



Where Automation Connects.



MVI69-DFNT

**CompactLogix or MicroLogix
Platform**

EtherNet/IP Client/Server
Communication Module

May 14, 2014

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

Copyright © 2014 ProSoft Technology, Inc., All rights reserved.

MVI69-DFNT User Manual

May 14, 2014

ProSoft Technology[®], ProLinX[®], inRAX[®], ProTalk[®], and RadioLinX[®] are Registered Trademarks of ProSoft Technology, Inc. All other brand or product names are or may be trademarks of, and are used to identify products and services of, their respective owners.

ProSoft Technology[®] Product Documentation

In an effort to conserve paper, ProSoft Technology no longer includes printed manuals with our product shipments. User Manuals, Datasheets, Sample Ladder Files, and Configuration Files are provided on the enclosed DVD in Adobe[®] Acrobat Reader file format (.PDFs). These product documentation files may also be freely downloaded from our web site: www.prosoft-technology.com

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:

WARNING - EXPLOSION HAZARD - SUBSTITUTION OF COMPONENTS MAY IMPAIR SUITABILITY FOR CLASS I, DIV. 2;

WARNING - EXPLOSION HAZARD - WHEN IN HAZARDOUS LOCATIONS, TURN OFF POWER BEFORE REPLACING OR WIRING MODULES

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

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 non-hazardous.
- 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 @ 5.1 Vdc
- Power Supply Distance Rating: 2
- Operating Temperature: 0°C to 60°C (32°F to 140°F)
- Storage Temperature: -40°C to 85°C (-40°F to 185°F)
- 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

E193122

Class I Div 2

Groups A,B,C,D T6

-30°C <= Ta <= 60°C

II 3 G

Ex nA IIC T6 X

0°C <= Ta <= +60°C

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

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

G - Equipment protected against explosive gasses.

Agency Approvals and Certifications

Agency	Applicable Standard(s)
ATEX	EN 60079-0:2006, EN 60079-15:2005
DNV	DET NORSKE VERITAS Test 2.4
CE	EMC-EN61326-1:2006; EN61000-6-4:2007
CB Safety	CA/10533/CSA, IEC 61010-1 Ed. 2, CB 243333-2056722 (2090408)
GOST-R	EN 61010
CSA	61010
cULus	UL508, UL1604, CSA 22.2 No 142 & 213



ME06



243333



E193122

Contents

Your Feedback Please	2
How to Contact Us	2
ProSoft Technology® Product Documentation	2
Important Installation Instructions	3
MVI (Multi Vendor Interface) Modules	3
Warnings	3
Battery Life Advisory	4
Markings.....	4

Guide to the MVI69-DFNT User Manual **9**

1 Start Here **11**

1.1	System Requirements	12
1.2	Package Contents	13
1.3	Installing ProSoft Configuration Builder Software	14
1.4	Setting Jumpers	15
1.5	Installing the Module	16
1.6	Connecting Your PC to the Processor	19
1.7	Downloading the Sample Program to the Processor	20
1.7.1	Configuring the RSLinx Driver for the PC COM Port	21
1.8	Connecting Your PC to the Module.....	23

2 Configuring the MVI69-DFNT Module **25**

2.1	Using ProSoft Configuration Builder	25
2.1.1	Setting Up the Project	26
2.1.2	Setting Module Parameters.....	28
2.2	[Module].....	30
2.2.1	Failure Flag Count.....	30
2.2.2	Initializing Output Data	30
2.2.3	DFNT Server File Size	31
2.2.4	Read Register Start.....	31
2.2.5	Read Register Count.....	31
2.2.6	Write Register Start.....	31
2.2.7	Write Register Count.....	31
2.2.8	Block Transfer Size	31
2.3	[DFNT Client x].....	32
2.3.1	Minimum Command Delay	32
2.3.2	Response Timeout	32
2.3.3	Retry Count	32
2.4	[DFNT Client x Commands]	33
2.4.1	Command List	33
2.4.2	Command Entry Formats	36
2.5	Ethernet Configuration	41
2.6	Downloading the Project to the Module Using a Serial COM Port.....	42

3	Ladder Logic	43
3.1	Module Data.....	43
3.2	Using the RSLogix 5000 v16 Add-On Instruction	44
3.2.1	Importing the Ladder Rung - MVI69-DFNT	45
3.3	Adding the Module to an Existing CompactLogix Project.....	51
3.4	Adding the Module to an Existing MicroLogix Project	55
4	Diagnostics and Troubleshooting	57
4.1	LED Status Indicators	58
4.1.1	Ethernet LED Indicators.....	59
4.1.2	Clearing a Fault Condition	60
4.1.3	Troubleshooting	60
4.2	Using ProSoft Configuration Builder (PCB) for Diagnostics	62
4.2.1	Using the Diagnostic Window in ProSoft Configuration Builder	62
4.2.2	Navigation	65
4.2.3	Main Menu	66
4.2.4	Database View Menu.....	70
4.2.5	Master Command Error List Menu.....	73
4.2.6	Master Command List Menu.....	74
4.2.7	Network Menu.....	75
4.3	Reading Status Data from the Module	76
5	Reference	77
5.1	Product Specifications	77
5.1.1	EtherNet/IP (Explicit Messaging) Compatible Devices.....	78
5.1.2	General Specifications	78
5.1.3	Hardware Specifications	79
5.1.4	Functional Specifications	80
5.2	Functional Overview	81
5.2.1	General Concepts	81
5.2.2	Normal Data Transfer	83
5.2.3	Module Control Blocks.....	89
5.2.4	Data Flow between MVI69-DFNT Module and CompactLogix or MicroLogix Processor	99
5.3	Server Driver.....	107
5.3.1	RSLink Software	107
5.3.2	ControlLogix (CLX) Processor	117
5.3.3	PLC5 Processor.....	125
5.3.4	SLC 5/05 Processor.....	128
5.3.5	RSView Software	131
5.4	Cable Connections	134
5.4.1	Ethernet Connection	134
5.4.2	RS-232 Configuration/Debug Port	136
5.4.3	DB9 to RJ45 Adaptor (Cable 14)	138
5.5	Error Codes.....	139
5.5.1	Local STS Error Codes	139
5.5.2	Remote STS Error Codes	140
5.5.3	Errors When EXT STS Is Present	141
5.5.4	Module Specific Error (not DFNT Compliant)	142
5.6	TCP/IP Interface Errors	143

5.6.1	Timeout Errors.....	143
5.6.2	Register Session Response Errors	143
5.6.3	Forward Open Response Errors	143
5.6.4	PCCC Response Errors	144
5.7	Command Function Codes.....	145
5.8	General Command Structure	146
5.8.1	Function Code #1 - Protected Write (Basic Command Set)	147
5.8.2	Function Code #2 - Unprotected Read (Basic Command Set)	147
5.8.3	Function Code #3 - Protected Bit Write (Basic Command Set)	148
5.8.4	Function Code #4 - Unprotected Bit Write (Basic Command Set)	148
5.8.5	Function Code #5 - Unprotected Write (Basic Command Set)	149
5.8.6	Function Code #100 - Word Range Write (PLC-5 Command) (Binary Address)..	150
5.8.7	Function Code #101 - Word Range Read (PLC-5 Command) (Binary Address) .	151
5.8.8	Function Code #102 - Read-Modify-Write (PLC-5 Command) (Binary Address) .	152
5.8.9	Function Code #150 - Word Range Write (PLC-5 Command) (ASCII Address) ..	153
5.8.10	Function Code #151 - Word Range Read (PLC-5 Command) (ASCII Address) ..	154
5.8.11	Function Code #152 - Read-Modify-Write (PLC-5 Command) (ASCII Address) ..	155
5.8.12	Function Code #501 - Protected Typed Logical Read (Two Address Fields)	156
5.8.13	Function Code #502 - Protected Typed Logical Read (Three Address Fields)	157
5.8.14	Function Code #509 - Protected Typed Logical Write (Two Address Fields)	158
5.8.15	Function Code #510 - Protected Typed Logical Write (Three Address Fields)	159
5.8.16	Function Code #511 - Protected Typed Logical Write with Mask (Three Address Fields)	160
5.9	MicroLogix Processor Specifics	161
5.9.1	SLC File Types.....	161
5.10	DFNT Command Entry Form	162
6	Support, Service & Warranty	163
6.1	Contacting Technical Support	163
6.2	Warranty Information	164
Index		165

Guide to the MVI69-DFNT User Manual

Function		Section to Read	Details
Introduction (Must Do)	→	Start Here (page 10)	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 57)	This section describes Diagnostic and Troubleshooting procedures.
Reference Product Specifications	→	Reference (page 77) Product Specifications (page 77)	These sections contain general references associated with this product and its Specifications..
Support, Service, and Warranty Index	→	Support, Service and Warranty (page 163) Index	This section contains Support, Service and Warranty information. Index of chapters.

1 Start Here

In This Chapter

❖ System Requirements	12
❖ Package Contents	13
❖ Installing ProSoft Configuration Builder Software	14
❖ Setting Jumpers	15
❖ Installing the Module	16
❖ Connecting Your PC to the Processor.....	19
❖ Downloading the Sample Program to the Processor	20
❖ Connecting Your PC to the Module	23

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 CompactLogix or MicroLogix devices to a power source and to the MVI69-DFNT module's application port(s)

1.1 System Requirements

The MVI69-DFNT module requires the following minimum hardware and software components:

- *Rockwell Automation CompactLogix* processors or *MicroLogix 1500 LRP* processor, with compatible power supply and one free slot in the rack, for the MVI69-DFNT module. The module requires 800 mA of available power.

Important: The MVI69-DFNT module has a power supply distance rating of 2 (L43 and L45 installations on first 2 slots of 1769 bus).

Important: For 1769-L23x processors, please make note of the following limitations.

- 1769-L23-QBFC1B = 800 mA at 5 Vdc (One MVI69-DFNT will use all 800 mA of available power. No other modules can be used with an MVI69 module connected to this processor.)
- 1769-L23E-QB1B = 1000 mA at 5 Vdc (One MVI69-DFNT will use 800 mA of available power. One other module can be used on this rack provided it consumes less than 200 mA at 5 Vdc.)
- 1769-L23E-QBFC1B = 450 mA at 5 Vdc (No MVI69 module can be used with this processor.)

- *Rockwell Automation RSLogix 5000 (CompactLogix)* or *RSLogix 500 (MicroLogix)* programming software
- *Rockwell Automation RSLinx* communication software
- *Pentium® II* 450 MHz minimum. *Pentium III* 733 MHz (or better) recommended
- Supported operating systems:
 - *Microsoft Windows XP Professional* with *Service Pack 1* or *2*
 - *Microsoft Windows 2000 Professional* with *Service Pack 1, 2, or 3*
 - *Microsoft Windows Server 2003*
- 128 Mbytes of RAM minimum, 256 Mbytes of RAM recommended
- 100 Mbytes of free hard disk space (or more based on application requirements)
- 256-color VGA graphics adapter, 800 x 600 minimum resolution (True Color 1024 × 768 recommended)
- DVD drive
- HyperTerminal or other terminal emulator program capable of file transfers using Ymodem protocol.

NOTE: MVI69/PS69 modules will not work with CompactLogix L4x processors using RSLogix 5000 v17. All other processor combinations and RSLogix versions will work correctly.

1.2 Package Contents

The following components are included with your MVI69-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	MVI69-DFNT Module	MVI69-DFNT	EtherNet/IP Client/Server Communication Module
1	Cable	Cable #15 - RS232 Null Modem	For RS232 between a Personal Computer (PC) and the CFG port of the module
1	Cable	Cable #14 - RJ45 to DB9 Male Adapter	For connecting the module's port to Cable #15 for RS-232 connections
1	ProSoft Solutions DVD	DVD-001	Contains sample programs, utilities and documentation for the MVI69-DFNT module.

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

1.3 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.

To install ProSoft Configuration Builder from the ProSoft Technology website

- 1 Open your web browser and navigate to <http://www.prosoft-technology.com/pcb>
- 2 Click the link at the *Current Release Version* section to download the latest version of *ProSoft Configuration Builder*.
- 3 Choose **SAVE** or **SAVE FILE** when prompted.
- 4 Save the file to your *Windows Desktop*, so that you can find it easily when you have finished downloading.
- 5 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 DVD*, included in the package with your module.

To install ProSoft Configuration Builder from the Product DVD

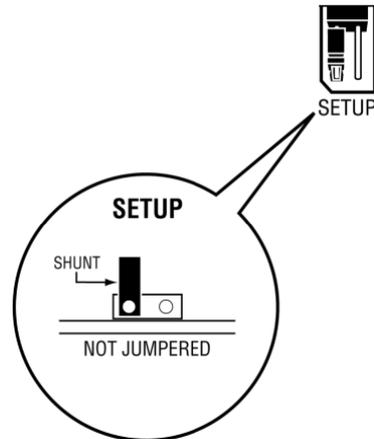
- 1 Insert the *ProSoft Solutions Product DVD* into the DVD drive of your PC. Wait for the startup screen to appear.
- 2 On the startup screen, click **PRODUCT DOCUMENTATION**. This action opens a *Windows Explorer* file tree window.
- 3 Click to open the **UTILITIES** folder. This folder contains all of the applications and files you will need to set up and configure your module.
- 4 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 DVD. You may wish to copy the files from the *Utilities* folder on the DVD to a convenient location on your hard drive.

1.4 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 MVI69-DFNT jumper configuration.



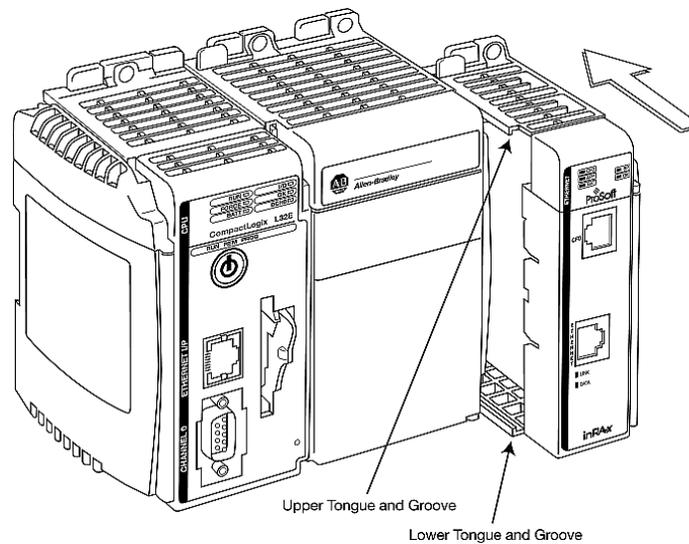
Note: If you are installing the module in a remote rack, you may prefer to leave the Setup pins jumpered. That way, you can update the module's firmware without requiring physical access to the module.

1.5 Installing the Module

Before you attempt to install the module, make sure that the bus lever of the adjacent module is in the unlocked (fully right) position.

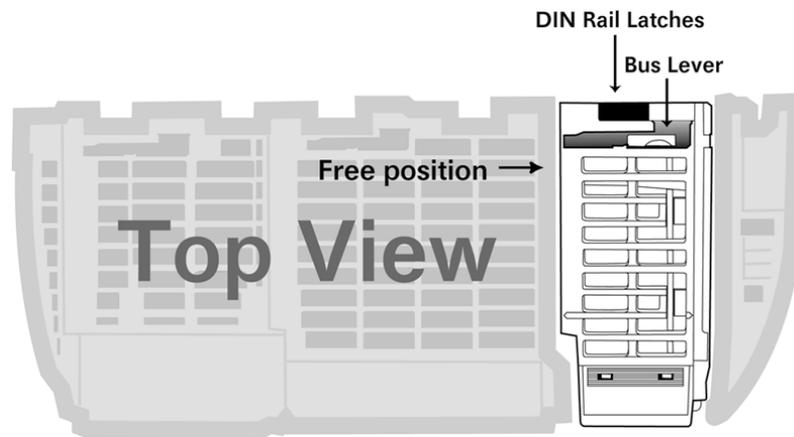
Warning: This module is not hot-swappable! Always remove power from the rack before inserting or removing this module, or damage may result to the module, the processor, or other connected devices.

- 1 Align the module using the upper and lower tongue-and-groove slots with the adjacent module and slide forward in the direction of the arrow.

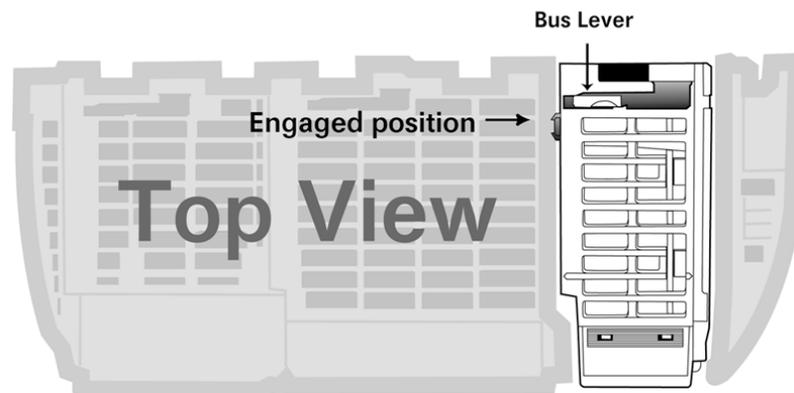


- 2 Move the module back along the tongue-and-groove slots until the bus connectors on the MVI69 module and the adjacent module line up with each other.

- 3 Push the module's bus lever back slightly to clear the positioning tab and move it firmly to the left until it clicks. Ensure that it is locked firmly in place.

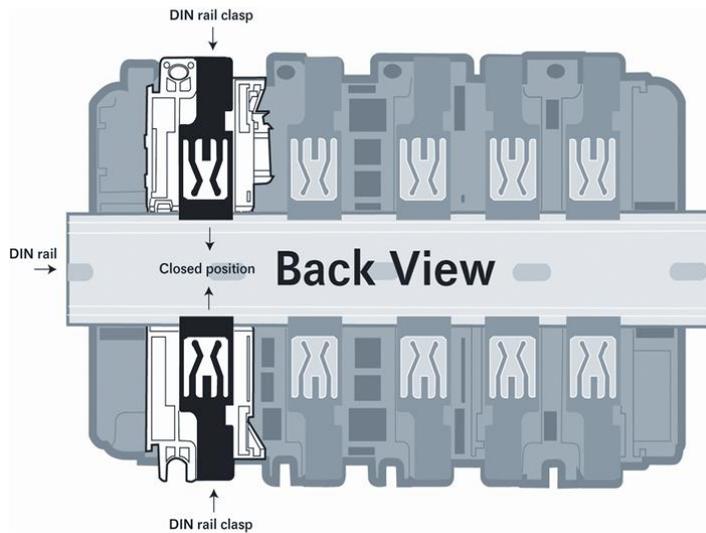
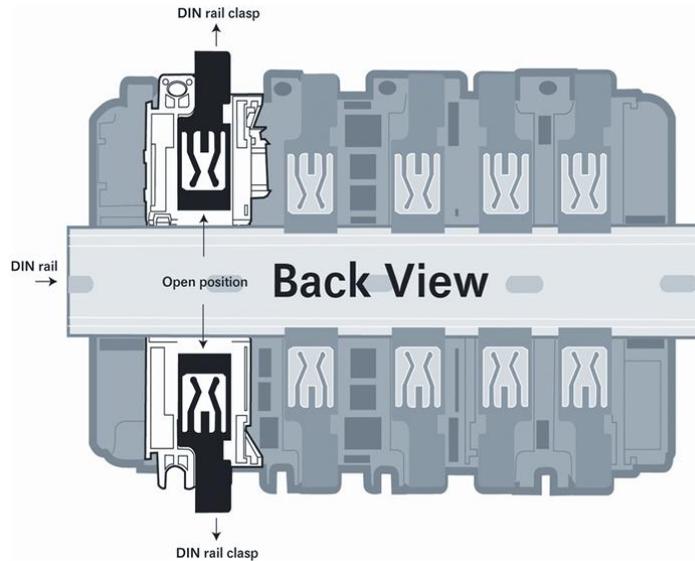


**Move the Bus Lever to the left
until it clicks**



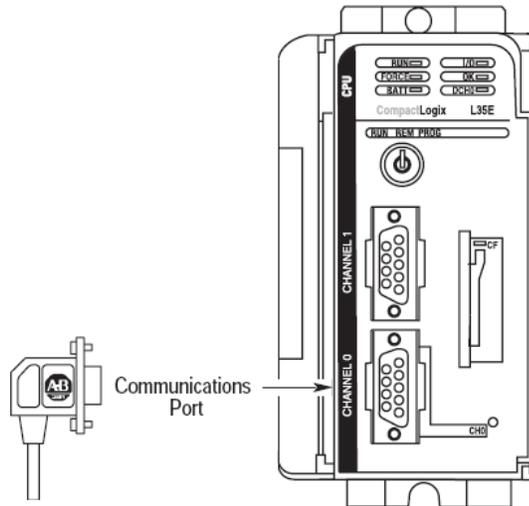
- 4 Close all DIN-rail latches.

- 5 Press the DIN-rail mounting area of the controller against the DIN-rail. The latches will momentarily open and lock into place.

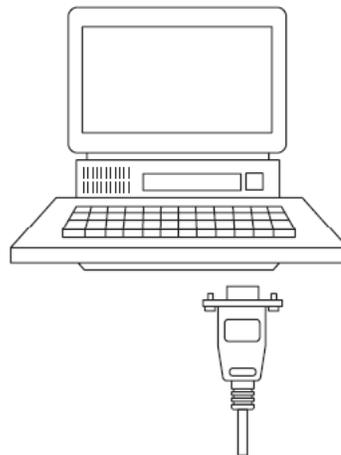


1.6 Connecting Your PC to the Processor

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



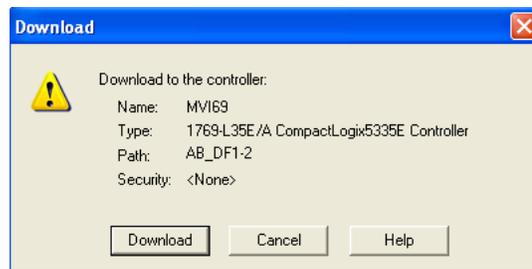
- 2 Connect the straight connector end of the cable to the serial port on your computer.



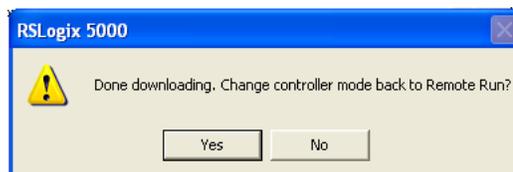
1.7 Downloading the Sample Program to the Processor

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

- 1 If you are not already online to the processor, open the **COMMUNICATIONS** menu, and then choose **DOWNLOAD**. RSLogix will establish communication with the processor.
- 2 When communication is established, RSLogix will open a confirmation dialog box. Click the **DOWNLOAD** button to transfer the sample program to the processor.



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

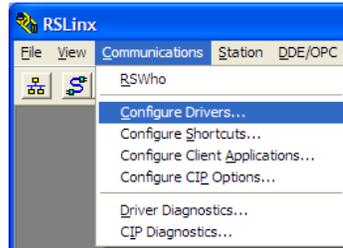


Note: If you receive an error message during these steps, refer to your RSLogix documentation to interpret and correct the error.

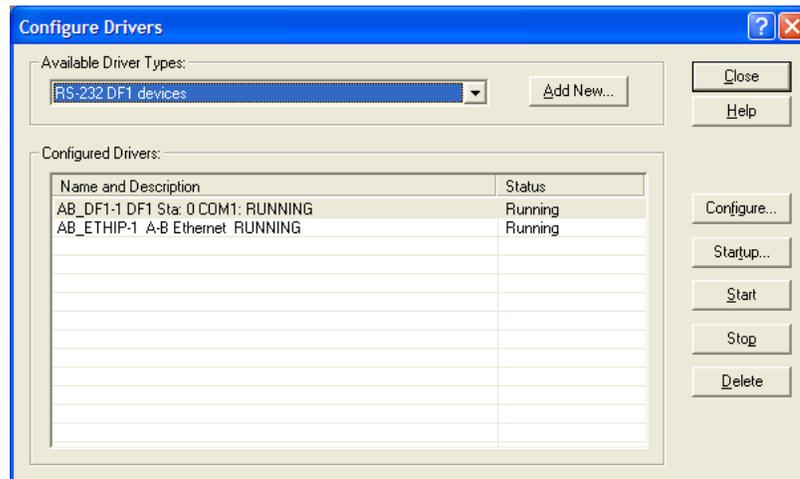
1.7.1 Configuring the RSLinx Driver for the PC COM Port

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

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

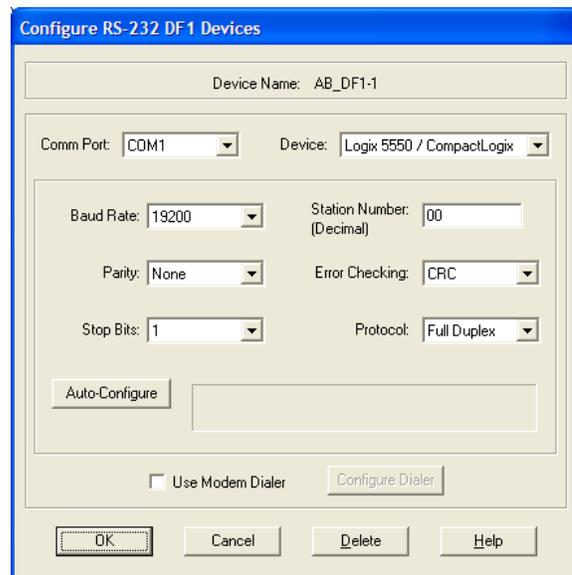


This action opens the *Configure Drivers* dialog box.



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.



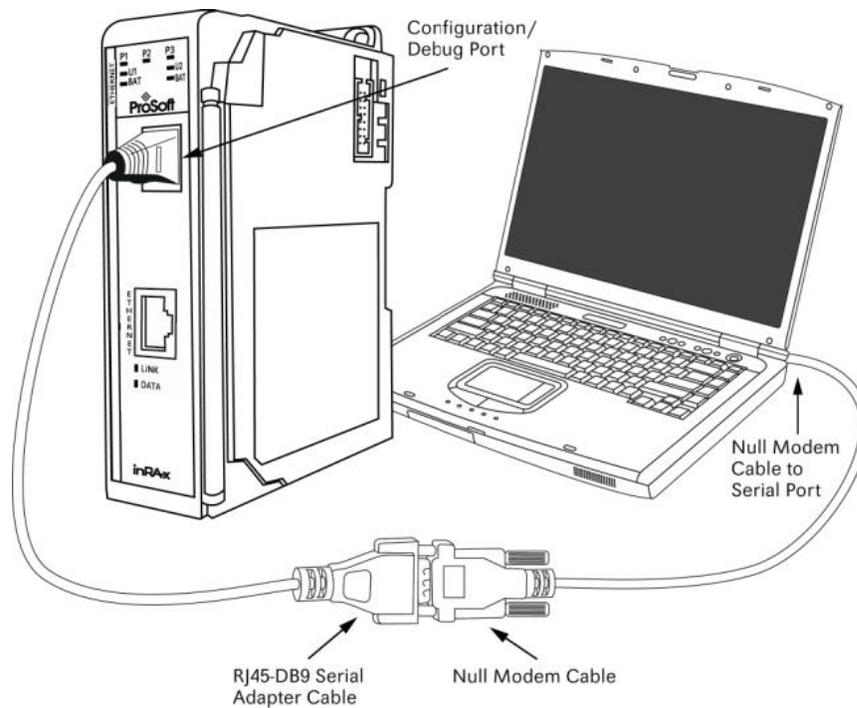
- 4 Click the **AUTO-CONFIGURE** button. RSLinx will attempt to configure your serial port to work with the selected driver.
- 5 When you see the message *Auto Configuration Successful*, click the **OK** button to dismiss the dialog box.

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.

1.8 Connecting Your PC to the Module

With the module securely mounted, connect your PC to the CFG (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 CFG port of the module.
- 3 Attach the other end to the serial port on your PC.



2 Configuring the MVI69-DFNT Module

In This Chapter

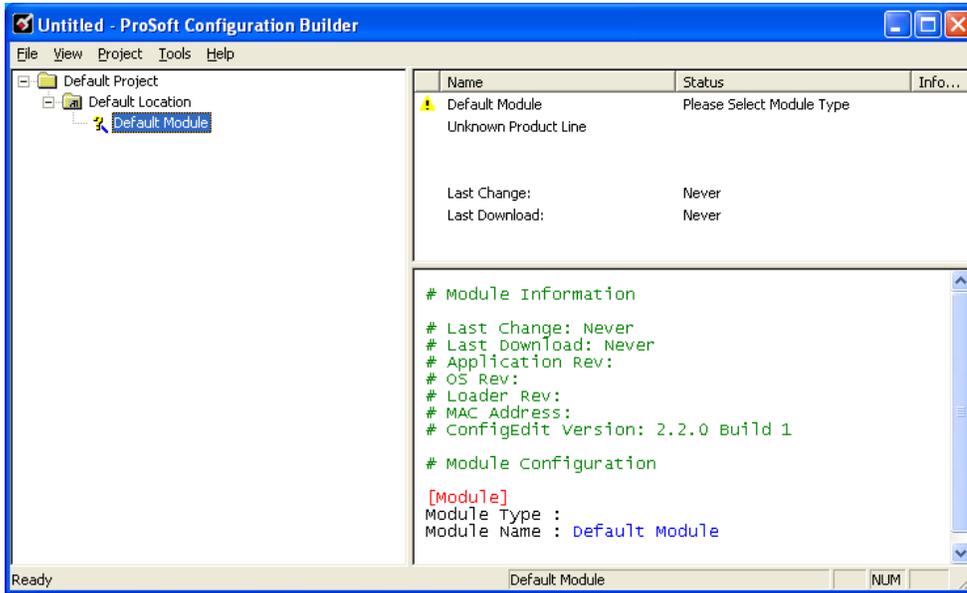
❖ Using ProSoft Configuration Builder.....	25
❖ [Module].....	30
❖ [DFNT Client x].....	32
❖ [DFNT Client x Commands].....	33
❖ Ethernet Configuration	41
❖ Downloading the Project to the Module Using a Serial COM Port.....	42

2.1 Using ProSoft Configuration Builder

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

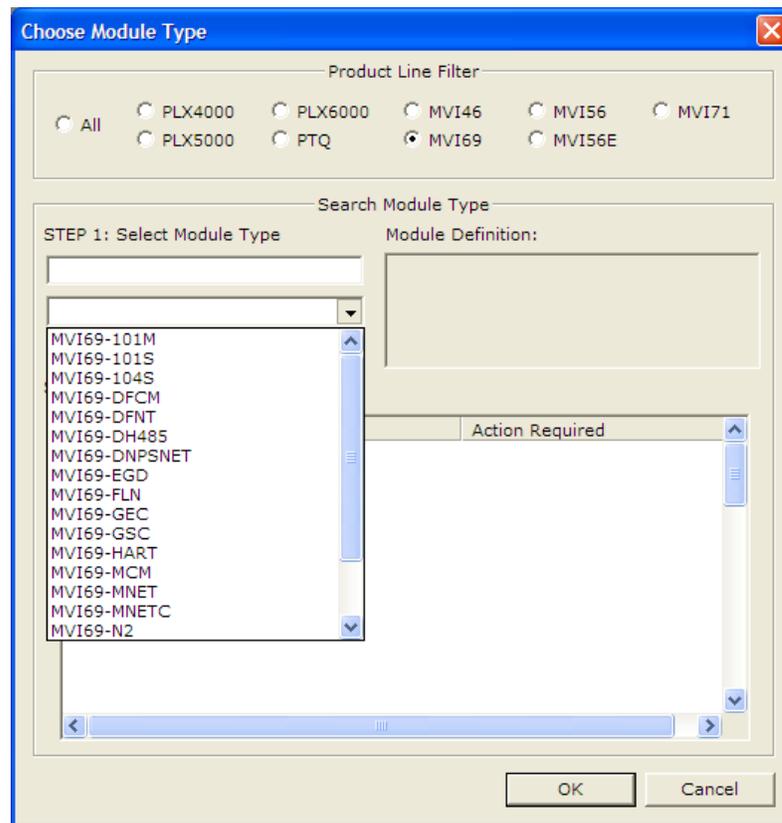
2.1.1 Setting Up the Project

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



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

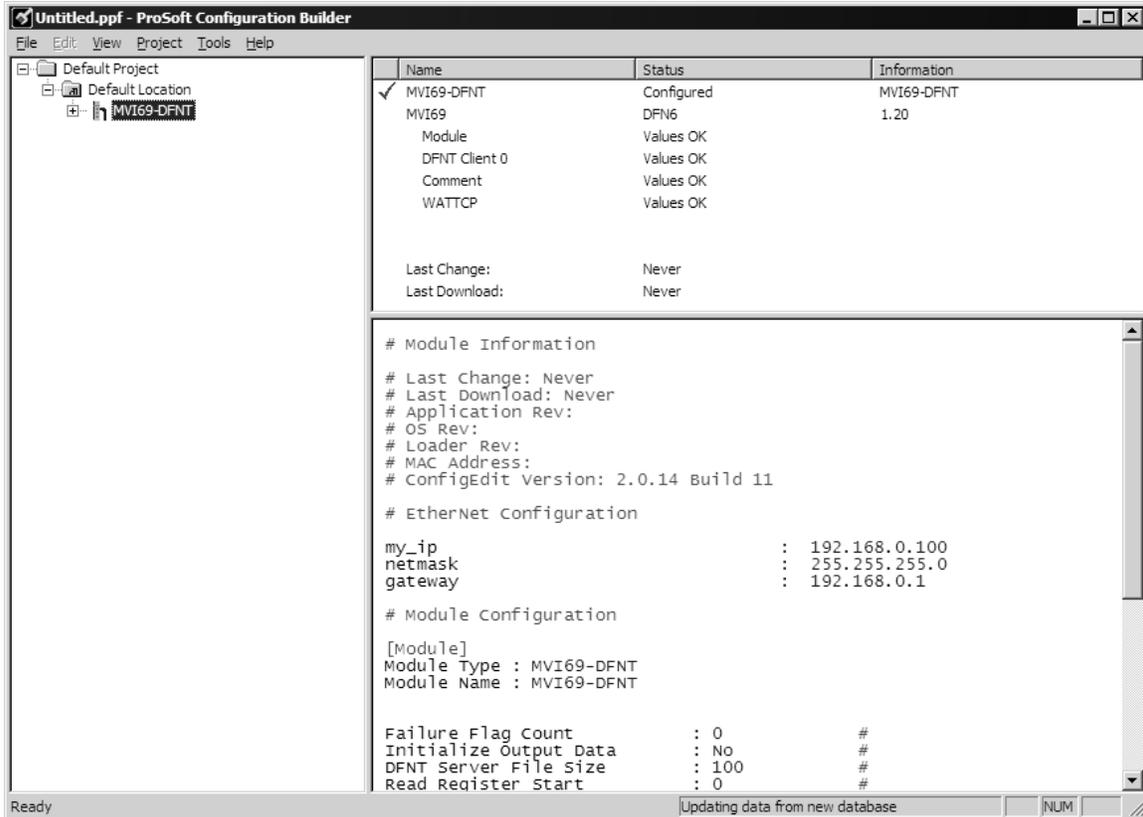
- 2 On the shortcut menu, select **CHOOSE MODULE TYPE**. This action opens the *Choose Module Type* dialog box.



- 3 In the *Product Line Filter* area of the dialog box, select **MVI69**. In the *Select Module Type* dropdown list, select **MVI69-DFNT**, and then click **OK** to save your settings and return to the *ProSoft Configuration Builder* window.

2.1.2 Setting Module Parameters

Notice that the contents of the information pane and the configuration pane changed when you added the MVI69-DFNT module to the project.



At this time, you may wish to rename the *Default Project* and *Default Location* folders in the tree view.

Renaming an Object

- 1 Select the object, and then click the right mouse button to open a shortcut menu. From the shortcut menu, choose **RENAME**.
- 2 Type the name to assign to the object.
- 3 Click away from the object to save the new name.

Configuring Module Parameters

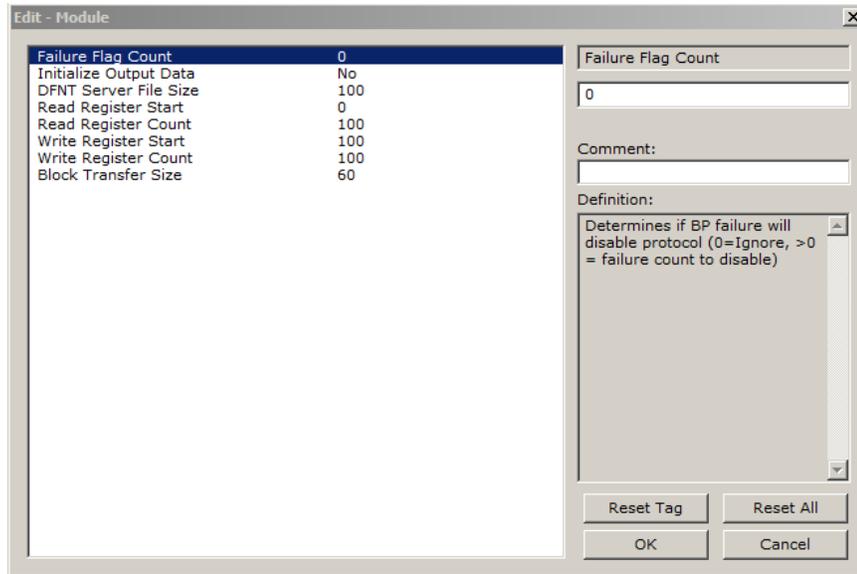
- 1 Click the **[+]** sign next to the module icon to expand module information.
- 2 Click the **[+]** sign next to any  icon to view module information and configuration options.
- 3 Double-click any  icon to open an *Edit* dialog box.
- 4 To edit a parameter, select the parameter in the left pane and make your changes in the right pane.
- 5 Click **OK** to save your changes.

Printing a Configuration File

- 1 Select the module icon, and then click the right mouse button to open a shortcut menu.
- 2 On the shortcut menu, choose **VIEW CONFIGURATION**. This action opens the *View Configuration* window.
- 3 In the *View Configuration* window, open the **FILE** menu, and choose **PRINT**. This action opens the *Print* dialog box.
- 4 In the *Print* dialog box, choose the printer to use from the drop-down list, select printing options, and then click **OK**.

2.2 [Module]

This section describes how to initialize the module upon startup.



2.2.1 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.

2.2.2 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. Setting this option to **YES** requires associated ladder logic to pass the data from the processor to the module.

2.2.3 DFNT Server File Size

100 or 1000

Sets the maximum file size (**100** or **1000**) for the servers. Servers emulate PCCC-type data table addressing. Use **100** to simulate SLC-type data tables, which are size-limited to 256 elements per table. Use **1000** to simulate PLC-type data tables, which are size-limited to 1000 elements per table. For use with ControlLogix or CompactLogix systems, either size may be used.

2.2.4 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. This will be used for data brought into the processor from other nodes on the network.

2.2.5 Read Register Count

0 to 3960

This parameter specifies the number of registers to be transferred from the module to the processor. This will be used for data brought into the processor from other nodes on the network.

2.2.6 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. This will be used for data to be sent from the processor to other nodes on the network.

2.2.7 Write Register Count

0 to 3960

This parameter specifies the number of registers to transfer from the processor to the module. This will be used for data to be sent from the processor to other nodes on the network.

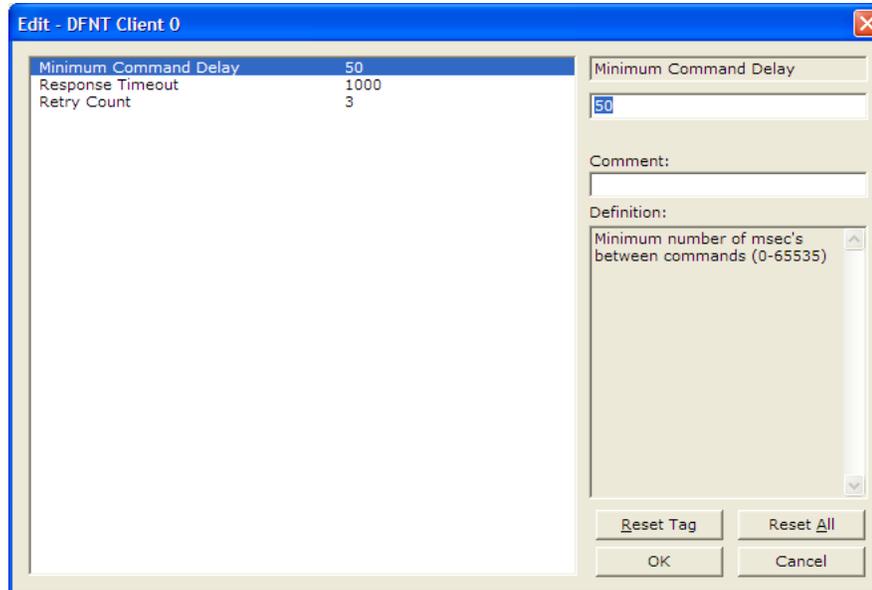
2.2.8 Block Transfer Size

60, 120 or 240

This read-only parameter specifies the number of words of user data transferred in each block between the module and processor.

2.3 [DFNT Client x]

This section defines the configuration for the DFNT Client (master) device simulated on network port



2.3.1 *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.

2.3.2 *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.

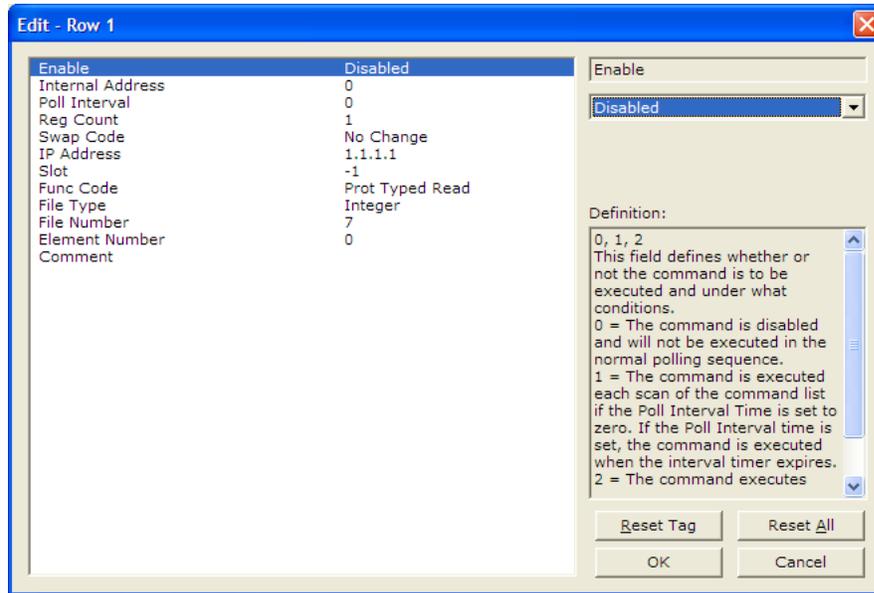
2.3.3 *Retry Count*

0 to 10

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

2.4 [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.



2.4.1 Command List

In order to interface the virtual database with Rockwell Automation Programmable Automation Controllers (PACs), Programmable Logic Controllers (PLCs), or other EtherNet/IP server devices, you must construct a command list.

The commands in the list specify:

- The message type to be utilized to access the server device
- The registers in the module's virtual database to be used as the source of data sent or destination of data received
- The polling interval of each command
- The number of data points to be read or written
- The byte-order of the data, controlled by a swap code
- The Ethernet IP address of the target server device
- The chassis slot number of the target server device
- The command and function code of the operation to be performed (read or write)
- The data area to access in the server device

There is a separate command list for each of the different message types supported by the driver. Each list is processed from top to bottom, one after the other, until all specified commands are completed, and then the polling process begins over again.

A poll interval parameter is included in 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 once every 10 seconds for the DF1 serial implementation and no more frequently than once per second for the DFNT Ethernet implementation.

Write commands can be set to execute on a continuous repeating basis or execute only if the data in the database area associated with the write command changes (called a "Conditional Write"). When configured as a "Conditional Write", if the data for the command has not changed since the command was last issued, the command will not be executed during the current poll cycle. If the data for the command has changed since the command was last issued, the command will be executed. Use this feature to reduce unnecessary network traffic and optimize network bandwidth usage. In order to implement this feature; set the enable code for the command to a value of 2.

The module supports numerous DF1 protocol command/function combinations. This permits the module to interface with a wide variety of DF1 protocol devices. This includes PLC2, PLC3, PLC5, SLC500, MicroLogix, CompactLogix, and ControlLogix processors. Additionally, other devices that use the DF1 and EtherNet/IP protocols 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.

2.4.2 Command Entry Formats

The format of each command in the list depends on the function being executed. Refer to Command Function Codes (page 36) for a complete discussion of the commands supported by the module and of the structure and content of each command.

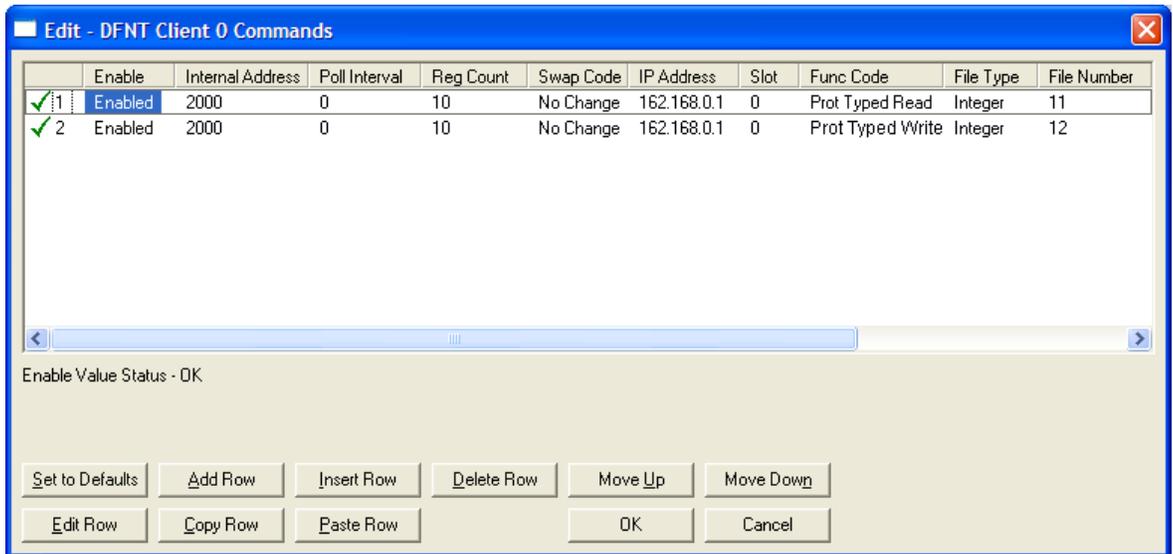
The following table shows the structure of the configuration data necessary for each of the supported commands.

Module Information Data							Device Information Data					
Column #	1	2	3	4	5	6	7	8	9	10	11	12
Function Code	Enable Code	Internal Addresses	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

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 511	Code	Register	Seconds	Count	Code	Node		511	File Type	File Number	Element	Sub-Element

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

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

Parameter	Range	Description								
Enable	0, 1, 2	This field defines whether the command is to be executed and under what conditions. <table border="1" data-bbox="609 982 1279 1276"> <thead> <tr> <th>Value</th> <th>Description</th> </tr> </thead> <tbody> <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> </tbody> </table>	Value	Description	0	The command is disabled and will not be executed in the normal polling sequence.	1	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.	2	The command executes only if the internal data associated with the command changes. This value is valid for write commands only.
Value	Description									
0	The command is disabled and will not be executed in the normal polling sequence.									
1	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.									
2	The command executes only if the internal data associated with the command changes. This value is valid for write commands only.									
Internal Address	0 to 3999	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.								
Poll Delay	0 to 1000	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.								
Count	Command dependent.	This parameter specifies the number of registers or digital points to be associated with the command. See Command Function Codes (page 36) for information								

Parameter	Range	Description										
Swap Code	0, 1, 2, 3	<p>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:</p> <table border="1"> <thead> <tr> <th>Swap Code</th> <th>Description</th> </tr> </thead> <tbody> <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> </tbody> </table> <p>The words should be swapped only when using an even number of words.</p>	Swap Code	Description	0	None - No Change is made in the byte ordering (1234 = 1234)	1	Words - The words are swapped (1234=3412)	2	Words & Bytes - The words are swapped then the bytes in each word are swapped (1234=4321)	3	Bytes - The bytes in each word are swapped (1234=2143)
Swap Code	Description											
0	None - No Change is made in the byte ordering (1234 = 1234)											
1	Words - The words are swapped (1234=3412)											
2	Words & Bytes - The words are swapped then the bytes in each word are swapped (1234=4321)											
3	Bytes - The bytes in each word are swapped (1234=2143)											
Node IP Address	xxx.xxx.xxx.xxx	The IP address of the device being addressed by the command.										
Slot		<p>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.</p> <p>Use a value of -1 when interfacing to an SLC 5/05 or a PLC5. These devices do not have a slot number.</p>										

Parameter	Range	Description																																
Function Code	See Command Function Codes (page 36)	<p>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.</p> <p>Function Code Listing</p> <p>Basic Command Set</p> <table border="1"> <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> <p>PLC-5 Command Set (0x0F)</p> <table border="1"> <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> <p>SLC Command Set (0x0F)</p> <table border="1"> <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>	1	Protected Write	2	Unprotected Read	3	Protected Bit Write	4	Unprotected Bit Write	5	Unprotected Write	100	Word Range Write (Binary Address)	101	Word Range Read (Binary Address)	102	Read-Modify-Write (Binary Address)	150	Word Range Write (ASCII Address)	151	Word Range Read (ASCII Address)	152	Read-Modify-Write (ASCII Address)	501	Prot Typed Read with 2 addr fields	502	Prot Typed Read with 3 addr fields	509	Prot Typed Write with 2 addr fields	510	Prot Typed Write with 3 addr fields	511	Prot Type Write with Mask 3 addr field
1	Protected Write																																	
2	Unprotected Read																																	
3	Protected Bit Write																																	
4	Unprotected Bit Write																																	
5	Unprotected Write																																	
100	Word Range Write (Binary Address)																																	
101	Word Range Read (Binary Address)																																	
102	Read-Modify-Write (Binary Address)																																	
150	Word Range Write (ASCII Address)																																	
151	Word Range Read (ASCII Address)																																	
152	Read-Modify-Write (ASCII Address)																																	
501	Prot Typed Read with 2 addr fields																																	
502	Prot Typed Read with 3 addr fields																																	
509	Prot Typed Write with 2 addr fields																																	
510	Prot Typed Write with 3 addr fields																																	
511	Prot Type Write with Mask 3 addr field																																	
Function Parameters	See Command Function Codes (page 36)	The number of auxiliary parameters required depends on the function code selected for the command.																																

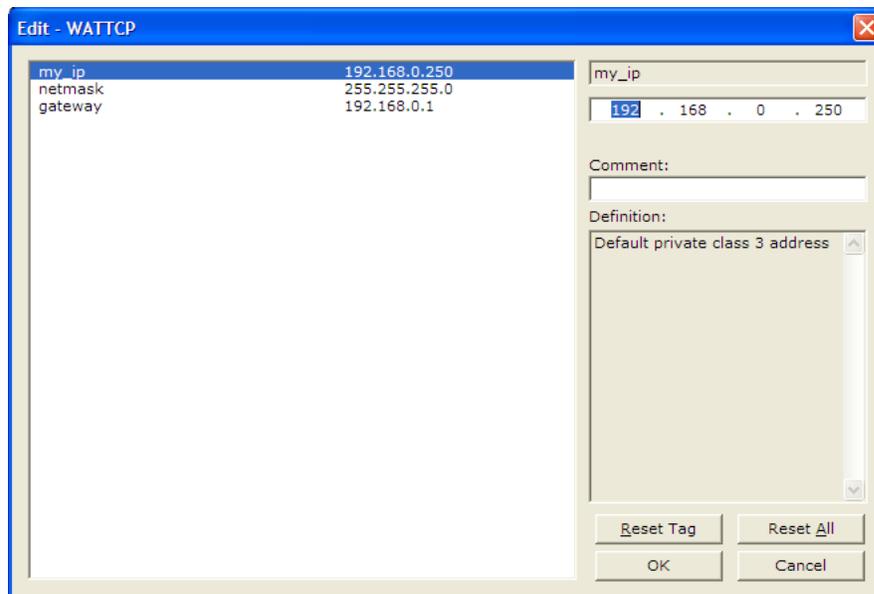
2.5 Ethernet Configuration

Use this procedure to configure the Ethernet settings for your module. You must assign an IP address, subnet mask and gateway address. After you complete this step, you can connect to the module with an Ethernet cable.

- 1 Determine the network settings for your module, with the help of your network administrator if necessary. You will need the following information:
 - IP address (fixed IP required) _____ . _____ . _____ . _____
 - Subnet mask _____ . _____ . _____ . _____
 - Gateway address _____ . _____ . _____ . _____

Note: The gateway address is optional, and is not required for networks that do not use a default gateway.

- 2 Double-click the **ETHERNET CONFIGURATION** icon. This action opens the *Edit* dialog box.



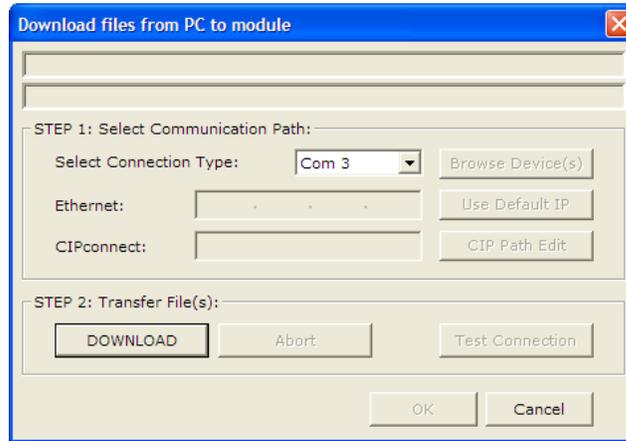
- 3 Edit the values for *my_ip*, *netmask* (subnet mask) and *gateway* (default gateway).
- 4 When you are finished editing, click **OK** to save your changes and return to the *ProSoft Configuration Builder* window.

2.6 Downloading the Project to the Module Using a Serial COM Port

For the module to use the settings you configured, you must download (copy) the updated *Project* file from your PC to the module.

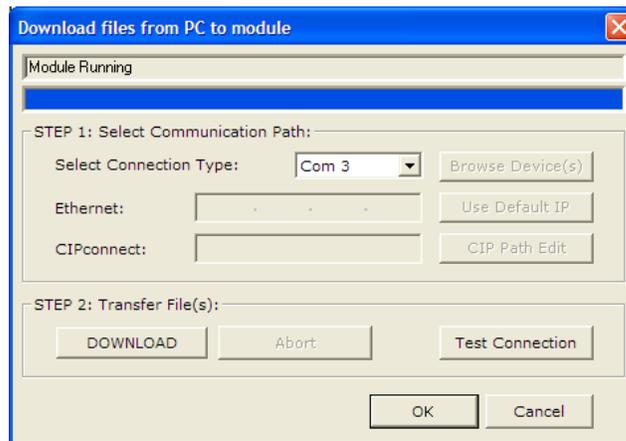
To download the project file

- 1 In the tree view in *ProSoft Configuration Builder*, click once to select the module.
- 2 Right-click the module icon to open a shortcut menu. From the shortcut menu, choose **DOWNLOAD FROM PC TO DEVICE**. The program will scan your PC for a valid com port (this may take a few seconds). When *PCB* has found a valid COM port, the *Download* dialog box will open.



- 3 Choose the COM port to use from the dropdown list, and then click the **DOWNLOAD** button.

The module will perform a platform check to read and load its new settings. When the platform check is complete, the status bar in the *Download* dialog box will display the message *Module Running*.



3 Ladder Logic

In This Chapter

- ❖ Module Data 43
- ❖ Using the RSLogix 5000 v16 Add-On Instruction 44
- ❖ Adding the Module to an Existing CompactLogix Project 51
- ❖ Adding the Module to an Existing MicroLogix Project..... 55

Ladder logic is required for the MVI69-DFNT module to work. 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 *inRAX DVD*, is extensively commented, to provide information on the purpose and function of each rung. For most applications, the sample ladder will work without modification.

For applications using RSLogix™ 5000, versions 16 and later, some sample ladder files include Add-On Instruction (AOI) samples and rung import files. These AOI samples compress all logic needed to use the module into an easily-managed, single logic rung, which can be used as the basis for creating your own custom application or can be quickly added to an existing application.

3.1 Module Data

All data related to the MVI69-DFNT used in Add-On Instruction (AOI) logic are stored in Controller Tag variables based on User-defined Data Type (UDT) structures. See the sample AOI program for details.

3.2 Using the RSLogix 5000 v16 Add-On Instruction

Important: If you are using an older version of RSLogix 5000 (version 15 or older), you will not be able to use the Add-On Instruction. Please see the ladder logic sample files on the distribution DVD for standard ladder logic examples.

If you have RSLogix 5000 version 16 or newer, you can use an Add-On Instruction to simplify the task of configuring the module, either as a new application, or within an existing application.

The ProSoft Solutions DVD included in the package with the module contains ladder logic, product manuals, and utility programs for all ProSoft Technology products.

Copy the manuals and sample program from the DVD

- 1 Insert the ProSoft Solutions DVD into the DVD drive of your PC. Wait for the startup screen to appear.
- 2 On the startup screen, click *Product Documentation*. This action opens an Explorer window. Files are arranged by type:
 - The Ladder Logic folder contains sample programs for each module, arranged by processor type, and then by product name. The sample programs for your module are in the CompactLogix or MicroLogix/MVI69/MVI69-DFNT folder.
 - The Manuals folder contains product manuals and datasheets in Adobe Acrobat Reader format (PDF) for each module, arranged in the same way as the Ladder Logic folder.
 - The Utilities folder contains additional programs and tools required for some ProSoft modules. Refer to your user manual to determine if you need to use or install any of these additional tools.
- 3 In the Explorer window, navigate to the files you need, and then copy them to a location on your hard drive.

Download the manuals and sample program from the ProSoft Technology web site

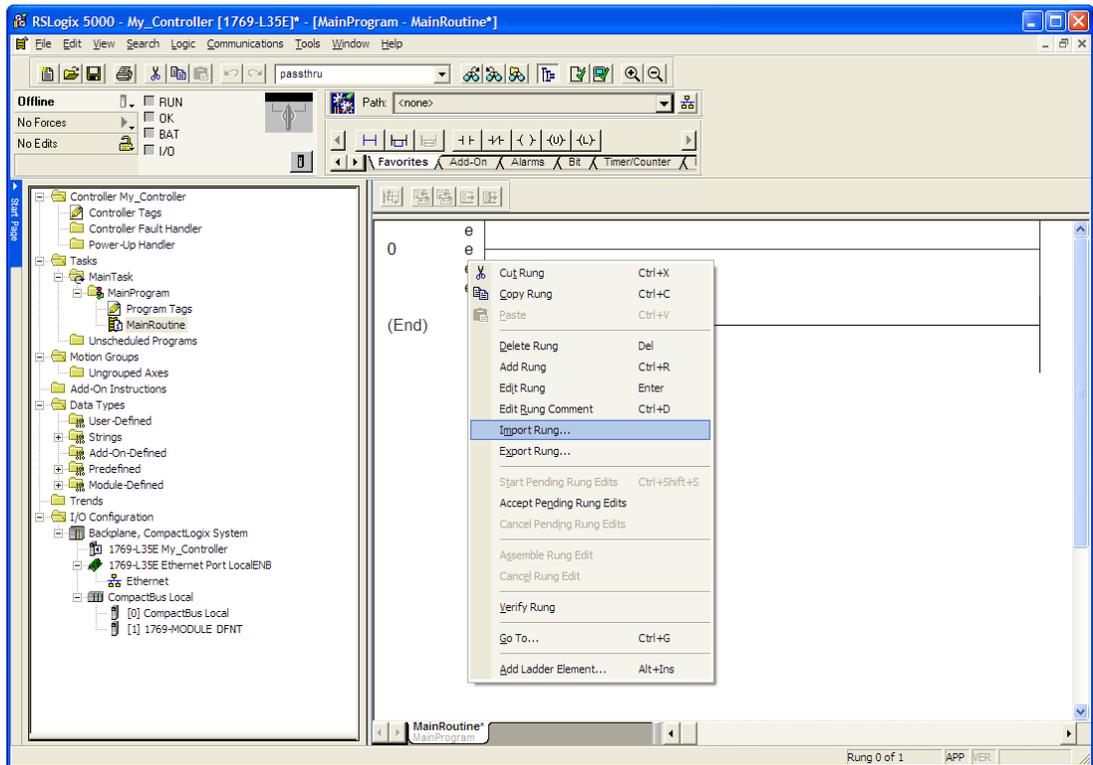
You can always download the latest version of the sample ladder logic and user manuals for the MVI69-DFNT module from the ProSoft Technology web site, at www.prosoft-technology.com/support/downloads

From that link, navigate to the download page for your module and choose the sample ladder program to download for your version of RSLogix 5000 and your processor.

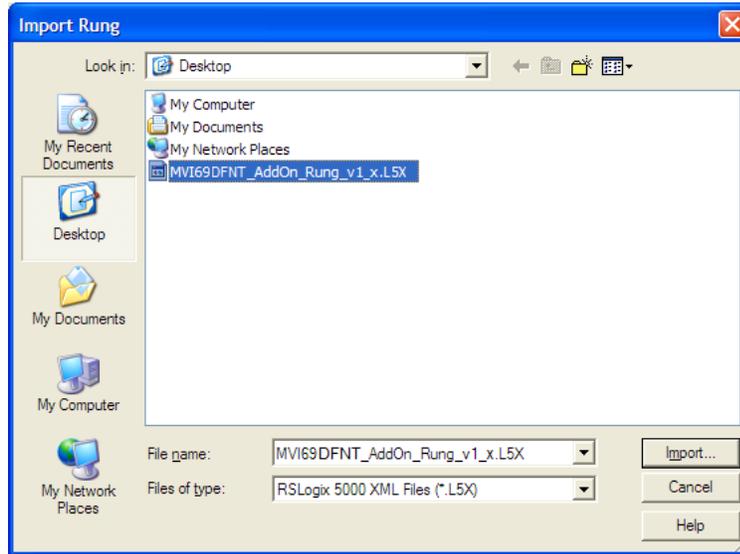
3.2.1 Importing the Ladder Rung - MVI69-DFNT

The Add-On Instruction (AOI) is supplied already installed as part of a complete sample .ACD program file and also as an .L5X ladder import file that can be imported into an existing or new application. Importing the AOI reduces module setup time and simplifies your application logic. The .L5X file automatically imports all necessary components into an application.

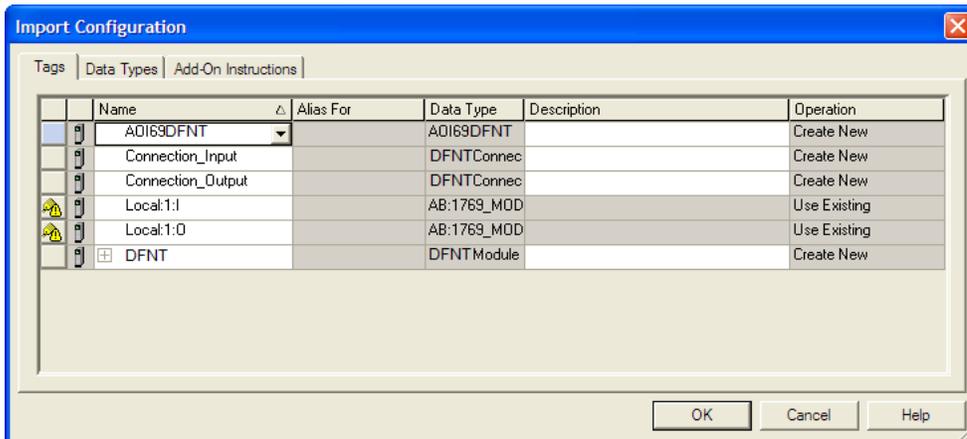
- 1 Expand the **TASKS** folder, and then expand the **MAIN TASK** folder.
- 2 Double-click to open the **MAINROUTINE** ladder.
- 3 Select an empty rung, and then click the right mouse button to open a shortcut menu. On the shortcut menu, choose **IMPORT RUNG**.



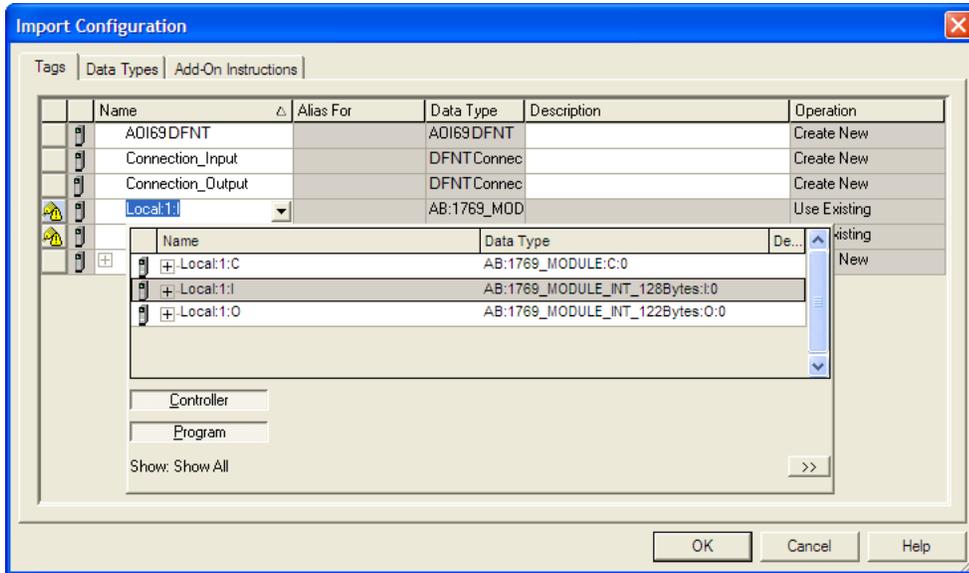
4 Select the **MVI69DFNT_ADDON_RUNG_VX_Y.L5X** file.



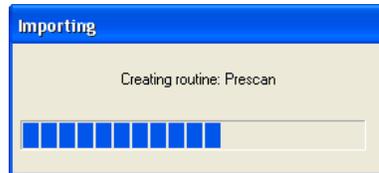
This action opens the *Import Configuration* dialog box, showing the controller tags to be created during the import procedure.



- If you are using the module in a different slot, select the correct connection input and output variables associated to the module. If your module is located in Slot 1, this step is not required.



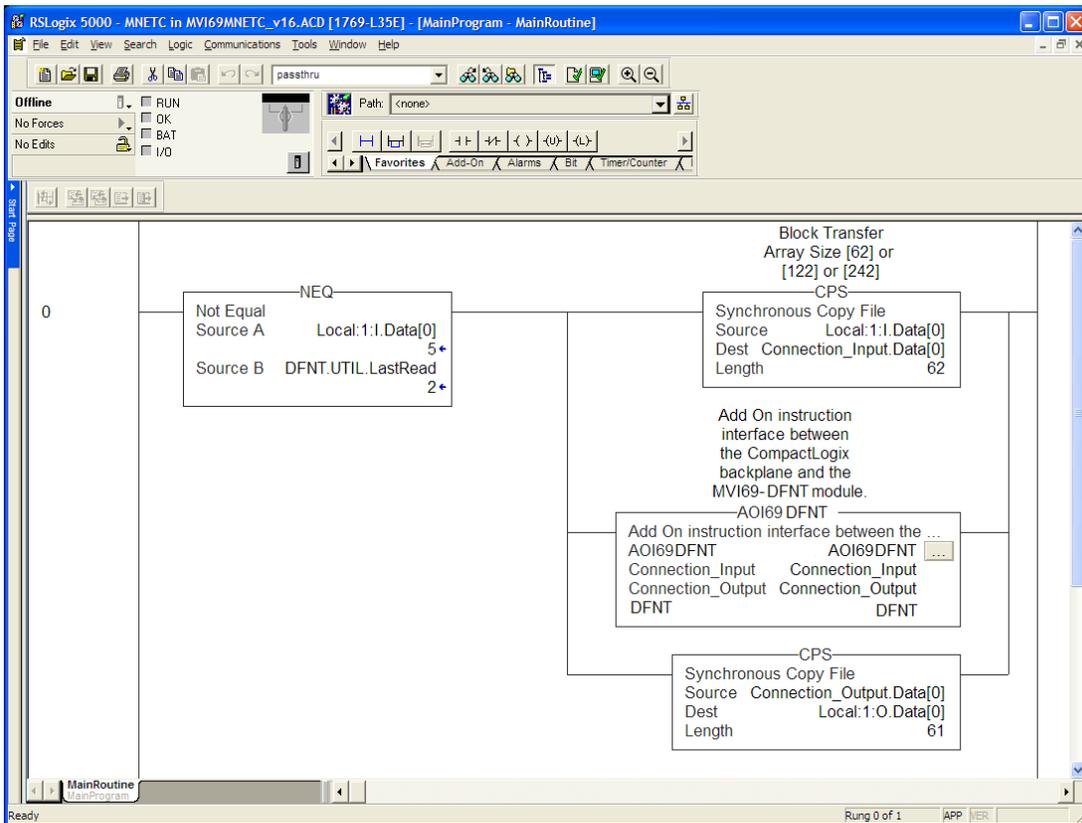
- Click **OK** to confirm the import. RSLogix will indicate that the import is under progress:



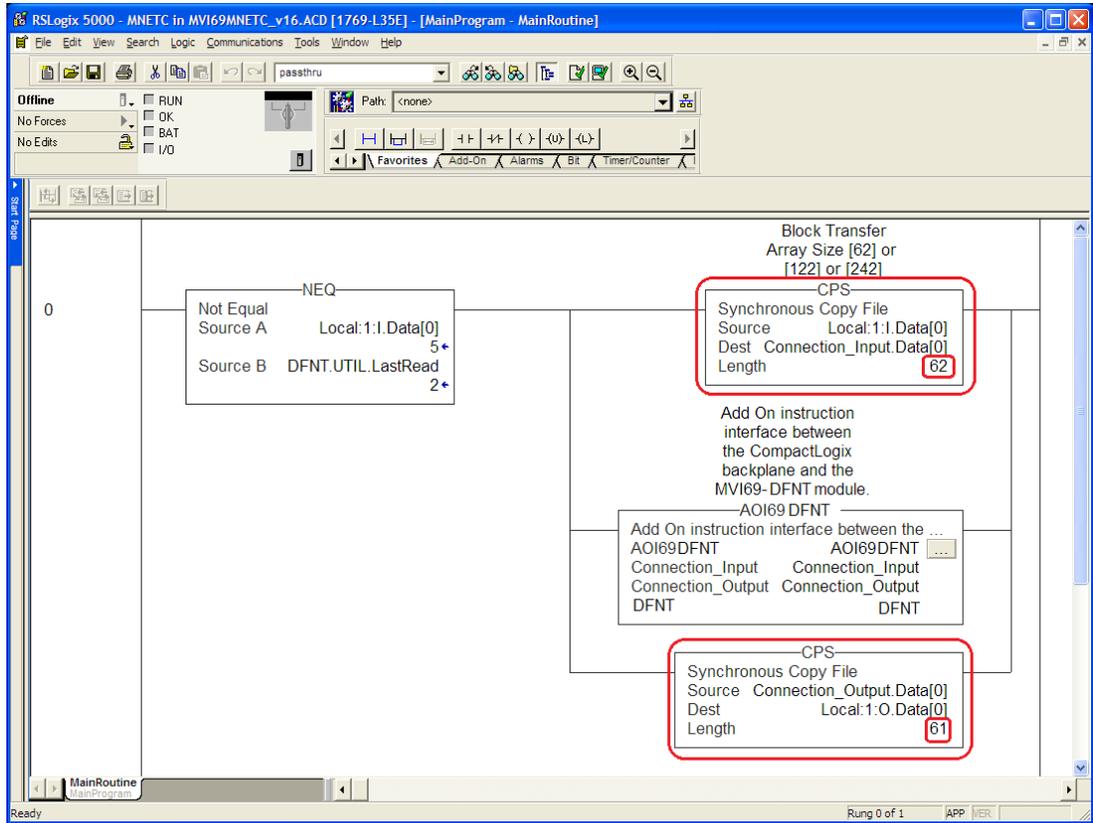
When the import is completed, the new rung with the Add-On instruction will be visible as shown in the following illustration.



The procedure has also imported new user-defined data types, data objects and the Add-On instruction to be used at your project.



- 7 The imported rung will contain the Add-On instruction with two CPS instructions as follows below. The CPS instructions are set by default for a length of 62/61 words.



Edit the above CPS instructions Length field values according to the following table.

**"Block Transfer Size" Ladder Routine window:
 Parameter
 (60/120/240 options)**

Connection Parameters: CPS instructions Length field values:

Block Transfer Size	Input Size	Output Size	CPS Synchronous Copy File Source Local:1:I.Data[0] Dest Connection_Input.Data[0] Length 62	CPS Synchronous Copy File Source Connection_Output.Data[0] Dest Local:1:O.Data[0] Length 61
60	62	61	62	61
120	122	121	122	121
240	242	241	242	241

- 8 Once the CPS statements have been modified, open the Controller Tags window and set the tag, **DFNT.CONFIG.BLOCKTRANSFERSIZE** to match the CPS settings (60, 120, or 240).

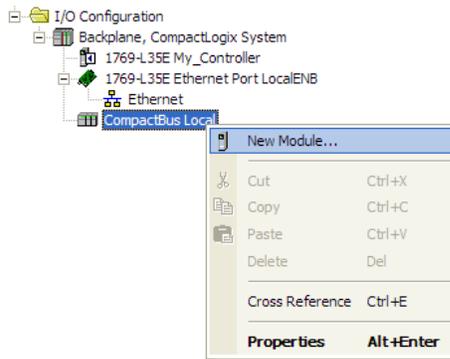
[-] DFNT	{...}
[-] DFNT.CONFIG	{...}
+ DFNT.CONFIG.BlockTransferSize	60

3.3 Adding the Module to an Existing CompactLogix Project

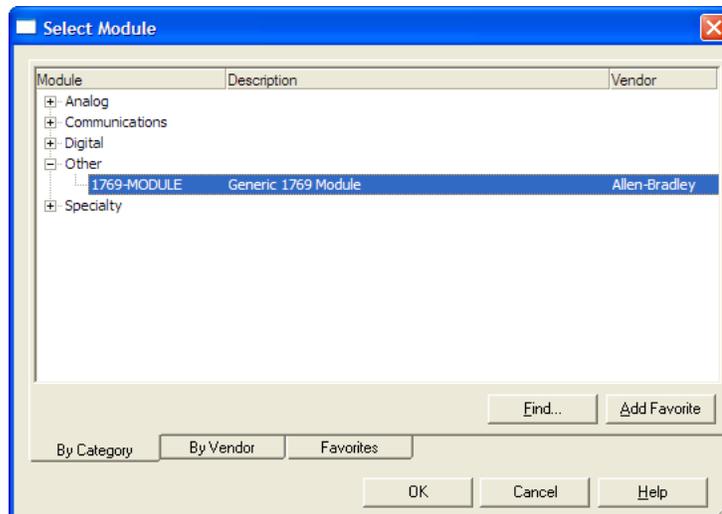
Important: The MVI69-DFNT module has a power supply distance rating of 2 (L43 and L45 installations on first 2 slots of 1769 bus, to the right of the processor).

If you are installing and configuring the module with a CompactLogix controller, and you decide to use the standard sample ladder logic rather than importing the Add-On Instruction (AOI) Rung, then follow these steps. If you are using a MicroLogix controller, refer to the next section (page 55).

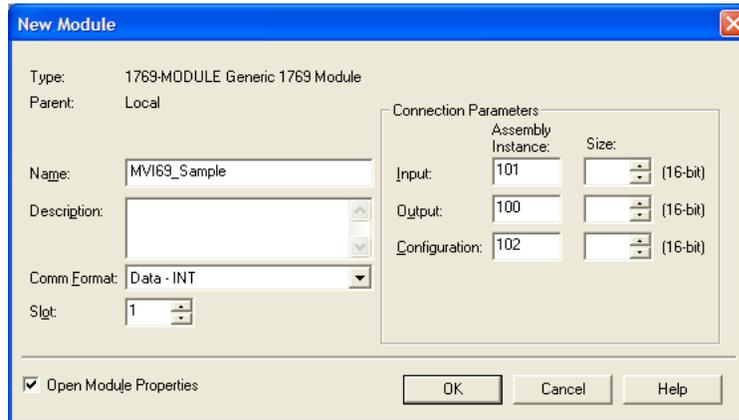
- 1 Add the MVI69-DFNT module to the project. Right-click the mouse button on the **I/O CONFIGURATION** option in the *Controller Organization* window to display a pop-up menu. Select the **NEW MODULE** option from the **I/O CONFIGURATION** menu.



This action opens the *Select Module* dialog box:



- 2 Select the *1769-Module* (Generic 1769 Module) from the list and click **OK**.



- 3 Enter the *Name*, *Description* and *Slot* options for your application, using the values in the illustration above. You must select the *Comm Format* as **DATA - INT** in the dialog box, otherwise the module will not communicate over the backplane of the CompactLogix rack.
- 4 Configure the *Connection Parameters* to match to the *Block Transfer Size* parameter in the configuration file. Use the values in the table corresponding with the block transfer size you configured.

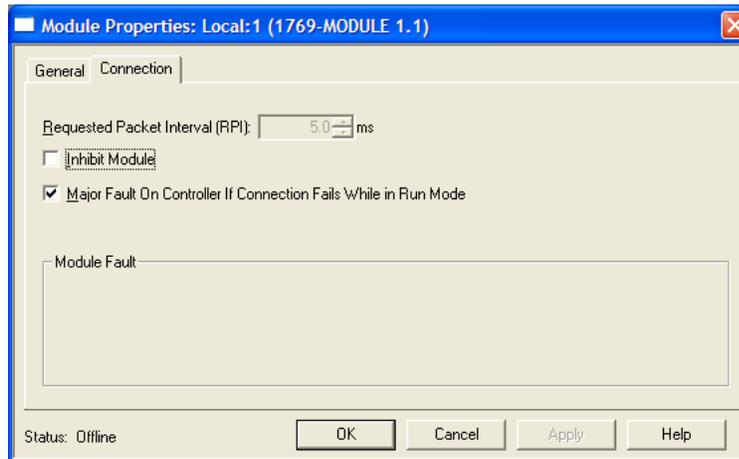
Block Transfer Size = 60

Field	Recommended Value
Type	1769-MODULE Generic 1769 Module
Parent	Local
Name	MVI69
Description	MVI69 Application Module
Comm Format	Data - INT
Slot	The slot number in the rack where the module is installed
Input Assembly Instance	101
Input Size	62
Output Assembly Instance	100
Output Size	61
Configuration Assembly Instance	102
Configuration Size	0

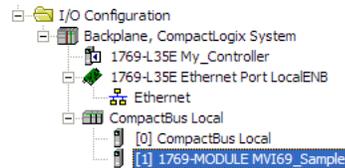
Block Transfer Size = 120	
Field	Recommended Value
Type	1769-MODULE Generic 1769 Module
Parent	Local
Name	MVI69
Description	MVI69 Application Module
Comm Format	Data - INT
Slot	The slot number in the rack where the module is installed
Input Assembly Instance	101
Input Size	122
Output Assembly Instance	100
Output Size	121
Configuration Assembly Instance	102
Configuration Size	0

Block Transfer Size = 240	
Field	Recommended Value
Type	1769-MODULE Generic 1769 Module
Parent	Local
Name	MVI69
Description	MVI69 Application Module
Comm Format	Data - INT
Slot	The slot number in the rack where the module is installed
Input Assembly Instance	101
Input Size	242
Output Assembly Instance	100
Output Size	241
Configuration Assembly Instance	102
Configuration Size	0

- 5 Click **NEXT** to continue.



- 6 Select the *Requested Packet Interval* value for scanning the I/O on the module. This value represents the minimum frequency at which the module will handle scheduled events. It should not be set to less than 1 millisecond. Values between 5 and 10 milliseconds should work with most applications.
- 7 **SAVE** the module. Click **OK** to dismiss the dialog box. The *Controller Organization* window now displays the module's presence. The following illustration shows the *Controller Organization* window:



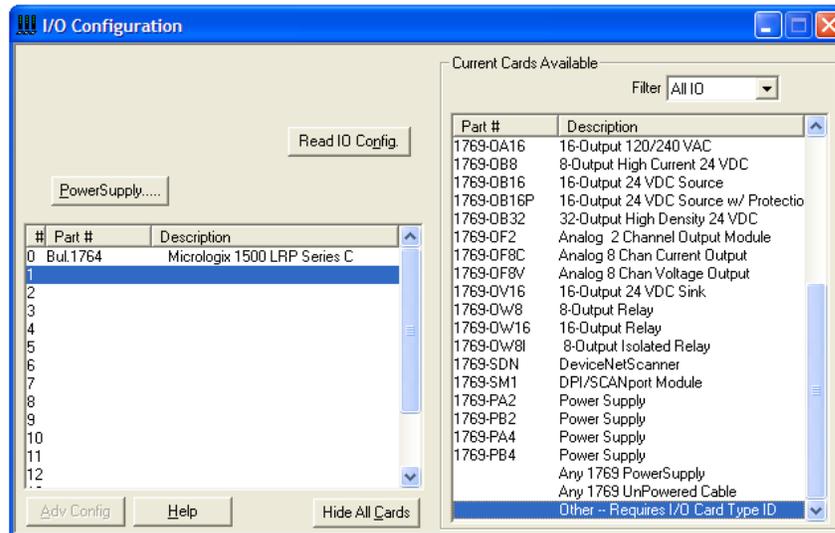
- 8 Copy the *User-defined Data Types* (UDTs) from the sample program.
- 9 Copy the *Controller Tags* from the sample program.
- 10 Copy the *Ladder Rungs* from the sample program.
- 11 Save and download the new application to the controller.

3.4 Adding the Module to an Existing MicroLogix Project

If you are installing and configuring the module with a MicroLogix controller, follow these steps. If you are using a CompactLogix controller, refer to the previous section.

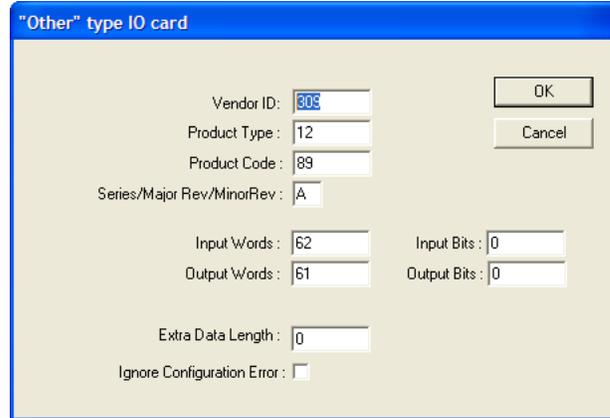
The first step in setting up the processor ladder file is to define the I/O type module to the system. Start RSLogix 500, and follow these steps:

- 1 In RSLogix, open your existing application, or start a new application, depending on your requirements.
- 2 Double-click the **I/O CONFIGURATION** icon located in the *Controller* folder in the project tree. This action opens the *I/O Configuration* dialog box.



- 3 In the *I/O Configuration* dialog box, select "**OTHER - REQUIRES I/O CARD TYPE ID**" at the bottom of the list in the right pane, and then double-click to open the "*Other*" type IO card dialog box.

- 4 Enter the values shown in the following illustration to define the module correctly for the MicroLogix processor, and then click **OK** to save your configuration.



The *Input Words* and *Output Words* parameters will depend on the *Block Transfer Size* parameter you specify in the configuration file. Use the values from the following table.

Block Transfer Size	Input Words	Output Words
60	62	61
120	122	121
240	242	241

- 5 Click **OK** to continue.
- 6 After completing the module setup, the *I/O Configuration* dialog box will display the module's presence.

The last step is to add the ladder logic. If you are using the example ladder logic, adjust the ladder to fit your application.

Download the new application to the controller. If you encounter errors when you attempt to run the program, refer to *Diagnostics and Troubleshooting* (page 57) for information on how to connect to the module's Config/Debug port to use its troubleshooting features.

4 Diagnostics and Troubleshooting

In This Chapter

- ❖ LED Status Indicators..... 58
- ❖ Using ProSoft Configuration Builder (PCB) for Diagnostics..... 62
- ❖ Reading Status Data from the Module 76

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 MVI69-DFNT is working normally.
		On	The MVI69-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

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 CompactLogix or MicroLogix 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. Verify that the slot in the rack configuration has been set up correctly in the ladder logic.
Processor I/O LED flashes	This indicates a problem with backplane communications. Verify that all modules in the rack are configured in the ladder logic. The module has a power supply distance rating of 2 on CompactLogix, meaning that there must not be more than one other module between the MVI69-DFNT module and the power supply. If the module is used in a MicroLogix system, verify that the backplane can supply the 800 mA required by the module.

Module Errors

Problem Description	Steps to take
BP ACT LED remains OFF or blinks slowly	<p>This indicates that backplane transfer operations are failing. Connect to the module's Configuration/Debug port to check this.</p> <p>To establish backplane communications, verify the following items:</p> <ul style="list-style-type: none">▪ The processor is in RUN mode▪ The backplane driver is loaded in the module▪ The module is configured for read and write block data transfer▪ The ladder logic handles all read and write block situations▪ The module is configured in the processor
OK LED remains RED	<p>The program has halted or a critical error has occurred. Connect to the Configuration/Debug port to see if the module is running. If the program has halted, turn off power to the rack, remove the card from the rack and re-insert the card in the rack, and then restore power to the rack.</p>

4.2 Using ProSoft Configuration Builder (PCB) for Diagnostics

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

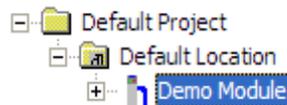
Because this is a text-based menu system, you enter commands by typing the [command letter] from your computer keyboard in the *Diagnostic* window in *ProSoft Configuration Builder (PCB)*. The module does not respond to mouse movements or clicks. The command executes as soon as you press the [COMMAND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

4.2.1 Using the Diagnostic Window in ProSoft Configuration Builder

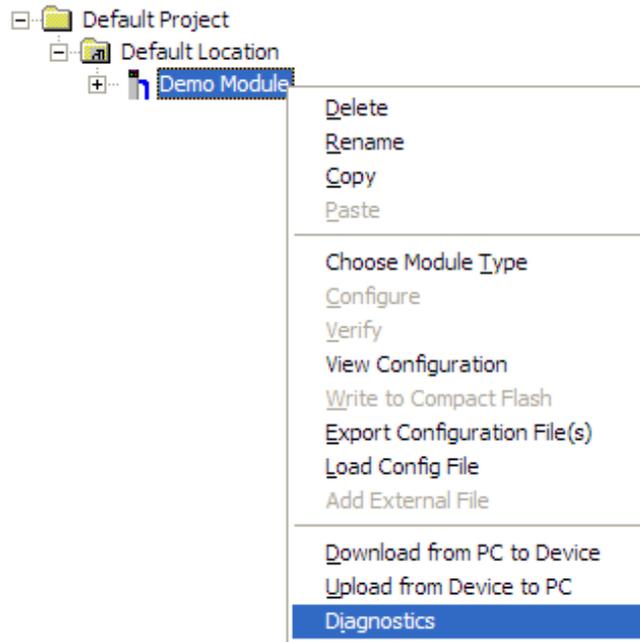
Tip: You can have a ProSoft Configuration Builder Diagnostics window open for more than one module at a time.

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

- 1 Start *PCB*, and then select the module to test. Click the right mouse button to open a shortcut menu.

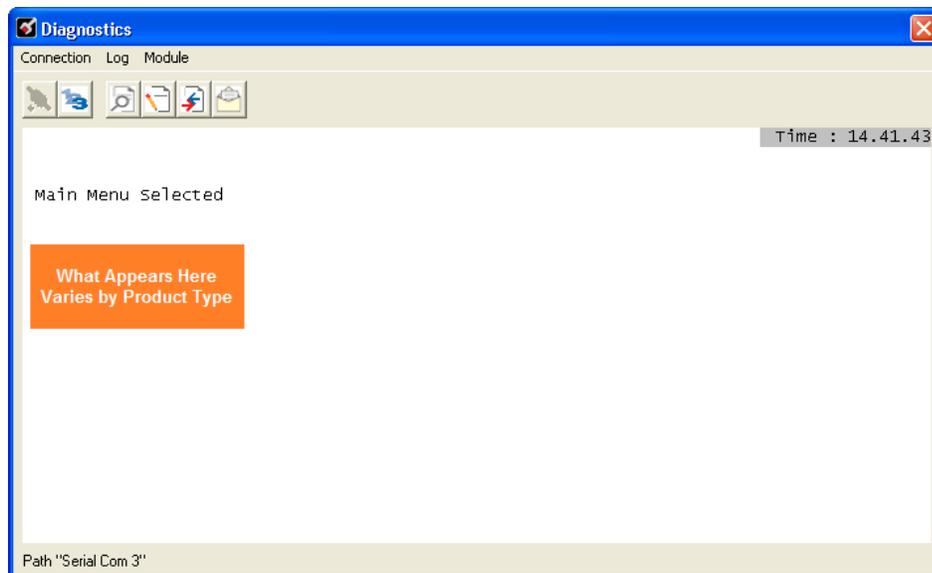


- 2 On the shortcut menu, choose **DIAGNOSTICS**.



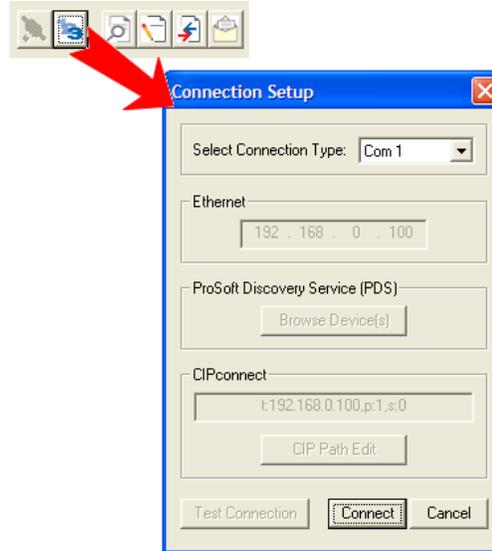
This action opens the *Diagnostics* dialog box.

- 3 Press [?] to open the *Main* menu.



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

- 1 Click to configure the connection. On the *Connection Setup* dialog box, select a valid com port or other connection type supported by the module.



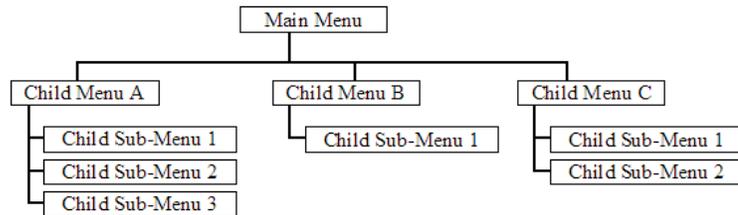
- 2 Verify that the null modem cable is connected properly between your computer's serial port and the module. A regular serial cable will not work.
- 3 On computers with more than one serial port, verify that your communication program is connected to the same port that is connected to the module.

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

4.2.2 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.

4.2.3 Main Menu

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

```
MVI69-DFNT COMMUNICATION MODULE MENU
?=Display Menu
B=Block Transfer Statistics
C=Module Configuration
D=Database View
E=Client 0 Command List Errors
I=Client 0 Command List
R=Transfer Configuration from PC to MVI Unit
S=Transfer Configuration from MVI Unit to PC
U=Reset diagnostic data
V=Version Information
W=Warm Boot Module
Communication Status: 1=Clients 2-5=DFNT Servers
Configuration:      8=Clients
@=Network Menu      Esc=Exit Program
```

Caution: Some of the commands available to you from this menu are designed for advanced debugging and system testing only, and can cause the module to stop communicating with the processor or with other devices, resulting in potential data loss or other communication failures. Use these commands only if you fully understand their potential effects, or if you are specifically directed to do so by ProSoft Technology Technical Support Engineers. There may be some special command keys that are not listed on the menu but that may activate additional diagnostic or debugging features. If you need these functions, you will be advised how to use them by Technical Support. Please be careful when pressing keys so that you do not accidentally execute an unwanted command.

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 70).

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

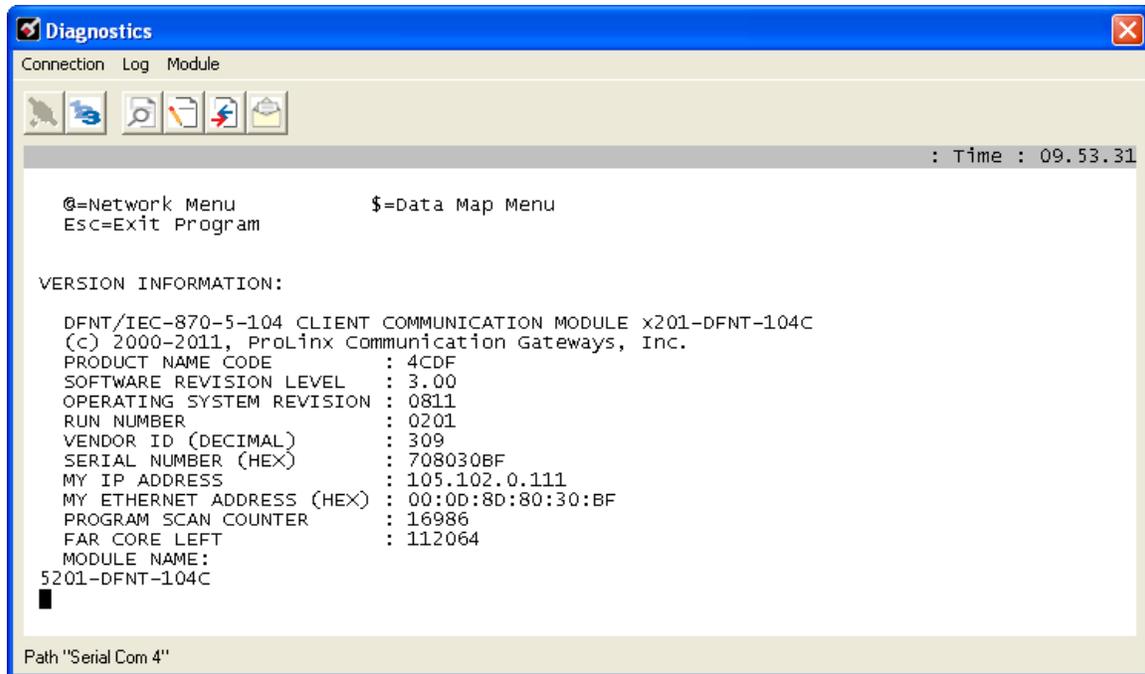
Press **[U]** to reset the status counters for the Client and/or server(s) in the module.

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 MVI69-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 Client Configuration

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

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 75).

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
0-3=Display data page 0-3000
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

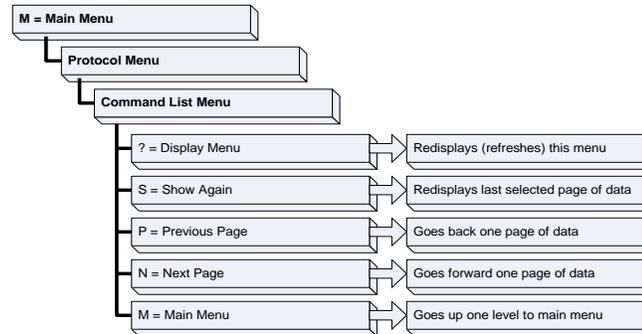
Press **[A]** from the *Database View* menu to display the data on the current page in ASCII format. This is useful for regions of the database that contain ASCII data.

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

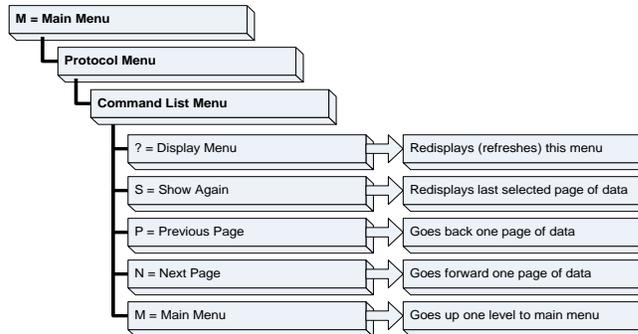
Press **[N]** to display the next page of commands.

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. 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.

```
WATTCP.CFG FILE:
# ProLinx Communication Gateways, Inc.
# Default private class 3 address
my_ip=192.168.0.75
# Default class 3 network mask
netmask=255.255.255.0
# name server 1 up to 9 may be included
# nameserver=xxx.xxx.xxx.xxx
# name server 2
# nameserver=xxx.xxx.xxx.xxx
# The gateway I wish to use
gateway=192.168.0.1
# some networks (class 2) require all three parameters
# gateway.network.subnetmask
# gateway 192.168.0.1,192.168.0.0,255.255.255.0
# The name of my network
# domainlist="mynetwork.name"
```

Returning to the Main Menu

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

4.3 Reading Status Data from the Module

The MVI69-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

❖ Product Specifications	77
❖ Functional Overview	81
❖ Server Driver	107
❖ Accessing a CompactLogix or MicroLogix Processor via Ethernet Using MVI69-DFNT	134
❖ Cable Connections	134
❖ Error Codes	139
❖ TCP/IP Interface Errors	143
❖ Command Function Codes	145
❖ General Command Structure.....	146
❖ MicroLogix Processor Specifics.....	161
❖ DFNT Command Entry Form.....	162

5.1 Product Specifications

The MVI69 EtherNet/IP™ Client/Server Communication Module allows processors to interface with EtherNet/IP protocol compatible devices and hosts. Devices commonly supporting the protocol include PLCs, PACs, field devices, and host HMI systems.

The module can send read or write commands to remote devices on the EtherNet/IP network. The module also has 20 servers that allow the module to receive commands from remote devices. Up to 100 commands can be configured by the user. The module provides support for an Ethernet port to be used in applications requiring a separate or dedicated network.

5.1.1 EtherNet/IP (Explicit Messaging) Compatible Devices

Rockwell Automation equipment that supports 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
- CompactLogix L23E
- RSLinx Gateway V1.7+
- ControlLogix 1756-ENET/ENBT/EN2T/EN2TR

5.1.2 General Specifications

- Single-slot, 1769 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. A sample ladder file with AOI is included.
- Configuration data obtained from configuration text file downloaded to module. A sample configuration file is included.
- Supports CompactLogix and MicroLogix 1500 LRP processors with 1769 I/O bus capability and at least 800 mA of 5 Vdc backplane current.

5.1.3 Hardware Specifications

Specification	Description
Dimensions	Standard 1769 single-slot module
Current Load	800 mA max @ 5 Vdc Power supply distance rating of 2 (L43 and L45 installations on first 2 slots of 1769 bus)
Operating Temp.	0°C to 60°C (32°F to 140°F)
Storage Temp.	-40°C to 85°C (-40°F to 185°F)
Relative Humidity	5% to 95% (with no condensation)
LED Indicators	Power and Module Status Application Status CFG Port Activity Ethernet Port Activity Error Status
CFG Port (CFG)	RJ45 (DB-9M with supplied cable) RS-232 only No hardware handshaking
App Port (Ethernet modules)	10/100 Base-T Ethernet compatible interface Electrical Isolation 1500 Vrms 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
Shipped with Unit	RJ45 to DB-9M cables for each port 6-foot RS-232 configuration cable

5.1.4 Functional Specifications

- Provides up to 4000 16-bit memory registers for user application data transfer
- User-definable module memory usage
- 10/100 Base-T Ethernet compatible interface
- Functions as a server and Client
- Status data is available in ladder logic or through the Configuration Diagnostics Serial Port

Server Specifications

- Supports EtherNet/IP explicit, connected, and unconnected class messaging
- 20 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 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

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

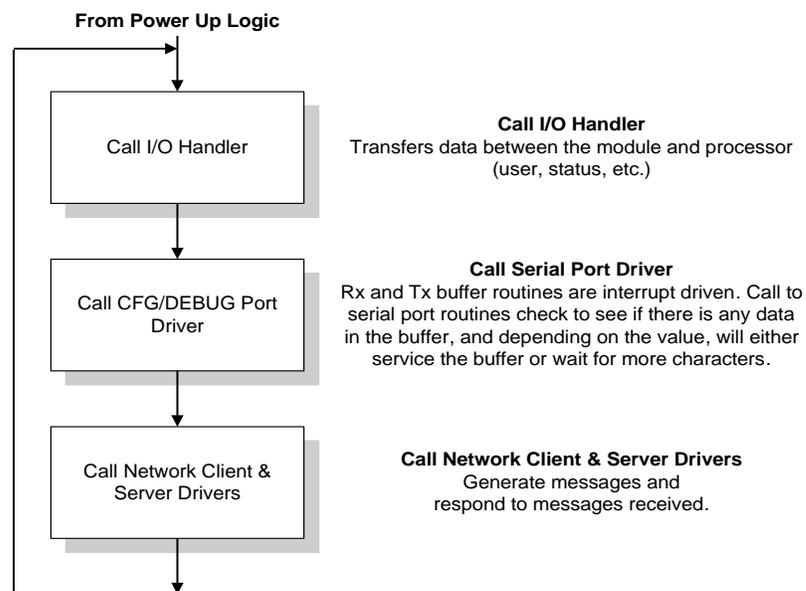
On power up the module begins performing the following logical functions:

- 1 Initialize hardware components
- 2 Initialize CompactLogix or MicroLogix 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

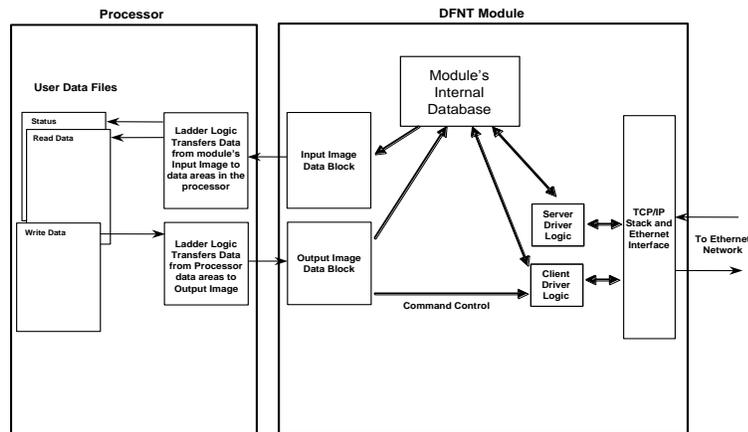
Upon completing the power-up configuration process, the module enters an infinite loop that performs the functions shown in the following diagram.



Block Transfer Backplane Data Transfer

The MVI69-DFNT module communicates directly over the CompactLogix or MicroLogix backplane. Data is paged between the module and the CompactLogix or MicroLogix processor across the backplane using Input Image and Output Image operations. Data is transferred from the module to the processor using the Input Image blocks, and data is transferred from the processor to the module using Output Image blocks.

The following illustration shows the data transfer method used to move data between the CompactLogix or MicroLogix processor, the MVI69-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 Input Image and Output Image blocks. Ladder logic must be written in the CompactLogix or MicroLogix 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



5.2.2 Normal Data Transfer

Normal data transfer includes the paging of the user data found in the module's internal database and the Status Data. These data are transferred through Read (input image) and Write (output image) Blocks. Refer to the *Module Configuration* section for a description of the data objects used with the blocks. 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 CompactLogix or MicroLogix. The structure of the input image used to transfer this data is shown below:

Offset	Description	Length
0	Read Block ID	1
1	Write Block ID	1
2 to (n+1)	Read Data	n

n=60, 120 or 240 depending on the Block Transfer Size parameter (refer to the configuration file).

The *Read Block ID* is an index value used to determine where the data will be placed in the CompactLogix or MicroLogix controller tag *ReadData* array.

The *Write Block ID* contained in the read block is used by ladder logic to decide what portion of the *WriteData* array to send to the module from the CompactLogix or MicroLogix.

Write Block

These blocks of data transfer information from the CompactLogix or MicroLogix to the module. The following table describes the structure of the output image.

Offset	Description	Length
0	Write Block ID	1
1 to n	Write Data	n

n=60, 120 or 240 depending on the Block Transfer Size parameter (refer to the configuration file).

The *Write Block ID* is an index value used to determine the location in the module's database where the data will be placed.

How Data is Transferred

In order to understand how the data is transferred between the processor and the module, you must understand the concept of Read Data and Write Data areas in the module's internal memory database. The Read Data Area will be transferred from the module to the CompactLogix or MicroLogix. The Write Data Area will be transferred from the CompactLogix or MicroLogix to the module.

The module's available database space can be divided into Read Data and Write Data Areas, as well as used to hold module hardware, driver status and application status data. These areas are defined by the user when the module configuration is created. The following is one example of how configuration parameters can be used to define Read and Write Data areas.

Read Register Start = 0

Read Register Count = 120

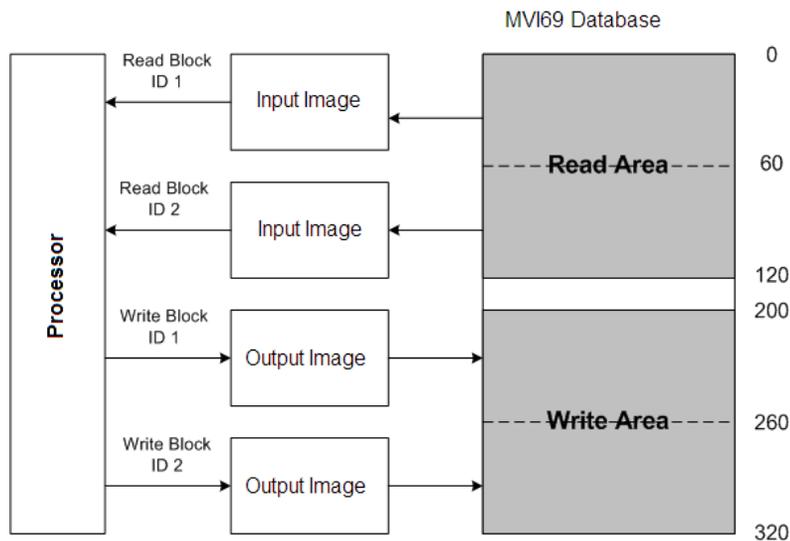
Write Register Start = 200

Write Register Count = 120

Let us also assume that the configuration parameter, *Block Transfer Size* is set to **60**.

In this case, backplane data transfers will hold only 60 16-bit words per block; so, each area will be broken down into blocks of 60 words.

The following example shows the resulting data flow:



Notice that the number of read or write blocks that will need to be transferred depends on the *Read Register Count* and *Write Register Count* values and may be calculated using the formulas:

Read Register Count divided by the *Block Transfer Size* = Total number of Read Blocks

Write Register Count divided by the *Block Transfer Size* = Total number of Write Blocks

Note: Any fractional portion of a block will be rounded up to the next larger whole block number, that is, 100 divided by 60 would result in 2 blocks to transfer; 500 divided by 60 would result in 9 blocks to transfer, 500 divided by 240 would result in 3 block to transfer, and so on. Therefore, it would be a good idea to enter the *Read Register Count* and *Write Register Count* parameters in multiples of the configured *Block Transfer Size*.

During normal program operation, the module sequentially sends read blocks and requests write blocks. For example, if the application uses three read and two write blocks, the sequence will be as follows:

R0W1→R1W2→R2W1→R3W2→R0W1→R1W2→R2W1→R3W2→

Notice that ReadBlock 0, the read block that contains status information, appears at the start of each read block cycle. This means status information in the controller tags will update only after all other read data blocks have been updated.

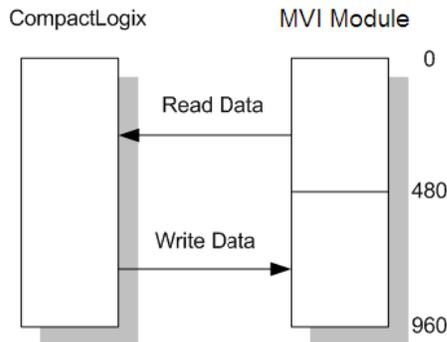
The sequence will continue unless interrupted by special function write block numbers sent by the controller or by a command request from a node on the DFNT network or operator control through the module's Configuration/Debug port. For more information, see Special Function Blocks.

Here are some additional examples of alternative backplane communication setups.

Assume that the backplane parameters are configured as follows:

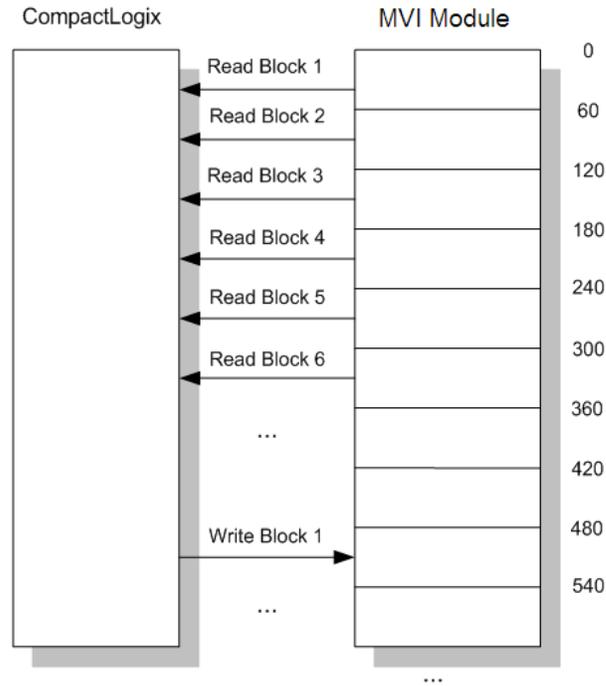
```
Read Register Start:    0
Read Register Count:  480
Write Register Start:  480
Write Register Count:  480
```

The module data areas would be allocated as follows:

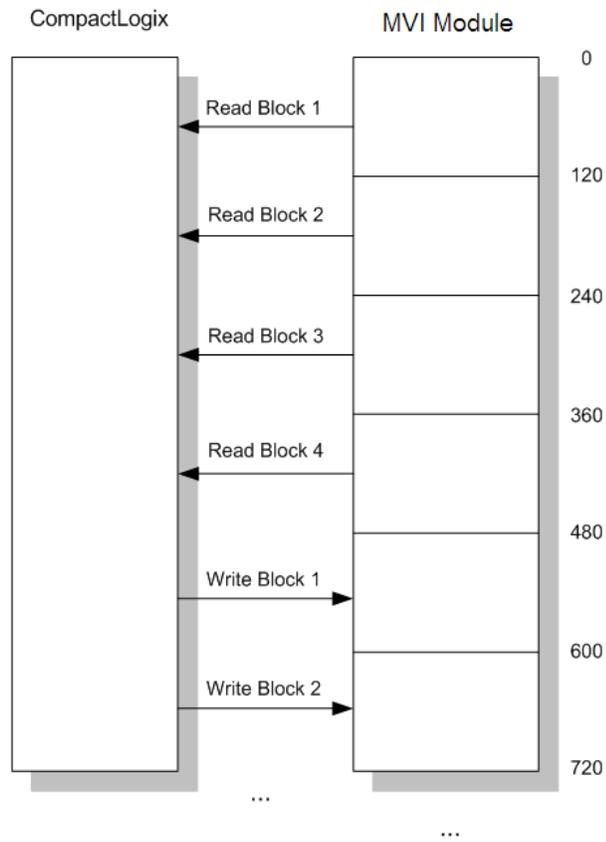


Database address 0 to 479 will be continuously transferred from the module to the processor. Database address 480 to 959 will be continuously transferred from the processor to the module.

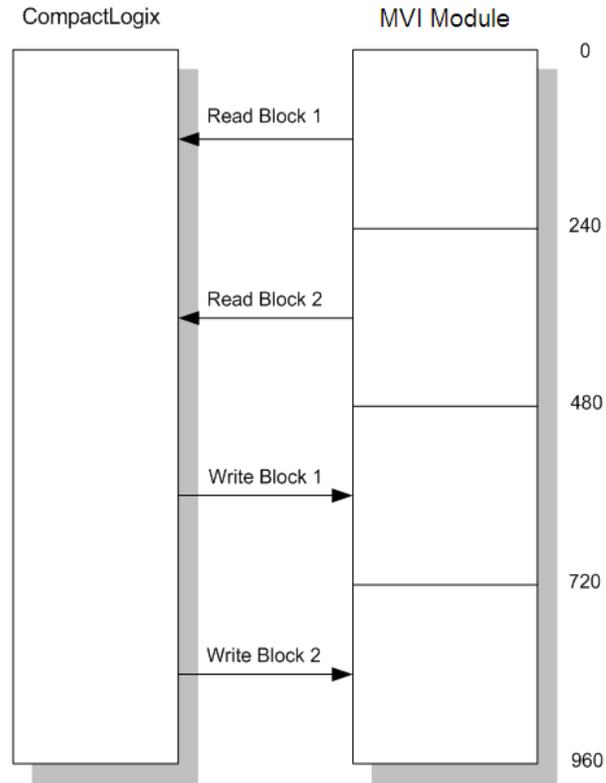
If *Block Transfer Size* = **60**:



If *Block Transfer Size* = **120**:



If *Block Transfer Size* = **240**:



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 to 254	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 Blocks (250 and 251)

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

Offset	Description
250	Module and Client Status
251	Status for servers 0 to 4

MVI69-DFNT Status Data Definition contains a complete listing of the data returned for the status blocks.

Block 250 Response Block Format (also Block -1 and 0)

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)

HTTP Port 80 Status

Offset	Content	Description
11	Not Used	Reserved
12	Not Used	Reserved
13	Not Used	Reserved
14	Number of Requests	Total number of http requests for the module
15	Number of Responses	Total number of http response messages generated.
16	Not Used	Reserved
17	Not Used	Reserved
18	Not Used	Reserved
19	Not Used	Reserved
20	Not Used	Reserved

DFNT 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.

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

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

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

Block 251 Response Block Format

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.

DFNT 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.

DFNT 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.

DFNT 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.

DFNT 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.

Offset	Content	Description
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.

DFNT 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.

Output Data Initialization Request 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 n	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 n	Data to place in database	n

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 MVI69-DFNT Status Data Definition. 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 52	Not used	50

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 53	Command error list returned	50

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 51	List of command indices on which to perform the operation.	50

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 52	Not used	50

Warm Boot Block (9998)

This block is sent from the CompactLogix or MicroLogix to the module when the module is required to perform a Warm Boot (software reset) operation. The 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 Warm Boot block.

Block Request from Processor to Module

Word Offset	Description	Length
0	9998	1
1 to <i>n</i>	Not used	<i>n</i>

n = 60, 120, or 240 depending on the Block Transfer Size parameter (refer to the configuration file).

Cold Boot Block (9999)

This block is sent from the CompactLogix or MicroLogix to the module when the module is required to perform the cold boot (hardware reset) operation. The 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 <i>x</i>	Not used	blk_size

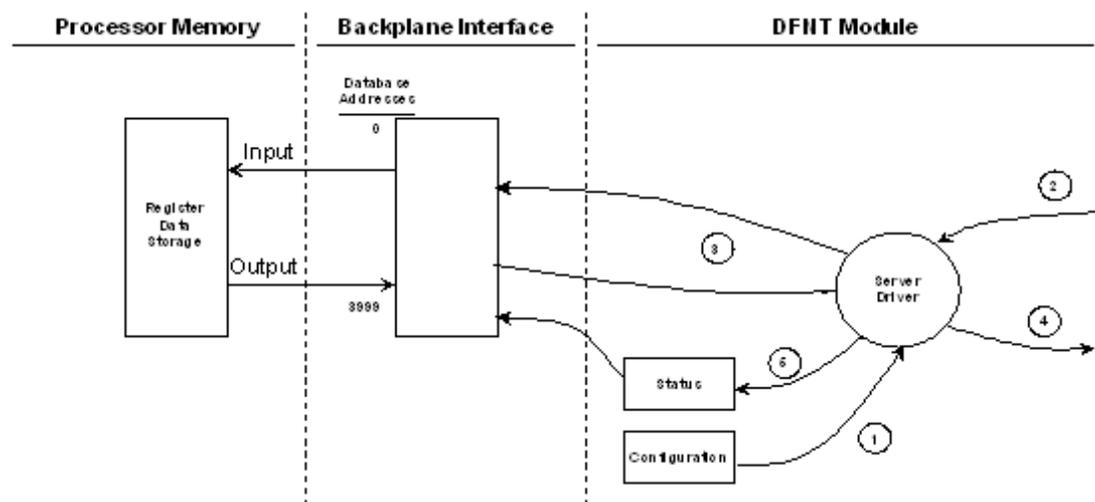
5.2.4 Data Flow between MVI69-DFNT Module and CompactLogix or MicroLogix Processor

The following discussion outlines the flow of data between the two pieces of hardware (CompactLogix or MicroLogix processor and MVI69-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

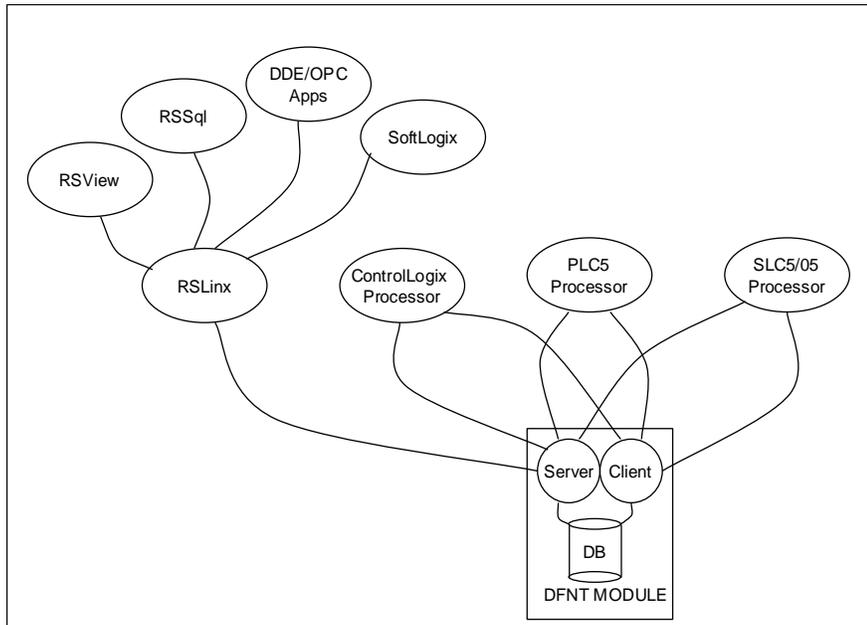
The Server Driver allows the MVI69-DFNT module to respond to data read and write commands issued by clients on the EtherNet/IP network using explicit messaging. The following flow chart and associated table describe the flow of data into and out of the module.



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 section 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 PLC	ControlLogix	
			PCCC	CIP Integer
0	0	N10:0	N10:0	Int_data[0]
999	999	N10:999	N10:999	Int_data[999]
1000	1000	N11:0	N11:0	Int_data[1000]
1999	1999	N11:999	N11:999	Int_data[1999]
2000	2000	N12:0	N12:0	Int_data[2000]
2999	2999	N12:999	N12:999	Int_data[2999]
3000	3000	N13:0	N13:0	Int_data[3000]
4000	4000	N14:0	N14:0	Int_data[4000]

When using PLC5 or PLC commands, access to the database is through simulated 'N' files. For example, to access database element 3012, use the file address of N13:12. The module simulates N-files in the internal database. The following table lists the relationship between the N-files and the module's internal database registers:

Internal	Simulated
Database	N-File
Register	
0	N10:0
1	N10:1

999	N10:999
1000	N11:0
1001	N11:1

1999	N11:999
2000	N12:0
2001	N12:1

2999	N12:999
3000	N13:0
3001	N13:1

3999	N13:999

Note: The way the data files are used will depend on the DFNT Server File Size value (100 or 1000). The previous example shows an example where this parameter is set with a value of 1000. The following table lists the PCCC functions supported by the module:

Basic Command Set Functions

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

CompactLogix or MicroLogix-5 Command Set Functions

Command	Function	Definition
0x0F	0x00	Word Range Write (Binary Address)
0x0F	0x01	Word Range Read (Binary Address)
0x0F	0x00	Word Range Write (ASCII Address)
0x0F	0x01	Word Range Read (ASCII Address)

SLC-500 Command Set Functions

Command	Function	Definition
0x0F	0xA1	Protected Typed Logical Read With Two Address Fields
0x0F	0XA2	Protected Typed Logical Read With Three Address Fields
0x0F	0XA9	Protected Typed Logical Write With Two Address Fields
0x0F	0XAA	Protected Typed Logical Write With Three Address Fields

Additionally, the module supports CIP data table read and write functions. These functions use controller tags to access data in the module's database. This is the preferred data access method as it directly specifies the data type used with the command. The following table lists the data access methods:

MSG Instruction Type									
Database Addresses	PL C2	PLC5 or SLC	ControlLogix						
			PCCC	CIP Boolean	CIP Bit Array	CIP Byte	CIP Integer	CIP Double Int	CIP Real
0	0	N10:0	N10:0	BoolData[0]	BitAData[0]	SintData[0]	Int_Data[0]	DintData[0]	RealData[0]
999	999	N10:99	N10:99	BoolData[1598]		SintData[1998]	Int_Data[999]		
1000	1000	N11:0	N11:0	BoolData[1600]	BitAData[500]	SintData[2000]	Int_Data[1000]	DintData[500]	RealData[500]
1999	1999	N11:99	N11:99	BoolData[3198]		SintData[3998]	Int_Data[1999]		
2000	2000	N12:0	N12:0	BoolData[3200]	BitAData[1000]	SintData[4000]	Int_Data[2000]	DintData[1000]	RealData[1000]
2999	2999	N12:99	N12:99	BoolData[4798]		SintData[5998]	Int_Data[2999]		
3000	3000	N13:0	N13:0	BoolData[4800]	BitAData[1500]	SintData[6000]	Int_Data[3000]	DintData[1500]	RealData[1500]
3998	3998	N13:98	N13:98	BoolData[6396]	BitAData[1999]	SintData[7996]	Int_Data[3998]	DintData[1999]	RealData[1999]
3999	3999	N13:99	N13:99	BoolData[6398]		SintData[7998]	Int_Data[3999]		

If the CIP data table read and write functions are utilized, the controller tag array names defined in the module must be used. The following table lists the controller tag names recognized by the module and the associated data types:

Tag Array	Data Type	Data Size
BoolData[]	Bit	1 bit
BitAData[]	Bit Array	32 bits
SintData[]	Byte	8 bits
Int_Data[]	Word	16 bits
DIntData[]	Double Word	32 bits
RealData[]	Floating-point	32 bits

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

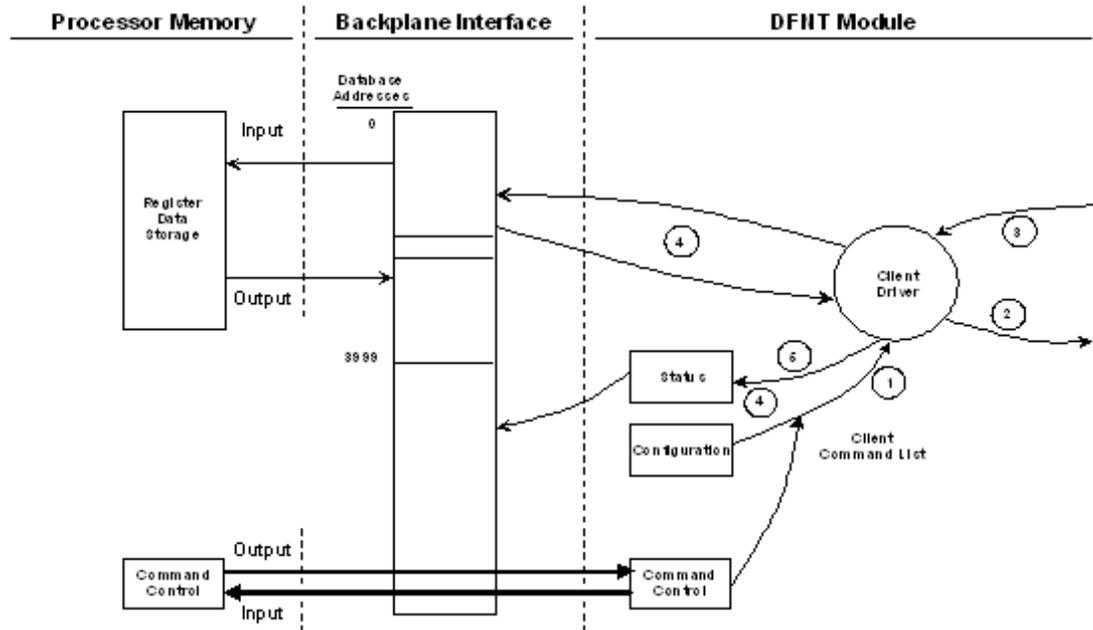
Command	Function	Definition	Supported in Slave
0x0F	0x00	Word Range Write (Binary Address)	X
0x0F	0x01	Word Range Read (Binary Address)	X
0x0F		Typed Range Read (Binary Address)	X
0x0F		Typed Range Write (Binary Address)	X
0x0F	0x26	Read-Modify-Write (Binary Address)	
0x0F	0x00	Word Range Write (ASCII 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 MVI69-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 CompactLogix or MicroLogix 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)

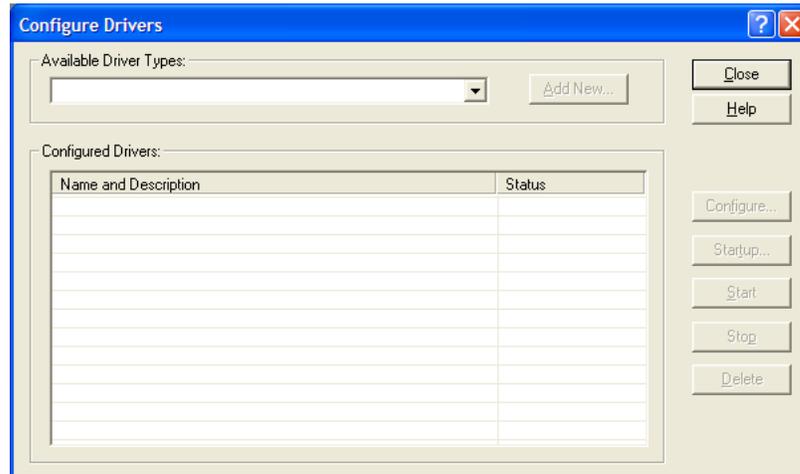
5.3 Server Driver

The Server Driver allows the MVI69-DFNT module to respond to data read and write commands issued by clients on the EtherNet/IP network using explicit messaging.

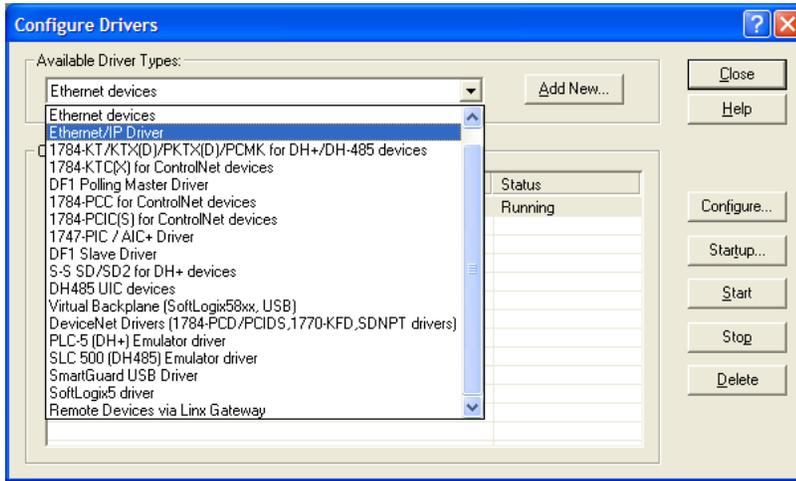
5.3.1 RSLinx Software

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.



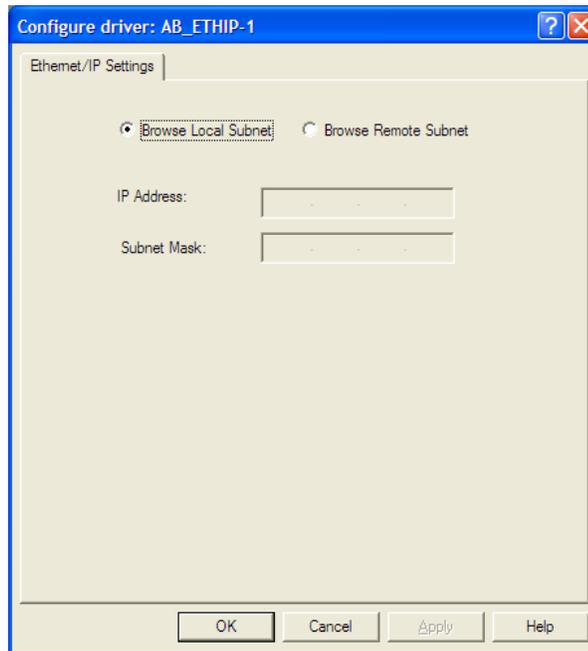
- 3 Click the arrow to the right of the Available Driver Types field, and then select **ETHERNET/IP DRIVER** from the dropdown list.



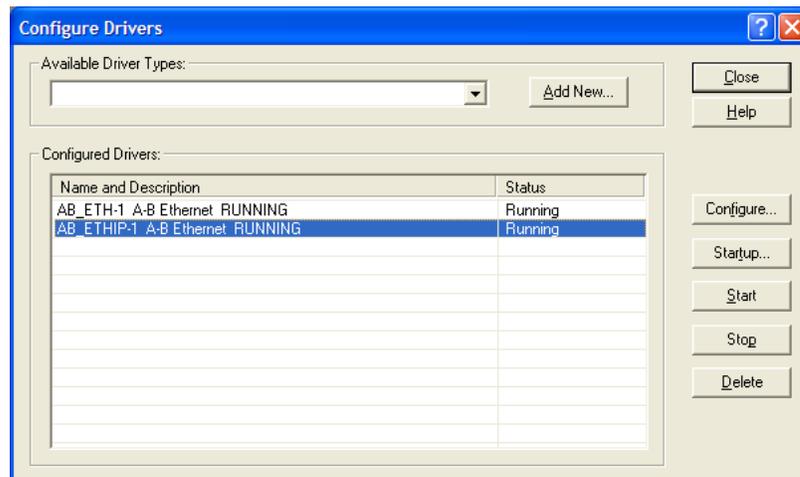
- 4 Click the **ADD/NEW** button, and then click **OK** in the **ADD NEW RSLINX DRIVER** dialog box.



- 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.

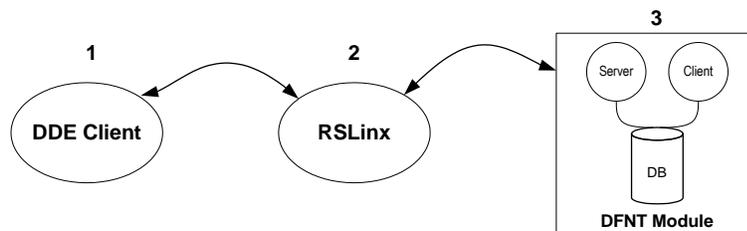


- 8 Close RSLinx software.

DDE Connection

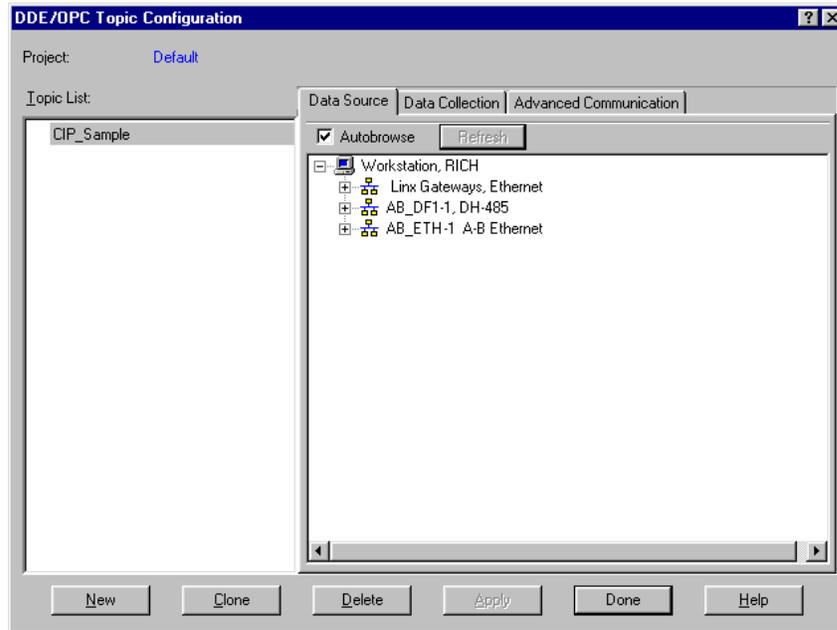
This section discusses setting up a DDE connection to the module in order to transfer data between the module and a DDE compliant application.

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.

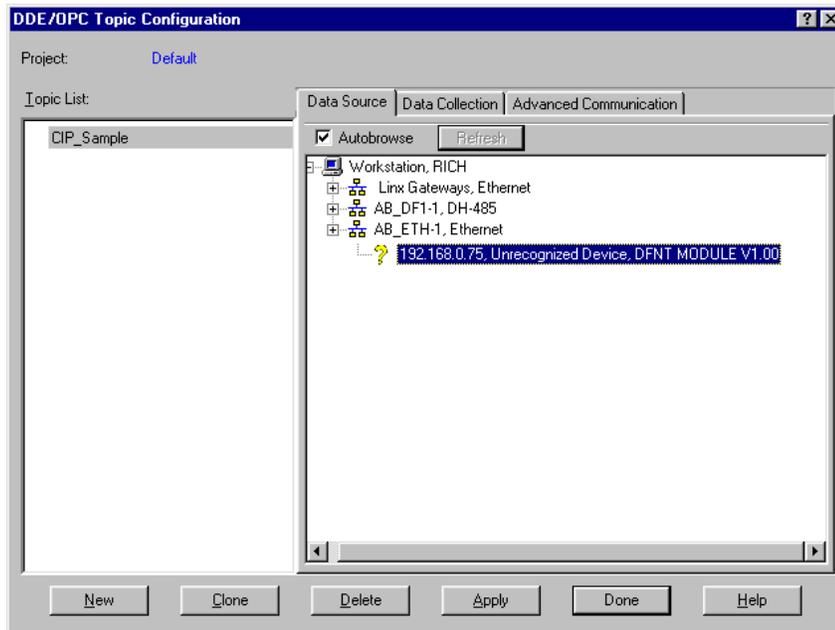


- 1 The DDE Client application specifies the DDE link by specifying the APPLICATION, TOPIC, and ITEM elements. For example, in Excel, enter =RSLINX[CIP_Sample! 'ReadData[0]'] into a cell.
- 2 RSLinx is the DDE Server: APPLICATION = RSLINX.
And
RSLinx defines the TOPIC.
This is the communication link 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]).

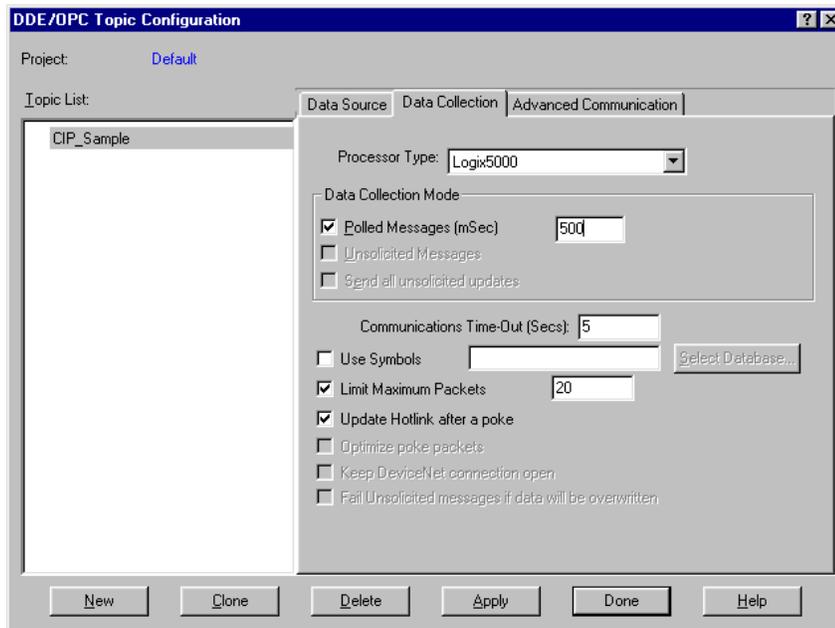
As discussed in the previous section, RSLinx must be used to define the Topic element for the DDE link definition. This is accomplished using the following procedure. It is assumed that the module can be seen in the RSLinx software. Refer to the RSLinx section of this manual to set up this connection. 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:



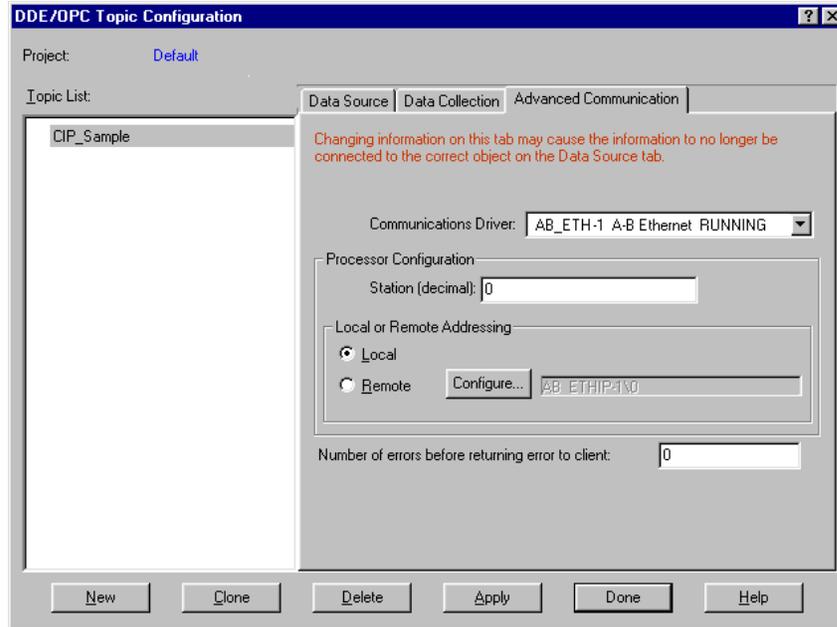
Click the **NEW** button to add a new Topic to the **TOPIC LIST**. This displays a default name. Edit the Topic name for the link to be formed. This name should reflect the unit or location to which the connection is being made. Do not press the Enter key. Instead, double-click the mouse on the DFNT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

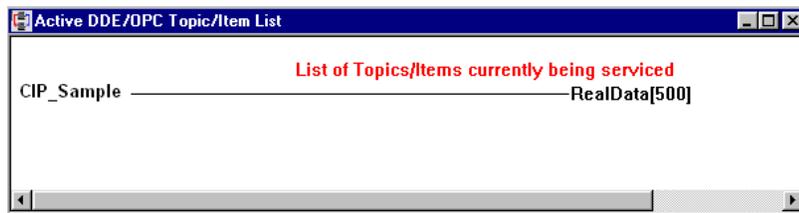
```
=APPLICATION|TOPIC!ITEM
```

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:

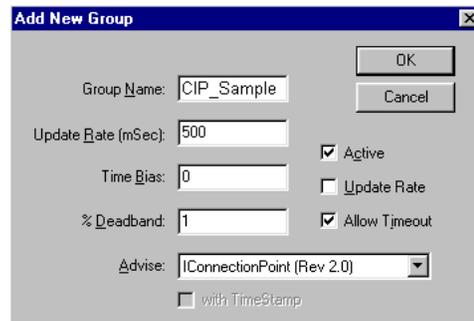


Any DDE compliant program can be used in the same manner. For maximum utility Visual Basic applications can be used to interface with module's database using DDE connectivity.

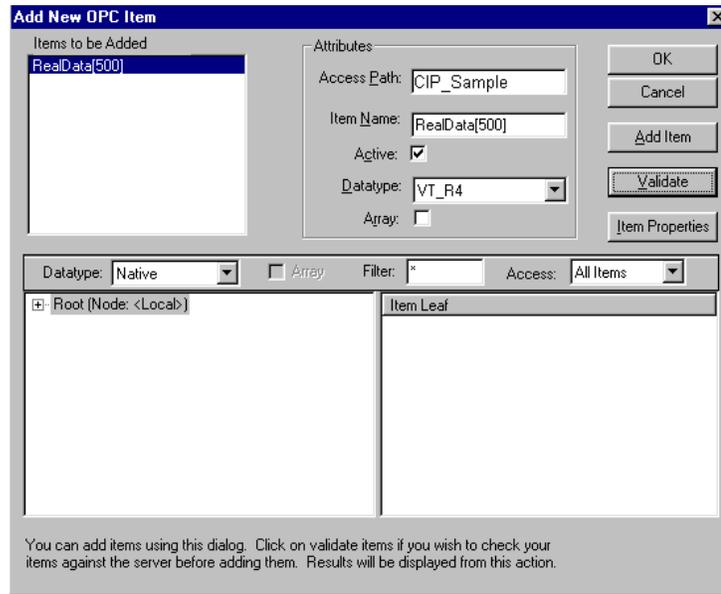
OPC Connection

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.

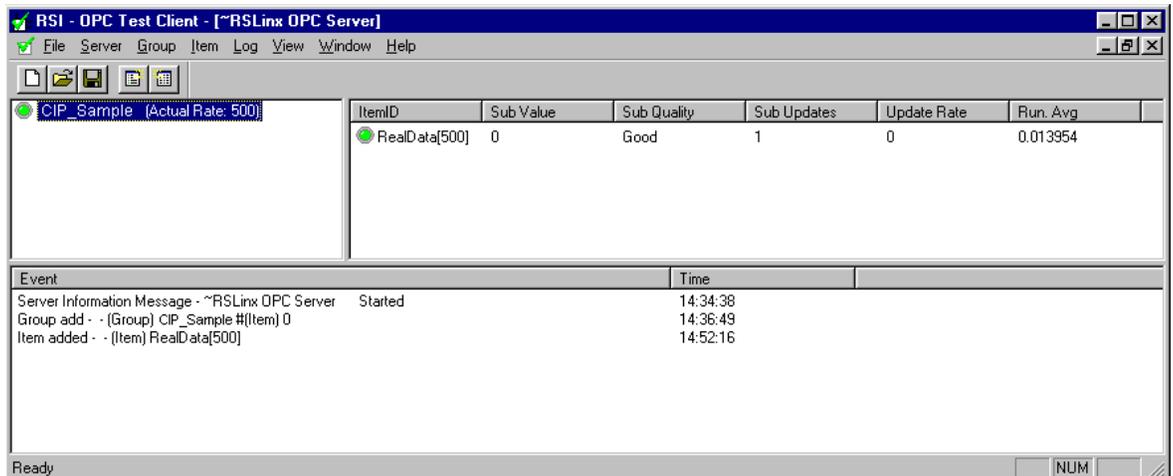
The example used in this section uses the OPC test client (opctest.exe) available from Rockwell Automation. First start the application and select the Connect... option from the Server Menu. In the dialog box shown, select the RSLinx OPC server. Next add a group using the Group menu option and fill in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



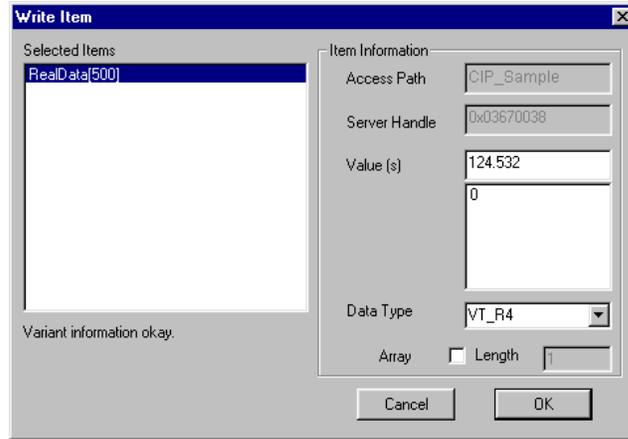
Next, add items to the client. The following shows an example dialog used to add a floating-point data item:



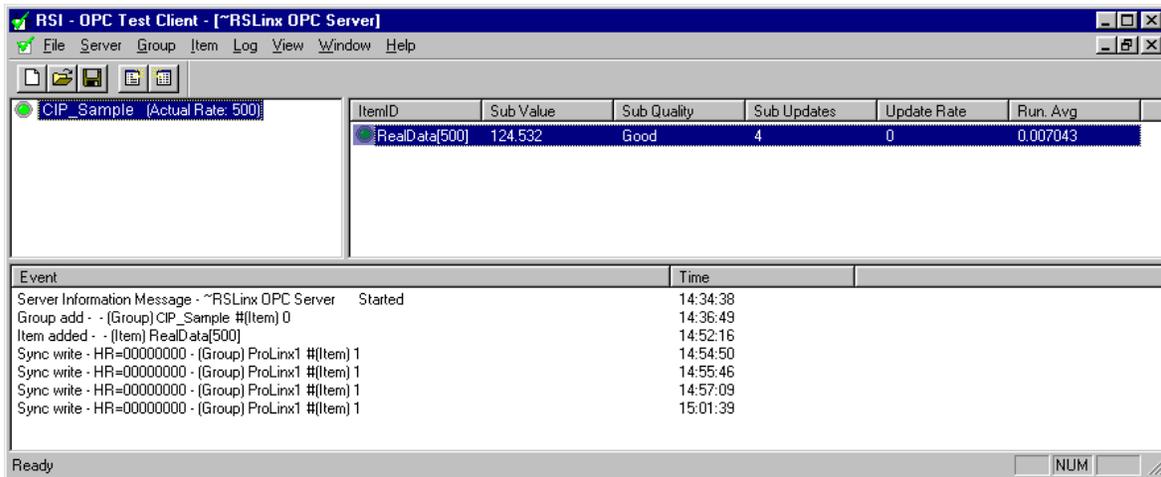
It is important to set the Access Path value to the Topic name assigned in RSLinx and to set the Item Name to a valid controller tag in the DFNT module. The Datatype parameter must be set to match that of the controller tag. In the example shown, the VT_R4 data type is selected for the floating-point 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.



5.3.2 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

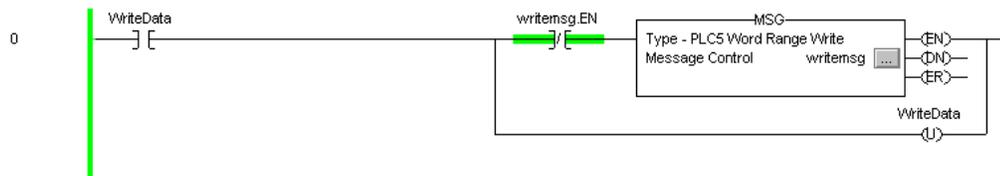
PLC5 and SLC5/05 processors containing an Ethernet interface use the encapsulated PCCC message method. The module simulates these devices and accepts both read and write commands. The following topics describe the support for the read and write operations.

Encapsulated PCCC Write Message

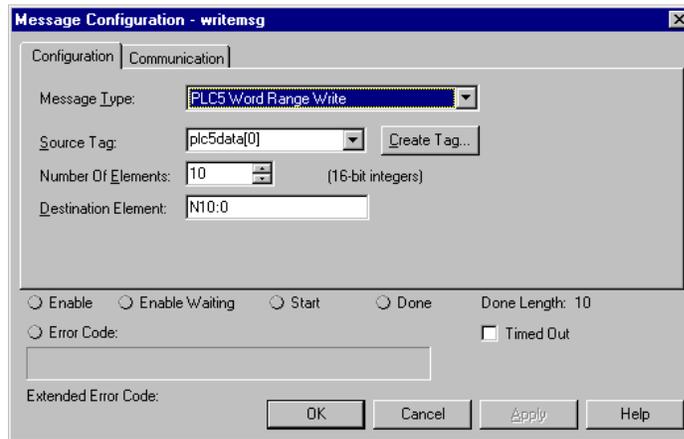
Write commands transfer data from the ControlLogix processor to the module. The following encapsulated PCCC commands are supported from a ControlLogix Processor:

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

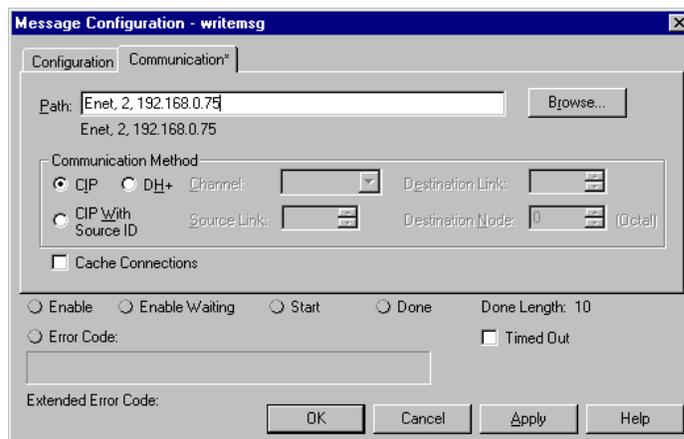
An example rung used to execute a write command is shown in the following diagram:



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.

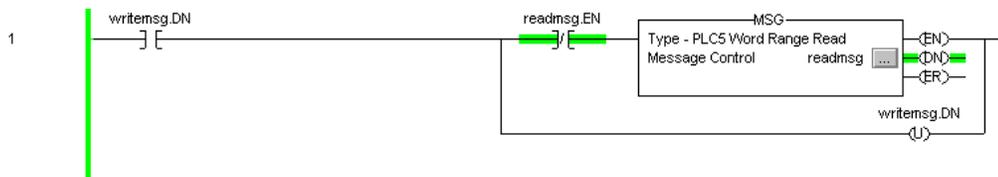
More complex paths are possible if routing to other networks using multiple 1756-ENET modules and racks. Refer to the Rockwell Automation Support Knowledgebase for more information on Ethernet routing and path definitions.

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

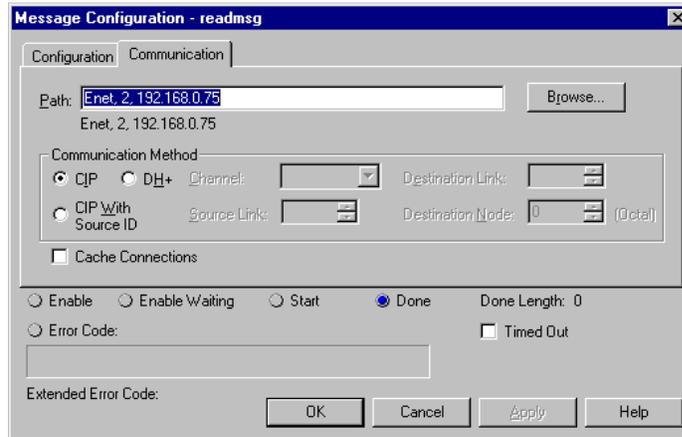
An example rung used to execute a read command is shown in the following diagram:



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.

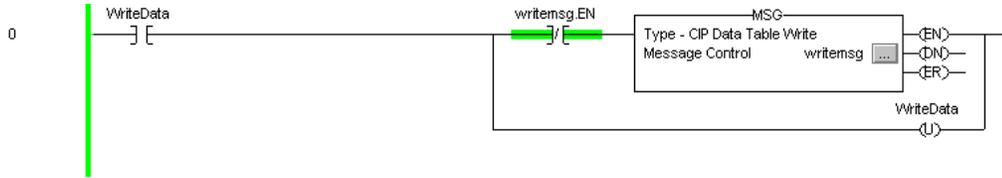
More complex paths are possible if routing to other networks using multiple 1756-ENET modules and racks. Refer to the Rockwell Automation Support Knowledgebase for more information on Ethernet routing and path definitions.

CIP Data Table Operations

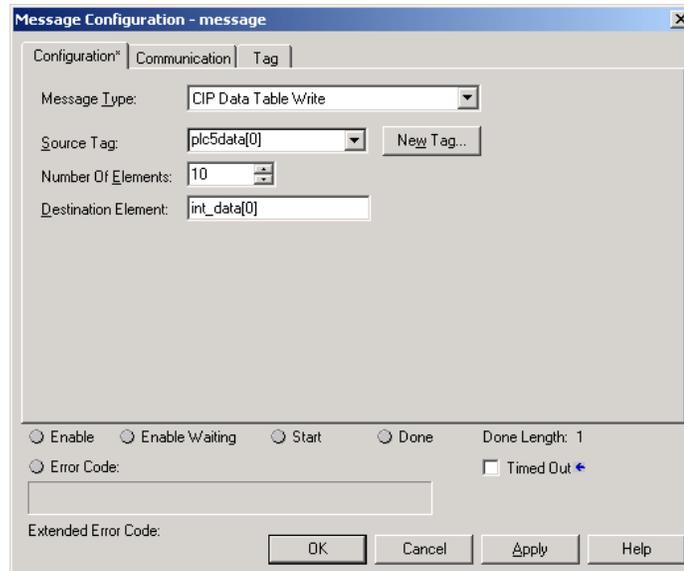
This method of data transfer uses CIP messages to transfer data between the ControlLogix processor and the module. Tag names define the elements to be transferred. The following topics describe the support for the read and write operations.

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:



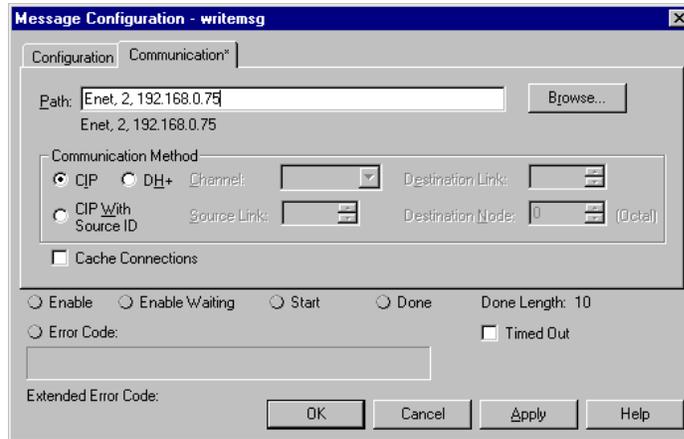
The **MESSAGE CONFIGURATION** dialog box must be completed to define the data set to be transferred from the processor to the module. An example of the dialog box follows:



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:

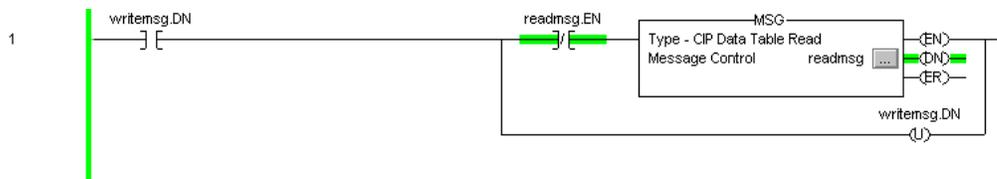


Verify that the **CIP** radio-button is selected as the **COMMUNICATION METHOD**. The **PATH** specifies the message route from the ControlLogix processor to the DFNT module. Path elements are separated by commas. In the example path shown, the first element is "Enet", which is the user-defined name 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.

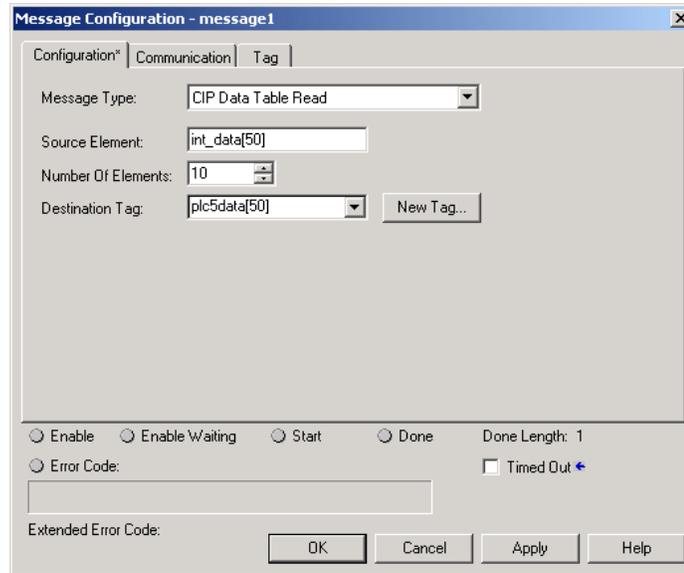
More complex paths are possible if routing to other networks using multiple 1756-ENET modules and racks. Refer to the Rockwell Automation Support Knowledgebase for more information on Ethernet routing and path definitions.

CIP Data Table Read

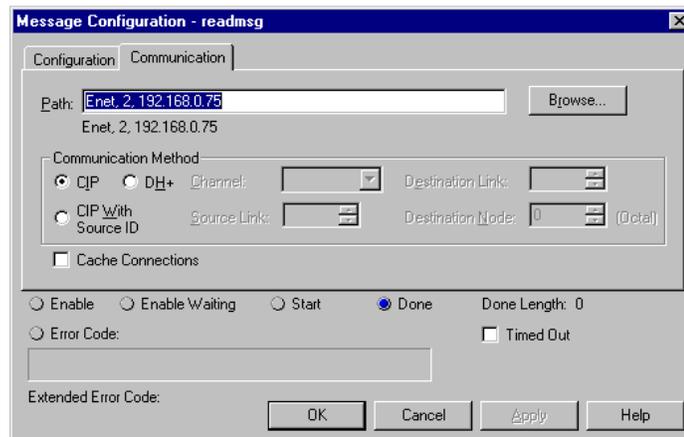
CIP data table read messages transfer data to the ControlLogix processor from the DFNT module. An example rung used to execute a read command is shown:



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.

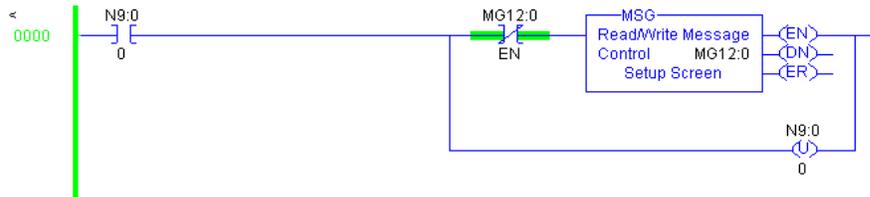
More complex paths are possible if routing to other networks using multiple 1756-ENET modules and racks. Refer to the Rockwell Automation Support Knowledgebase for more information on Ethernet routing and path definitions.

5.3.3 PLC5 Processor

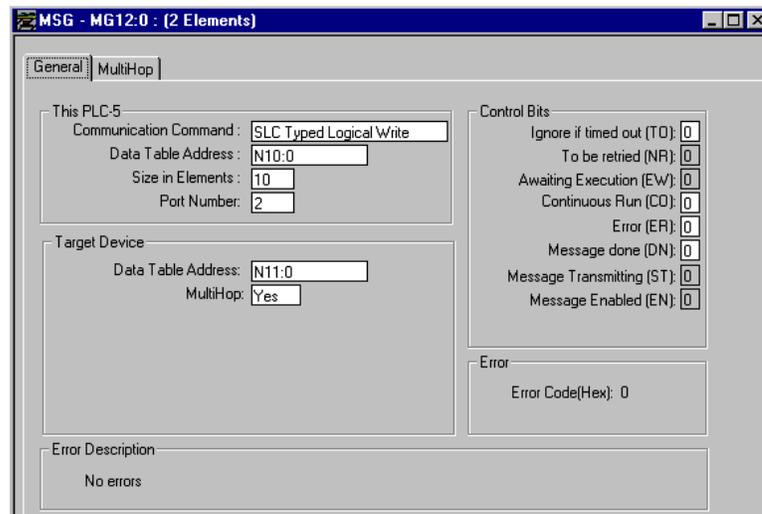
The module can be used to receive messages from a PLC5 containing an Ethernet interface. The module supports both read and write commands. A discussion of each operation is provided in the following topics:

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.

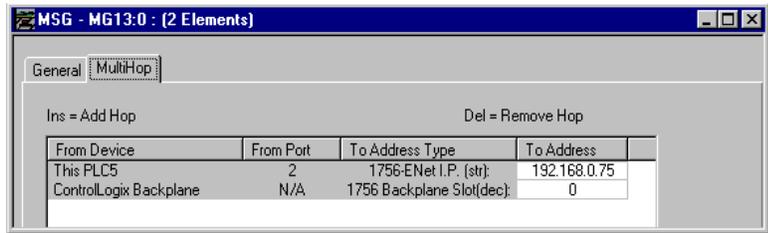


Select the **COMMUNICATION COMMAND** to execute from the following list of supported commands.

- 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.

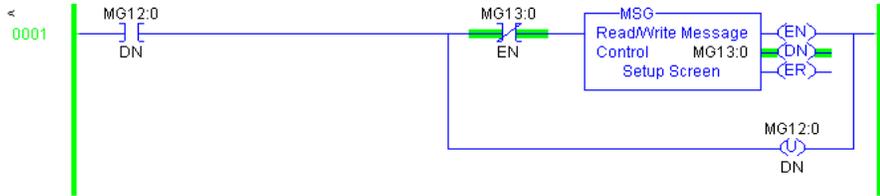
The **MULTIHOP** option must be set to **YES**. The **MULTIHOP** tab portion of the dialog box must be completed as shown in the following window:



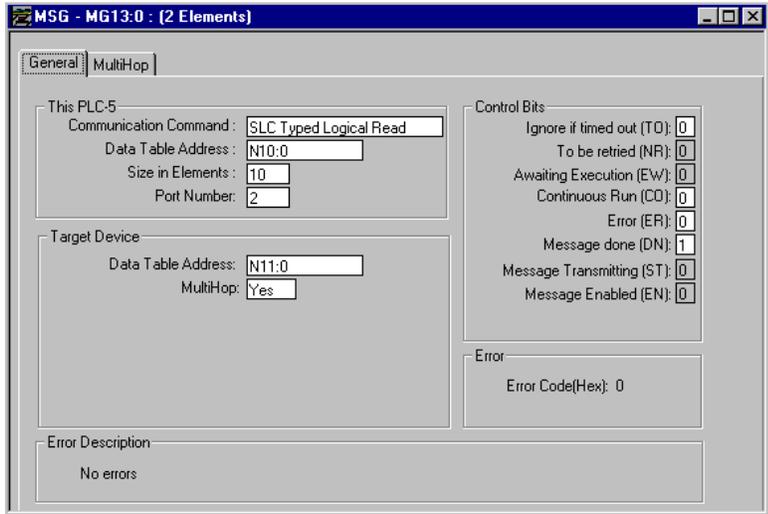
Set the IP address value to the module's Ethernet IP address. You must press the "Insert" key to add the second line for ControlLogix Backplane and set the slot number to zero.

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.

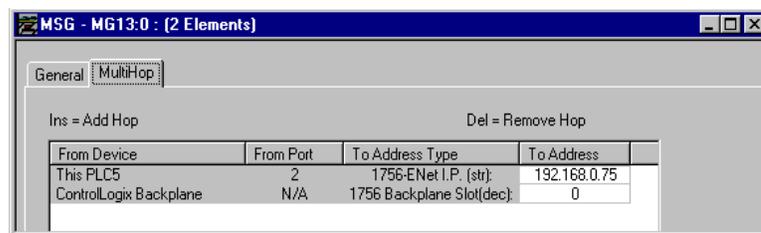


Select the **COMMUNICATION COMMAND** to execute from the following list of supported commands.

- 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:



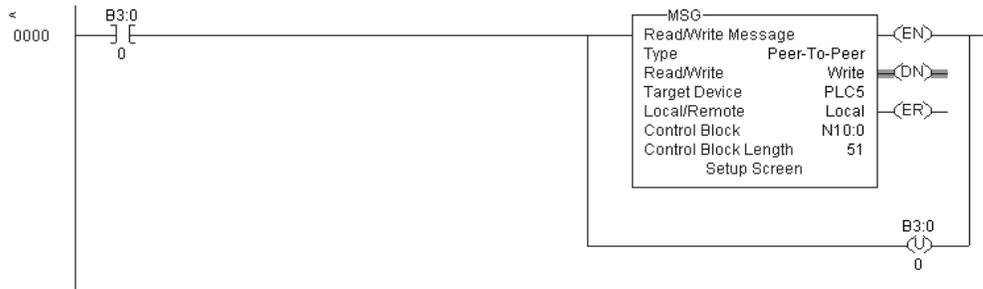
Set the IP address value to the module's Ethernet IP address. You must press the "Insert" key to add the second line for ControlLogix Backplane and set the slot number to zero.

5.3.4 SLC 5/05 Processor

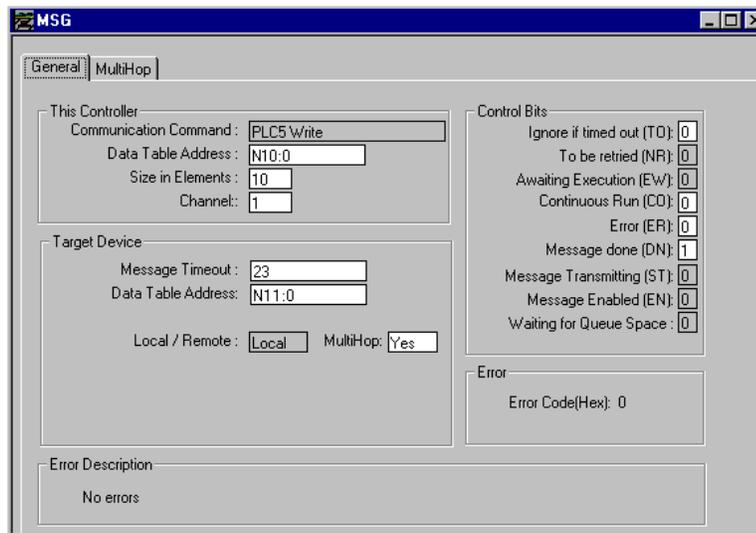
The module can be used to receive messages from a SLC 5/05 containing an Ethernet interface. The module supports both read and write commands. A discussion of each operation is provided in the following topics.

SLC5/05 Write Commands

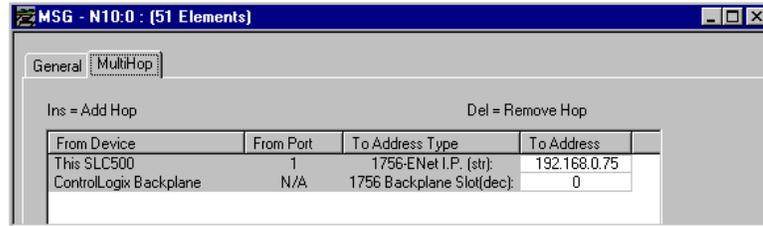
Write commands transfer data from the SLC processor to the DFNT module. An example rung used to execute a write command is shown in the following diagram:



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.



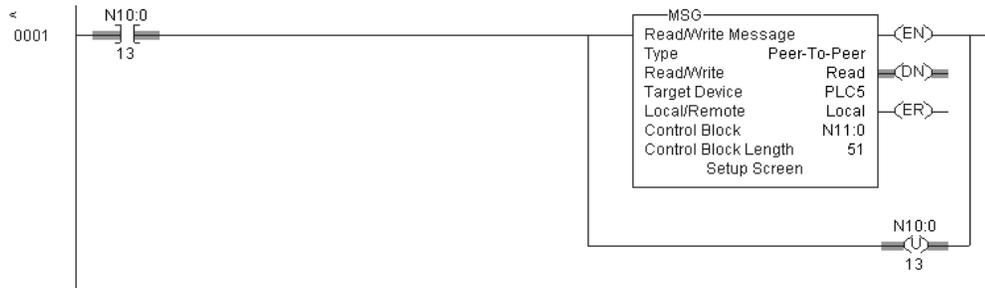
The **TARGET DEVICE DATA TABLE ADDRESS** must be set to a valid file element (such as, N11:0) for SLC and PLC5 messages. The **MULTIHOP** option must be set to **YES**. The **MULTIHOP** tab portion of the dialog box must be completed as displayed in the following window:



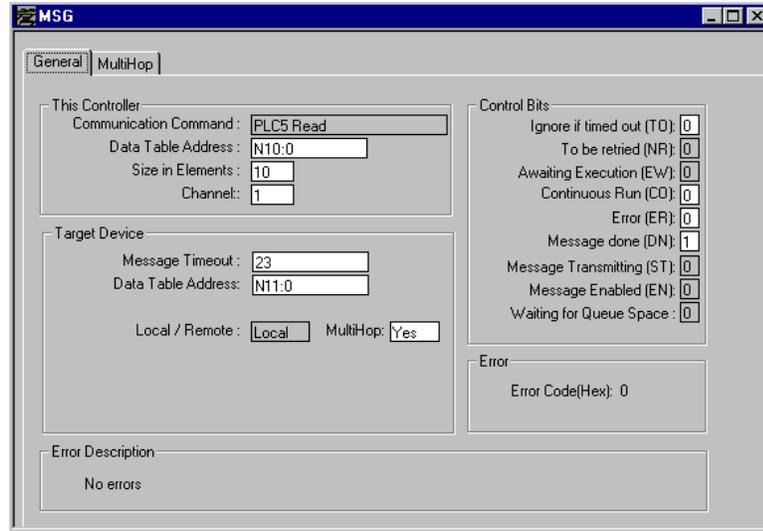
Set the IP address value to the module's Ethernet IP address. You must press the "Insert" key to add the second line for ControlLogix Backplane and set the slot number to zero.

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:

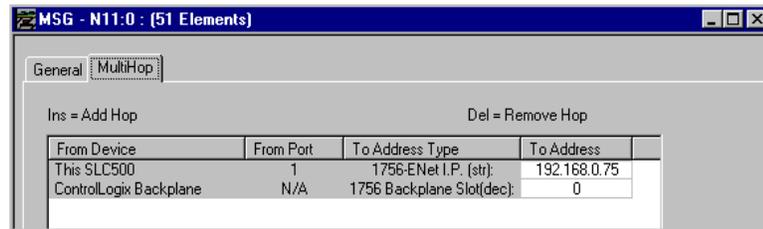


Set the **READ/WRITE** parameter to **READ**. The module supports a **TARGET DEVICE** parameter value of **500CPU** or **PLC5**. In order to complete the configuration of the MSG instruction, select the **SETUP SCREEN** area of the MSG object. This displays the following dialog box.



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.

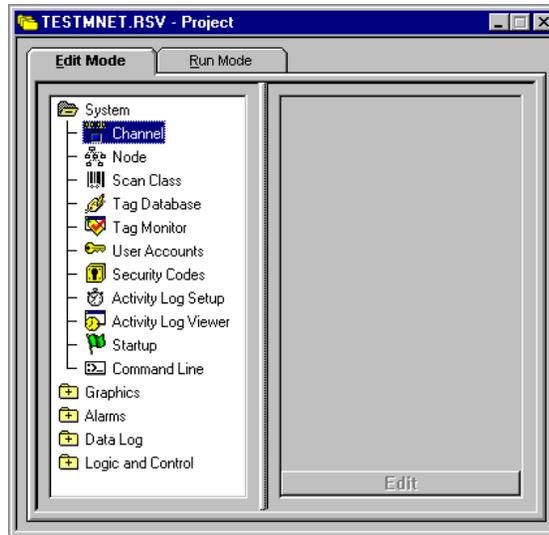


Set the IP address value to the module's Ethernet IP address. You must press the "Insert" key to add the second line for ControlLogix Backplane and set the slot number to zero.

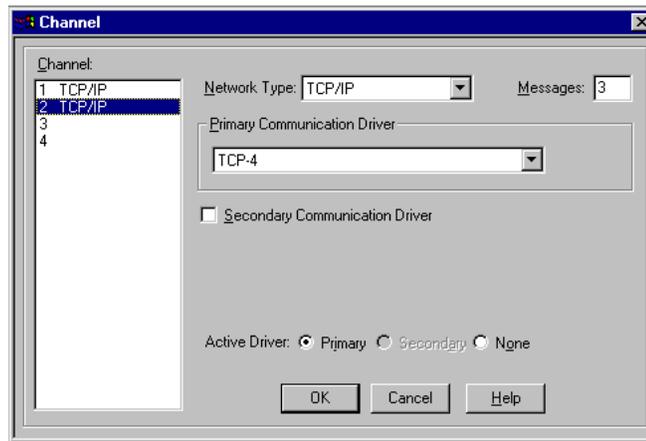
5.3.5 RSView Software

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:



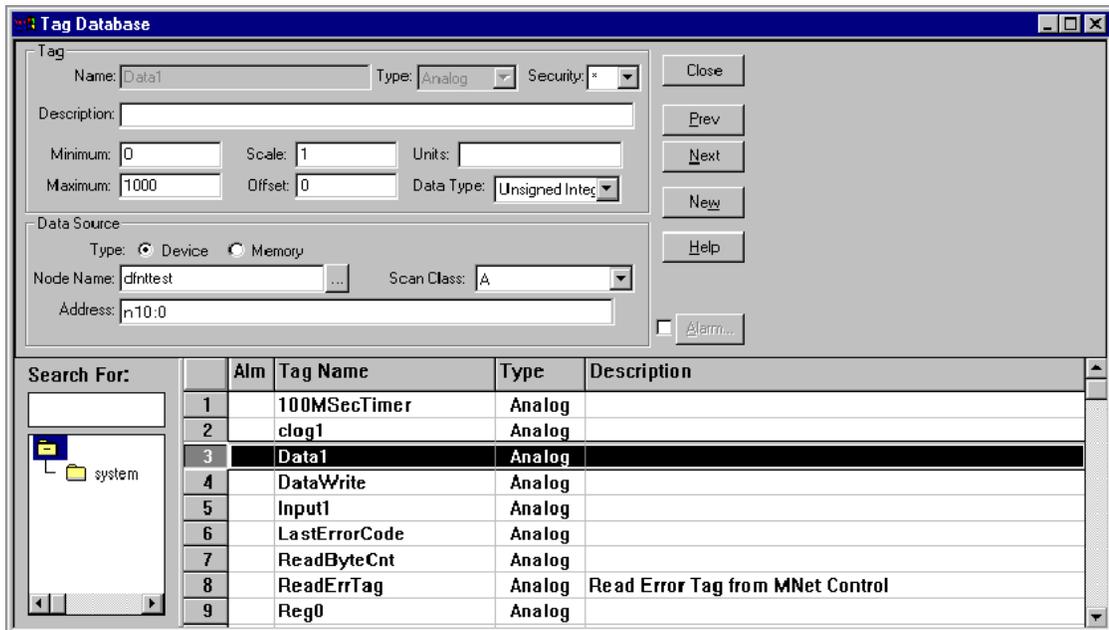
Select the **NETWORK TYPE** and **PRIMARY COMMUNICATION DRIVER** for the channel to be associated with the DFNT module. The **NETWORK TYPE** should be TCP/IP and the **PRIMARY COMMUNICATION DRIVER** name should match that set up in RSLinx. Click OK to save the information.

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

	Name	Data Source	Device	Channel	Station or Server	Timeout	OPC Server
1	clogix	Direct Driver	PLC-5 [Enhanced]	1 - TCP/IP	192.168.0.103	3.000	
2	dfnttest	Direct Driver	PLC-5	2 - TCP/IP	192.168.0.100	3.000	
3							
4							
5							
6							
7							
8							
9							

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.

5.4 Cable Connections

The MVI69-DFNT module has the following functional communication connections installed:

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

5.4.1 Ethernet Connection

The MVI69-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.

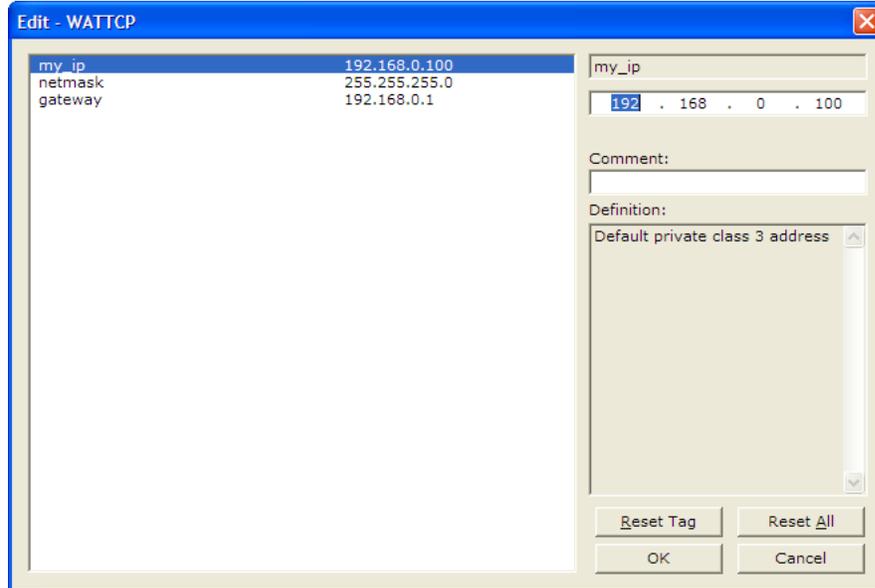
Note: Depending on hardware configuration, you may see more than one RJ45 port on the module. The Ethernet port is labeled *Ethernet*.

Warning: The MVI69-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

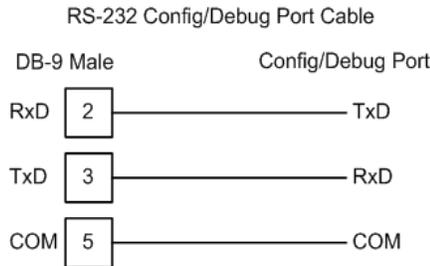
The wattcp.cfg file must be set up properly in order to use a TCP/IP network connection. You can view the current network configuration in *ProSoft Configuration Builder (PCB)*, as shown:



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 57).

5.4.2 RS-232 Configuration/Debug Port

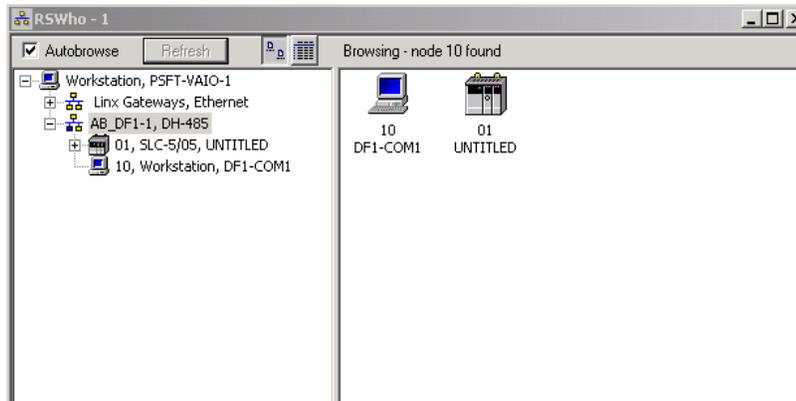
This port is physically an RJ45 connection. An RJ45 to DB-9 adapter cable is included with the module. This port permits a PC-based terminal emulation program to view configuration and status data in the module and to control the module. The cable pinout for communications on this port is shown in the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



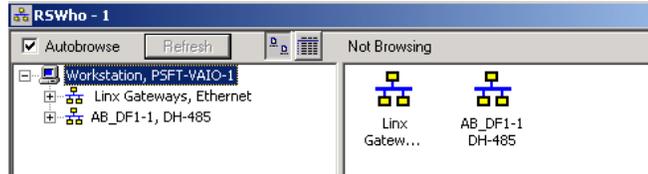
Disabling the RSLinx Driver for the Com Port on the PC

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.



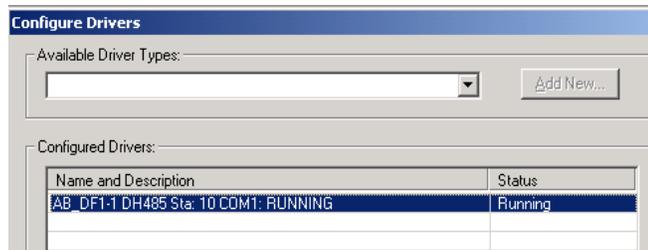
- 3 Notice how the DF1 driver is opened, and the driver is looking for a processor on node 1. If the network is being browsed, then you will not be able to stop this driver. To stop the driver your *RSWho* screen should look like this:



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



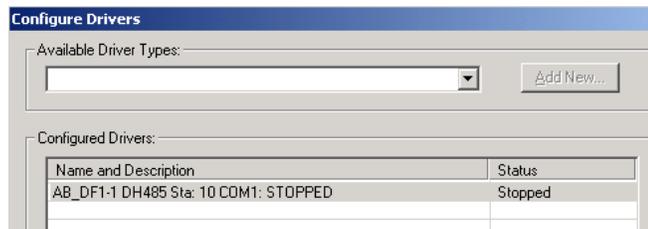
- 4 When you have verified that the driver is not being browsed, go to **COMMUNICATIONS > CONFIGURE DRIVERS**. You may see something like this:



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:



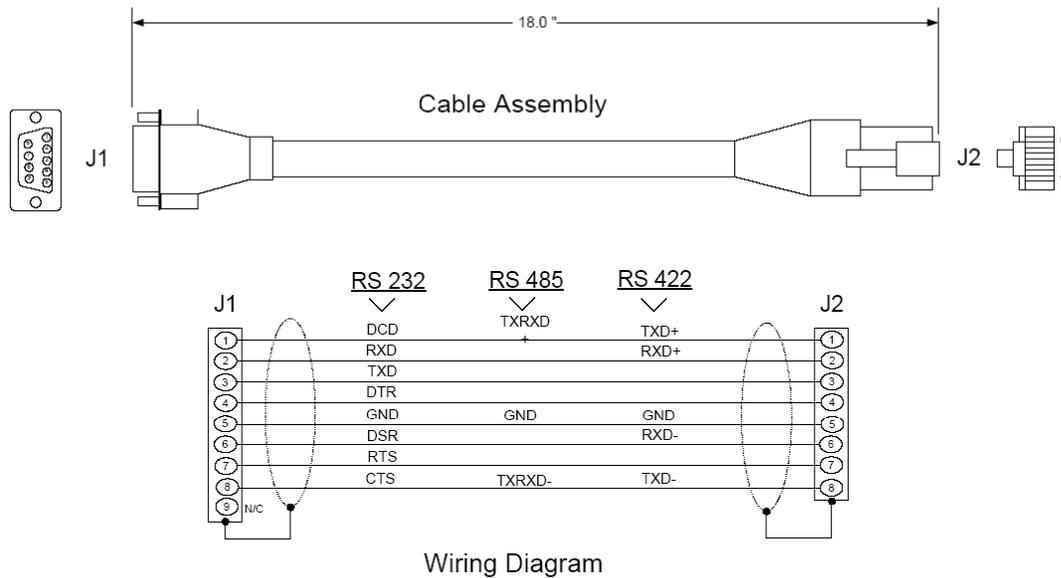
- 5 After you have stopped the driver you will see the following.



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.

5.4.3 DB9 to RJ45 Adaptor (Cable 14)



5.5 Error Codes

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

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

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

5.5.1 Local STS Error Codes

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

5.5.2 Remote STS Error Codes

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

5.5.3 Errors When EXT STS Is Present

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

5.5.4 Module Specific Error (not 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

5.6 TCP/IP Interface Errors

5.6.1 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.

5.6.2 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

5.6.3 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

5.6.4 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

5.7 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			

5.8 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.

5.8.1 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.

5.8.2 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.

5.8.3 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.

5.8.4 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.

5.8.5 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.

**5.8.6 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.

**5.8.7 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.

**5.8.8 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.

**5.8.9 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.

**5.8.10 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.

**5.8.11 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.

5.8.12 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.

5.8.13 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.

5.8.14 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.

5.8.15 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.

**5.8.16 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.

5.9 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:

5.9.1 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.

6 Support, Service & Warranty

In This Chapter

- ❖ Contacting Technical Support 163
- ❖ Warranty Information 164

6.1 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

6.2 Warranty Information

For complete details regarding ProSoft Technology's TERMS & CONDITIONS OF SALE, WARRANTY, SUPPORT, SERVICE AND RETURN MATERIAL AUTHORIZATION INSTRUCTIONS please see the documents on the Product DVD or go to www.prosoft-technology.com/warranty.

Documentation is subject to change without notice.

Index

I

[DFNT Client x Commands] • 33
[DFNT Client x] • 32
[Module] • 30

A

Adding the Module to an Existing CompactLogix Project • 51
Adding the Module to an Existing MicroLogix Project • 51, 55

B

Basic Command Set Functions • 34
Battery Life Advisory • 4
Block 250 Response (BTR Block) • 89
Block 251 Response (BTR Block) • 92
Block Transfer Backplane Data Transfer • 82
Block Transfer Size • 31

C

Cable Connections • 134
CIP Data Table Operations • 120
CIP Data Table Read • 122
CIP Data Table Write • 121
Clearing a Fault Condition • 60
Client Command List • 106
Client Configuration Error Word • 91
Client Driver • 105
Cold Boot Block (9999) • 98
Command Control Blocks (3000 to 3002) • 97
Command Entry Formats • 36, 38, 40
Command Error List Request Block (2000) • 96
Command Function Codes • 145
Command List • 33
Configuring Module Parameters • 28
Configuring the MVI69-DFNT Module • 25
Configuring the RSLinx Driver for the PC COM Port • 21
Connecting Your PC to the Module • 23
Connecting Your PC to the Processor • 19
Contacting Technical Support • 163
ControlLogix (CLX) Processor • 117

D

Data Flow between MVI69-DFNT Module and CompactLogix or MicroLogix Processor • 99
Database View Menu • 67, 70
DB9 to RJ45 Adaptor (Cable 14) • 138
DDE Connection • 110
Defining a DDE link in Excel • 113
DFNT Command Entry Form • 162

DFNT Server File Size • 31
Diagnostics and Troubleshooting • 9, 56, 57, 135
Disabling the RSLinx Driver for the Com Port on the PC • 136
Displaying the Current Page of Registers Again • 71
Downloading the Project to the Module Using a Serial COM Port • 42
Downloading the Sample Program to the Processor • 20

E

Encapsulated PCCC Messages • 117
Encapsulated PCCC Read Message • 119
Encapsulated PCCC Write Message • 117
Error Codes • 139
Errors When EXT STS Is Present • 141
Ethernet Configuration • 41
Ethernet Connection • 134
Ethernet LED Indicators • 59
Ethernet Port Configuration - wattcp.cfg • 135
EtherNet/IP (Explicit Messaging) Compatible Devices • 78
Exiting the Program • 70

F

Failure Flag Count • 30
Forward Open Response Errors • 143
Function Code #1 - Protected Write (Basic Command Set) • 147
Function Code #100 - Word Range Write (PLC-5 Command) (Binary Address) • 150
Function Code #101 - Word Range Read (PLC-5 Command) (Binary Address) • 151
Function Code #102 - Read-Modify-Write (PLC-5 Command) (Binary Address) • 152
Function Code #150 - Word Range Write (PLC-5 Command) (ASCII Address) • 153
Function Code #151 - Word Range Read (PLC-5 Command) (ASCII Address) • 154
Function Code #152 - Read-Modify-Write (PLC-5 Command) (ASCII Address) • 155
Function Code #2 - Unprotected Read (Basic Command Set) • 147
Function Code #3 - Protected Bit Write (Basic Command Set) • 148
Function Code #4 - Unprotected Bit Write (Basic Command Set) • 148
Function Code #5 - Unprotected Write (Basic Command Set) • 149
Function Code #501 - Protected Typed Logical Read (Two Address Fields) • 156
Function Code #502 - Protected Typed Logical Read (Three Address Fields) • 157
Function Code #509 - Protected Typed Logical Write (Two Address Fields) • 158
Function Code #510 - Protected Typed Logical Write (Three Address Fields) • 159
Function Code #511 - Protected Typed Logical Write with Mask (Three Address Fields) • 160

Functional Overview • 81
Functional Specifications • 80

G

General Command Structure • 146
General Concepts • 81
General Specifications • 78
Guide to the MVI69-DFNT User Manual • 9

H

Hardware Specifications • 79
How Data is Transferred • 83
How to Contact Us • 2

I

Important Installation Instructions • 3
Importing the Ladder Rung - MVI69-DFNT • 45
Initializing Output Data • 30
Installing ProSoft Configuration Builder Software • 14
Installing the Module • 16

K

Keystrokes • 65

L

Ladder Logic • 43
LED Status Indicators • 58
Local STS Error Codes • 139

M

Main Logic Loop • 81
Main Menu • 66
Markings • 4
Master Command Error List Menu • 73
Master Command List Menu • 74
MicroLogix Processor Specifics • 161
Minimum Command Delay • 32
Module Control Blocks • 89
Module Data • 43
Module Power Up • 81
Module Specific Error (not DFNT Compliant) • 142
Moving Back Through 5 Pages of Commands • 73
Moving Back Through 5 Pages of Registers • 71
Moving Forward (Skipping) Through 5 Pages of Commands • 73
Moving Forward (Skipping) Through 5 Pages of Registers • 71
MVI (Multi Vendor Interface) Modules • 3

N

Navigation • 65
Network Menu • 70, 75
Normal Data Transfer • 83

O

OPC Connection • 114
Opening the Client Command Error List Menu • 67

Opening the Client Command List Menu • 67
Opening the Database View Menu • 67
Opening the Network Menu • 70
Output Data Initialization Request Blocks (1000 to 1066) • 95

P

Package Contents • 13
PCCC Response Errors • 144
Pinouts • 3, 134, 138
PLC-5 Command Set Functions • 34
PLC5 Processor • 125
PLC5 Read Commands • 126
PLC5 Write Commands • 125
Printing a Configuration File • 29
Product Specifications • 9, 77
ProSoft Technology® Product Documentation • 2

R

Read Block • 83
Read Register Count • 31
Read Register Start • 31
Reading Status Data from the Module • 76
Redisplaying the Current Page • 73, 74
Reference • 9, 77
Register Session Response Errors • 143
Remote STS Error Codes • 140
Renaming an Object • 28
Resetting Diagnostic Data • 68
Response Timeout • 32
Retry Count • 32
Returning to the Main Menu • 72, 73, 74, 75
RS-232 Configuration/Debug Port • 136
RSLinx Software • 107
RSView Software • 131

S

Server Driver • 99, 107
Setting Jumpers • 15
Setting Module Parameters • 28
Setting Up the Project • 26
SLC 5/05 Processor • 128
SLC File Types • 161
SLC5/05 Read Commands • 129
SLC5/05 Write Commands • 128
SLC-500 Command Set Functions • 35
Start Here • 9, 11
Status Data Request Blocks (250 and 251) • 89
Support, Service & Warranty • 9, 163
System Requirements • 12

T

TCP/IP Interface Errors • 143
Timeout Errors • 143
Transferring the Configuration File from The Module to the PC • 67
Transferring the Configuration File from the PC to the Module • 67

Transferring WATTCP.CFG to the Module • 75
Transferring WATTCP.CFG to the PC • 75
Troubleshooting • 60

U

Using ProSoft Configuration Builder • 25
Using ProSoft Configuration Builder (PCB) for
Diagnostics • 62
Using the Diagnostic Window in ProSoft Configuration
Builder • 62
Using the RSLogix 5000 v16 Add-On Instruction • 44

V

Viewing Block Transfer Statistics • 66
Viewing Client Communication Status • 69
Viewing Client Configuration • 69
Viewing Data in ASCII (Text) Format • 72
Viewing Data in Decimal Format • 71
Viewing Data in Floating-Point Format • 72
Viewing Data in Hexadecimal Format • 71
Viewing Module Configuration • 67
Viewing Register Pages • 70
Viewing Server Status Data • 69
Viewing the Next 50 Commands • 74
Viewing the Next Page of Commands • 73, 74
Viewing the Next Page of Registers • 71
Viewing the Previous 50 Commands • 74
Viewing the Previous Page of Commands • 73, 74
Viewing the Previous Page of Registers • 71
Viewing the WATTCP.CFG File on the module • 75
Viewing Version Information • 68

W

Warm Boot Block (9998) • 98
Warm Booting the Module • 68
Warnings • 3
Warranty Information • 164
Write Block • 83
Write Register Count • 31
Write Register Start • 31

Y

Your Feedback Please • 2