- PLC5 Typed Read
- PLC5 Word Range Read
- PLC Typed Read

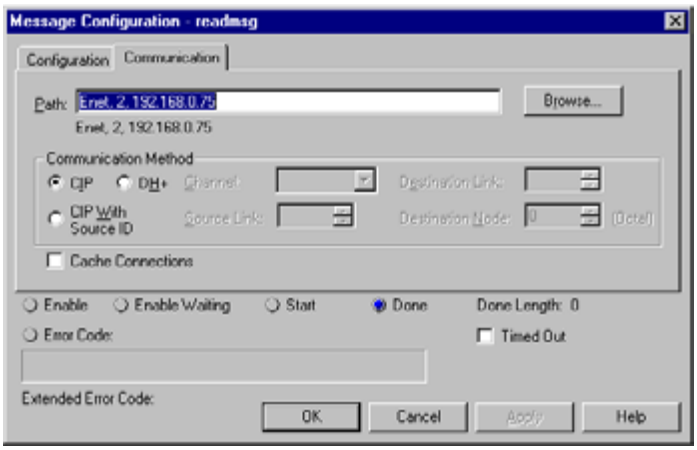
An example rung used to execute a read command is shown in the following diagram:



The **MESSAGE CONFIGURATION** dialog box must be completed to define the data set to transfer to the processor from the module. An example of the dialog box follows:



Complete the dialog box for the data area to be transferred. For PLC5 and SLC messages, the **SOURCE ELEMENT** should be an element in a data file (such as, N10:0). For the PLC2 Unprotected Read message, the **SOURCE ELEMENT** is the address in the module's internal database and cannot be set to value less than ten. This is not a limitation of the module but of the RSLogix software. Additionally, the **COMMUNICATION** information must also be configured. An example of the dialog box follows:



Verify that the **CIP** radio-button is selected as the **COMMUNICATION METHOD**. The **PATH** specifies the message route from the ControlLogix processor to the DFNT module. Path elements are separated by commas. In the example path shown, the first element is "Enet", which is the user-defined name given to the 1756-ENET module in the chassis (you could substitute the slot number of the ENET module for the name), the second element, "2", represents the Ethernet port on the 1756-ENET module, and the last element of the path, "192.168.0.75", is the IP address of the DFNT module, the target for the message.

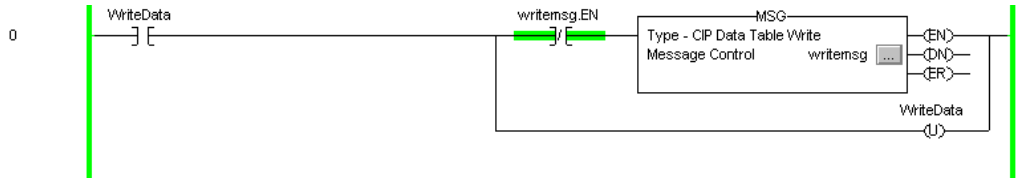
More complex paths are possible if routing to other networks using multiple 1756-ENET modules and racks. Refer to the Rockwell Automation Support Knowledgebase for more information on Ethernet routing and path definitions.

CIP Data Table Operations

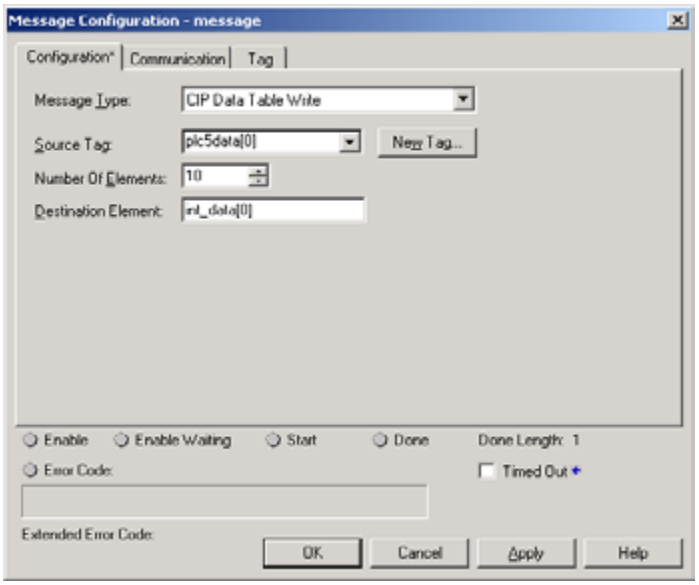
This method of data transfer uses CIP messages to transfer data between the ControlLogix processor and the module. Tag names define the elements to be transferred. The following topics describe the support for the read and write operations.

CIP Data Table Write

CIP data table write messages transfer data from the ControlLogix processor to the DFNT module. An example rung used to execute a write command is shown in the following diagram:



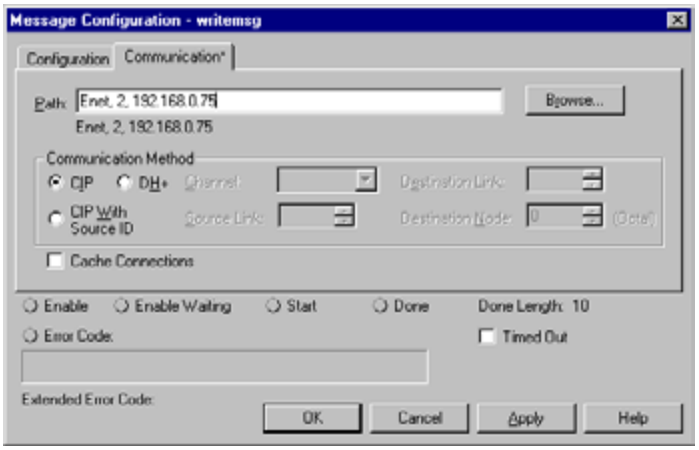
The **MESSAGE CONFIGURATION** dialog box must be completed to define the data set to be transferred from the processor to the module. An example of the dialog box follows:



Complete the dialog box for the data area to be transferred. CIP Data Table messages require a tag database element for both the source and destination. The **SOURCE TAG** is a tag defined in the ControlLogix Tag database. The **DESTINATION ELEMENT** is the tag element in the DFNT module.

The module simulates a tag database as an array of elements defined by the maximum register size for the module (user configuration parameter "Maximum Register" in the [Module] section) with the tag name **INT_DATA.**

In the previous example, the first element in the database is the starting location for the write operation of ten elements. Additionally, the **COMMUNICATION** information must also be configured. An example of the dialog box follows:

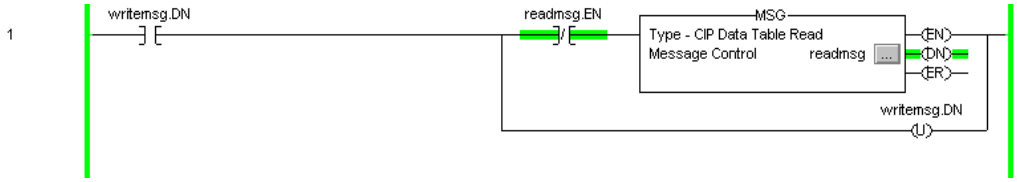


Verify that the **CIP** radio-button is selected as the **COMMUNICATION METHOD**. The **PATH** specifies the message route from the ControlLogix processor to the DFNT module. Path elements are separated by commas. In the example path shown, the first element is "Enet", which is the user-defined name given to the 1756-ENET module in the chassis (you could substitute the slot number of the ENET module for the name), the second element, "2", represents the Ethernet port on the 1756-ENET module, and the last element of the path, "192.168.0.75", is the IP address of the DFNT module, the target for the message.

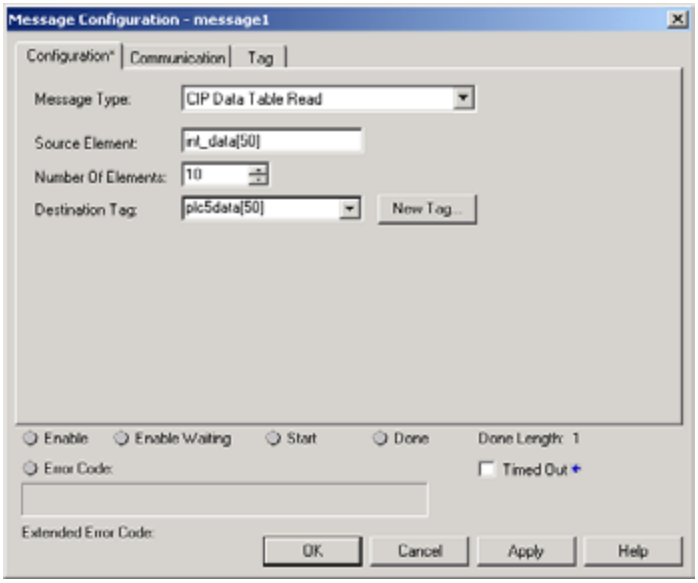
More complex paths are possible if routing to other networks using multiple 1756-ENET modules and racks. Refer to the Rockwell Automation Support Knowledgebase for more information on Ethernet routing and path definitions.

CIP Data Table Read

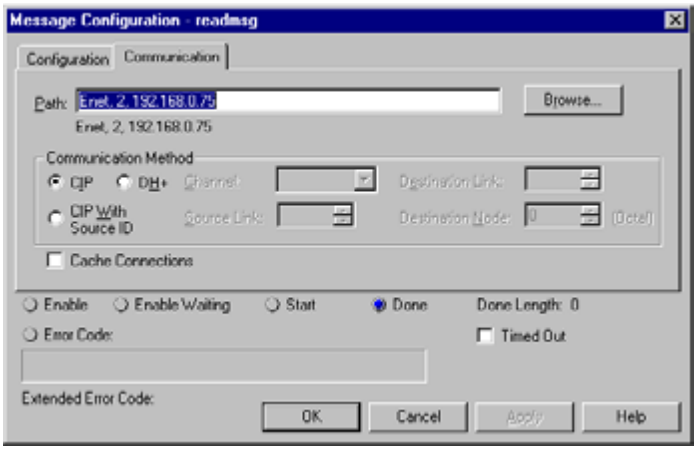
CIP data table read messages transfer data to the ControlLogix processor from the DFNT module. An example rung used to execute a read command is shown:



The **MESSAGE CONFIGURATION** dialog box must be completed to define the data set to transfer to the processor from the module. An example of the dialog box follows:



Complete the dialog box for the data area to be transferred. CIP Data Table messages require a tag database element for both the source and destination. The **DESTINATION TAG** is a tag defined in the ControlLogix Tag database. The **SOURCE ELEMENT** is the tag element in the DFNT module. The module simulates a tag database as an array of elements defined by the maximum register size for the module (user configuration parameter "Maximum Register" in the [Module] section) with the tag name **INT_DATA.** In the example above, the first element in the database is the starting location for the read operation of ten elements. Additionally, the **COMMUNICATION** information must also be configured. An example of the dialog box follows:



Verify that the **CIP** radio-button is selected as the **COMMUNICATION METHOD**. The **PATH** specifies the message route from the ControlLogix processor to the DFNT module. Path elements are separated by commas. In the example path shown, the first element is "Enet", which is the user-defined name given to the 1756-ENET module in the chassis (you could substitute the slot number of the ENET module for the name), the second element, "2", represents the Ethernet port on the 1756-ENET module, and the last element of the path, "192.168.0.75", is the IP address of the DFNT module, the target for the message.

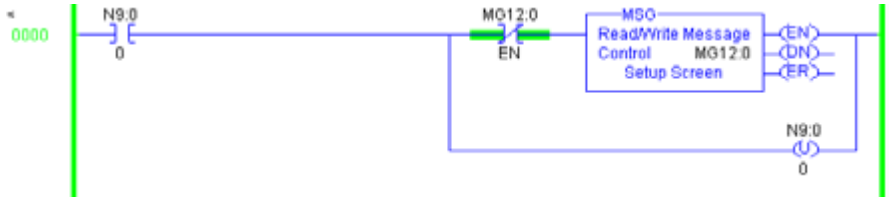
More complex paths are possible if routing to other networks using multiple 1756-ENET modules and racks. Refer to the Rockwell Automation Support Knowledgebase for more information on Ethernet routing and path definitions.

5.16.3 PLC5 Processor

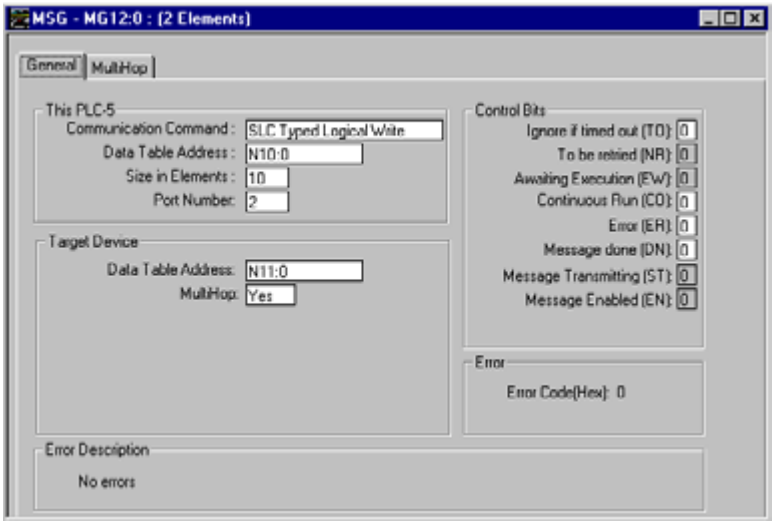
The module can be used to receive messages from a PLC5 containing an Ethernet interface. The module supports both read and write commands. A discussion of each operation is provided in the following topics:

PLC5 Write Commands

Write commands transfer data from the PLC5 processor to the DFNT module. An example rung used to execute a write command is shown in the following diagram:



In order to complete the configuration of the MSG instruction, select the **SETUP SCREEN** area of the MSG object. This displays the following dialog box.

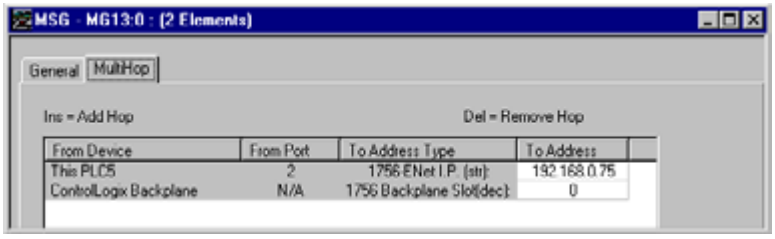


Select the **COMMUNICATION COMMAND** to execute from the following list of supported commands.

- PLC5 Type Write
- PLC2 Unprotected Write
- PLC5 Typed Write to PLC
- PLC Typed Logical Write

The **TARGET DEVICE DATA TABLE ADDRESS** must be set to a valid file element (such as, N11:0) for SLC and PLC5 messages. For the PLC2 Unprotected Write message, set the address to the database index (such as, 1000) to consider with the command.

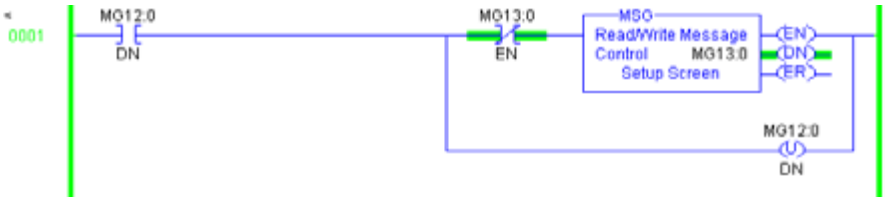
The **MULTIHOP** option must be set to **YES.** The **MULTIHOP** tab portion of the dialog box must be completed as shown in the following window:



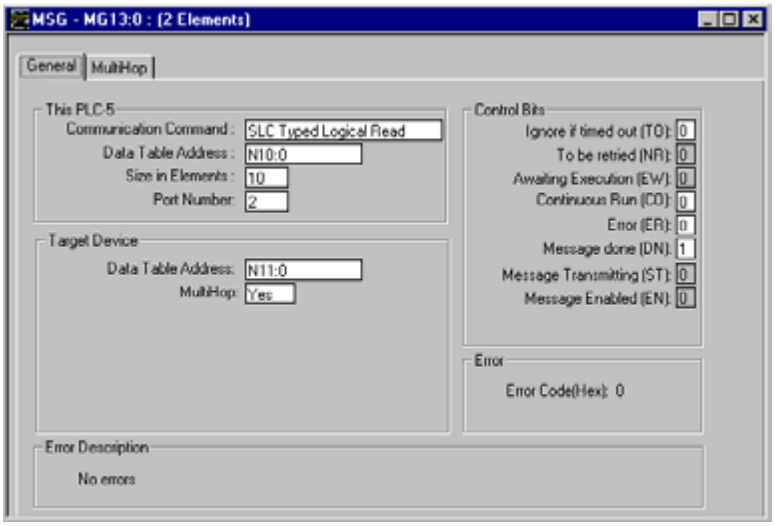
Set the IP address value to the module's Ethernet IP address. You must press the "Insert" key to add the second line for ControlLogix Backplane and set the slot number to zero.

PLC5 Read Commands

Read commands transfer data to the PLC5 processor from the DFNT module. An example rung used to execute a read command is shown in the following diagram:



In order to complete the configuration of the MSG instruction, select the **SETUP SCREEN** area of the MSG object. This displays the following dialog box.

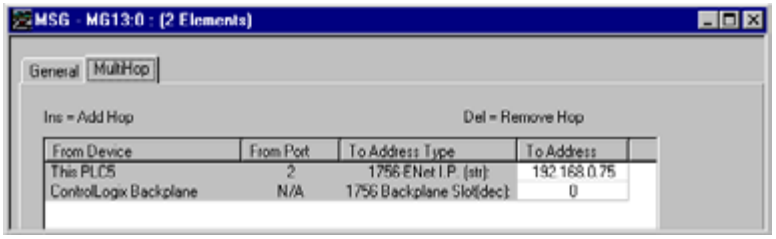


Select the **COMMUNICATION COMMAND** to execute from the following list of supported commands.

- PLC5 Type Read
- PLC2 Unprotected Read
- PLC5 Typed Read to PLC
- PLC Typed Logical Read

The **TARGET DEVICE DATA TABLE ADDRESS** must be set to a valid file element (such as, N11:0) for SLC and PLC5 messages. For the PLC2 Unprotected Read message, set the address to the database index (such as, 1000) to consider with the command.

The **MULTIHOP** option must be set to **YES.** The **MULTIHOP** tab portion of the dialog box must be completed as shown in the following window:



Set the IP address value to the module's Ethernet IP address. You must press the "Insert" key to add the second line for ControlLogix Backplane and set the slot number to zero.

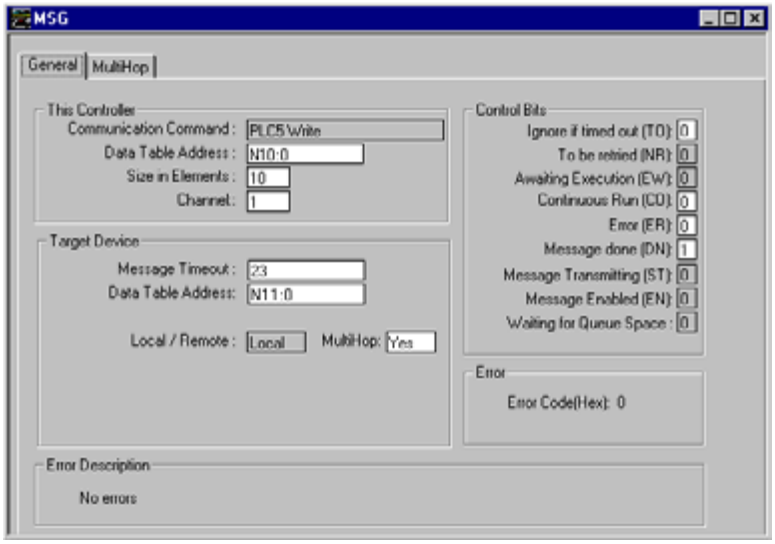
5.16.4 SLC 5/05 Processor

The module can be used to receive messages from a SLC 5/05 containing an Ethernet interface. The module supports both read and write commands. A discussion of each operation is provided in the following topics.

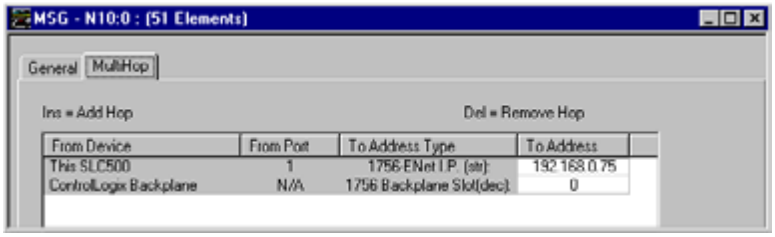
SLC5/05 Write Commands

Write commands transfer data from the SLC processor to the DFNT module. An example rung used to execute a write command is shown in the following diagram:

Set the **READ/WRITE** parameter to **WRITE.** The module supports a **TARGET DEVICE** parameter value of **500CPU** or **PLC5.** In order to complete the configuration of the MSG instruction, select the **SETUP SCREEN** area of the MSG object. This displays the following dialog box.



The **TARGET DEVICE DATA TABLE ADDRESS** must be set to a valid file element (such as, N11:0) for SLC and PLC5 messages. The **MULTIHOP** option must be set to **YES.** The **MULTIHOP** tab portion of the dialog box must be completed as displayed in the following window:

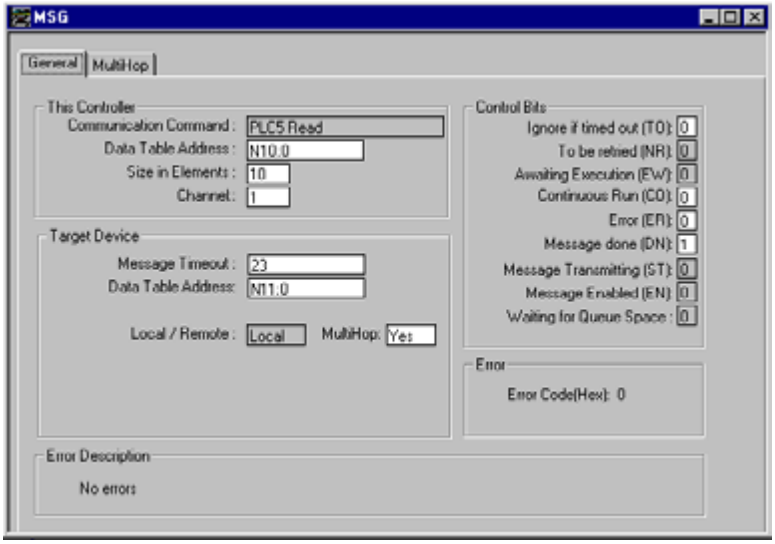


Set the IP address value to the module's Ethernet IP address. You must press the "Insert" key to add the second line for ControlLogix Backplane and set the slot number to zero.

SLC5/05 Read Commands

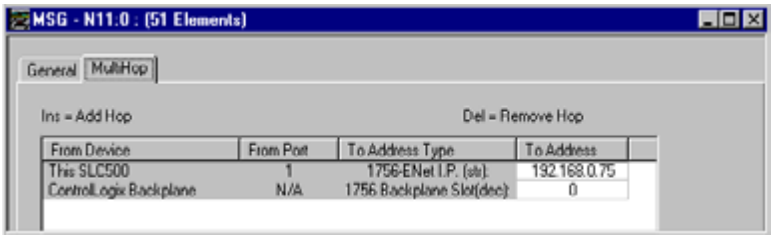
Read commands transfer data to the SLC processor from the DFNT module. An example rung used to execute a read command is shown in the following diagram:

Set the **READ/WRITE** parameter to **READ.** The module supports a **TARGET DEVICE** parameter value of **500CPU** or **PLC5.** In order to complete the configuration of the MSG instruction, select the **SETUP SCREEN** area of the MSG object. This displays the following dialog box.



The **TARGET DEVICE DATA TABLE ADDRESS** must be set to a valid file element (such as, N11:0) for SLC and PLC5 messages. The **MULTIHOP** option must be set to **YES.**

Fill in the **MULTIHOP** tab portion of the dialog box as shown in the following illustration.

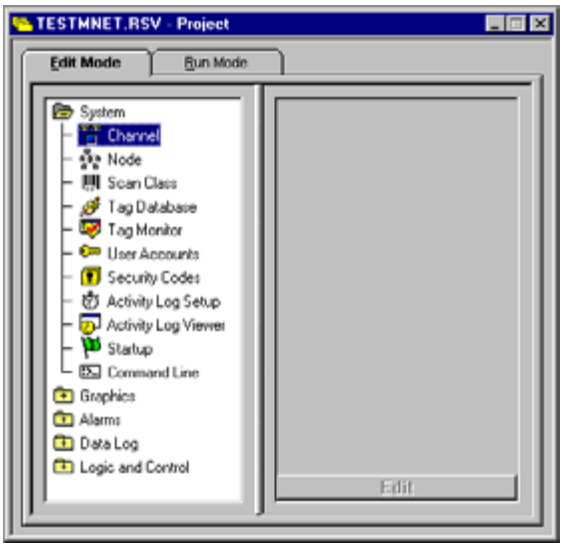


Set the IP address value to the module's Ethernet IP address. You must press the "Insert" key to add the second line for ControlLogix Backplane and set the slot number to zero.

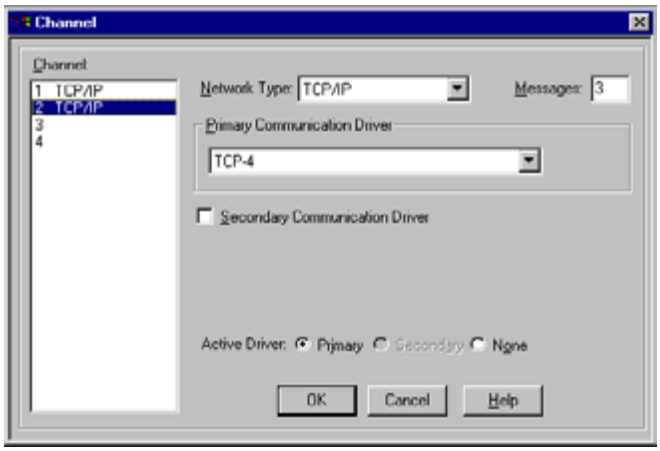
5.16.5 RSView Software

RSView is a client application for building user interfaces to control systems. This tool requires RSLinx to be loaded and operational (refer to the RSLinx section of this document). In order to interface RSView to a DFNT module, the following steps are required:

First select the **CHANNEL** option from the **EDIT MODE** tab as shown in the following window.

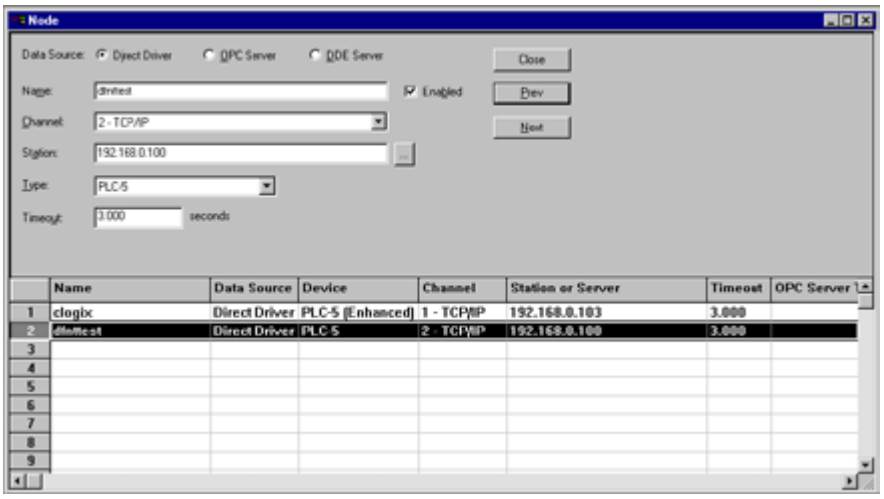


After selecting the option, the following dialog box is displayed:



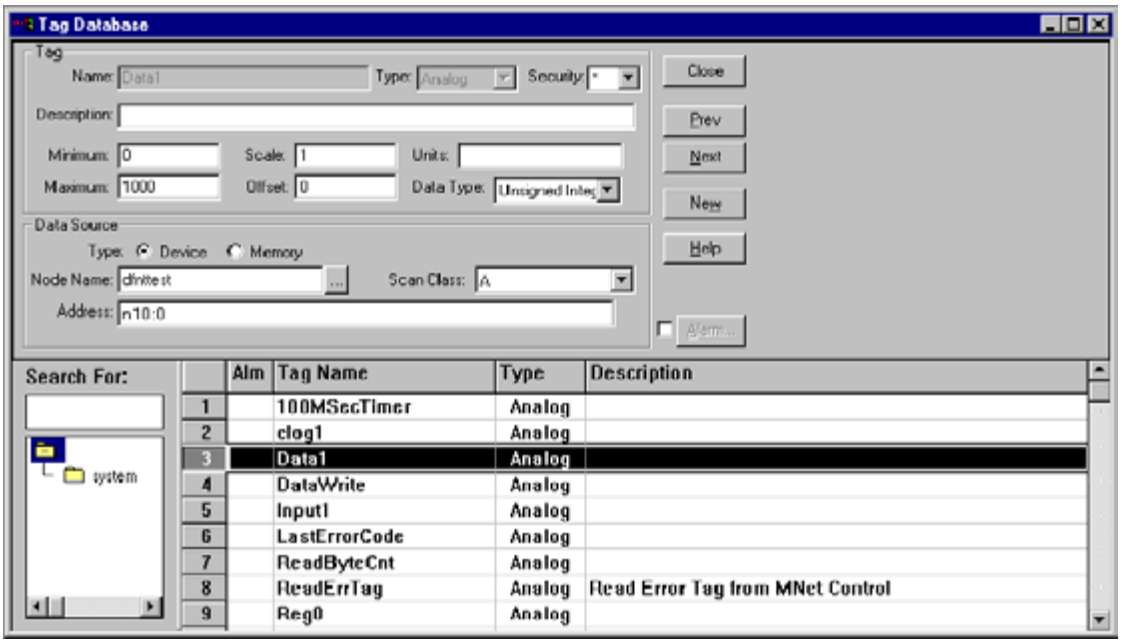
Select the **NETWORK TYPE** and **PRIMARY COMMUNICATION DRIVER** for the channel to be associated with the DFNT module. The **NETWORK TYPE** should be TCP/IP and the **PRIMARY COMMUNICATION DRIVER** name should match that set up in RSLinx. Click OK to save the information.

Next select the **NODE** option from the **EDIT MODE** tab. After selecting the option, the following dialog box is displayed:



Enter a record in the dialog box for the DFNT module to be addressed. The **NAME** field identifies the module to the RSView system. The **CHANNEL** parameter should be that defined in the channel set up defined above. The **STATION** parameter should be set to the IP address of the DFNT module. Select the **CLOSE** button after completing the node entry. If your version of the RSView supports ControlLogix controller tag read and write operations, select the device type consistent with the ControlLogix processor. This will permit direct access to the controller tags simulated in the module. Use of controller tags simplifies handling of the data in RSView as the data will be passing in the correct data format (that is, bit, word, float).

Next select the **TAG DATABASE** option from the **EDIT MODE** tab. After selecting the option, the following dialog box is displayed:



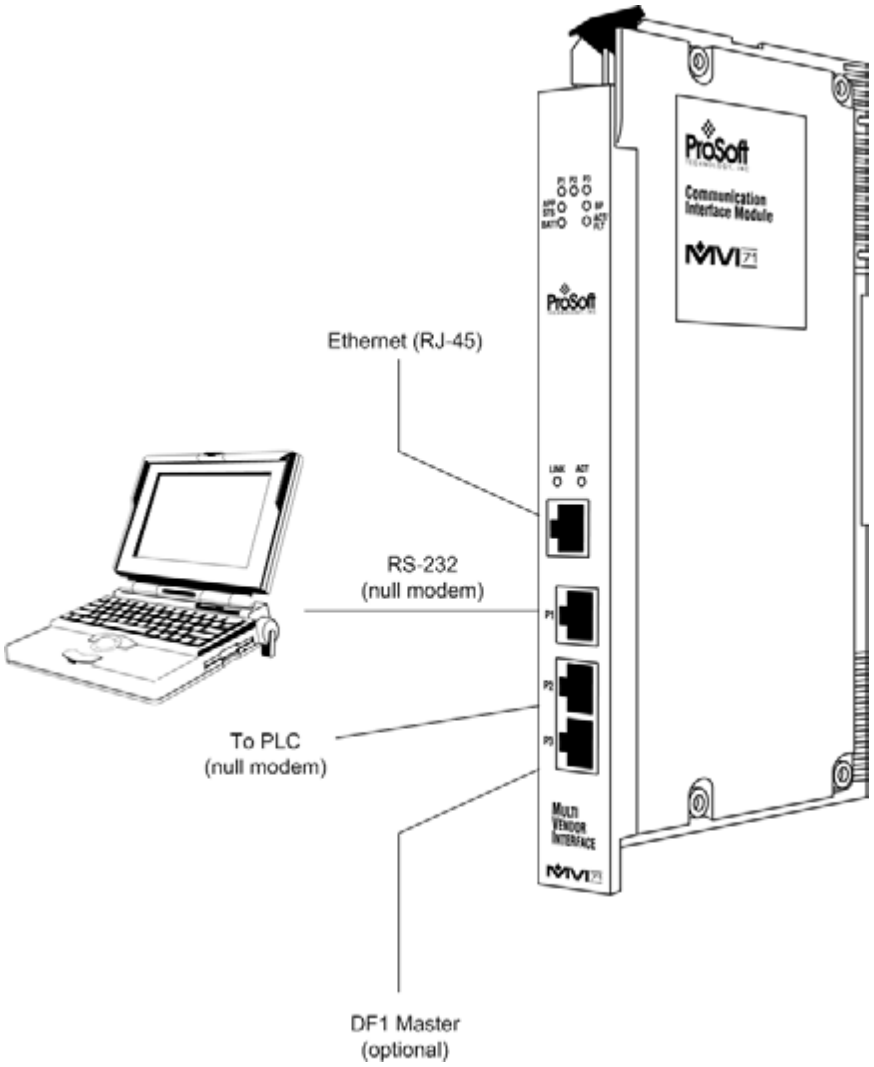
Set up tags for each element to be transferred between RSView and the DFNT module. In the example above, **DATA1** is associated with the first element in the DFNT module's database (N10:0). A tag should be setup for each register in the module's database to be interfaced. If RSView is set in run mode, values for the tags should match those in the module's database. Use the module controller tag names if using CIP data table read and write operations.

Refer to the RSView documentation for a full discussion of database tags and reading and writing data between RSView and a processor.

5.17 Accessing a PLC Processor via Ethernet Using MVI71-DFNT

This section explains how to set up Ethernet access to a PLC 5 processor using the MVI71-DFNT module. Port 2 (middle port) should be connected to the PLC RS-232 port using an RS-232 null modem cable. Optionally you can use Port 3 to emulate Channel 0 on the PLC. A Master DF1 device can be attached to Port 3 to monitor and control the PLC data files.

The following network should be used:



Conversion software (user-supplied) is required in order to convert serial data to TCP/IP format. The software will typically select a serial port that is not being used and direct to it to an IP address and port number (MVI71-DFNT Pass-Through Server).

Perform the following steps to set up your network.

Step 1- Configure the Pass-Through Server

The MVI71-DFNT Pass-Through Server connects a local computer serial port to the module's Ethernet driver. Refer to the configuration file to configure the following section:

```
[DF1 Pass-Through Server Port 1]
Enabled: Yes #Y=Use server, N=Do not use server
Service Port Number: 15000 #TCP service port for this server
Busy Timeout: 500 #Time to wait for not Busy (100-65535 milliseconds)
Baud Rate: 19200 #Baud rate for port 110-115200
Parity: None #N=None,O=Odd,E=Even,M=Mark,S=Space
Data Bits: 8 #5, 6, 7 or 8
Stop Bits: 1 #1 or 2
```

Select Y (or Yes) for the Enabled parameter and then configure each communication parameter.

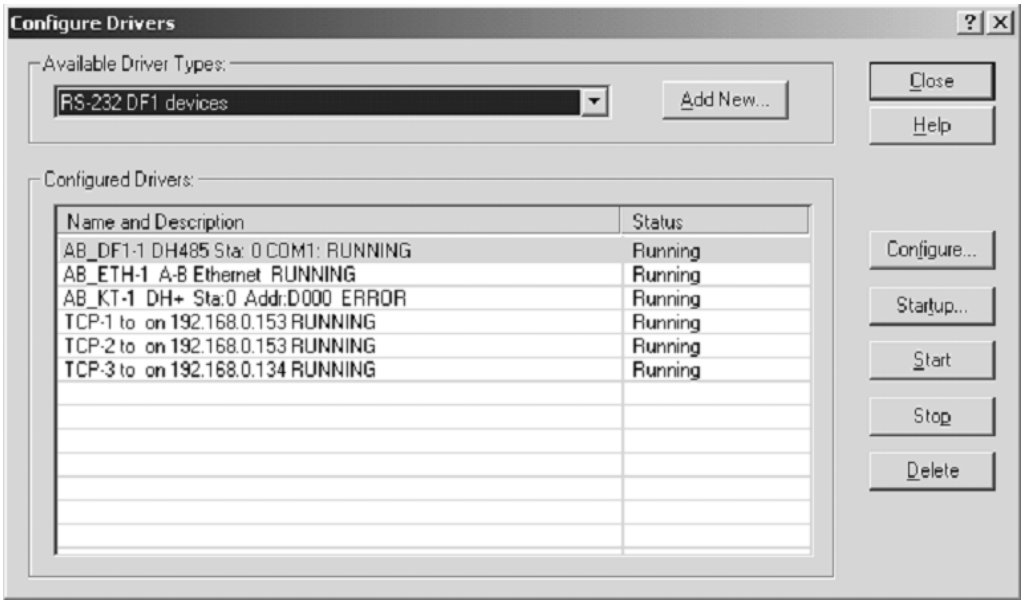
Step 2- Configure the Serial/IP Converter Software

Conversion software is required in order to convert serial data to TCP/IP format. The software will select a serial port that is not being used and direct to it to an IP address and port number (MVI71-DFNT Pass-Through Server). It is essential to select the MVI71-DFNT IP address and the port number configured on the previous step.

The serial communication parameters should also match the parameters configured in the previous step.

Step 3- Configure a RS-232 driver at RS-LINX

Open the RS-LINX software and create a new RS-232 driver:



Select the COM Port (configured during the previous step) and click on Auto-Config.

Step 4- Use the New Driver to Connect to the PLC

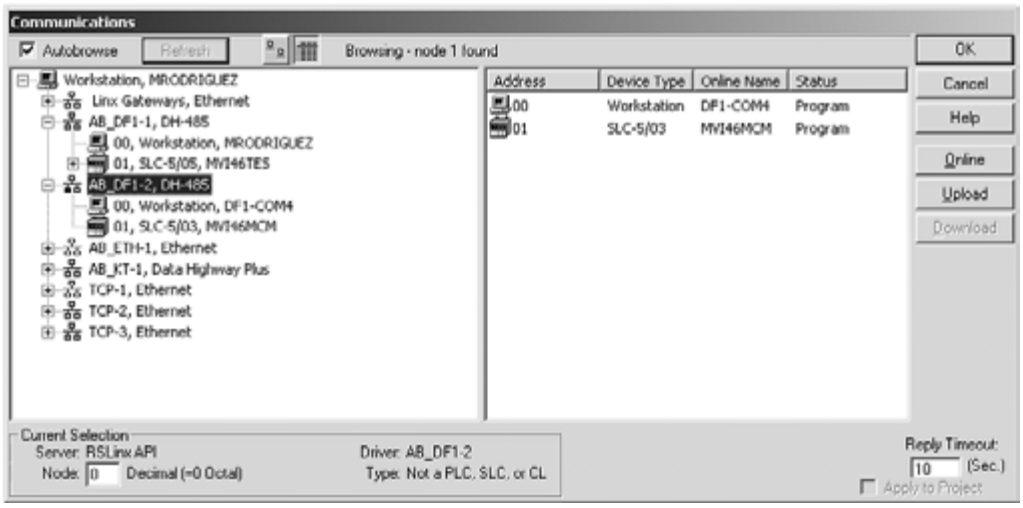
You can now use the new driver created during the previous step to access the PLC processor.

Open the RSLogix 5 and click on **Comms** > **System Comms** in order to view all configured drivers.

Click on the newly created driver. The PLC 5 processor appears on the screen.

Select the processor and click **Online**.

You now have an Ethernet connection to a PLC 5 processor.



Step 5 - (Optional) Configure the Port 3 Pass-Through

After you configure the MVI71-DFNT pass-through server, you may configure Port 3 to receive DF1 commands in order to allow access from a DF1 Master to the PLC data file.

In order to accomplish this, refer to the DF1-Pass-Through Port Section at the MVI71-DFNT configuration file.

```
[DF1 Pass-Through Port]
Enabled: Y #Y=Use port, N=Do not use port
Local Station ID: 1 #DF1 node address
Protocol: Full #F=Full-Duplex, H=Half-Duplex
Termination Type: CRC #B=BCC, C=CRC
Baud Rate: 19200 #Baud rate for port 110-115200
Parity: None #N=None,O=Odd,E=Even,M=Mark,S=Space
Data Bits: 8 #5, 6, 7 or 8
Stop Bits: 1 #1 or 2
RTS On: 0 #0-65536 mSec before message
RTS Off: 0 #0-65536 mSec after message
Use CTS Line: No #Use CTS modem control line (Y/N)
Retry Count: 3 #Response failure retry count
Request Timeout: 1000 #Request message timeout (0-65535 milliseconds)
Busy Timeout: 500 #Port Busy timeout (0-65535 milliseconds)
ACK Timeout: 100 #DLE-ACK timeout (0-65535 milliseconds)
```

Note: The Port 3 pass-through feature will only work properly if the user is not accessing the PLC program using the pass-through server.

5.17.1 Troubleshooting

If you are unable to connect the software converter to the MVI71-DFNT you should configure the communication parameter used in the network.

To establish communication between the local PC and the PLC processor, all communication parameters at each part of the network must match. This means that the communication parameters configured at the following parts of the network should be the same:

- PLC processor Channel 0 (Use RSLogix 5000)
- MVI71-DFNT configuration
- Serial/IP Software Converter
- RSLinx RS-232 driver (use the auto-config feature)

6 Support, Service & Warranty

In This Chapter

Contacting Technical Support

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 associated ladder files, if any
2. Module operation and any unusual behavior
3. Configuration/Debug status information
4. LED patterns
5. Details about the serial, Ethernet or fieldbus devices interfaced to the module, if any.

Note: *For technical support calls within the United States, an after-hours answering system allows 24-hour/7-days-a-week pager access to one of our qualified Technical and/or Application Support Engineers. Detailed contact information for all our worldwide locations is available on the following page.*

Internet	Web Site: www.prosoft-technology.com/support E-mail address: support@prosoft-technology.com
Asia Pacific (location in Malaysia)	Tel: +603.7724.2080, E-mail: asiapc@prosoft-technology.com Languages spoken include: Chinese, English
Asia Pacific (location in China)	Tel: +86.21.5187.7337 x888, E-mail: asiapc@prosoft-technology.com Languages spoken include: Chinese, English
Europe (location in Toulouse, France)	Tel: +33 (0) 5.34.36.87.20, E-mail: support.EMEA@prosoft-technology.com Languages spoken include: French, English
Europe (location in Dubai, UAE)	Tel: +971-4-214-6911, E-mail: mea@prosoft-technology.com Languages spoken include: English, Hindi
North America (location in California)	Tel: +1.661.716.5100, E-mail: support@prosoft-technology.com Languages spoken include: English, Spanish
Latin America (Oficina Regional)	Tel: +1-281-2989109, E-Mail: latinam@prosoft-technology.com Languages spoken include: Spanish, English
Latin America (location in Puebla, Mexico)	Tel: +52-222-3-99-6565, E-mail: soporte@prosoft-technology.com Languages spoken include: Spanish
Brasil (location in Sao Paulo)	Tel: +55-11-5083-3776, E-mail: brasil@prosoft-technology.com Languages spoken include: Portuguese, English

6.1 Return Material Authorization (RMA) Policies and Conditions

The following Return Material Authorization (RMA) Policies and Conditions (collectively, "RMA Policies") apply to any returned product. These RMA Policies are subject to change by ProSoft Technology, Inc., without notice. For warranty information, see Limited Warranty (page 169). In the event of any inconsistency between the RMA Policies and the Warranty, the Warranty shall govern.

6.1.1 Returning Any Product

a) In order to return a Product for repair, exchange, or otherwise, the Customer must obtain a Return Material Authorization (RMA) number from ProSoft Technology and comply with ProSoft Technology 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 165). 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 Technology, 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 Technology using a shipment method other than that specified by ProSoft Technology, or shipped without an RMA number will be returned to the Customer, freight collect. Contact ProSoft Technical Support for further information.
d) A 10% restocking fee applies to all warranty credit returns, whereby a Customer has an application change, ordered too many, does not need, etc. Returns for credit require that all accessory parts included in the original box (i.e.; antennas, cables) be returned. Failure to return these items will result in a deduction from the total credit due for each missing item.

6.1.2 Returning Units Under Warranty

A Technical Support Engineer must approve the return of Product under ProSoft Technology'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 Technology at designated location referenced on the Return Material Authorization
   i. If a defect is found and is determined to be customer generated, or if the defect is otherwise not covered by ProSoft Technology s warranty, there will be no credit given. Customer will be contacted and can request module be returned at their expense;
   ii. If defect is customer generated and is repairable, customer can authorize ProSoft Technology to repair the unit by providing a purchase order for 30% of the current list price plus freight charges, duties and taxes as applicable.

6.1.3 Returning Units Out of Warranty

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

The following is a list of non-repairable units:

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

6.2 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 Technology, Incorporated (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.

6.2.1 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 (3) years from the date of shipment for Product purchased **on or after** January 1st, 2008, or one (1) year from the date of shipment for Product purchased **before** January 1st, 2008 (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 from date of original shipment. 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 remanufactured replacement parts. ProSoft reserves the right, and at its sole discretion, may replace unrepairable units with new or remanufactured equipment. All replacement units will be covered under warranty for the 3 year period commencing from the date of original equipment purchase, not the date of shipment of the replacement unit. Such repair, including both parts and labor, will be performed at ProSoft's expense. All warranty service will be performed at service centers designated by ProSoft.

b) *Warranty On Services*: Materials and labor performed by ProSoft to repair a verified malfunction or defect are warranteed in the terms specified above for new Product, provided said warranty will be for the period remaining on the original new equipment warranty or, if the original warranty is no longer in effect, for a period of 90 days from the date of repair.

6.2.2 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.

6.2.3 Disclaimer Regarding High Risk Activities

Product manufactured or supplied by ProSoft is not fault tolerant and is not designed, manufactured or intended for use in hazardous environments requiring fail-safe performance including and without limitation: the operation of nuclear facilities, aircraft navigation of communication systems, air traffic control, direct life support machines or weapons systems in which the failure of the product could lead directly or indirectly to death, personal injury or severe physical or environmental damage (collectively, "high risk activities"). ProSoft specifically disclaims any express or implied warranty of fitness for high risk activities.

6.2.4 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.

6.2.5 Disclaimer of all Other Warranties

The Warranty set forth in What Is Covered By This Warranty (page 169) 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.

6.2.6 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 include, but are not limited to: Loss of profits; loss of savings or revenue; loss of use of the product or any associated equipment; loss of data; cost of capital; cost of any substitute equipment, facilities, or services; downtime; the claims of third parties including, customers of the Purchaser; and, injury to property.

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

6.2.7 Time Limit for Bringing Suit

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

6.2.8 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.

6.2.9 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.

6.2.10 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

G

H

I

K

L

M

N

O

P

R

S

T

U

V

W

Y