




ProTalk[®]
PTQ-DFNT

Quantum Platform

EtherNet/IP Network Interface Module
for Quantum

September 10, 2010

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 © 2010 ProSoft Technology, Inc., all rights reserved.

PTQ-DFNT User Manual

September 10, 2010

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

ProSoft Technology[®] Product Documentation

In an effort to conserve paper, ProSoft Technology no longer includes printed manuals with our product shipments. User Manuals, Datasheets, Sample Ladder Files, and Configuration Files are provided on the enclosed CD-ROM, and are available at no charge from our web site: www.prosoft-technology.com

Printed documentation is available for purchase. Contact ProSoft Technology for pricing and availability.

North America: +1.661.716.5100

Asia Pacific: +603.7724.2080

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

Latin America: +1.281.298.9109

Information for ProTalk® Product Users

The statement "power, input and output (I/O) wiring must be in accordance with Class I, Division 2 wiring methods Article 501-10(b) of the National Electrical Code, NFPA 70 for installations in the U.S., or as specified in section 18-1J2 of the Canadian Electrical Code for installations within Canada and in accordance with the authority having jurisdiction".

The following or equivalent warnings shall be included:

- 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 Wiring Modules, and
- C** Warning - Explosion Hazard - Do not Disconnect Equipment unless Power has been switched Off or the Area is known to be Nonhazardous.
- D** Caution: The Cell used in this Device may Present a Fire or Chemical Burn Hazard if Mistreated. Do not Disassemble, Heat above 100 °C (212 °F) or Incinerate.

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.

Electrical Ratings

- Backplane Current Load: 1100 mA maximum @ 5 Vdc ± 5%
- Operating Temperature: 0 °C to 60 °C (32 °F to 140 °F)
- Storage Temperature: -40 °C to 85 °C (-40 °F to 185 °F)
- Shock: 30 g operational; 50 g non-operational; Vibration: 5 g from 10 to 150 Hz
- Relative Humidity: 5% to 95% (without condensation)
- All phase conductor sizes must be at least 1.3 mm(squared) and all earth ground conductors must be at least 4mm(squared).

Markings:

CSA/cUL	C22.2 No. 213-1987
CSA CB Certified	IEC61010
ATEX	EN60079-0 Category 3, Zone 2 EN60079-15

**Important Notice:**

CAUTION: THE CELL USED IN THIS DEVICE MAY PRESENT A FIRE OR CHEMICAL BURN HAZARD IF MISTREATED. DO NOT DISASSEMBLE, HEAT ABOVE 100 °C (212 °F) OR INCINERATE.

Maximum battery load = 200 μ A.

Maximum battery charge voltage = 3.4 Vdc.

Maximum battery charge current = 500 μ A.

Maximum battery discharge current = 30 μ A.

Contents

Your Feedback Please.....	2
How to Contact Us.....	2
ProSoft Technology® Product Documentation.....	2
Information for ProTalk® Product Users.....	3
Warnings.....	3
Important Notice:.....	4

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

1 Start Here **11**

1.1	Hardware and Software Requirements.....	12
1.1.1	Package Contents.....	12
1.1.2	Quantum Hardware.....	12
1.1.3	PC and PC Software.....	13
1.2	Installing ProSoft Configuration Builder Software.....	14

2 Configuring the Processor with Unity Pro **15**

2.1	Creating a New Project.....	16
2.2	Adding the PTQ Module to the Project.....	18
2.3	Building the Project.....	19
2.4	Connect Your PC to the Processor.....	20
2.4.1	Connecting to the Processor with TCP/IP.....	22
2.5	Downloading the Project to the Processor.....	23

3 Configuring the Processor with Concept **25**

3.1	Information for Concept Version 2.6 Users.....	26
3.1.1	Installing MDC Configuration Files.....	26
3.2	Creating a New Project.....	28
3.3	Adding the PTQ Module to the Project.....	30
3.4	Setting up Data Memory in Project.....	32
3.5	Downloading the Project to the Processor.....	34
3.6	Verifying Successful Download.....	36

4 Configuring the Processor with ProWORX **39**

5 Setting Up the ProTalk Module **43**

5.1	Installing the ProTalk Module in the Quantum Rack.....	44
5.1.1	Verifying Jumper Settings.....	44
5.1.2	Installing the ProTalk Module in the Quantum Rack.....	44
5.2	Connect the PC to the ProTalk Configuration/Debug Port.....	46
5.3	Verifying Communication Between the Processor and the Module.....	48

6	Configuring the Module	49
6.1	Using ProSoft Configuration Builder	50
6.1.1	Set Up the Project.....	50
6.1.2	Renaming PCB Objects	51
6.2	[Module]	53
6.3	[Backplane Data Exchange]	54
6.3.1	Setting Up Command Function 1 (Read Data from the Quantum).....	55
6.3.2	Setting Up Command Function 2 (Write Data to the Quantum)	56
6.3.3	Setting Up Command Function 3 (Special Functions).....	57
6.4	[DFNT Client x]	64
6.4.1	Minimum Command Delay	64
6.4.2	Response Timeout.....	64
6.4.3	Retry Count.....	64
6.5	[DFNT Client x Commands].....	65
6.5.1	Command List.....	65
6.5.2	Command Entry Formats.....	68
6.6	Ethernet Configuration.....	72
6.7	Downloading the Project to the Module Using a Serial COM port.....	73
7	Diagnostics and Troubleshooting	75
7.1	LED Status Indicators	76
7.1.1	Ethernet LED Indicators.....	76
7.1.2	Clearing a Fault Condition	77
7.1.3	Troubleshooting	77
7.1.4	Error Status Table.....	78
7.2	Using ProSoft Configuration Builder (PCB) for Diagnostics	79
7.2.1	Using the Diagnostic Window in ProSoft Configuration Builder	79
7.2.2	Navigation	81
7.2.3	Main Menu	81
7.2.4	Database View Menu.....	86
7.2.5	Master Command Error List Menu.....	88
7.2.6	Network Menu.....	89
7.3	Reading Status Data from the Module	91
8	Reference	93
8.1	Product Specifications	94
8.1.1	EtherNet/IP (Explicit Messaging) Compatible Devices.....	94
8.1.2	General Specifications	94
8.1.3	Hardware Specifications	95
8.1.4	Functional Specifications - PTQMDC/DFNT	96
8.2	Functional Overview	97
8.2.1	Backplane Data Exchange	97
8.2.2	DFNT Server.....	102
8.2.3	DFNT Client	104
8.3	PTQ-DFNT Error Status Table	106
8.4	Cable Connections	107
8.4.1	Ethernet Connection	107
8.4.2	RS-232 Configuration/Debug Port.....	108
8.5	PTQ-DFNT Status Data.....	109
8.5.1	HTTP Port 80 Status.....	109

8.5.2	DFNT Client Status	110
8.5.3	Block Status	110
8.5.4	Response Block (251)	110
8.5.5	DFNT Server 0 Status	111
8.5.6	DFNT Server 1 Status	112
8.6	Error Codes	113
8.6.1	Local STS Error Codes	113
8.6.2	Remote STS Error Codes	114
8.6.3	Errors When EXT STS Is Present.....	115
8.6.4	Module Specific Error (not DFNT Compliant).....	116
8.7	TCP/IP Interface Errors	117
8.7.1	Timeout Errors.....	117
8.7.2	Register Session Response Errors	117
8.7.3	Forward Open Response Errors	117
8.7.4	PCCC Response Errors	118
8.8	Command Function Codes.....	119
8.9	General Command Structure	120
8.9.1	Function Code #1 - Protected Write (Basic Command Set)	121
8.9.2	Function Code #2 - Unprotected Read (Basic Command Set)	121
8.9.3	Function Code #3 - Protected Bit Write (Basic Command Set)	122
8.9.4	Function Code #4 - Unprotected Bit Write (Basic Command Set)	123
8.9.5	Function Code #5 - Unprotected Write (Basic Command Set)	124
8.9.6	Function Code #100 - Word Range Write (PLC-5 Command) (Binary Address)..	125
8.9.7	Function Code #101 - Word Range Read (PLC-5 Command) (Binary Address) .	126
8.9.8	Function Code #102 - Read-Modify-Write (PLC-5 Command) (Binary Address) .	127
8.9.9	Function Code #150 - Word Range Write (PLC-5 Command) (ASCII Address) ..	128
8.9.10	Function Code #151 - Word Range Read (PLC-5 Command) (ASCII Address) ..	129
8.9.11	Function Code #152 - Read-Modify-Write (PLC-5 Command) (ASCII Address) ..	130
8.9.12	Function Code #501 - Protected Typed Logical Read (Two Address Fields).....	131
8.9.13	Function Code #502 - Protected Typed Logical Read (Three Address Fields)	132
8.9.14	Function Code #509 - Protected Typed Logical Write (Two Address Fields)	133
8.9.15	Function Code #510 - Protected Typed Logical Write (Three Address Fields)	134
8.9.16	Function Code #511 - Protected Typed Logical Write with Mask (Three Address Fields)	135
8.10	PLC-5 Processor Specifics.....	136
8.10.1	PLC-5 Sub-Element Codes	136
8.11	SLC Processor Specifics.....	138
8.11.1	SLC File Types	138
8.12	MicroLogix Processor Specifics	139
8.12.1	SLC File Types	139
8.13	ControlLogix Processor Specifics	140
8.14	DFNT Command Entry Form	141
8.15	Frequently Asked Questions	142

9 Support, Service & Warranty 145

	Contacting Technical Support.....	145
9.1	Return Material Authorization (RMA) Policies and Conditions.....	147
9.1.1	All Product Returns:	147
9.1.2	Procedures for Return of Units Under Warranty:	148
9.1.3	Procedures for Return of Units Out of Warranty:	148
9.2	LIMITED WARRANTY.....	149
9.2.1	What Is Covered By This Warranty	149

9.2.2	What Is Not Covered By This Warranty.....	150
9.2.3	Disclaimer Regarding High Risk Activities.....	150
9.2.4	Intellectual Property Indemnity	151
9.2.5	Disclaimer of all Other Warranties	151
9.2.6	Limitation of Remedies **	152
9.2.7	Time Limit for Bringing Suit.....	152
9.2.8	No Other Warranties.....	152
9.2.9	Allocation of Risks	152
9.2.10	Controlling Law and Severability	153

Index	155
--------------	------------

Guide to the PTQ-DFNT User Manual

Function		Section to Read	Details
Introduction (Must Do)	→	Start Here (page 11)	This section introduces the customer to the module. Included are: package contents, system requirements, hardware installation, and basic configuration.
Diagnostic and Troubleshooting	→	Diagnostics and Troubleshooting (page 75)	This section describes Diagnostic and Troubleshooting procedures.
Reference Product Specifications Functional Overview	→	Reference (page 93) Product Specifications (page 94) Functional Overview (page 97, page 87)	These sections contain general references associated with this product, Specifications, and the Functional Overview.
Support, Service, and Warranty Index	→	Support, Service and Warranty (page 145) Index	This section contains Support, Service and Warranty information. Index of chapters.

1 Start Here

In This Chapter

- ❖ Hardware and Software Requirements 12
- ❖ Installing ProSoft Configuration Builder Software 14

This guide is intended to guide you through the ProTalk module setup process, from removing the module from the box to exchanging data with the processor. In doing this, you will learn how to:

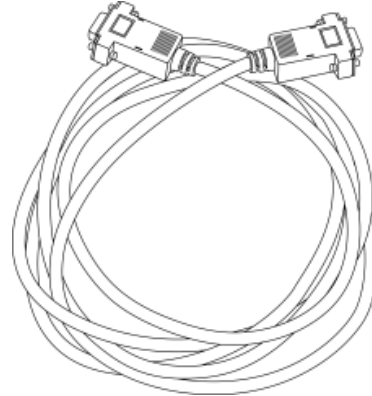
- Set up the processor environment for the PTQ module
- View how the PTQ module exchanges data with the processor
- Edit and download configuration files from your PC to the PTQ module
- Monitor the operation of the PTQ module

1.1 Hardware and Software Requirements

1.1.1 Package Contents



ProTalk Module



Null Modem Serial Cable



1454-9F DB-9 Female to 9 Pos Screw Terminal adapter (Serial protocol modules only)



ProSoft Solutions CD

Note: The DB-9 Female to 5 Pos Screw Terminal adapter is not required on Ethernet modules and is therefore not included in the carton with these types of modules.

1.1.2 Quantum Hardware

This guide assumes that you are familiar with the installation and setup of the Quantum hardware. The following should be installed, configured, and powered up before proceeding:

- Quantum Processor
- Quantum rack
- Quantum power supply
- Quantum Modbus Plus Network Option Module (NOM Module) (optional)
- Quantum to PC programming hardware
- NOM Ethernet or Serial connection to PC

1.1.3 PC and PC Software

ProSoft Technology recommends the following minimum hardware to use the module:

- Windows PC with 80486 based processor (Pentium preferred) with at least one COM, USB, or Ethernet port
- 1 megabyte of system memory
- Unity™ Pro PLC Programming Software, version 3.0 or later
or
Concept™ PLC Programming Software, version 2.6 or later
or
Other Quantum Programming Software

Note: ProTalk module configuration files are compatible with common Quantum programming applications, including Unity Pro and Concept. For all other programming applications, please contact technical support.

1.2 Installing ProSoft Configuration Builder Software

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

Installing ProSoft Configuration Builder from the ProSoft website

- 1 Open your web browser and navigate to *http://www.prosoft-technology.com/pcb*
- 2 Click the **DOWNLOAD HERE** link 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 CD-ROM*, included in the package with your module.

Installing ProSoft Configuration Builder from the Product CD-ROM

- 1 Insert the *ProSoft Solutions Product CD-ROM* into the CD-ROM 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 CD-ROM. You may wish to copy the files from the Utilities folder on the CD-ROM to a convenient location on your hard drive.

2 Configuring the Processor with Unity Pro

In This Chapter

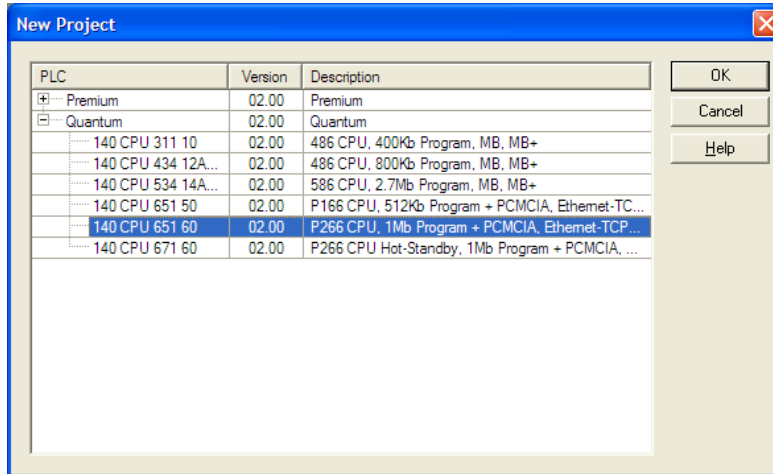
- ❖ Creating a New Project..... 16
- ❖ Adding the PTQ Module to the Project 18
- ❖ Building the Project 19
- ❖ Connect Your PC to the Processor 20
- ❖ Downloading the Project to the Processor..... 23

The following steps are designed to ensure that the processor (Quantum or Unity) is able to transfer data successfully with the PTQ module. As part of this procedure, you will use Unity Pro to create a project, add the PTQ module to the project, set up data memory for the project, and then download the project to the processor.

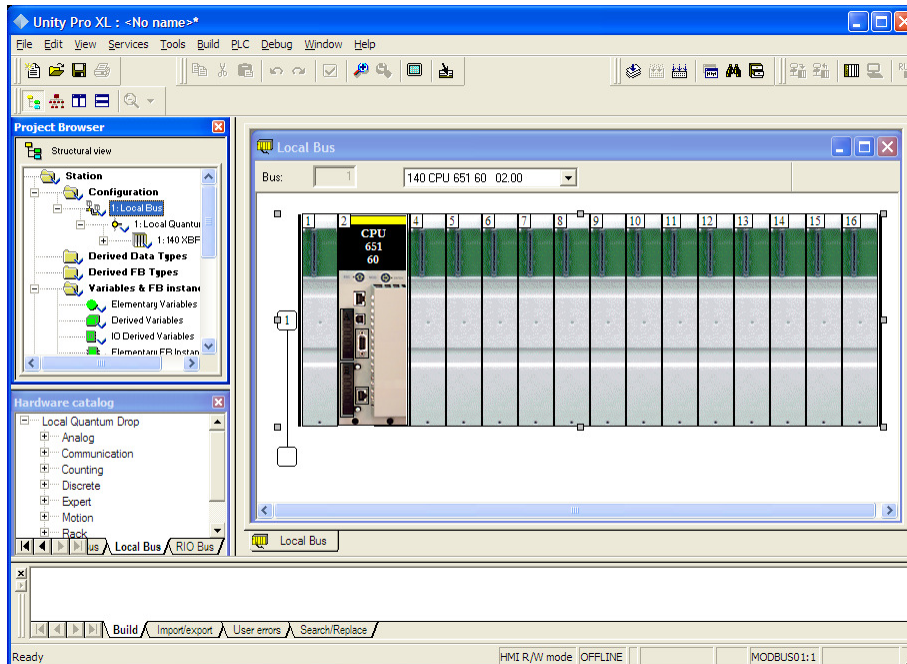
2.1 Creating a New Project

The first step is to open Unity Pro and create a new project.

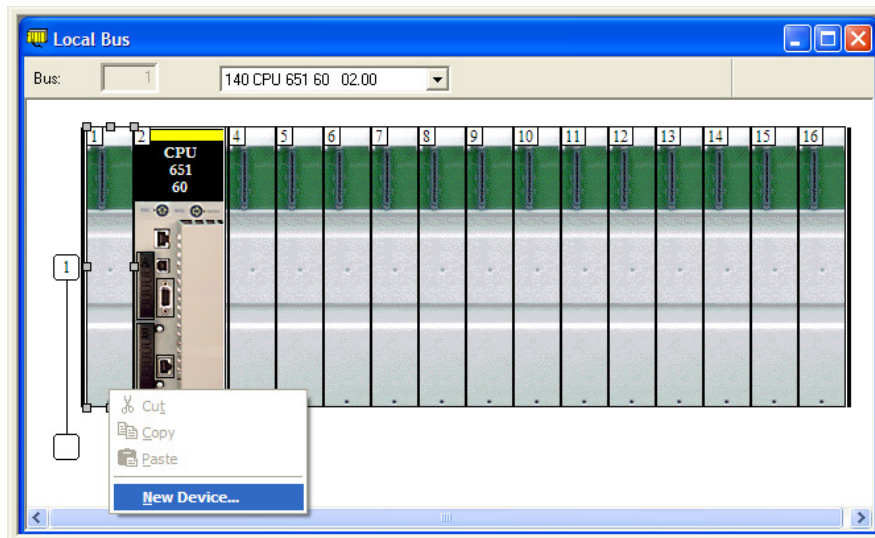
- 1 In the *New Project* dialog box, choose the CPU type. In the following illustration, the CPU is 140 CPU 651 60. Choose the processor type that matches your own hardware configuration, if it differs from the example. Click **OK** to continue.



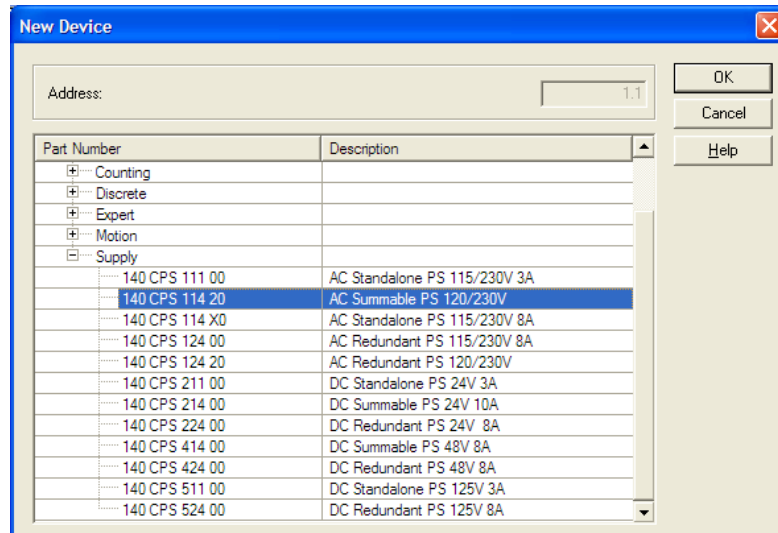
- 2 Next, add a power supply to the project. In the *Project Browser*, expand the *Configuration* folder, and then double-click the **1:LOCALBUS** icon. This action opens a graphical window showing the arrangement of devices in your Quantum rack.



- 3 Select the rack position for the power supply, and then click the right mouse button to open a shortcut menu. On the shortcut menu, choose **NEW DEVICE**.



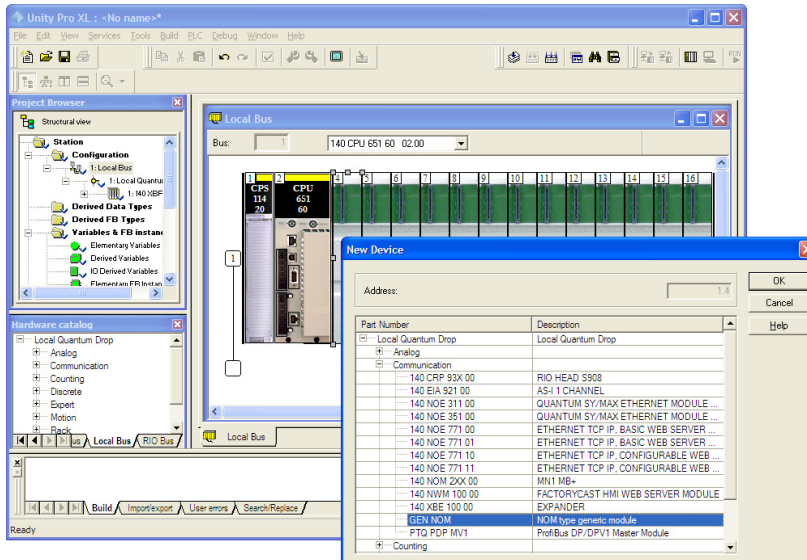
- 4 Expand the *Supply* folder, and then select your power supply from the list. Click **OK** to continue.



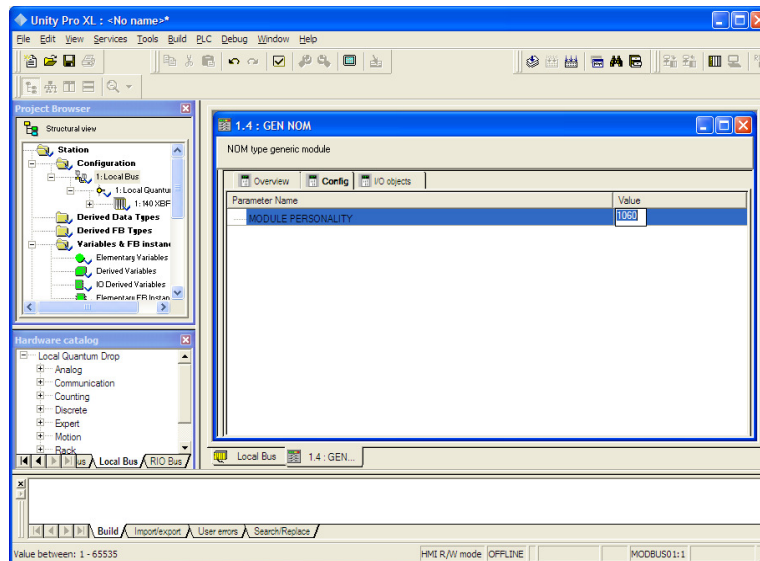
- 5 Repeat these steps to add any additional devices to your Quantum Rack.

2.2 Adding the PTQ Module to the Project

- 1 Expand the *Communication* tree, and select **GEN NOM**. This module type provides extended communication capabilities for the Quantum system, and allows communication between the PLC and the PTQ module without requiring additional programming.



- 2 Next, enter the module personality value. The correct value for ProTalk modules is 1060 decimal (0424 hex).



- 3 Before you can save the project in Unity Pro, you must validate the modifications. Open the **EDIT** menu, and then choose **VALIDATE**. If no errors are reported, you can save the project.
- 4 **SAVE** the project.

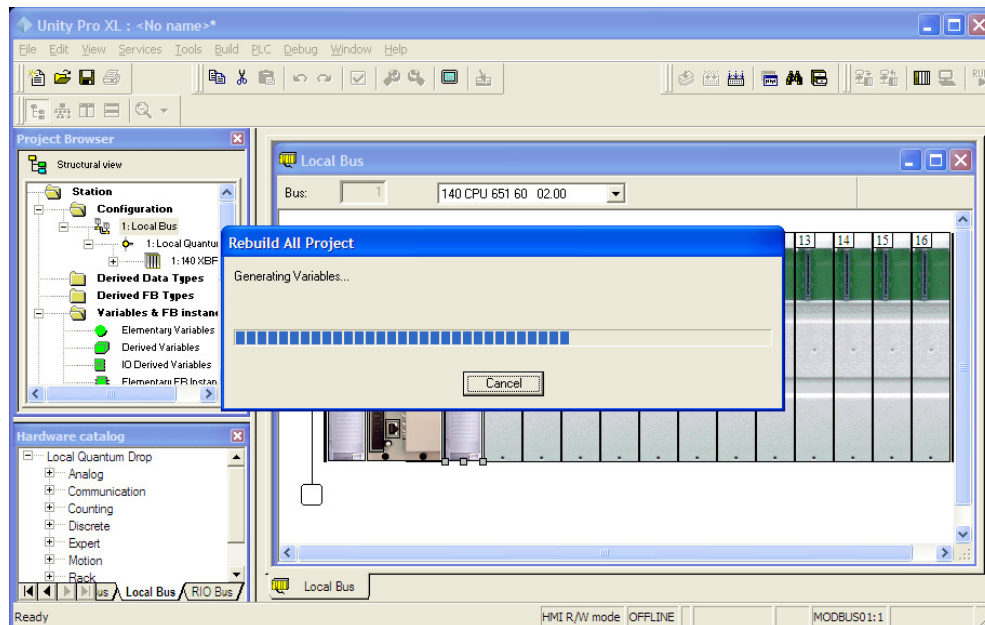
2.3 Building the Project

Whenever you update the configuration of your PTQ module or the processor, you must import the changed configuration from the module, and then build (compile) the project before downloading it to the processor.

Note: The following steps show you how to build the project in Unity Pro. This is not intended to provide detailed information on using Unity Pro, or debugging your programs. Refer to the documentation for your processor and for Unity Pro for specialized information.

To build (compile) the project

- 1 Review the elements of the project in the *Project Browser*.
- 2 When you are satisfied that you are ready to download the project, open the **BUILD** menu, and then choose **REBUILD ALL PROJECT**. This action builds (compiles) the project into a form that the processor can use to execute the instructions in the project file. This task may take several minutes, depending on the complexity of the project and the resources available on your PC.
- 3 As the project is built, Unity Pro reports its process in a *Progress* dialog box, with details appearing in a pane at the bottom of the window. The following illustration shows the build process under way.



After the build process is completed successfully, the next step is to download the compiled project to the processor.

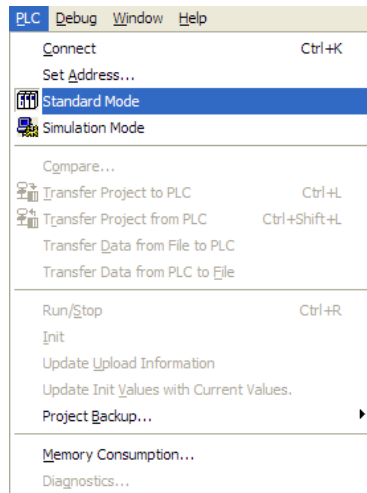
2.4 Connect Your PC to the Processor

The next step is to connect to the processor so that you can download the project file. The processor uses this project file to communicate over the backplane to modules identified in the project file.

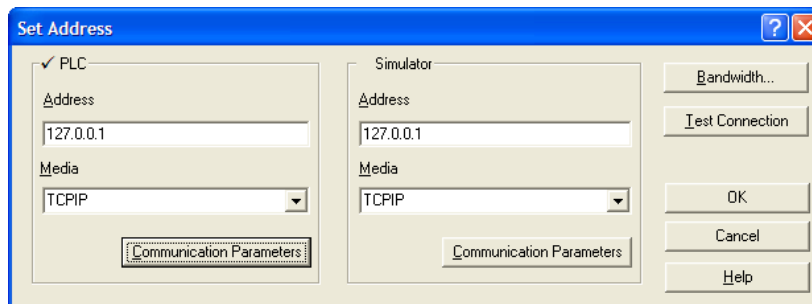
Note: If you have never connected from the PC to your processor before, you must verify that the necessary port drivers are installed and available to Unity Pro.

To verify address and driver settings in Unity Pro

- 1 Open the **PLC** menu, and choose **STANDARD MODE**. This action turns off the PLC Simulator, and allows you to communicate directly with the Quantum or Unity hardware.



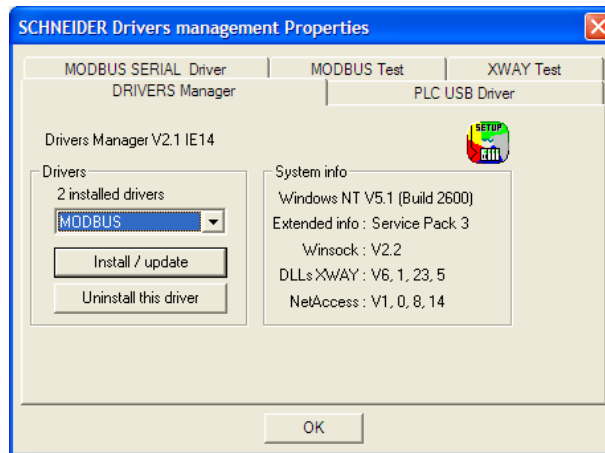
- 2 Open the **PLC** menu, and choose **SET ADDRESS...** This action opens the *Set Address* dialog box. Open the **MEDIA** dropdown list and choose the connection type to use (*TCPIP* or *USB*).



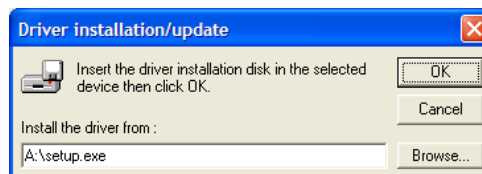
- 3 If the **MEDIA** dropdown list does not contain the connection method you wish to use, click the **COMMUNICATION PARAMETERS** button in the PLC area of the dialog box. This action opens the *PLC Communication Parameters* dialog box.



- 4 Click the **DRIVER SETTINGS** button to open the *SCHNEIDER Drivers management Properties* dialog box.



- 5 Click the **INSTALL/UPDATE** button to specify the location of the Setup.exe file containing the drivers to use. You will need your Unity Pro installation disks for this step.



- 6 Click the **BROWSE** button to locate the *Setup.exe* file to execute, and then execute the setup program. After the installation, restart your PC if you are prompted to do so. Refer to your Schneider Electric documentation for more information on installing drivers for Unity Pro.

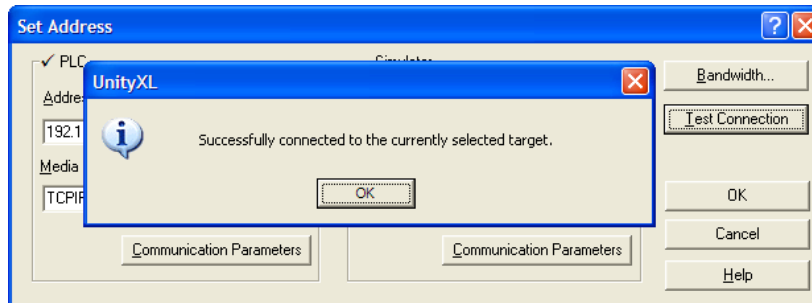
2.4.1 Connecting to the Processor with TCP/IP

The next step is to download (copy) the project file to the processor. The following steps demonstrate how to use an Ethernet cable connected from the Processor to your PC through an Ethernet hub or switch. Other connection methods may also be available, depending on the hardware configuration of your processor, and the communication drivers installed in Unity Pro.

- 1 If you have not already done so, connect your PC and the processor to an Ethernet hub.
- 2 Open the **PLC** menu, and then choose **SET ADDRESS**.

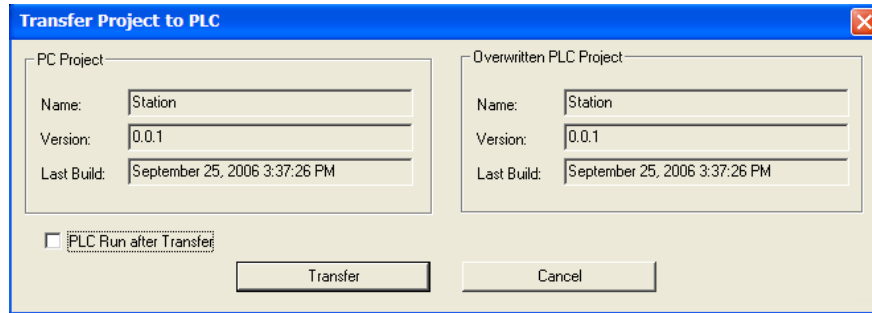
▪ **Important:** Notice that the *Set Address* dialog box is divided into two areas. Enter the address and media type in the *PLC* area of the dialog box, not the *Simulator* area.

- 3 Enter the IP address in the address field. In the **MEDIA** dropdown list, choose **TCP/IP**.
- 4 Click the **TEST CONNECTION** button to verify that your settings are correct.



2.5 Downloading the Project to the Processor

- 1 Open the **PLC** menu and then choose **CONNECT**. This action opens a *connection* between the Unity Pro software and the processor, using the address and media type settings you configured in the previous step.
- 2 On the **PLC** menu, choose **TRANSFER PROJECT TO PLC**. This action opens the *Transfer Project to PLC* dialog box. If you would like the PLC to go to *Run* mode immediately after the transfer is complete, select (check) the **PLC RUN AFTER TRANSFER** check box.



- 3 Click the **TRANSFER** button to download the project to the processor. As the project is transferred, Unity Pro reports its process in a *Progress* dialog box, with details appearing in a pane at the bottom of the window.

When the transfer is complete, place the processor in *Run* mode.

3 Configuring the Processor with Concept

In This Chapter

- ❖ Information for Concept Version 2.6 Users..... 26
- ❖ Creating a New Project..... 28
- ❖ Adding the PTQ Module to the Project 30
- ❖ Setting up Data Memory in Project..... 32
- ❖ Downloading the Project to the Processor..... 34
- ❖ Verifying Successful Download 36

The following steps are designed to ensure that the processor is able to transfer data successfully with the PTQ module. As part of this procedure, you will use Concept configuration software from Schneider Electric to create a project, add the PTQ module to the project, set up data memory for the project, and then download the project to the processor.

Important Note: Concept software does not report whether the PTQ module is present in the rack, and therefore is not able to report the health status of the module when the module is online with the Quantum processor. Please consider this when monitoring the status of the PTQ module.

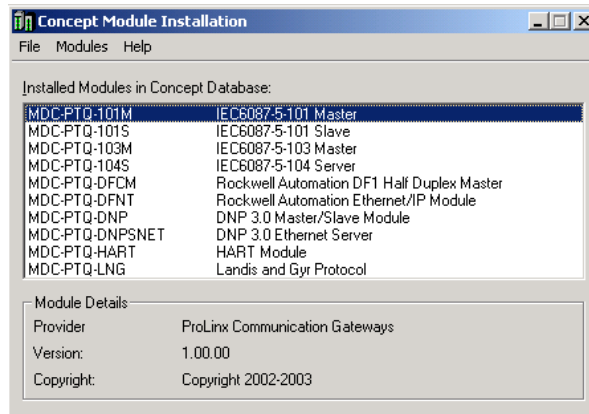
3.1 Information for Concept Version 2.6 Users

This guide uses Concept PLC Programming Software version 2.6 to configure the Quantum PLC. The ProTalk installation CD includes MDC module configuration files that help document the PTQ installation. Although not required, these files should be installed before proceeding to the next section.

3.1.1 Installing MDC Configuration Files

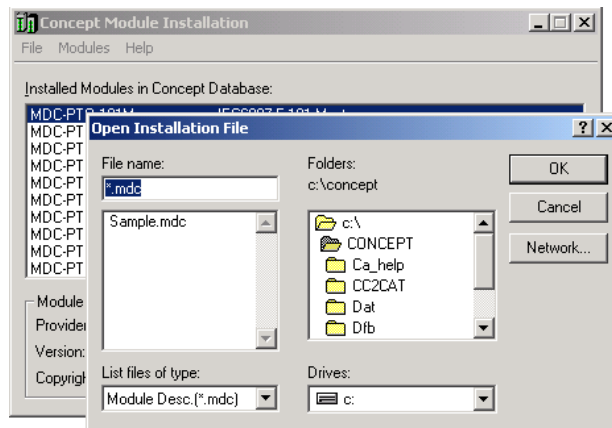
- 1 From a PC with Concept 2.6 installed, choose **START / PROGRAMS / CONCEPT / MODCONNECT TOOL**.

This action opens the *Concept Module Installation* dialog box.



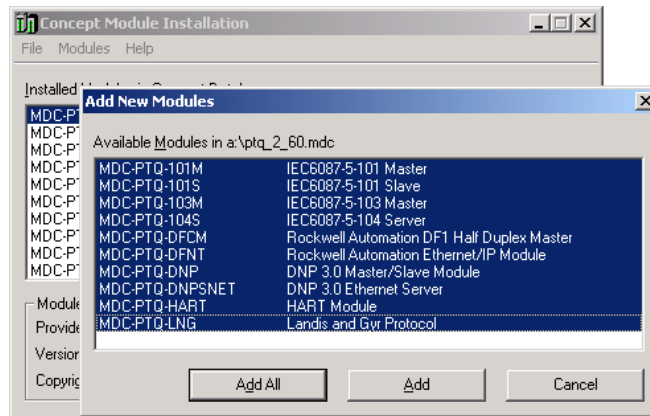
- 2 Choose **FILE / OPEN INSTALLATION FILE**.

This action opens the *Open Installation File* dialog box:



- 3 If you are using a Quantum processor, you will need the MDC files. In the *Open Installation File* dialog box, navigate to the *MDC Files* directory on the ProTalk CD.
- 4 Choose the MDC file and help file for your version of Concept:
 - Concept 2.6 users: select PTQ_2_60.mdc and PTQMDC.hlp
 - Concept 2.5 users: select PTQ_2_50.mdc and PTQMDC.hlp.

Select the files that go with the Concept version you are using, and then click **OK**. This action opens the *Add New Modules* dialog box.

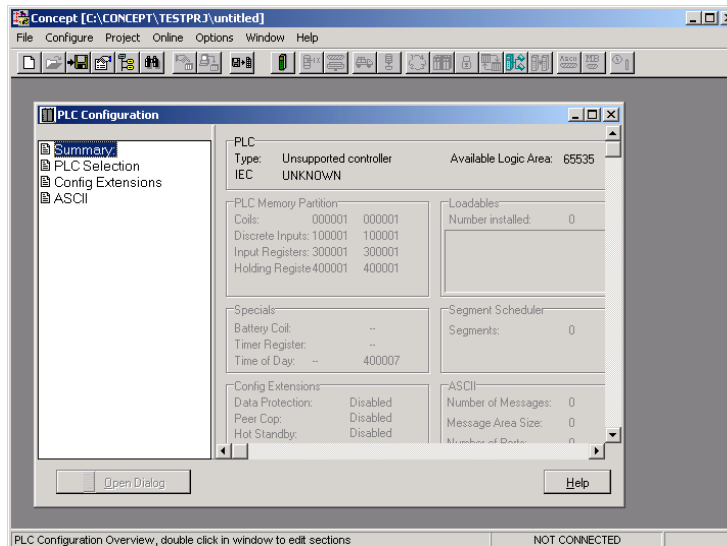


- 5 Click the **ADD ALL** button. A series of message boxes may appear during this process. Click **YES** or **OK** for each message that appears.
- 6 When the process is complete, open the **FILE** menu and choose **EXIT** to save your changes.

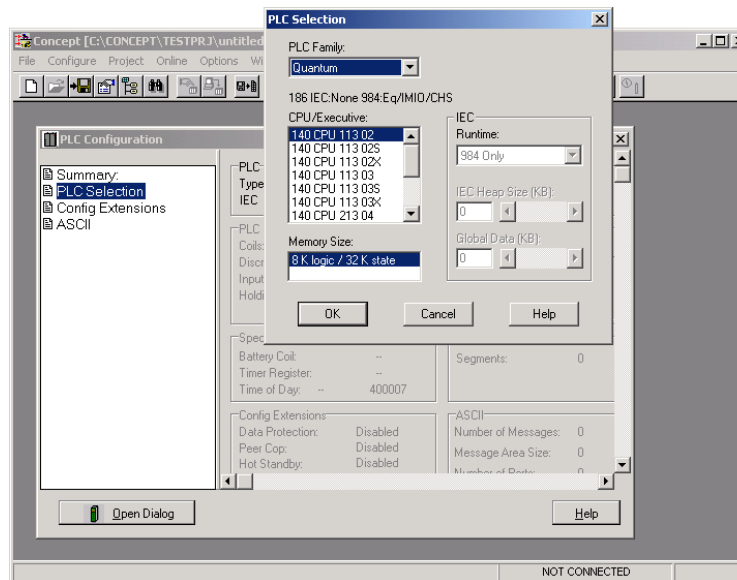
3.2 Creating a New Project

This phase of the setup procedure must be performed on a computer that has the Concept configuration software installed.

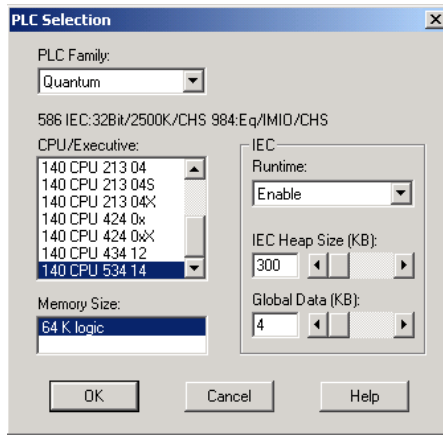
- 1 From your computer, choose **START / PROGRAMS / CONCEPT V2.6 XL.EN / CONCEPT**. This action opens the *Concept* window.
- 2 Open the File menu, and then choose **NEW PROJECT**. This action opens the *PLC Configuration* dialog box.



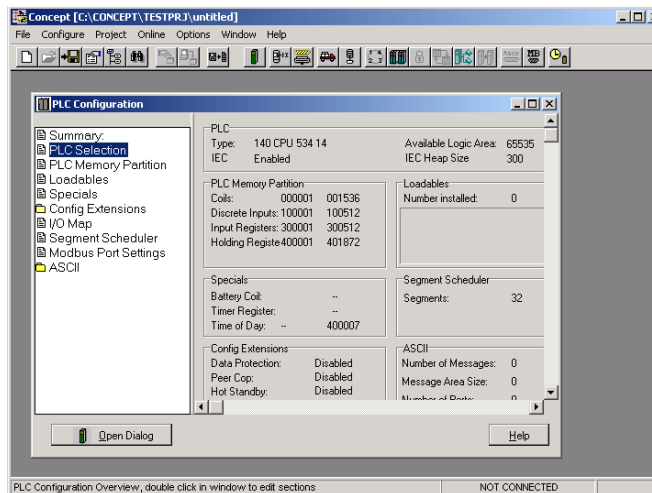
- 3 In the list of options on the left side of this dialog box, double-click the **PLC SELECTION** folder. This action opens the *PLC Selection* dialog box.



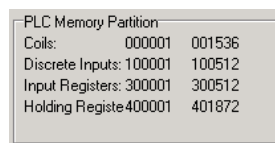
- In the *CPU/Executive* pane, use the scroll bar to locate and select the **PLC** to configure.



- Click **OK**. This action opens the *PLC Configuration* dialog box, populated with the correct values for the PLC you selected.

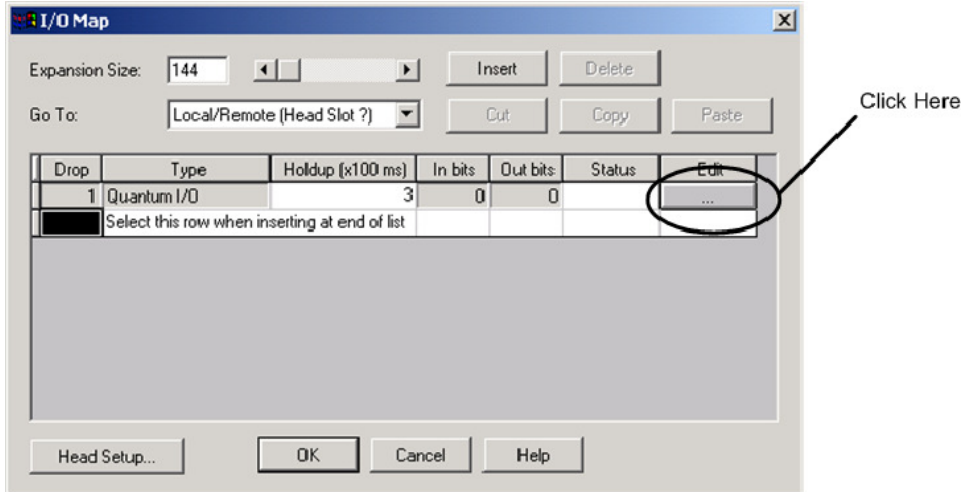


- Make a note of the holding registers for the module. You will need this information when you modify your application. The Holding Registers are displayed in the *PLC Memory Partition* pane of the *PLC Configuration* dialog box.

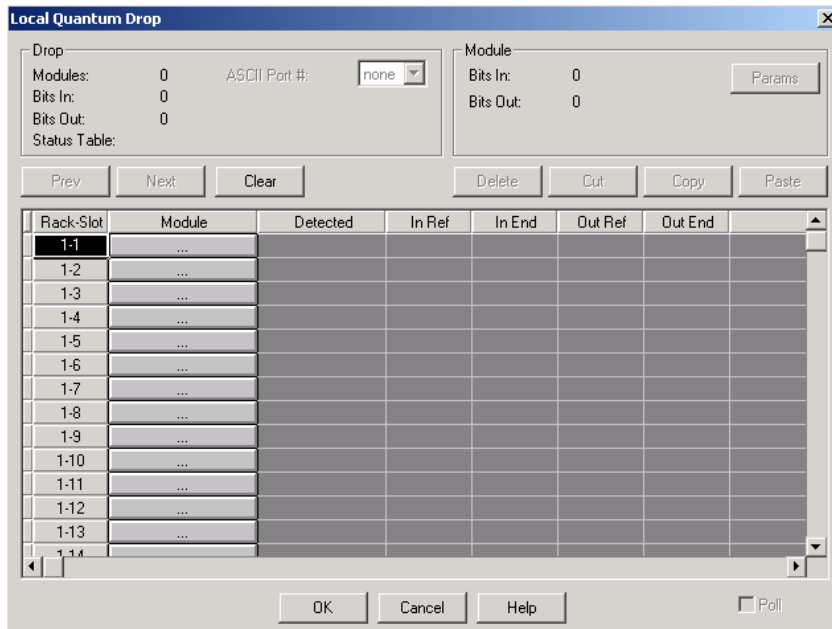


3.3 Adding the PTQ Module to the Project

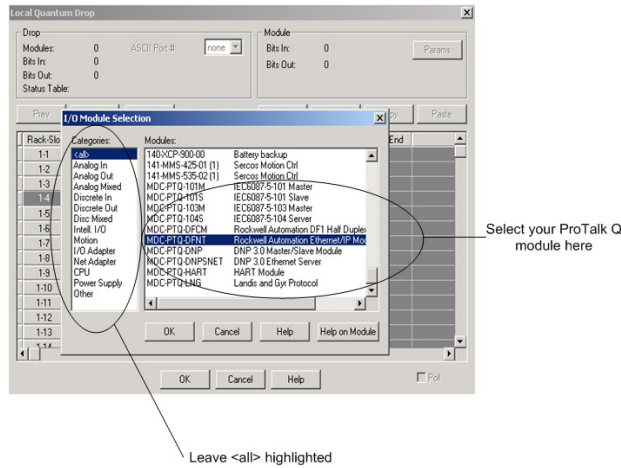
- 1 In the list of options on the left side of the *PLC Configuration* dialog box, double-click **I/O MAP**. This action opens the *I/O Map* dialog box.



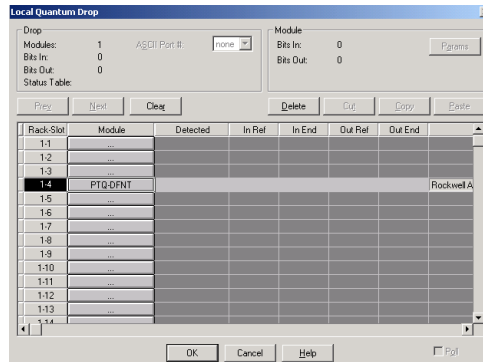
- 2 Click the **EDIT** button to open the *Local Quantum Drop* dialog box. This dialog box is where you identify rack and slot locations.



- Click the **MODULE** button next to the rack/slot position where the ProTalk module will be installed. This action opens the *I/O Module Selection* dialog box.



- In the *Modules* pane, use the scroll bar to locate and select the ProTalk module, and then click **OK**. This action copies the description of the ProTalk module next to the assigned rack and slot number of the *Local Quantum Drop* dialog box.



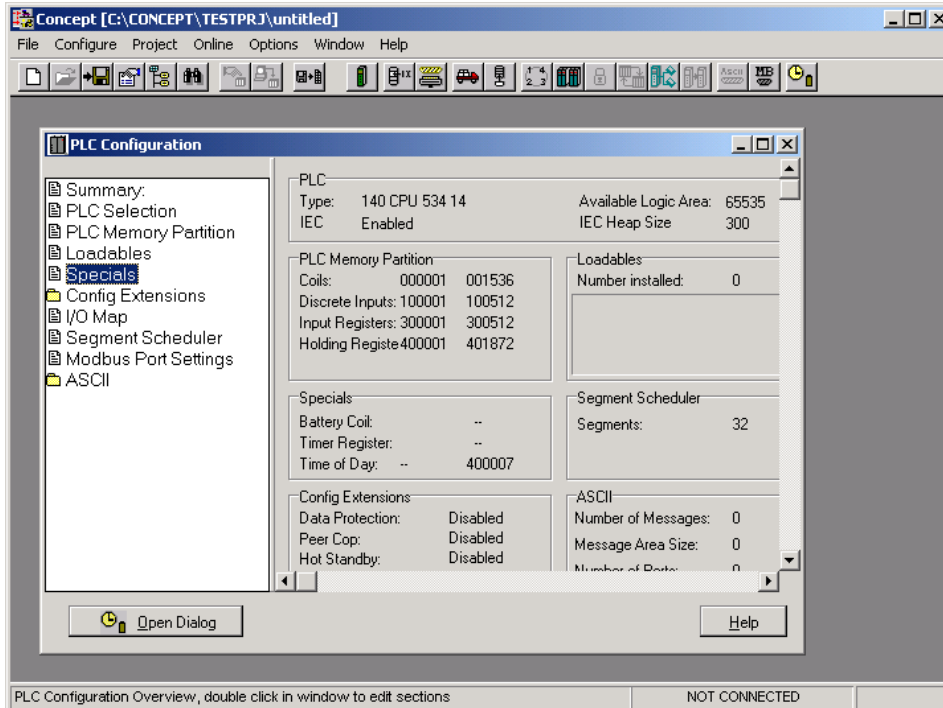
- Repeat steps 3 through 5 for each ProTalk module you plan to install. When you have finished installing your ProTalk modules, click **OK** to save your settings. Click **YES** to confirm your settings.

Tip: Select a module, and then click the Help on Module button for help pages.

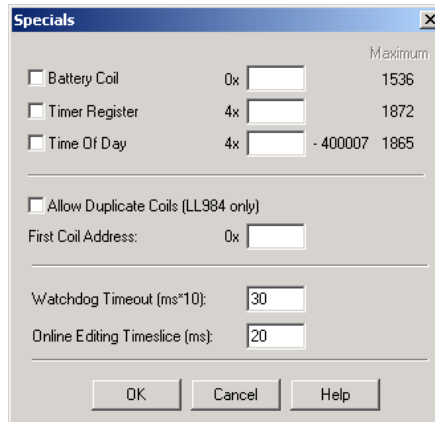


3.4 Setting up Data Memory in Project

- 1 In the list of options on the left side of the *PLC Configuration* dialog box, double-click **SPECIALS**.

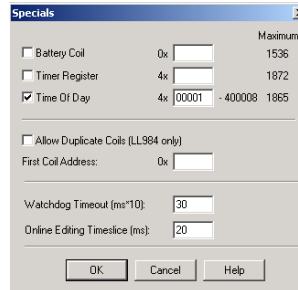


- 2 This action opens the *Specials* dialog box.



Selecting the Time of Day

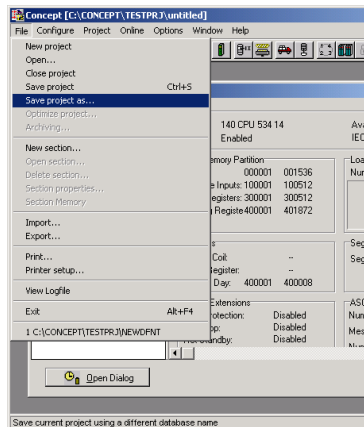
- 1 Select (check) the *Time of Day* box, and then enter the value 00001 as shown in the following illustration. This value sets the first time of day register to 400001.



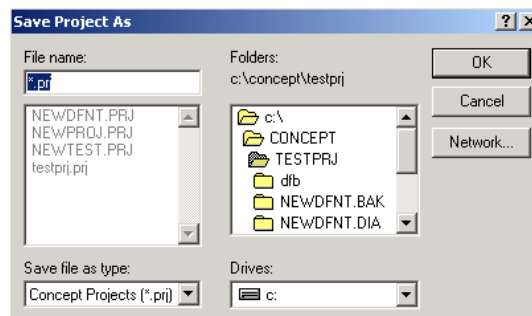
- 2 Click **OK** to save your settings and close the *Specials* dialog box.

Saving your project

- 1 In the *PLC Configuration* dialog box, choose **FILE / SAVE PROJECT AS**.



- 2 This action opens the *Save Project As* dialog box.

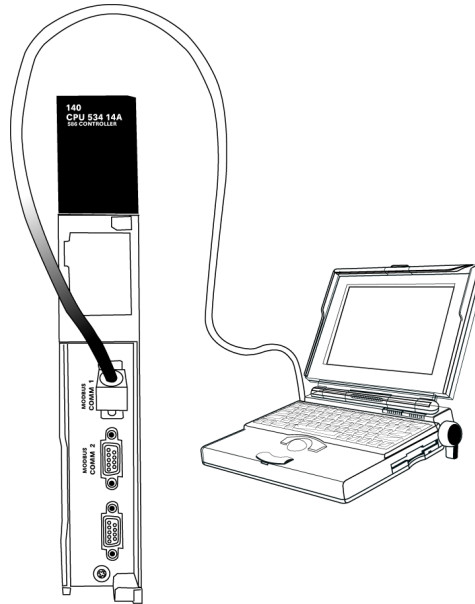


- 3 Name the project, and then click **OK** to save the project to a file.

3.5 Downloading the Project to the Processor

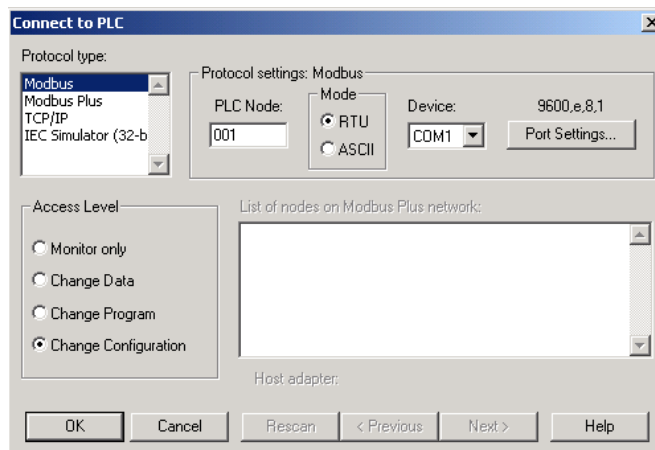
Next, download (copy) the project file to the Quantum Processor.

- 1 Use the null modem cable to connect your PC's serial port to the Quantum processor, as shown in the following illustration.



Note: You can use a Modbus Plus Network Option Module (NOM Module) module in place of the serial port if necessary.

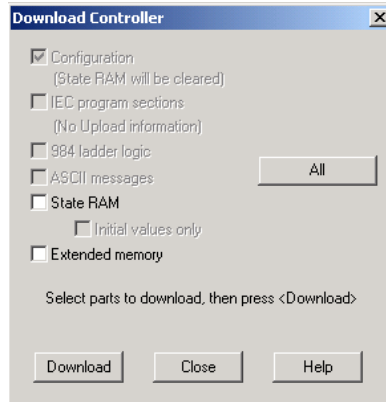
- 2 Open the **PLC** menu, and then choose **CONNECT**.
- 3 In the *PLC Configuration* dialog box, open the **ONLINE** menu, and then choose **CONNECT**. This action opens the *Connect to PLC* dialog box.



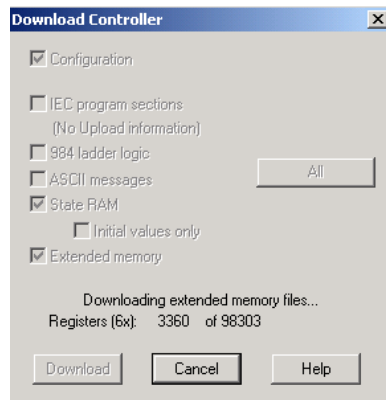
- 4 Leave the default settings as shown and click **OK**.

Note: Click **OK** to dismiss any message boxes that appear during the connection process.

- 5 In the *PLC Configuration* window, open the **ONLINE** menu, and then choose **DOWNLOAD**. This action opens the *Download Controller* dialog box.



- 6 Click **ALL**, and then click **DOWNLOAD**. If a message box appears indicating that the controller is running, click **YES** to shut down the controller. The *Download Controller* dialog box displays the status of the download as shown in the following illustration.

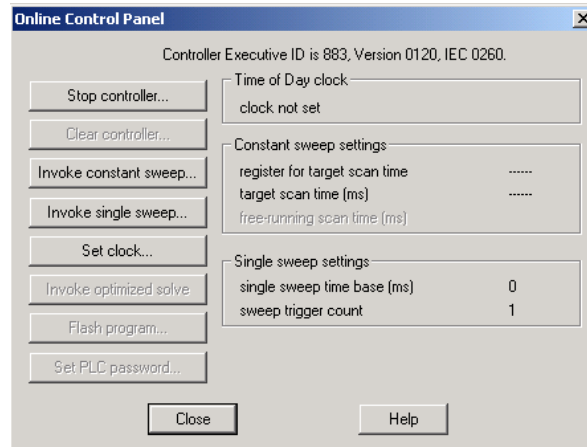


- 7 When the download is complete, you will be prompted to restart the controller. Click **YES** to restart the controller.

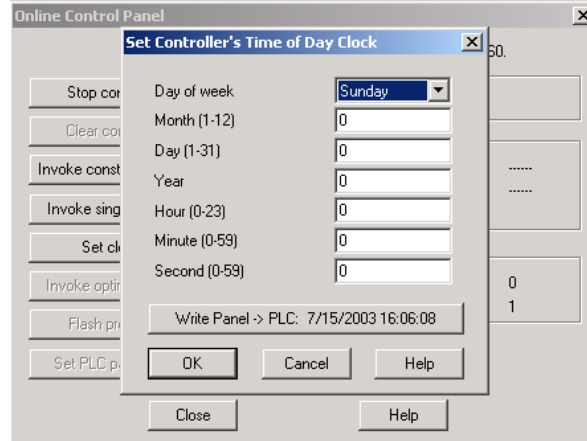
3.6 Verifying Successful Download

The final step is to verify that the configuration changes you made were received successfully by the module, and to make some adjustments to your settings.

- 1 In the *PLC Configuration* window, open the **ONLINE** menu, and then choose **ONLINE CONTROL PANEL**. This action opens the *Online Control Panel* dialog box.



- 2 Click the **SET CLOCK** button to open the *Set Controller's Time of Day Clock* dialog box.



- 3 Click the **WRITE PANEL** button. This action updates the date and time fields in this dialog box. Click **OK** to close this dialog box and return to the previous window.
- 4 Click **CLOSE** to close the *Online Control Panel* dialog box.
- 5 In the *PLC Configuration* window, open the **ONLINE** menu, and then choose **REFERENCE DATA EDITOR**. This action opens the *Reference Data Editor* dialog box. On this dialog box, you will add preset values to data registers that will later be monitored in the ProTalk module.

- Place the cursor over the first address field, as shown in the following illustration.

	Variable Name	Data Type	Address	Value	Set Value
1					
2					
3					
4					
5					
6					
7					
8					
9					
10					
11					
12					

- In the *PLC Configuration* window, open the **TEMPLATES** menu, and then choose **INSERT ADDRESSES**. This action opens the Insert addresses dialog box.
- On the *Insert Addresses* dialog box, enter the values shown in the following illustration, and then click **OK**.

Insert Addresses

First Reference To Insert: 400001

Last Reference To Insert: 400010

Number of References to Insert: 10

Display Format: Dec

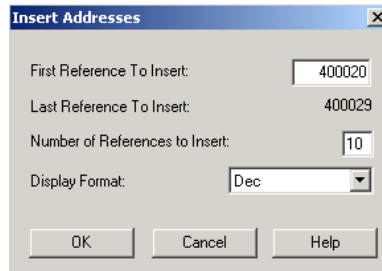
OK Cancel Help

- Notice that the template populates the address range, as shown in the following illustration. Place your cursor as shown in the first blank address field below the addresses you just entered.

Place cursor here

	Variable Name	Data Type	Address	Value	Set Value
2			400002		
3			400003		
4			400004		
5			400005		
6			400006		
7			400007		
8			400008		
9			400009		
10			400010		
11					
12					
13					

10 Repeat steps 6 through 9, using the values in the following illustration:



11 In the *PLC Configuration* window, open the **ONLINE** menu, and then choose **ANIMATE**. This action opens the *RDE Template* dialog box, with animated values in the *Value* field.

	Variable Name	Data Type	Address	Value	Set Value
3			400003	7	
4			400004	17	
5			400005	3	
6			400006	15	
7			400007	2	
8			400008	49	
9			400009	0	
10			400010	0	
11					
12			400020	24576	
13			400021	5	
14			400022	7	

12 Verify that values shown are cycling, starting from address 400065 and up.

13 In the *PLC Configuration* window, open the **TEMPLATES** menu, and then choose **SAVE TEMPLATE AS**. Name the template *ptqclock*, and then click **OK** to save the template.

14 In the *PLC Configuration* window, open the **ONLINE** menu, and then choose **DISCONNECT**. At the disconnect message, click **YES** to confirm your choice.

At this point, you have successfully

- Created and downloaded a Quantum project to the PLC
- Preset values in data registers that will later be monitored in the ProTalk module.

You are now ready to complete the installation and setup of the ProTalk module.

4 Configuring the Processor with ProWORX

When you use ProWORX 32 software to configure the processor, use the example SAF file provided on the ProTalk Solutions CD-ROM.

Important Note: ProWORX software does not report whether the PTQ module is present in the rack, and therefore is not able to report the health status of the module when the module is online with the Quantum processor. Please consider this when monitoring the status of the PTQ module.

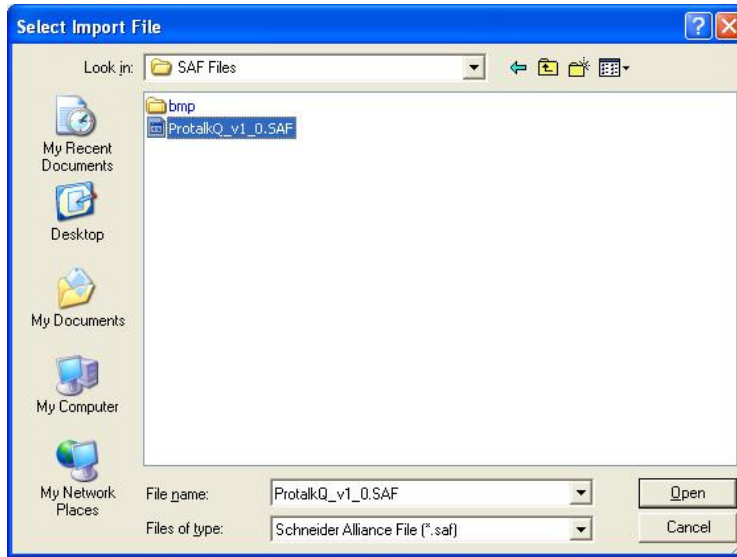
- 1 Run the **SCHNEIDER_ALLIANCES.EXE** application that is installed with the ProWORX 32 software:



- 2 Click on **IMPORT...**



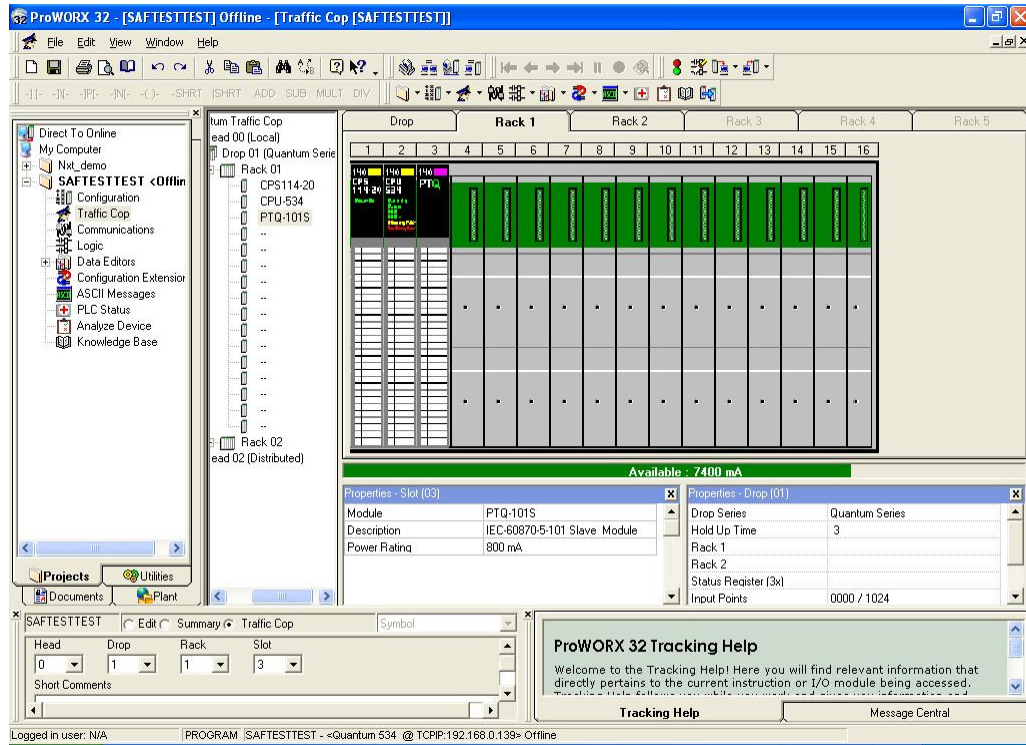
- 3 Select the **.SAF** File that is located on the CD-ROM shipped with the PTQ module.



- 4 After you click on **OPEN** you should see the PTQ modules imported (select **I/O SERIES** as **QUANTUM**):



Now you can close the Schneider alliances application and run the ProWORX 32 software. At the *Traffic Cop* section, select the PTQ module to be inserted at the slot:



5 Setting Up the ProTalk Module

In This Chapter

- ❖ Installing the ProTalk Module in the Quantum Rack..... 44
- ❖ Connect the PC to the ProTalk Configuration/Debug Port..... 46
- ❖ Verifying Communication Between the Processor and the Module 48

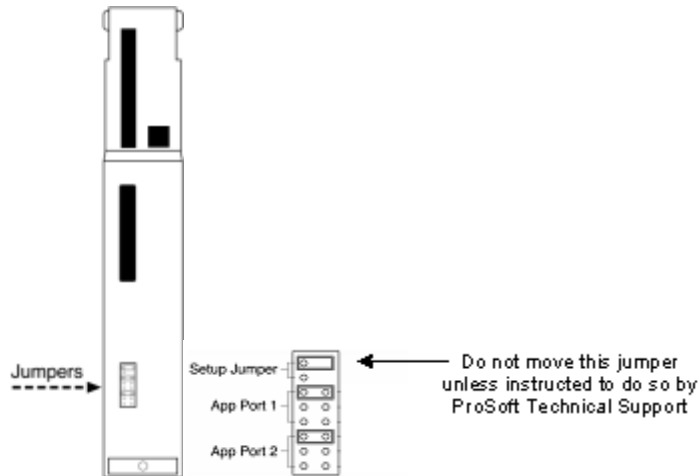
After you complete the following procedures, the ProTalk module will actively be transferring data bi-directionally with the processor.

5.1 Installing the ProTalk Module in the Quantum Rack

5.1.1 Verifying Jumper Settings

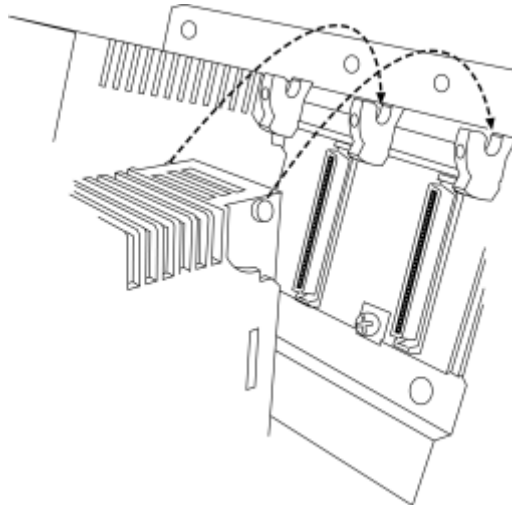
ProTalk modules are configured for RS-232 serial communications by default. To use RS-422 or RS-485, you must change the jumpers.

The jumpers are located on the back of the module as shown in the following illustration:

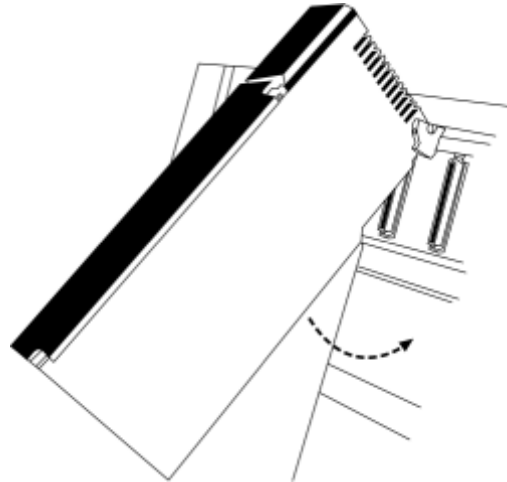


5.1.2 Installing the ProTalk Module in the Quantum Rack

- 1 Place the Module in the Quantum Rack. The ProTalk module must be placed in the same rack as the processor.
- 2 Tilt the module at a 45° angle and align the pegs at the top of the module with slots on the backplane.



- 3 Push the module into place until it seats firmly in the backplane.

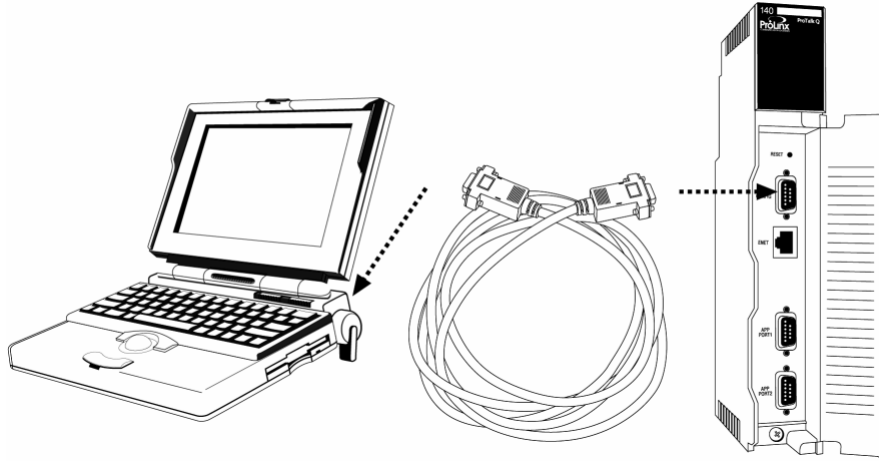


Caution: The PTQ module is hot-swappable, meaning that you can install and remove it while the rack is powered up. You should not assume that this is the case for all types of modules unless the user manual for the product explicitly states that the module is hot-swappable. Failure to observe this precaution could result in damage to the module and any equipment connected to it.

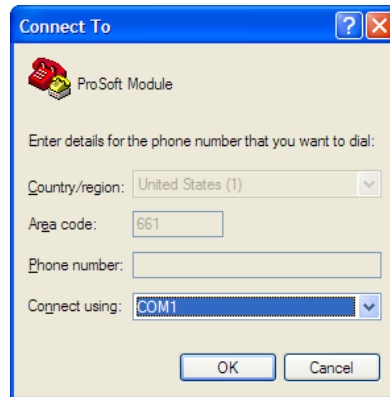
5.2 Connect the PC to the ProTalk Configuration/Debug Port

Make sure you have exited the Quantum programming software before performing these steps. This action will avoid serial port conflict.

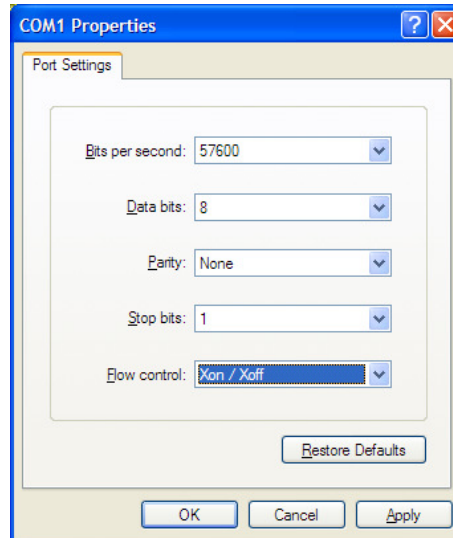
- 1 Using the supplied Null Modem cable, connect your PC to the Configuration/Debug port on the ProTalk module as shown



- 2 Click the Windows **START** button, then choose **PROGRAMS / ACCESSORIES / COMMUNICATIONS / HYPERTERMINAL**.
- 3 In the HyperTerminal window, enter a connection name, for example *ProSoft Module*, and then click **OK**. This action opens the *Connect To* dialog box.



- 4 In the *Connect Using* field, ensure that the com port matches the port on your PC to which you connected the Null Modem cable, and then click **OK**. This action opens the *COMx Properties* dialog box.

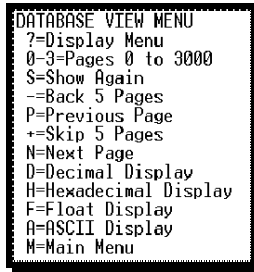


- 5 Verify that the settings match those shown in the example above, and then click **OK**. If your port settings are configured correctly, you will return to the *HyperTerminal* window.
- 6 In the *HyperTerminal* window, press **[?]**. This action opens the module's Configuration/Debug menu.

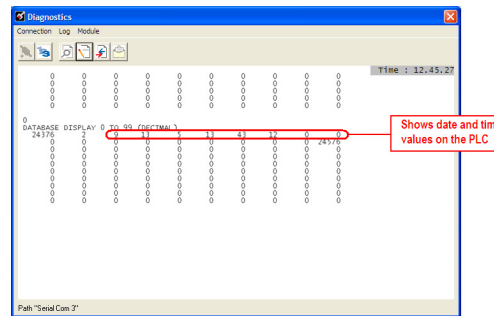
5.3 Verifying Communication Between the Processor and the Module

This procedure will verify that the clock values we entered in the processor’s data memory (page 32) can be read into the ProTalk module.

- 1 From the **CONFIGURATION/DEBUG MENU**, type **[D]**, then press **[?]**. This action opens the *Database View Menu*.



- 2 Type **[0]** (zero). This displays values present in the ProTalk database for 0 to 99.



Value	Description
9	Month (September)
13	Day of the Month
5	Year (2005)
13	Hour (13:00 or 1:00 P.M.)
43	Minutes
12	Seconds

In this example, the register values read from the PLC indicate that the date and time returned is September, 13, 2005, 1:43:12 p.m.

- 3 Type **[0]** again. The values should be different from those shown in the previous view. For example, the minute and second values should be incrementing just as the values on the PLC are also incrementing.

At this point, you have successfully:

- Installed and set up the ProTalk module
- Verified Data Read access between the processor and the ProTalk module

You are now ready to proceed with implementation of your application.

6 Configuring the Module

In This Chapter

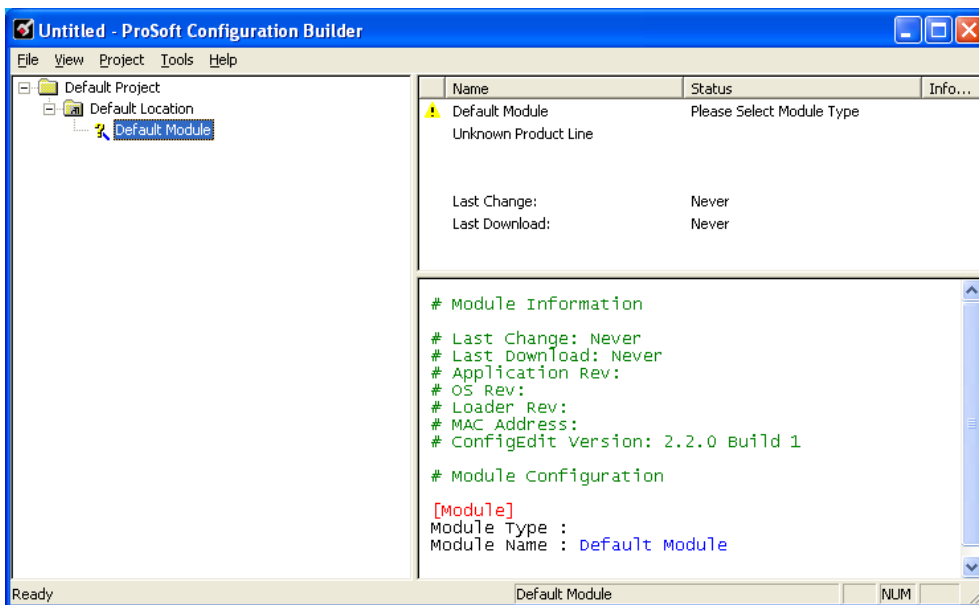
❖ Using ProSoft Configuration Builder.....	50
❖ [Module].....	53
❖ [Backplane Data Exchange].....	54
❖ [DFNT Client x].....	64
❖ [DFNT Client x Commands].....	65
❖ Ethernet Configuration	72
❖ Downloading the Project to the Module Using a Serial COM port	73

6.1 Using ProSoft Configuration Builder

ProSoft Configuration Builder (PCB) provides a quick and easy 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.

6.1.1 Set Up the Project

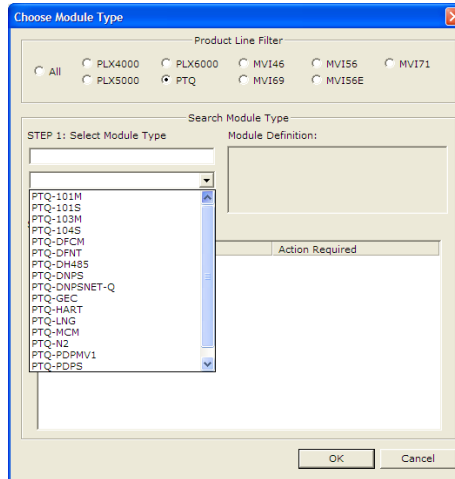
To begin, start *ProSoft Configuration Builder*. If you have used other Windows configuration tools before, you will find the screen layout familiar. *ProSoft Configuration Builder's* window consists of a tree view on the left, an information pane and a configuration pane on the right side of the window. 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.



Your first task is to add the PTQ-DFNT module to the 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, choose **CHOOSE MODULE TYPE**. This action opens the *Choose Module Type* dialog box.

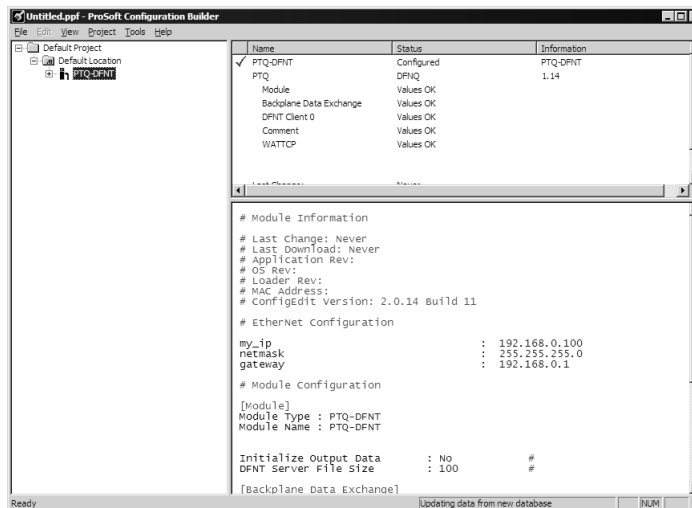


- 3 In the **PRODUCT LINE FILTER** area of the dialog box, select **PTQ**. In the **SELECT MODULE TYPE** dropdown list, select **PTQ-DFNT**, and then click **OK** to save your settings and return to the *ProSoft Configuration Builder* window.

The next task is to set the module parameters.

6.1.2 Renaming PCB Objects



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



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

- 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 on the **[+]** sign next to the module icon to expand module information.
- 2 Click on 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**.

6.2 [Module]

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

The following example shows a sample [Module] section:

```
# This section defines the configuration for the Module level
# data.
#
[Module]
Module Name: Test Example of ProTalk-DFNT Communication Module

Initialize Output Data: No #Initialize the database file on startup
DFNT Server File Size : 100 #100 or 1000 as max file size for DFNT server
```

Modify each of the parameters based on the needs of your application.

Module Name Parameter

The Module Name parameter assigns a name to the module that can be viewed using the configuration/debug port. Use this parameter to identify the module and the configuration file. You can enter a name from 0 to 80 characters.

Initialize Output Data Parameter

The Initialize Output Data parameter determines if the output data for the module should be initialized with values from the processor. If the value is set to N, the output data will be initialized to 0. If the value is set to Y during initialization, the module will invert (for this scan only) all backplane commands (Type 2).

DFNT Server File Size

This parameter defines the file size for DFNT Server. Enter a value of 100 to 1000. For example, if this parameter is set to 100, a DFNT write command to the module will be interpreted as:

Command	Description
N10:0	protocol database address 0
N10:50	protocol database address 50
N11:0	protocol database address 100
N12:0	protocol database address 200

If this parameter is set to 1000, a DFNT write command to the module will be interpreted as:

Command	Description
N10:0	protocol database address 0
N10:50	protocol database address 50
N11:0	protocol database address 1000
N12:0	protocol database address 2000

6.3 [Backplane Data Exchange]

The following is an example of a typical [Backplane Data Exchange] section:

```
# This section is used by the PTQ module to define the data transferred
# between the module and processor.
#
# Cmd Type      --> 0=Disable
#                1=Quantum to PTQ (Read from Quantum)
#                2=PTQ to Quantum (Write to Quantum)
#                3=Control data block for module
# DB Address    --> address of starting word in database
# Point Type    --> 0=0:x
#                1=1:x
#                3=3:x
#                4=4:x
# Point Address --> point address (1 based)
# (0x and 1x must be at start of word (that is, 1, 17, 33, ...))
# Word Count    --> number of words to transfer (1 to 130)
#                CMD TYPE is ALWAYS 64 words in length
```

```
[Backplane Data Exchange]
#   Cmd      PTQ      Point      QUANTUM      Word
#   Type     Address   Type       Address      Count
START
    1         0         4         1           100
# move data from Quantum to the PTQ
    1        100        4        101          50
# move data from Quantum to the PTQ
    2         500        4        200          50
# move data from PTQ to the Quantum
END
```

This example shows an application that reads 150 words from the Quantum to the module, and writes 50 words from the module to the Quantum.

The [Backplane Data Exchange] section is a series of messages that instruct the module how to transfer data to/from the Quantum. What is missing from the message is the ability to schedule its execution. This ability is normally left to the programmer in the PLC environment, however in the PTQ module this is not included so that the commands may run as fast as possible to maintain the synchronization of the two databases. One command from the list will execute during each I/O service interval at the end of the PLC ladder logic evaluation. So as an example if your configuration contains 10 "Backplane Data Exchange" commands it will require 10 PLC scans to process the entire list.

This section may contain up to 100 individual commands used in any combination to transfer data to/from the Quantum. The following topics provide information on the use of the commands as well as simple examples.

6.3.1 Setting Up Command Function 1 (Read Data from the Quantum)

This section provides information on how to read data from the Quantum into the module.

Command Function 1 (one) is designed to transfer data from the Quantum to the module on a continuous basis. The command(s) required to transfer your application data should be entered in the [BACKPLANE DATA EXCHANGE] section of your configuration file as required.

This command takes the following parameters:

- *Command type:* 1 (Read data from the Quantum)
- *PTQ Database Address:* The destination for the data retrieved from the Quantum.
- *Point Type:* The type of register within the Quantum (0:x = 0, 10:x = 1 30:x = 3 or 40:x = 4)
- *Quantum Address:* The source of the data within the Quantum. The address is expressed without the use of the register range, for example 400001 would be entered as 1 (400001 - 400000 = 1 or 40001 - 40000 = 1)
- *Word Count:* The number of words to copy. The length of this copy may be any length of 1 to 130 inclusive. If your application requires the movement of additional data you may enter additional commands.

Example 30:x or 40:x Register Transfer

The following example shows a typical command used to read 40:x data from the Quantum. In this example, registers 400001 to 400100 from the Quantum will be transferred to registers 0 to 99 within the module.

```
# Word Count          Number of words to transfer (1 to 130)
#
#   Cmd   PTQ          Point   Quantum   Word
#   Type  Address      Type    Address   Count
[Backplane Data Exchange]
START
      1           0           4           1           100
END
```

Example 0:x or 10:x Register Transfer

The transfer of Coils and Input bits require some forethought as the command transfers words and not bits. This means that if you want to transfer bits 000005 to 000007 from the Quantum to word 21 in the module you would have to transfer the word within the Quantum containing bits 000001 to 000016 to a word within the modules memory.

Important: Take care with the transfer of bits while planning the application so as to optimize the usage of the available bits and to preserve the integrity of your information.

The following example shows how this could be accomplished.

```
#   Cmd   PTQ  DB      Point   Quantum   Word
#   Type  Address      Type     Address   Count
START
      1     21          0         1         1
END
```

6.3.2 Setting Up Command Function 2 (Write Data to the Quantum)

This section provides information on how to write data from the module to the Quantum.

Command Function 2 (two) is designed to transfer data from the module to the Quantum on a continuous basis. The command(s) required to transfer your application data should be entered in the [BACKPLANE DATA EXCHANGE] section of your configuration file as required.

This command takes the following parameters:

- *Command type:* 2 (Write data to the Quantum)
- *PTQ Database Address:* The source of the data within the PTQ to be sent to the Quantum.
- *Point Type:* The type of register within the Quantum (0:x = 0, 10:x = 1 30:x = 3 or 40:x = 4)
- *Quantum Address:* The destination register within the Quantum. The address is expressed without the use of the register range, for example 400001 would be entered as 1 (400001 to 400000 = 1 or 40001 to 40000 = 1)
- *Word Count:* The number of words to copy. The length of this copy may be any length of 1 to 130 inclusive. If your application requires the movement of additional data you may enter additional commands.

Example 30:x or 40:x Register Transfer

The following example shows a typical command used to write to the 40:x registers within the Quantum. In this example, registers 500 to 549 from the PTQ will be transferred to registers 400200 to 400249 within the Quantum.

```
# Word Count          Number of words to transfer (1 to 130)
#
#   Cmd   PTQ  DB      Point   Quantum   Word
#   Type  Address      Type     Address   Count

[Backplane Data Exchange]
START
      2     500          4         200         50
END
```


Example 0:x or 10:x Register Transfer

The transfer of Coils and Inputs require some forethought as the command transfers words and not bits. This means that if you wanted to transfer the word containing the bits 805 to 806 from the module to the Quantum you would transfer the entire 50th word of the modules memory into the destination register in the Quantum. The following command transfer bits 800 to 815 (Word x Bits = Bit Address or 50 * 16 = 800) from the modules memory to word 1 of the coils (000001 to 000016) within the Quantum.

Important: Take care with the transfer of bits while planning the application so as to optimize the usage of the available bits and to preserve the integrity of your information.

The following example shows how this could be accomplished.

#	Cmd	PTQ DB	Point	Quantum	Word
#	Type	Address	Type	Address	Count
START					
	1	49	0	1	1
END					

6.3.3 Setting Up Command Function 3 (Special Functions)

This section provides information on how to request the module to perform special non-typical functions that may be required by an application.

Command Function 3 (three) if required should be the first item entered in the [BACKPLANE DATA EXCHANGE] section of your configuration file.

This may be used with all modules to implement the following functionality:

- Force a reboot of the PTQ module (Special Function 9998 or 9999 available on all products)
- Set / Retrieve Time and Date (DNP and IEC only!)
- Register events with the protocol (DNP and IEC only!)

Other modules may implement additional functionality, which will be described in the Special Functions section of this manual.

This command takes the following parameters:

- *Command type:* 3 (Write data to the Quantum)
- *PTQ Database Address:* This value is **ALWAYS** 0.

Note: This will **NOT** overwrite your application database in the PTQ but merely serves as an additional flag to notify the module of the unique nature of the command.

- *Point Type:* The type of register within the Quantum (0:x = 0, 10:x = 1 30:x = 3 or 40:x = 4)
- *Quantum Address:* The source register within the Quantum. The address is expressed without the use of the register range, for example 400001 would be entered as 1 (400001 - 400000 = 1 or 40001 - 40000 = 1)

- *Word Count:* This value is **ALWAYS** 64. Care should be taken to assure that 64 words of memory are available within the Quantum.

Example 30:x or 40:x Register Transfer

The following example shows a typical command used to retrieve a special function command from the Quantum. In this example, registers 400600 to 400663 from the Quantum will be used to provide the information required by the module.

```
# Word Count          Number of words to transfer (1 to 130)
#
#   Cmd   PTQ  DB      Point      Quantum      Word
#   Type  Address  Type      Address      Count
[Backplane Data Exchange]
START
      3      0          4          600          64
END
```

The following section shows the functions that may be performed by using the Command Function 3.

Special Functions

The PTQ-DFCM module supports the following special functions:

Block ID	Descriptions
1000	Event Port 1
2000	Event Port 2
3000 to 3001	Port 1 slave polling control
3002 to 3006	Port 1 slave status
3100 to 3101	Port 2 slave polling control
3102 to 3106	Port 2 slave status
5000 to 5006	Port 1 command control
5100 to 5106	Port 2 command control
9940	Clears/Sets bits in the database (mask in bits)
9941	Forces new values in database
9998	Warm-boot control block
9999	Cold-boot control block

Event Port Block (1000/2000)

This special function allows commands to be sent directly from the Quantum processor to the remote DF1 slave. Block ID 1000 sends commands to DF1 slaves connected to Port 1. Block ID 2000 sends commands to DF1 slaves connected to Port 2.

Event Request (Write Block)

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

After the PTQ-DFCM processes the event request, the following event response block will be generated:

Event Response (Read Block)

Offset	Description	Length
0	Set to zero when complete	1
1	1000 or 2000	1
2	0=Fail, 1=Success	1

Block IDs 3000 to 3001/3100 to 3101 - Port 1 & 2 Slave Polling Control

Blocks IDs 3000 (Port 1) and 3100 (Port 2) disable the polling of the slaves listed in the block, even if a configured command exists that already polls these slaves. In this case, this command would not be sent until the slave is enabled again (using Block IDs 3001 or 3101).

The following table shows the slave polling control block used to disable the polling of specific slaves:

Slave Polling Disable Request (Write Block)

Offset	Description	Length
0	3000 or 3100	1
1	Number of Slaves in Block	1
2 to 61	Slave indexes	60

After the module processes the block, it generates the following response block:

Slave Polling Disable Response (Read Block)

Offset	Description	Length
0	Set to zero when complete	1
1	3000 or 3100	1
2	Number of slaves processed	1

Blocks IDs 3001 (Port 1) and 3101 (Port 2) enable the polling of the slaves listed in the block, even if a configured command exists that already polls these slaves. In this case, this command would not be sent until the slave is disabled again (using Block IDs 3000 or 3100).

The following table shows the slave polling control block used to enable the polling of specific slaves:

Slave Polling Enable Request (Write Block)

Offset	Description	Length
0	3001 or 3101	1
1	Number of Slaves in Block	1
2 to 61	Slave indexes	60

After the module processes the block, it generates the following response block:

Slave Polling Enable Response (Read Block)

Offset	Description	Length
0	Set to zero when complete	1
1	3001 or 3101	1
2	Number of slaves processed	1

Block IDs 3002 to 3006/3102 to 3106 - Slave Polling Status

The slave polling status function retrieves the status of all the slaves in the DF1 network. The possible slave status codes are:

- 0 = slave is not used
- 1 = slave being actively polled
- 2 = slave suspended
- 3 = slave disabled

Use the block IDs to poll the status for different slave address ranges as described in the following table.

Block ID	Port Number	Start DF1 Slave Address	End DF1 Slave Address	Number of DF1 Slaves
3002	1	0	59	60
3003	1	60	119	60
3004	1	120	179	60
3005	1	180	239	60
3006	1	240	255	16
3102	2	0	59	60
3103	2	60	119	60
3104	2	120	179	60
3105	2	180	239	60
3106	2	240	255	16

Slave Polling Status List Request (Write Block)

Offset	Description	Length
0	3002 to 3006 or 3102 to 3106	1

After the module processes the block, it generates the following response block:

Slave Polling Status List Response (Read Block)

Offset	Description	Length
0	Set to zero when complete	1
1	3002 to 3006 or 3102 to 3106	1
2 to 61	Slave Poll Status Data	60

Block IDs 5001-5006/5101-5106 - Command Control Request

This special function adds commands from the configured command list to the command queue. This not only gives a higher priority to these commands, but also may send commands that are currently disabled.

The following table shows the block request:

Command Control Request (Write Block)

Offset	Description	Length
0	5001-5006 or 5101-5106	1
1	Command index	1
2	Command index	1
3	Command index	1
4	Command index	1
5	Command index	1
6	Command index	1

Block ID	Port Number	Number of Commands
5001	1	1
5002	1	2
5003	1	3
5004	1	4
5005	1	5
5006	1	6
5101	2	1
5102	2	2
5103	2	3
5104	2	4
5105	2	5
5106	2	6

After the module processes the block, it generates the following response block:

Command Control Response (Read Block)

Offset	Description	Length
0	Set to zero when complete	1
1	5000 to 5006 or 5100 to 5106	1
2	Number of commands added to command queue	1

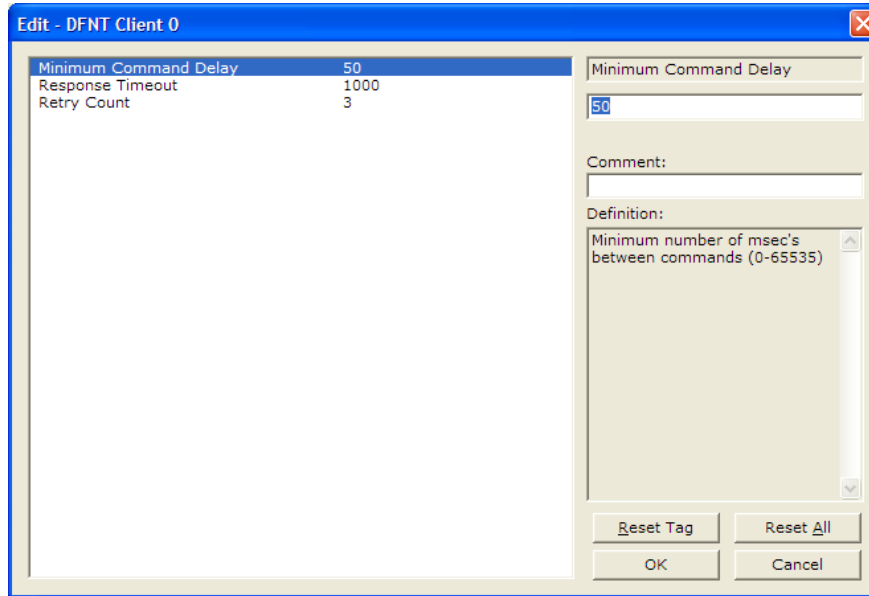
Warm/Cold boot Module Block (9999/9998)

If the Quantum processor sends a block number 9998 or 9999, the module will reset the contents of the data block to zero and perform a complete reboot operation.

Word Offset in Block	Example Address	Data Field(s)	Description
0	400600	9998 or 9999	Block ID to reboot module

6.4 [DFNT Client x]

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



6.4.1 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.

6.4.2 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 upon the type of communication network used, and the expected response time of the slowest device on the network.

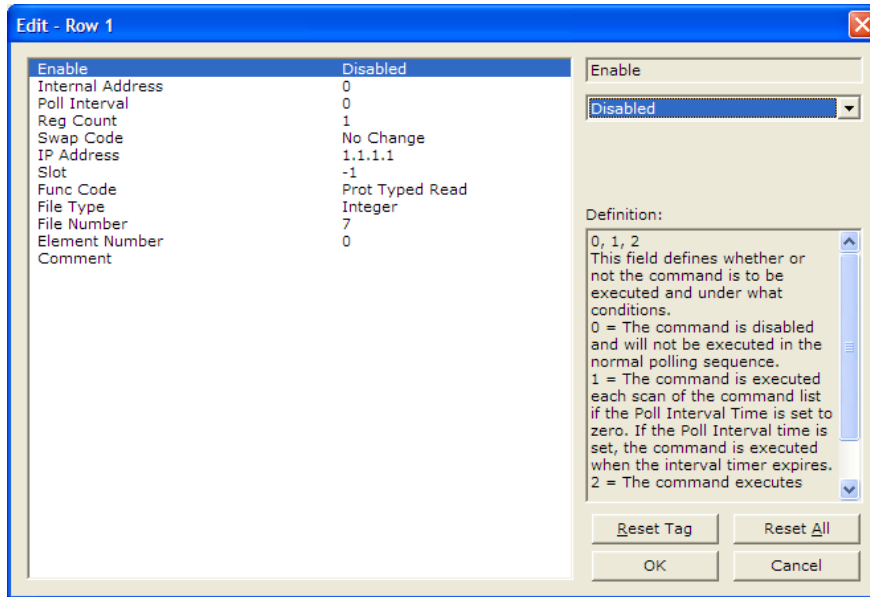
6.4.3 Retry Count

0 to 10

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

6.5 [DFNT Client x Commands]

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



6.5.1 Command List

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

Write commands have a special feature, as they can be set to execute only if the data in the write command changes. If the data in the command has not changed since the command was last issued, the command will not be executed. If the data in the command has changed since the command was last issued, the command will be executed. Use of this feature can lighten the load on the DF1 network. In order to implement this feature; set the enable code for the command to a value of 2.

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

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

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

Basic Command Set Functions

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

PLC-5 Command Set Functions

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

SLC-500 Command Set Functions

Function Code	Command	Function	Definition	PLC5	SLC500 & MicroLogix	Power-monitor II	ControlLogix
501	0x0F	0xA1	Protected Typed Logical Read With Two Address Fields		X		X
502	0x0F	0xA2	Protected Typed Logical Read With Three Address Fields		X	X	X
509	0x0F	0xA9	Protected Typed Logical Write With Two Address Fields		X		X
510	0x0F	0XAA	Protected Typed Logical Write With Three Address Fields		X	X	X
511	0x0F	0XAB	Protected Typed Logical Write With Mask (Three Address Fields)		X		X

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

6.5.2 Command Entry Formats

The format of each command in the list depends on the function being executed. Refer to Command Function Codes (page 68) for a complete discussion of the commands supported by the module and of the structure and content of each command.

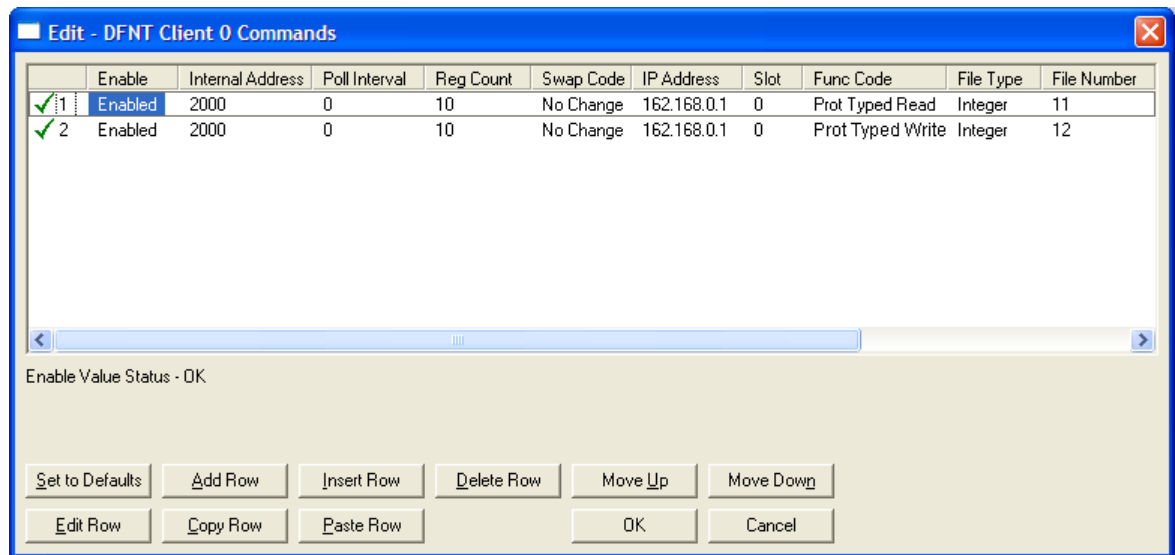
The following table shows the structure of the configuration data necessary for each of the supported commands.

Module Information Data							Device Information Data					
Column #	1	2	3	4	5	6	7	8	9	10	11	12
Function Code	Enable Code	Internal Address	Poll Interval Time	Count	Swap Code	IP Address	Slot Number	Function Code	Function Parameters			
FC 1	Code	Register	Seconds	Count	Code	Node	Slot	1	Word Address			
FC 2	Code	Register	Seconds	Count	Code	Node		2	Word Address			
FC 3	Code	Register	Seconds	Count	Code	Node		3	Word Address			
FC 4	Code	Register	Seconds	Count	Code	Node		4	Word Address			
FC 5	Code	Register	Seconds	Count	Code	Node		5	Word Address			
FC 100	Code	Register	Seconds	Count	Code	Node		100	File Number	Element	Sub-Element	
FC 101	Code	Register	Seconds	Count	Code	Node		101	File Number	Element	Sub-Element	
FC 102	Code	Register	Seconds	Count	Code	Node		102	File Number	Element	Sub-Element	
FC 150	Code	Register	Seconds	Count	Code	Node		150	File String			
FC 151	Code	Register	Seconds	Count	Code	Node		151	File String			
FC 152	Code	Register	Seconds	Count	Code	Node		152	File String			
FC 501	Code	Register	Seconds	Count	Code	Node		501	File Type	File Number	Element	
FC 502	Code	Register	Seconds	Count	Code	Node		502	File Type	File Number	Element	Sub-Element
FC 509	Code	Register	Seconds	Count	Code	Node		509	File Type	File Number	Element	

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 510	Code	Register	Seconds	Count	Code	Node		510	File Type	File Number	Element	Sub-Element
FC 511	Code	Register	Seconds	Count	Code	Node		511	File Type	File Number	Element	Sub-Element

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"> <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 68) for information										
Swap Code	0, 1, 2, 3	This parameter defines if the data received from the Server is to be ordered differently than that received from the Server device. This parameter is helpful when dealing with floating-point or other multi-register values, as there is no standard method of storage of these data types in Server devices. This parameter can be set to order the register data received in an order useful by other applications. The following table defines the values and their associated operations: <table 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)											

Parameter	Range	Description																																
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>																																
Function Code	See Command Function Codes (page 68)	<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"> <tbody> <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> </tbody> </table> <p>PLC-5 Command Set (0x0F)</p> <table border="1"> <tbody> <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> </tbody> </table> <p>SLC Command Set (0x0F)</p> <table border="1"> <tbody> <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> </tbody> </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 68)	The number of auxiliary parameters required depends on the function code selected for the command.																																

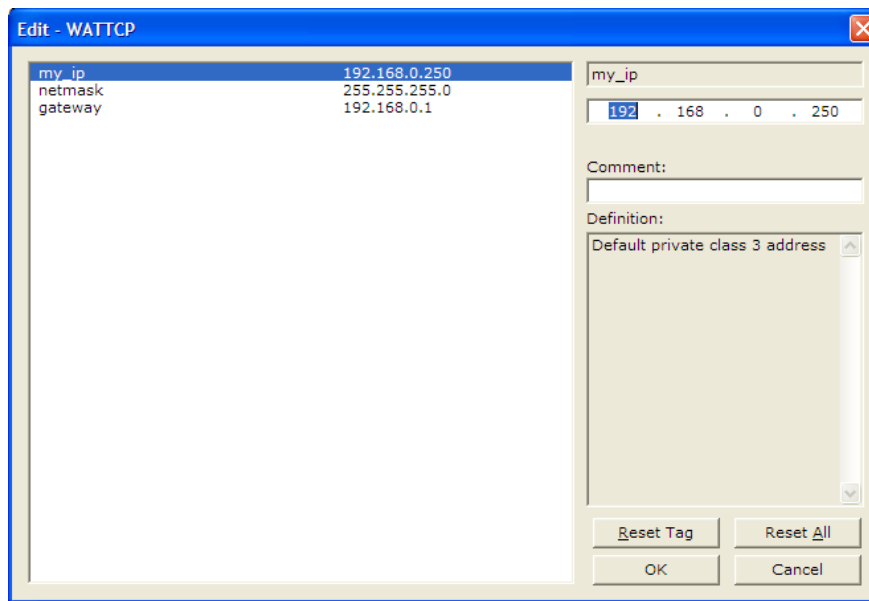
6.6 Ethernet Configuration

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.

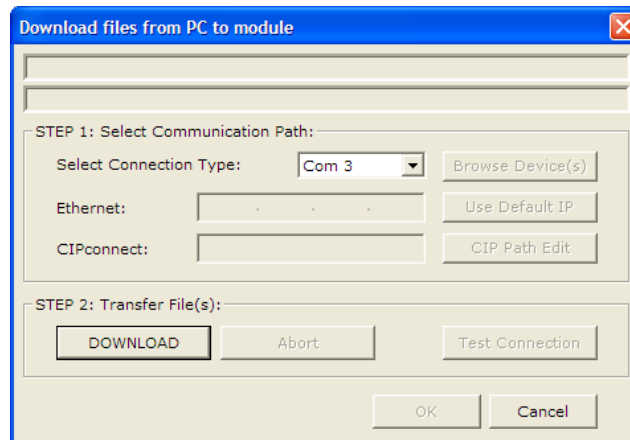


- 3 Edit the values for *my_ip*, *netmask* (subnet mask) and *gateway* (default gateway).
- 4 When you are finished editing, click **OK** to save your changes and return to the *ProSoft Configuration Builder* window.

6.7 Downloading the Project to the Module Using a Serial COM port

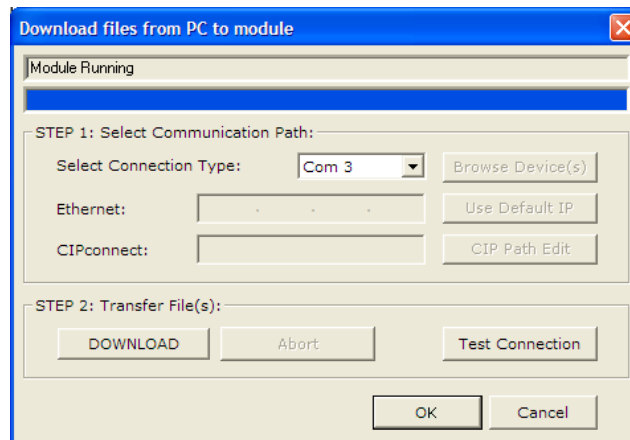
For the module to use the settings you configured, you must download (copy) the updated *Project* file from your PC to the module.

- 1 In the tree view in *ProSoft Configuration Builder*, click once to select the module.
- 2 Open the *Project* menu, and then choose **MODULE/DOWNLOAD**. 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*.



7 Diagnostics and Troubleshooting

In This Chapter

- ❖ LED Status Indicators..... 76
- ❖ Using ProSoft Configuration Builder (PCB) for Diagnostics..... 79
- ❖ Reading Status Data from the Module 91

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. For details on Status Data values, see PTQ-DFNT Status Data Area.

7.1 LED Status Indicators

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

Module	Color	Status	Indication
Active	Green	On	The LED is on when the module recognizes a processor and is able to communicate if the [Backplane Data Movement] section specifies data transfer commands.
		Off	The LED is off when the module is unable to speak with the processor. The processor either absent or not running.
E-Link	Green	On	The Ethernet port is connected to the TCP/IP network
		Off	No Connection
E-Data	Green	On	There is data being transferred through the Ethernet port.
		Off	No data transfer
BAT Low	Red	Off	The battery voltage is OK and functioning.
		On	The battery voltage is low or the battery is not present. The battery LED will illuminate briefly upon the first installation of the module or if the unit has been un-powered for an extended period of time. This behavior is normal, however should the LED come on in a working installation please contact ProSoft Technology.
DEBUG	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.
CFG ERR		On	
		Off	
PRT1	Green	On	Port not used in application
		Off	Port not used in application
PRT2	Green	On	Port not used in application
		Off	Port not used in application
ERR1	Red	Off	The PTQ-DFNT is working normally.
		On	The PTQ-DFNT module program has recognized an application error. This LED will also be turned on if any command presents an error.
ERR2	N/A		Not used in application

If your module is not operating, and the status LEDs are not illustrated in the table above, please contact ProSoft Technology for technical assistance.

7.1.1 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.

7.1.2 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 Quantum 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.

7.1.3 Troubleshooting

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

Processor Errors

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

Module Errors

Problem description	Steps to take
BP ACT LED (not present on MVI56E modules) remains OFF or blinks slowly MVI56E modules with scrolling LED display: <Backplane Status> condition reads ERR	This indicates that backplane transfer operations are failing. Connect to the module's Configuration/Debug port to check this. To establish backplane communications, verify the following items: <ul style="list-style-type: none">▪ The processor is in RUN or REM RUN mode.▪ The backplane driver is loaded in the module.▪ The module is configured for read and write data block transfer.▪ The ladder logic handles all read and write block situations.▪ The module is properly configured in the processor I/O configuration and ladder logic.
OK LED remains RED	The program has halted or a critical error has occurred. Connect to the Configuration/Debug port to see if the module is running. If the program has halted, turn off power to the rack, remove the card from the rack and re-insert it, and then restore power to the rack.

7.1.4 Error Status Table

The program maintains an error/status table that is transferred to the processor in each read block. You can use the error/status data to determine the "health" of the module. Refer to Error Status Table for data block structure.

7.2 Using ProSoft Configuration Builder (PCB) for Diagnostics

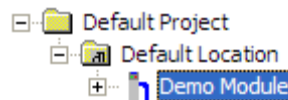
The *Configuration and Debug* menu for this module is arranged as a tree structure, with the *Main* menu at the top of the tree, and one or more submenus for each menu command. The first menu you see when you connect to the module is the *Main* menu.

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.

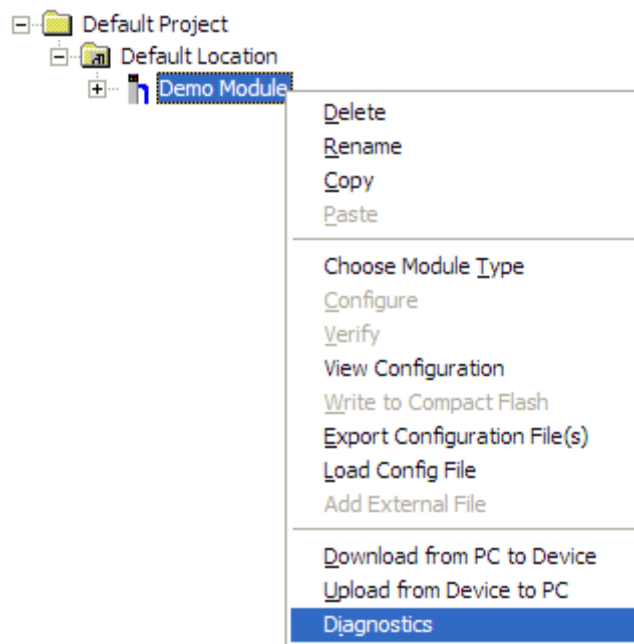
7.2.1 Using the Diagnostic Window in ProSoft Configuration Builder

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.

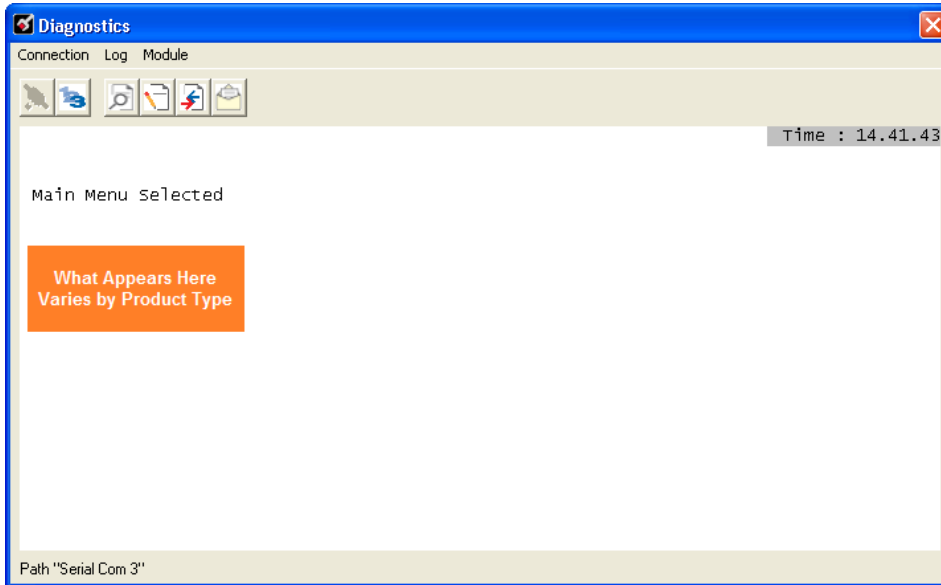


- 2 On the shortcut menu, choose **DIAGNOSTICS**.



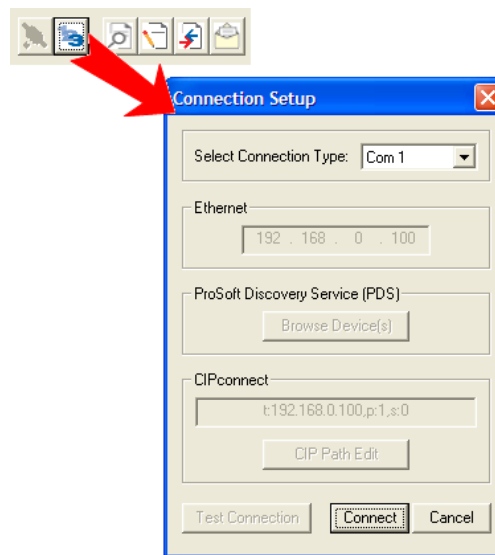
This action opens the *Diagnostics* dialog box.

- 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.



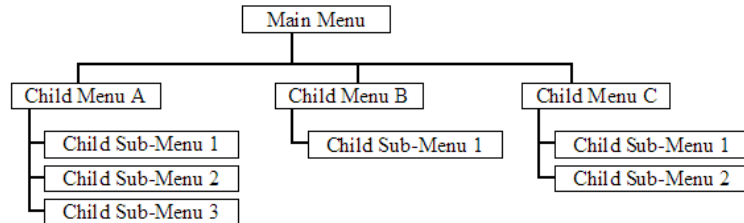
- 2 Verify that the null modem cable is connected properly between your computer's serial port and the module. A regular serial cable will not work.
- 3 On computers with more than one serial port, verify that your communication program is connected to the same port that is connected to the module.

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

7.2.2 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.

7.2.3 Main Menu

Features available through the use of the configuration/debug port on the PTQ-DFNT module are all reached using single keystrokes on your computer. There is a single main menu and several sub-menus presented on the port. To view the current selections available, press **[?]**.

All facilities offered by the configuration/debugger are shown on the main menu. You make selections from this menu based on the type of information you want to view.

```
PTQ-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
 P=Backplane 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
```

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 86).

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.

Viewing the Backplane Command List

Press **[P]** from the Main Menu to view the Backplane Data Exchange List. Use this command to display the configuration and statistics of the backplane data transfer operations.

```
BACKPLANE DATA EXCHANGE LIST -- COMMANDS 0 TO 9
TYPE  DBREG  DBTYPE  ADDRESS  COUNT  LASTERR
 0      0      0        0        0      0X0000
 0      0      0        0        0      0X0000
 0      0      0        0        0      0X0000
 0      0      0        0        0      0X0000
 0      0      0        0        0      0X0000
 0      0      0        0        0      0X0000
 0      0      0        0        0      0X0000
 0      0      0        0        0      0X0000
 0      0      0        0        0      0X0000
 0      0      0        0        0      0X0000
```

Tip: Repeat this command at one-second intervals to determine the number of blocks transferred each second.

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 servers 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 PTQ-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 Client Configuration

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

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
- 4 = Display servers 15 to 19
- 5

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)

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 89).

Exiting the Configuration/Debug Program

Note: The following option should only be selected if instructed to do so by the ProSoft Technology Technical Support Group.

Press the **[ESC]** key. Use this command to exit the program and display the operating system prompt. This action causes the module to cease operation. Data will no longer be transferred between the ports and the module and between the processor and the module. This may interrupt a currently running process.

7.2.4 Database View Menu

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

```
DB Menu selected

DATABASE VIEW MENU
?=Display Menu
0-9=Display 0-9000
S=Show Again
-=Back 5 Pages
P=Previous Page
+=Skip 5 Pages
N=Next Page
D=Decimal Display
H=Hexadecimal Display
F=Float Display
A=ASCII Display
M=Main Menu
```

Viewing Register Pages

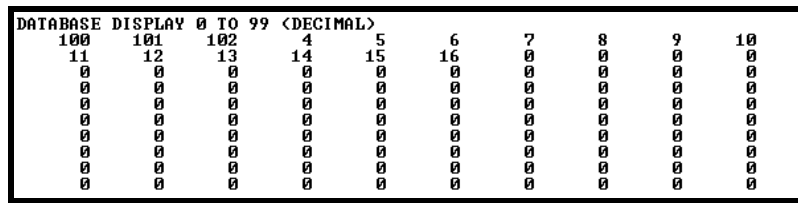
To view sets of register pages, use the keys described below:

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

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

Displaying the Current Page of Registers Again

Press **[S]** from the *Database View* menu to show the current page of registers again.



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

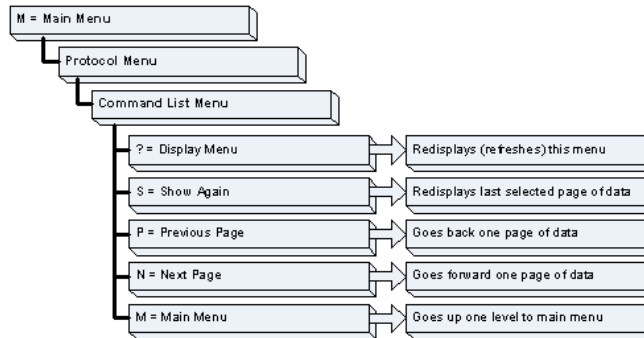
Press **[A]** from the *Database View* menu to display the data on the current page in ASCII format. This is useful for regions of the database that contain ASCII data.

Returning to the Main Menu

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

7.2.5 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

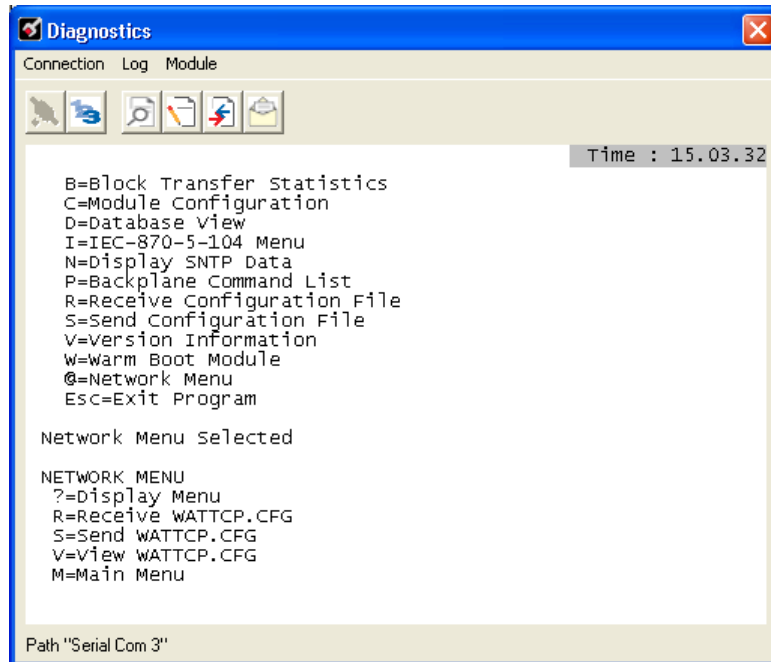
Press **[N]** to display the next page of commands.

Returning to the Main Menu

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

7.2.6 Network Menu

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



Transferring WATTCP.CFG to the Module

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

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

Transferring WATTCP.CFG to the PC

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

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

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

Viewing the WATTCP.CFG File on the module

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

```
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.

7.3 Reading Status Data from the Module

The PTQ-DFNT module provides the status data in each read block. This data can also be located in the module's database.

8 Reference

In This Chapter

❖ Product Specifications	94
❖ Functional Overview	97
❖ PTQ-DFNT Error Status Table	106
❖ Cable Connections	107
❖ PTQ-DFNT Status Data.....	109
❖ Error Codes	113
❖ TCP/IP Interface Errors	117
❖ Command Function Codes.....	119
❖ General Command Structure.....	120
❖ PLC-5 Processor Specifics	136
❖ SLC Processor Specifics	138
❖ MicroLogix Processor Specifics.....	139
❖ ControlLogix Processor Specifics.....	140
❖ DFNT Command Entry Form.....	141
❖ Frequently Asked Questions	142

8.1 Product Specifications

The PTQ EtherNet/IP (DF1 for Ethernet) Client/Server Communication Module is a Schneider Electric Quantum backplane compatible module that allows Quantum processors to interface easily with EtherNet/IP protocol (Explicit Messaging) compatible devices and hosts. Devices commonly supporting the protocol include Schneider Electric PLCs, and host HMI systems.

The PTQ-DFNT module is a single slot solution that provides a powerful connection between Schneider Electric's Quantum processor and EtherNet/IP (DF1 for Ethernet) network applications.

The module can send read or write commands to remote devices on the EtherNet/IP network. The PTQ-DFNT also has 20 servers that allow the module to receive commands from remote devices. Up to 100 commands can be configured by the user.

8.1.1 EtherNet/IP (Explicit Messaging) Compatible Devices

List of Rockwell Automation material that support EPIC:

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

8.1.2 General Specifications

- Single Slot - Quantum backplane compatible
- The module is recognized as an Options module and has access to PLC memory for data transfer
- Configuration data is stored in non-volatile memory in the ProTalk[®] module
- Up to six modules can be placed in a rack
- Local rack - The module must be placed in the same rack as processor
- Quantum data types supported: 0x, 1x, 3x, 4x
- High speed data transfer across backplane provides quick data update times
- Does not currently support Hot-Standby processors or applications

8.1.3 Hardware Specifications

Specification	Value
Backplane Current Load	1100 mA maximum @ 5 Vdc ± 5%
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% (without condensation)
Vibration	Sine vibration 4-100 Hz in each of the 3 orthogonal axes
Shock	30G, 11 mSec. in each of the 3 orthogonal axes
Dimensions (HxWxD), Approx.	250 x 103.85 x 40.34 mm 9.84 x 4.09 x 1.59 in
LED Indicators	Module Status Backplane Transfer Status Serial Port Activity LED Serial Activity and Error LED Status
Debug/Configuration Port (Debug)	
CFG Port (DEBUG)	DB-9M PC Compatible RS-232 only No hardware handshaking
Application Ports	
Application Ethernet port	RJ45 Connector Link and Activity LED indicators Electrical Isolation 1500 V rms at 50 Hz to 60 Hz for 60 s, applied as specified in section 5.3.2 of IEC 60950: 1991 Ethernet Broadcast Storm Resiliency = less than or equal to 5000 [ARP] frames-per-second and less than or equal to 5 minutes duration

8.1.4 Functional Specifications - PTQMDC/DFNT

Type	Specifications
Internal Database	4000 registers (words) available for transfer to and from Quantum processor
Modes	Client and Server supported
Messaging	PCCC on CIP - See command list that follows CIP Services (Server Only) - See command list below
Service Port	AF12Hex (44818)
Compatibility	Compatible devices include a wide assortment of other client/server devices such as: <ul style="list-style-type: none"> ▪ PLC: PLC5/E Rev. C/N, D/E, E/D or later, 1785-ENET Series A, Rev. D ▪ SLC: SLC5/05 Series A, OS501 fm5 or later ▪ CLX: 1756-ENET, 1756-ENBT ▪ CPLX: 1769-L5335E Rev. 12.26 Interchange (limited - Contact Tech Support) ▪ RSLinx: Gateway V1.7 or later
Miscellaneous	125 word read and write data lengths Floating point data supported

EtherNet/IP Server Specifications

When in the Server mode, the module accepts commands from a Client(s) to read/write data stored in the module's internal registers. This data is easily transferred to the Quantum processor's data registers.

The EtherNet/IP functionality supported by the module includes:

EtherNet/IP Server

Connections	Twenty independent TCP server sockets 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 Quantum processor
Data File	Data Table File Start - Fixed at N10 Data Table File Size - 100 or 1000 words Data Table location in PTQ - Start Register 0

EtherNet/IP Client Specifications

When configured in the client mode the DFNT module is capable of reading and writing data to remote EtherNet/IP devices, enabling the Quantum platform to act as a SCADA sub-master.

The EtherNet/IP Client functionality supported by the module includes:

EtherNet/IP Client Specifications

Messaging	PCCC on CIP Explicit Messaging supported
General	One client configurable for timing and commands
Command List	Up to 100 command, each fully configurable for function, slave address, register to/from addressing and word/bit count
Polling of command list	User configurable polling of commands, including disabled, continuous and on change of data (write only)

8.2 Functional Overview

8.2.1 Backplane Data Exchange

Before modifying the [Backplane Data Exchange] section of the configuration file, you must understand some important concepts. The following topics describe these concepts.

If you have used the parameters defined in the [Module] section, you have created the following memory map. We will use this map to explain how data transfer works between the processor and the ProTalk module.

PTQ Memory Address		Application Memory Address
0	=	0
10	=	10
20	=	20
30	=	30
40	=	40
50	=	50
60	=	60
70	=	70
80	=	80
90	=	90
100	=	100
110	=	110
120	=	120
130	=	130
140	=	140
149	=	149
...		...
500	=	500
510	=	510
520	=	520
530	=	530
540	=	540
549	=	549

A thorough understanding of the information contained in this section is required for successful implementation of the module in a user application.

Data Transfer

The module uses a concept referred to as "Command Functions". The command functions reside in the [Backplane Data Exchange] section of the configuration file (page 97, page 54). This method of data transfer is probably different from other methods you might have used, but does offer some unique advantages:

- The amount of ladder logic required will be substantially reduced; in fact in many applications no ladder logic will be required.
- The module may be placed in any position in the chassis containing the PLC and will operate with no modifications.

Defining Data to be Sent to the PTQ Database

You might be asked to provide access to 207 words of information for other devices on the network. This information resides in the PLC at addresses 400001 to 400207 and you must make this the first 207 words of the database inside the module. This would require the use of "**Command Function 1**".

Because the total amount of data exceeds the maximum length of any single command function, you will need two entries in the [Backplane Data Exchange] section of your configuration file. This might look like the following:

```
[Backplane Data Exchange]
#   Cmd      PTQ      Point      QUANTUM      Word
#   Type     Address   Type        Address       Count
START
      1          0          4           1           100
# move data from Quantum to the PTQ
      1        100          4          101           107
# move data from Quantum to the PTQ
END
```

The first command states:

Field	Value	Meaning
CMD TYPE	1	The type of operation to perform 1 = Read data from the Quantum into the PTQ
PTQ Address	0	The destination address within the PTQ
Point Type	4	The range of registers to read from the Quantum 4 = 40:x style register
Quantum Address	1	The starting address of the data within the Quantum This would be Point Type + offset Example: 40000 + 1 = 40001
Word Count	100	The number of registers to transfer

The second command states:

Field	Value	Meaning
CMD TYPE	1	The type of operation to perform 1 = Read data from the Quantum into the PTQ
PTQ Address	100	The destination address within the PTQ
Point Type	4	The range of registers to read from the Quantum 4 = 40:x style register

Field	Value	Meaning
Quantum Address	101	The starting address of the data within the Quantum This would be Point Type + Quantum Address Example: 40000 + 101 = 40101
Word Count	107	The number of registers to transfer

The following diagram shows the result of this example.

Quantum Memory Address		PTQ Memory Address
40001		0
40010	First Command	10
40020		20
40030		30
40040		40
40050		50
40060	→	60
40070		70
40080		80
40090		90
40100		100
40110		110
40120	Second Command	120
40130	→	130
40140		140
40150		149
...		...

Defining Data to be Retrieved from the PTQ Database.

Your application may need to retrieve 50 words of data from other devices on the network. These devices have either sent you the data if you are a slave, or you have obtained it for your use if you happen to be a master in your application. Assuming that the data resides in registers 500 to 549 within the PTQ modules memory and you wish to place this information in addresses 400200 to 400249 within the Quantum you could use **Command Function 2** to transfer the information.

Because the total amount of data does not exceed 130 words in length, a single command can be used to transfer the data. This additional entry will be added to our [Backplane Data Exchange] section and it would look like the third command below:

```
[Backplane Data Exchange]
#   Cmd      PTQ      Point      QUANTUM      Word
#   Type     Address   Type        Address       Count
START
      1         0         4           1           100
# move data from Quantum to the PTQ
      1        100         4          101           50
# move data from Quantum to the PTQ
      2        500         4          200           50
# move data from PTQ to the Quantum

END
```

The third command states:

Field	Value	Meaning
CMD TYPE	2	The type of operation to perform 2 = Write data from the PTQ to the Quantum
PTQ Address	500	The destination address within the PTQ
Point Type	4	The range of registers to read from the Quantum 4 = 40:x style register
Quantum Address	200	The starting address of the data within the Quantum This would be Point Type + Quantum Address Example: 40000 + 200 = 40200
Word Count	50	The number of registers to transfer

The following diagram shows the result of this example:

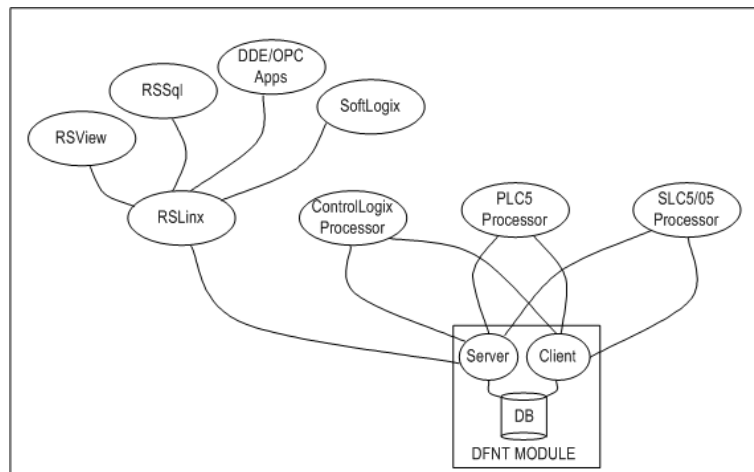
Quantum Memory Address		PTQ Memory Address
40001		0
40010	First Command	10
40020		20
40030		30
40040		40
40050		50
40060	→	60
40070		70
40080		80
40090		90
40100		100
40110		110
40120	Second Command	120
40130	→	130
40140		140
40150		149
...		...
40200		500
40210	Third Command	510
40220		520
40230	←	530
40240		540
40249		549
...		...

8.2.2 DFNT Server

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

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

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



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

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

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

MSG Instruction Type

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

MSG Instruction Type

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

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

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

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

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

Basic Command Set Functions

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

PLC-5 Command Set Functions

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

SLC-500 Command Set Functions

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

8.2.3 DFNT Client

In the client driver, the PTQ-DFNT module is responsible for issuing read or write commands to servers on the DFNT 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 (PROSOFT.CFG). Command status is returned to the processor for each individual command in the command list status block in the command control data area.

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)

8.3 PTQ-DFNT Error Status Table

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.

If the APP, BP ACT and OK LEDs blink at a rate of every one-second, this indicates a serious problem with the module. Contact ProSoft Technology for repair or replacement, as this is not a user serviceable item.

8.4 Cable Connections

The PTQ-DFNT module has the following communication connections on the module:

- One Ethernet port (RJ45 connector)
- One RS-232 Configuration/Debug port (DB9 connector)

8.4.1 Ethernet Connection

The PTQ-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.

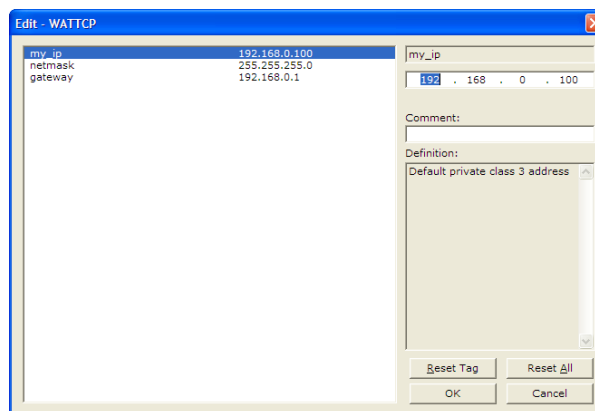
Note: Depending on hardware configuration, you may see more than one RJ45 port on the module. The Ethernet port is labeled *Ethernet*.

Warning: The PTQ-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

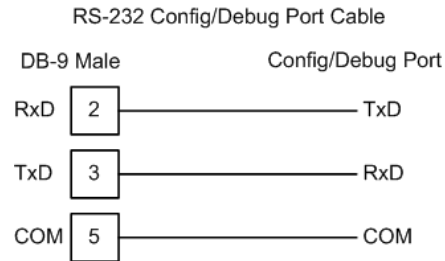
The wattcp.cfg file must be set up properly in order to use a TCP/IP network connection. You can view the current network configuration in *ProSoft Configuration Builder (PCB)*, as shown:



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 75).

8.4.2 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 for communications on this port is shown in the following diagram:



8.5 PTQ-DFNT Status Data

The status data block starts at address 4000 in the PTQ-DFNT database. The status data block is divided as follows:

Address	Block Description
4000 to 4008	Module information
4010 to 4019	Reserved
4020 to 4025	Client status area
4030 to 4035	Block transfer status area
4100	Server status area
4400	Client command status area

The status information can be copied into the Quantum database using a backplane command type 2. Each block is described in the following topics.

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

Offset	Content	Description
4000	Program Scan Count	This value is incremented each time a complete program cycle occurs in the module.
4001 to 4002	Product	Product Name (ASCII)
4003 to 4004	Rev	Revision (ASCII)
4005 to 4006	Op Sys	Operating System (ASCII)
4007 to 4008	Run	Production Run Number (ASCII)

8.5.1 HTTP Port 80 Status

Offset	Content	Description
4010	Not Used	Reserved
4011	Not Used	Reserved
4012	Not Used	Reserved
4013	Not Used	Reserved
4014	Not Used	Reserved
4015	Not Used	Reserved
4016	Not Used	Reserved
4017	Not Used	Reserved
4018	Not Used	Reserved
4019	Not Used	Reserved

8.5.2 DFNT Client Status

Offset	Content	Description
4020	Client Cmd Request	This value is incremented each time a command request is issued.
4021	Client Cmd Response	This value is incremented each time a command response is received.
4022	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.
4023	Client Request Count	This value is incremented each time a request message is issued.
4024	Client Response Count	This value is incremented each time a response message is received.
4025	Client Error Sent Count	This value is incremented each time an error is sent from the client.
4026	Client Error Received Count	This value is incremented each time an error is received from a remote unit.
4027	Client Cfg Error Word	This word contains a bit map that defines configuration errors in the configuration file for the client.
4028	Client Current Error Code	This value corresponds to the current error code for the client.
4029	Client Last Error Code	This value corresponds to the last error code recorded for the client.

8.5.3 Block Status

Offset	Content	Description
4030	Read Block Count	This field contains the total number of read blocks transferred from the module to the processor.
4031	Write Block Count	This field contains the total number of write blocks transferred from the processor to the module.
4032	Parse Block Count	This field contains the total number of blocks successfully parsed that were received from the processor.
4033	Command Event Block Count	This field contains the total number of command event blocks received from the processor.
4034	Command Block Count	This field contains the total number of command blocks received from the processor.
4035	Error Block Count	This field contains the total number of block errors recognized by the module.

8.5.4 Response Block (251)

Note: Each server occupies 12 words in this block.

8.5.5 DFNT Server 0 Status

Offset	Content	Description
4100	Socket State	
4101	Connection State	
4102	Open Count	Total number of times the server has performed an open operation.
4103	Establish Count	Total number of times a connection has been established on the server.
4104	Close Count	Total number of times the server has performed a close operation.
4105	Read	Total number of packets received by the server.
4106	Message	Total number of message receive by the server.
4107	Write	Total number of packets sent from the server to the client.
4108	Timeout	Total number of times the server as reached a connection timeout condition.
4109	Host Port	Service port on client connected to the server.
4110 to 4111	Host IP Address	IP address of the client connected to the server.

8.5.6 DFNT Server 1 Status

Offset	Content	Description
4112	Socket State	
4113	Connection State	
4114	Open Count	Total number of times the server has performed an open operation.
4115	Establish Count	Total number of times a connection has been established on the server.
4116	Close Count	Total number of times the server has performed a close operation.
4117	Read	Total number of packets received by the server.
4118	Message	Total number of message receive by the server.
4119	Write	Total number of packets sent from the server to the client.
4120	Timeout	Total number of times the server as reached a connection timeout condition.
4121	Host Port	Service port on client connected to the server.
4122 to 4123	Host IP Address	IP address of the client connected to the server.

The structure for DFNT servers 2 through 19 is identical to the structure shown for DFNT Servers 0 and 1.

Address ranges are as follows:

Server	Address Range
Server 2	4124 to 4135
Server 3	4136 to 4147
Server 4	4148 to 4159
Server 5	4160 to 4171
Server 6	4172 to 4183
Server 7	4184 to 4195
Server 8	4196 to 4207
Server 9	4208 to 4219
Server 10	4220 to 4231
Server 11	4232 to 4243
Server 12	4244 to 4255
Server 13	4256 to 4267
Server 14	4268 to 4279
Server 15	4280 to 4291
Server 16	4292 to 4303
Server 17	4304 to 4315
Server 18	4316 to 4327
Server 19	4328 to 4339

8.6 Error Codes

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

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

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

8.6.1 Local STS Error Codes

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

8.6.2 Remote STS Error Codes

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

8.6.3 Errors When EXT STS Is Present

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

8.6.4 Module Specific Error (not DFNT Compliant)

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

8.7 TCP/IP Interface Errors

8.7.1 Timeout Errors

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

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

8.7.2 Register Session Response Errors

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

8.7.3 Forward Open Response Errors

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

8.7.4 PCCC Response Errors

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

8.8 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			

8.9 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.

8.9.1 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.

8.9.2 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.

8.9.3 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.

8.9.4 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.

8.9.5 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.

8.9.6 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.

**8.9.7 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.

8.9.8 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.

**8.9.9 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.

**8.9.10 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.

**8.9.11 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.

8.9.12 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.

8.9.13 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.

8.9.14 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.

8.9.15 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.

8.9.16 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.

8.10 PLC-5 Processor Specifics

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

8.10.1 PLC-5 Sub-Element Codes

Timer / Counter

Code	Description
0	Control
1	Preset
2	Accumulated

Control

Code	Description
0	Control
1	Length
2	Position

PD*

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

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

BT

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

MG

Code	Description
0	Control
1	Error

Code	Description
2	RLEN
3	DLEN

8.11 SLC Processor Specifics

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

8.11.1 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.

8.12 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:

8.12.1 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.

8.13 ControlLogix Processor Specifics

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

8.15 Frequently Asked Questions

How fast do the "Backplane Data Exchange" commands run?

The "Backplane Data Exchange" commands will execute one at a time during the I/O service interval of the processor. What this means is that if you had a list of 10 commands at the end of every processor scan one command would execute. This would mean that it would take 10 processor scans to execute the 10 commands contained within the "Backplane Data Exchange" section of the configuration file.

What is the maximum number of words I can transfer with a "Backplane Data Exchange" command?

For command types 1 & 2 you may move up to 130 words with each command. Function 3 is somewhat different in that it provides only 64 words of data movement BUT since it is intended to solve very specialized operations its size must be restricted.

Do I need to use "Backplane Data Exchange" function 3?

The only time you should need it is if you are using the DNPSNET, DNP or one of the IEC protocols. If you are using one of these protocols then you can find sample structured text examples included in the manual for these protocols. In all other instances you should not need to use this function.

How much data can I transfer between the processor and the Module.

You can enter up to 100 commands in the [BACKPLANE DATA EXCHANGE] section of the configuration file. The limit for any single execution of a Function 1 or 2 is 130 words but you may enter multiple commands to transfer more data.

How do I configure the module?

The PTQ requires a simple text based configuration file to make it operational. For a really quick tutorial on the modules communications with the processor you should review the [QUICK START GUIDE] or for more in depth information the chapter on "Backplane Data Exchange" should answer most questions.

What software application is required for my Ladder Logic?

The design of the module should be software independent and for many installations minimal or possibly no ladder will be required. The section on "Backplane Data Exchange" offers to samples to help in the few instances where ladder is required.

What kind of data transfer rates can I expect between the processor and the module?

Data transfer rates between the processor and the module depend on a number of variables, among them the number of words being transferred per command, the amount of other network traffic at the time data is being transferred, and overall processor scan times.

Is a .MDC available for configuration of the Module?

Yes. The CD-ROM that ships with the module should have a version for both Concept 2.5 and 2.6 in the ProTalk directory.

Does the module work in a remote rack?

The module is designed to be located in the chassis with the processor and will not operate in a remote chassis. If your application requires remote placement of the communication device you should investigate the other members of the ProLinx family such as the 4202-MNET-DFCM (if you require DF1 connectivity for example although many others are available). This module for example would allow you to communicate with DF1 devices and allow you to map the contents of its memory using Modbus TCP/IP.

Can I use the PTQ module in a hot backup system?

Support for Hot Backup is not currently implemented in the module. We are currently investigating the addition of this functionality but until this development can be finalized it may be possible to use one of the 4000 series of ProLinx Communication products. Please call our technical support technicians when considering this application.

9 Support, Service & Warranty

In This Chapter

- ❖ How to Contact Technical Support 145
- ❖ Return Material Authorization (RMA) Policies and Conditions..... 147
- ❖ LIMITED WARRANTY..... 149

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.

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

9.1 Return Material Authorization (RMA) Policies and Conditions

The following RMA Policies and Conditions (collectively, "RMA Policies") apply to any returned Product. These RMA Policies are subject to change by ProSoft without notice. For warranty information, see Limited Warranty (page 149). In the event of any inconsistency between the RMA Policies and the Warranty, the Warranty shall govern.

9.1.1 All Product Returns:

- a) In order to return a Product for repair, exchange or otherwise, the Customer must obtain a Return Material Authorization (RMA) number from ProSoft and comply with ProSoft shipping instructions.
- b) In the event that the Customer experiences a problem with the Product for any reason, Customer should contact ProSoft Technical Support at one of the telephone numbers listed above (page 145). A Technical Support Engineer will request that you perform several tests in an attempt to isolate the problem. If after completing these tests, the Product is found to be the source of the problem, we will issue an RMA.
- c) All returned Products must be shipped freight prepaid, in the original shipping container or equivalent, to the location specified by ProSoft, and be accompanied by proof of purchase and receipt date. The RMA number is to be prominently marked on the outside of the shipping box. Customer agrees to insure the Product or assume the risk of loss or damage in transit. Products shipped to ProSoft using a shipment method other than that specified by ProSoft, or shipped without an RMA number will be returned to the Customer, freight collect. Contact ProSoft Technical Support for further information.
- d) A 10% restocking fee applies to all warranty credit returns, whereby a Customer has an application change, ordered too many, does not need, etc. Returns for credit require that all accessory parts included in the original box (i.e.; antennas, cables) be returned. Failure to return these items will result in a deduction from the total credit due for each missing item.

9.1.2 Procedures for Return of Units Under Warranty:

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

- a) A replacement module will be shipped and invoiced. A purchase order will be required.
- b) Credit for a product under warranty will be issued upon receipt of authorized product by ProSoft at designated location referenced on the Return Material Authorization
 - i. If a defect is found and is determined to be customer generated, or if the defect is otherwise not covered by ProSoft's warranty, there will be no credit given. Customer will be contacted and can request module be returned at their expense;
 - ii. If defect is customer generated and is repairable, customer can authorize ProSoft repair the unit by providing a purchase order for 30% of the current list price plus freight charges, duties and taxes as applicable.

9.1.3 Procedures for Return of Units Out of Warranty:

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

The following is a list of non-repairable units:

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

9.2 LIMITED WARRANTY

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

9.2.1 What Is Covered By This Warranty

- a) *Warranty On New Products:* ProSoft warrants, to the original purchaser, that the Product that is the subject of the sale will (1) conform to and perform in accordance with published specifications prepared, approved and issued by ProSoft, and (2) will be free from defects in material or workmanship; provided these warranties only cover Product that is sold as new. This Warranty expires three (3) years from the date of shipment for Product purchased **on or after** January 1st, 2008, or one (1) year from the date of shipment for Product purchased **before** January 1st, 2008 (the "Warranty Period"). If the Customer discovers within the Warranty Period a failure of the Product to conform to specifications, or a defect in material or workmanship of the Product, the Customer must promptly notify ProSoft by fax, email or telephone. In no event may that notification be received by ProSoft later than 39 months from date of original shipment. Within a reasonable time after notification, ProSoft will correct any failure of the Product to conform to specifications or any defect in material or workmanship of the Product, with either new or remanufactured replacement parts. ProSoft reserves the right, and at its sole discretion, may replace unrepairable units with new or remanufactured equipment. All replacement units will be covered under warranty for the 3 year period commencing from the date of original equipment purchase, not the date of shipment of the replacement unit. Such repair, including both parts and labor, will be performed at ProSoft's expense. All warranty service will be performed at service centers designated by ProSoft.
- b) *Warranty On Services:* Materials and labor performed by ProSoft to repair a verified malfunction or defect are warranted in the terms specified above for new Product, provided said warranty will be for the period remaining on the original new equipment warranty or, if the original warranty is no longer in effect, for a period of 90 days from the date of repair.

9.2.2 What Is Not Covered By This Warranty

- a) ProSoft makes no representation or warranty, expressed or implied, that the operation of software purchased from ProSoft will be uninterrupted or error free or that the functions contained in the software will meet or satisfy the purchaser's intended use or requirements; the Customer assumes complete responsibility for decisions made or actions taken based on information obtained using ProSoft software.
- b) This Warranty does not cover the failure of the Product to perform specified functions, or any other non-conformance, defects, losses or damages caused by or attributable to any of the following: (i) shipping; (ii) improper installation or other failure of Customer to adhere to ProSoft's specifications or instructions; (iii) unauthorized repair or maintenance; (iv) attachments, equipment, options, parts, software, or user-created programming (including, but not limited to, programs developed with any IEC 61131-3, "C" or any variant of "C" programming languages) not furnished by ProSoft; (v) use of the Product for purposes other than those for which it was designed; (vi) any other abuse, misapplication, neglect or misuse by the Customer; (vii) accident, improper testing or causes external to the Product such as, but not limited to, exposure to extremes of temperature or humidity, power failure or power surges; or (viii) disasters such as fire, flood, earthquake, wind and lightning.
- c) The information in this Agreement is subject to change without notice. ProSoft shall not be liable for technical or editorial errors or omissions made herein; nor for incidental or consequential damages resulting from the furnishing, performance or use of this material. The user guide included with your original product purchase from ProSoft contains information protected by copyright. No part of the guide may be duplicated or reproduced in any form without prior written consent from ProSoft.

9.2.3 Disclaimer Regarding High Risk Activities

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

9.2.4 Intellectual Property Indemnity

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

- a) Any documentation included with Product purchased from ProSoft is protected by copyright and may not be duplicated or reproduced in any form without prior written consent from ProSoft.
- b) ProSoft's technical specifications and documentation that are included with the Product are subject to editing and modification without notice.
- c) Transfer of title shall not operate to convey to Customer any right to make, or have made, any Product supplied by ProSoft.
- d) Customer is granted no right or license to use any software or other intellectual property in any manner or for any purpose not expressly permitted by any license agreement accompanying such software or other intellectual property.
- e) Customer agrees that it shall not, and shall not authorize others to, copy software provided by ProSoft (except as expressly permitted in any license agreement accompanying such software); transfer software to a third party separately from the Product; modify, alter, translate, decode, decompile, disassemble, reverse-engineer or otherwise attempt to derive the source code of the software or create derivative works based on the software; export the software or underlying technology in contravention of applicable US and international export laws and regulations; or use the software other than as authorized in connection with use of Product.
- f) **Additional Restrictions Relating To Software And Other Intellectual Property**

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

9.2.5 Disclaimer of all Other Warranties

The Warranty set forth in What Is Covered By This Warranty (page 149) are in lieu of all other warranties, express or implied, including but not limited to the implied warranties of merchantability and fitness for a particular purpose.

9.2.6 Limitation of Remedies **

In no event will ProSoft or its Dealer be liable for any special, incidental or consequential damages based on breach of warranty, breach of contract, negligence, strict tort or any other legal theory. Damages that ProSoft or its Dealer will not be responsible for include, but are not limited to: Loss of profits; loss of savings or revenue; loss of use of the product or any associated equipment; loss of data; cost of capital; cost of any substitute equipment, facilities, or services; downtime; the claims of third parties including, customers of the Purchaser; and, injury to property.

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

9.2.7 Time Limit for Bringing Suit

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

9.2.8 No Other Warranties

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

9.2.9 Allocation of Risks

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

9.2.10 Controlling Law and Severability

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

Index

I

[Backplane Data Exchange] • 54, 98
[DFNT Client x Commands] • 65
[DFNT Client x] • 64
[Module] • 53

A

Adding the PTQ Module to the Project • 18, 30
All Product Returns: • 147
Allocation of Risks • 152

B

Backplane Data Exchange • 97, 98
Basic Command Set Functions • 66
Block IDs 3000 to 3001/3100 to 3101 - Port 1 & 2
 Slave Polling Control • 60
Block IDs 3002 to 3006/3102 to 3106 - Slave Polling
 Status • 61
Block IDs 5001-5006/5101-5106 - Command Control
 Request • 62
Block Status • 110
BT • 136
Building the Project • 19

C

Cable Connections • 107
Clearing a Fault Condition • 77
Client Command List • 105
Command Entry Formats • 68, 70, 71
Command Function Codes • 119
Command List • 65
Configuring Module Parameters • 52
Configuring the Module • 49
Configuring the Processor with Concept • 25
Configuring the Processor with ProWORX • 39
Configuring the Processor with Unity Pro • 15
Connect the PC to the ProTalk Configuration/Debug
 Port • 46
Connect Your PC to the Processor • 20
Connecting to the Processor with TCPIP • 22
Control • 136
Controlling Law and Severability • 153
ControlLogix Processor Specifics • 140
Creating a New Project • 16, 28

D

Data Transfer • 98
Database View Menu • 82, 86
Defining Data to be Retrieved from the PTQ Database.
 • 100
Defining Data to be Sent to the PTQ Database • 98

DFNT Client • 104
DFNT Client Status • 110
DFNT Command Entry Form • 141
DFNT Server • 102
DFNT Server 0 Status • 111
DFNT Server 1 Status • 112
Diagnostics and Troubleshooting • 9, 75, 107
Disclaimer of all Other Warranties • 151
Disclaimer Regarding High Risk Activities • 150
Displaying the Current Page of Registers Again • 86
Downloading the Project to the Module Using a Serial
 COM port • 73
Downloading the Project to the Processor • 23, 34

E

Error Codes • 113
Error Status Table • 78
Errors When EXT STS Is Present • 115
Ethernet Configuration • 72
Ethernet Connection • 107
Ethernet LED Indicators • 76
Ethernet Port Configuration - wattcp.cfg • 107
EtherNet/IP (Explicit Messaging) Compatible Devices •
 94
Event Port Block (1000/2000) • 59
Exiting the Configuration/Debug Program • 85

F

Forward Open Response Errors • 117
Frequently Asked Questions • 142
Function Code #1 - Protected Write (Basic Command
 Set) • 121
Function Code #100 - Word Range Write (PLC-5
 Command) (Binary Address) • 125
Function Code #101 - Word Range Read (PLC-5
 Command) (Binary Address) • 126
Function Code #102 - Read-Modify-Write (PLC-5
 Command) (Binary Address) • 127
Function Code #150 - Word Range Write (PLC-5
 Command) (ASCII Address) • 128
Function Code #151 - Word Range Read (PLC-5
 Command) (ASCII Address) • 129
Function Code #152 - Read-Modify-Write (PLC-5
 Command) (ASCII Address) • 130
Function Code #2 - Unprotected Read (Basic
 Command Set) • 121
Function Code #3 - Protected Bit Write (Basic
 Command Set) • 122
Function Code #4 - Unprotected Bit Write (Basic
 Command Set) • 123
Function Code #5 - Unprotected Write (Basic
 Command Set) • 124
Function Code #501 - Protected Typed Logical Read
 (Two Address Fields) • 131
Function Code #502 - Protected Typed Logical Read
 (Three Address Fields) • 132
Function Code #509 - Protected Typed Logical Write
 (Two Address Fields) • 133

Function Code #510 - Protected Typed Logical Write (Three Address Fields) • 134
Function Code #511 - Protected Typed Logical Write with Mask (Three Address Fields) • 135
Functional Overview • 9, 97
Functional Specifications - PTQMDC/DFNT • 96

G

General Command Structure • 120
General Specifications • 94
Guide to the PTQ-DFNT User Manual • 9

H

Hardware and Software Requirements • 12
Hardware Specifications • 95
How to Contact Technical Support • 145, 147
How to Contact Us • 2
HTTP Port 80 Status • 109

I

Information for Concept Version 2.6 Users • 26
Information for ProTalk® Product Users • 3
Installing MDC Configuration Files • 26
Installing ProSoft Configuration Builder Software • 14
Installing the ProTalk Module in the Quantum Rack • 44
Intellectual Property Indemnity • 151

K

Keystrokes • 81

L

LED Status Indicators • 76
Limitation of Remedies ** • 152
LIMITED WARRANTY • 147, 149
Local STS Error Codes • 113

M

Main Menu • 81
Master Command Error List Menu • 88
MG • 136
MicroLogix Processor Specifics • 139
Minimum Command Delay • 64
Module Specific Error (not DFNT Compliant) • 116
Moving Back Through 5 Pages of Commands • 88
Moving Back Through 5 Pages of Registers • 87
Moving Forward (Skipping) Through 5 Pages of Commands • 88
Moving Forward (Skipping) Through 5 Pages of Registers • 87

N

Navigation • 81
Network Menu • 85, 89
No Other Warranties • 152

O

Opening the Client Command Error List Menu • 82
Opening the Client Command List Menu • 83
Opening the Database View Menu • 82
Opening the Network Menu • 85

P

Package Contents • 12
PC and PC Software • 13
PCCC Response Errors • 118
PD* • 136
Pinouts • 3, 107
PLC-5 Command Set Functions • 66
PLC-5 Processor Specifics • 136
PLC-5 Sub-Element Codes • 136
Printing a Configuration File • 52
Procedures for Return of Units Out of Warranty: • 148
Procedures for Return of Units Under Warranty: • 148
Product Specifications • 9, 94
ProSoft Technology® Product Documentation • 2
PTQ-DFNT Error Status Table • 106
PTQ-DFNT Status Data • 109

Q

Quantum Hardware • 12

R

Reading Status Data from the Module • 91
Redisplaying the Current Page • 88
Reference • 9, 93
Register Session Response Errors • 117
Remote STS Error Codes • 114
Renaming PCB Objects • 51
Resetting Diagnostic Data • 84
Response Block (251) • 110
Response Timeout • 64
Retry Count • 64
Return Material Authorization (RMA) Policies and Conditions • 147
Returning to the Main Menu • 87, 88, 90
RS-232 Configuration/Debug Port • 108

S

Set Up the Project • 50
Setting Up Command Function 1 (Read Data from the Quantum) • 55
Setting Up Command Function 2 (Write Data to the Quantum) • 56
Setting Up Command Function 3 (Special Functions) • 57
Setting up Data Memory in Project • 32, 48
Setting Up the ProTalk Module • 43
SLC File Types • 138, 139
SLC Processor Specifics • 138
SLC-500 Command Set Functions • 67
Special Functions • 58
Start Here • 9, 11

Support, Service & Warranty • 9, 145

T

TCP/IP Interface Errors • 117
Time Limit for Bringing Suit • 152
Timeout Errors • 117
Timer / Counter • 136
Transferring the Configuration File from The Module to the PC • 83
Transferring the Configuration File from the PC to the Module • 83
Transferring WATTCP.CFG to the Module • 89
Transferring WATTCP.CFG to the PC • 89
Troubleshooting • 77

U

Using ProSoft Configuration Builder • 50
Using ProSoft Configuration Builder (PCB) for Diagnostics • 79
Using the Diagnostic Window in ProSoft Configuration Builder • 79

V

Verifying Communication Between the Processor and the Module • 48
Verifying Jumper Settings • 44
Verifying Successful Download • 36
Viewing Block Transfer Statistics • 82
Viewing Client Communication Status • 84
Viewing Client Configuration • 84
Viewing Data in ASCII (Text) Format • 87
Viewing Data in Decimal Format • 9, 87
Viewing Data in Floating-Point Format • 87
Viewing Data in Hexadecimal Format • 87
Viewing Module Configuration • 82
Viewing Register Pages • 86
Viewing Server Status Data • 85
Viewing the Backplane Command List • 83
Viewing the Next Page of Commands • 88
Viewing the Next Page of Registers • 87
Viewing the Previous Page of Commands • 88
Viewing the Previous Page of Registers • 87
Viewing the WATTCP.CFG File on the module • 90
Viewing Version Information • 84

W

Warm Booting the Module • 84
Warm/Cold boot Module Block (9999/9998) • 63
Warnings • 3
What Is Covered By This Warranty • 149, 151
What Is Not Covered By This Warranty • 150

Y

Your Feedback Please • 2