ProSoft® TECHNOLOGY

Where Automation Connects.

ProTalk®

PTQ-DFCM

Quantum / Unity Platform

DF1 Communication Module

11/11/2008

USER MANUAL

Please Read This Notice

Successful application of this module requires a reasonable working knowledge of the Schneider Electric Quantum / Unity hardware, the PTQ-DFCM Module and the application in which the combination is to be used. For this reason, it is important that those responsible for implementation satisfy themselves that the combination will meet the needs of the application without exposing personnel or equipment to unsafe or inappropriate working conditions.

This manual is provided to assist the user. Every attempt has been made to ensure that the information provided is accurate and a true reflection of the product's installation requirements. In order to ensure a complete understanding of the operation of the product, the user should read all applicable Schneider Electric documentation on the operation of the Schneider Electric hardware.

Under no conditions will ProSoft Technology be responsible or liable for indirect or consequential damages resulting from the use or application of the product.

Information in this manual is subject to change without notice and does not represent a commitment on the part of ProSoft Technology Improvements and/or changes in this manual or the product may be made at any time. These changes will be made periodically to correct technical inaccuracies or typographical errors.

PTQ Installation and Operating Instructions

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.

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 µA.

Maximum battery charge voltage = 3.4 VDC.

Maximum battery charge current = 500 µA.

Maximum battery discharge current = 30 µA.

Your Feedback Please

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

ProSoft Technology
1675 Chester Avenue, Fourth Floor
Bakersfield, CA 93301
+1 (661) 716-5100
+1 (661) 716-5101 (Fax)
http://www.prosoft-technology.com

PTQ-DFCM User Manual
11/11/2008

ProSoft® Product Documentation

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

Contents

Guide to the PTQ-DFCM User Manual

Function		Section to Read	Details
Introduction (Must Do)	→	Start Here (page 9)	This Section introduces the customer to the module. Included are: package contents, system requirements, hardware installation, and basic configuration.
Verify Communication, Diagnostic and Troubleshooting	→	Verifying Communication (page 47) Diagnostics and Troubleshooting (page 77)	This section describes how to verify communications with the network. Diagnostic and Troubleshooting procedures.
Reference Product Specifications Functional Overview Glossary	→	Reference (page 95) Functional Overview (page 98) Product Specifications (page 95)	These sections contain general references associated with this product, Specifications, and the Functional Overview.
Support, Service, and Warranty Index	→	Support, Service and Warranty (page 133)	This section contains Support, Service and Warranty information. Index of chapters.

1 Start Here

In This Chapter

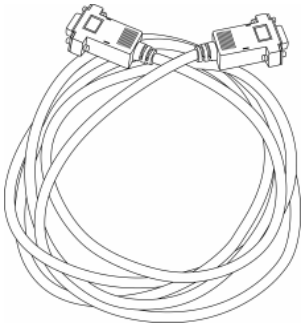
This guide is intended to guide you through the ProTalk module setup process, from removing the module from the box to exchanging data with the processor. In doing this, you will learn how to:

- 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

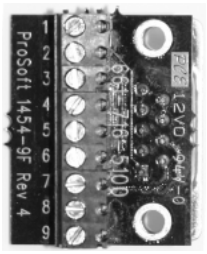
1.1 Hardware and Software Requirements

1.1.1 ProTalk Module Carton 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 / Unity Hardware

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

- Quantum or Unity 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

- Windows-based PC with at least one COM port
- Quantum programming software installed on machine

  or

- Concept™ PLC Programming Software version 2.6

  or
  ProWORX PLC Programming Software
  or
  UnityPro XL PLC Programming Software

- HyperTerminal (used in this guide) This is a communication program that is included with Microsoft Windows. You can normally find it in **Start / Programs / accessories / Communications**.

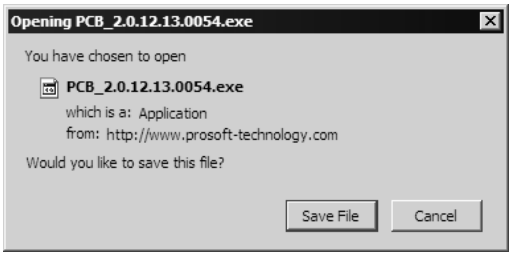
Note: ProTalk modules are compatible with common Quantum / Unity programming applications, including Concept and UnityPro XL. For all other programming applications, please contact technical support.

1.2 Install ProSoft Configuration Builder Software

You must install the ProSoft Configuration Builder (PCB) software in order to configure the PTQ-DFCM module. You can always get the newest version of ProSoft Configuration Builder from the ProSoft Technology web site.

To install ProSoft Configuration Builder from the ProSoft Web Site

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. The following illustrations show the file download prompt for two of the most common web browsers.



4. Make a note of the location where you saved the file, for example "Desktop", or "My Documents", so you can start the installation program.
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 CD-ROM, included in the package with your PTQ-DFCM module.

To install ProSoft Configuration Builder from the CD-ROM

1. Insert the ProSoft Solutions CD-ROM into the CD drive of your PC. Wait for the startup screen to appear.
2. On the startup screen, click *Product Documentation*. This action opens an explorer window.

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 *ProSoft Configuration Builder Setup* program and follow the instructions on your screen to install the software on your PC.

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 Concept

In This Chapter

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 take this into account when monitoring the status of the PTQ module.

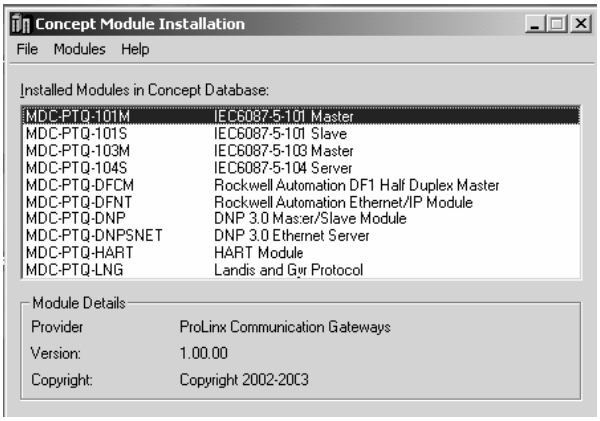
2.1 Information for Concept Version 2.6 Users

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.

2.1.1 Installing MDC Configuration Files

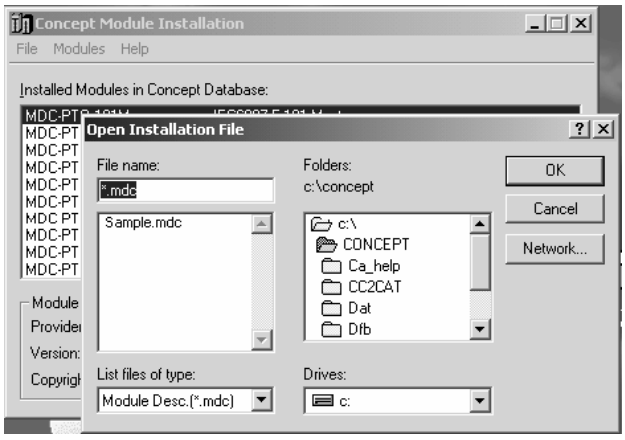
1 From a PC with Concept 2.6 installed, choose **Start / Programs / Concept / ModConnect Tool**.

This action opens the Concept Module Installation dialog box.



2 Choose **File / Open Installation File**.

This action opens the Open Installation File dialog box:

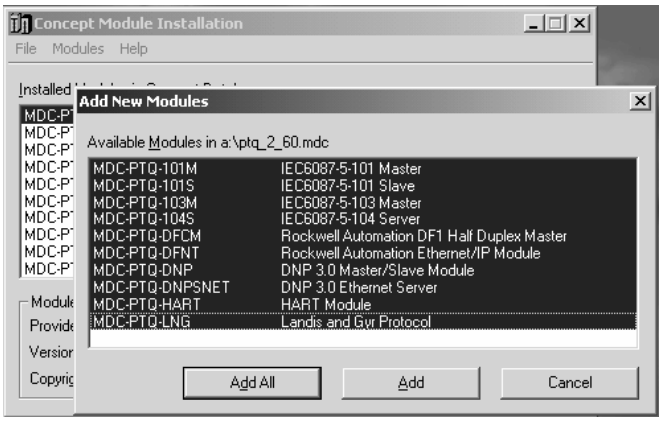


3 If you are using a Quantum processor, you will need the MDC files. In the Open Installation File dialog box, navigate to the **MDC Files** directory on the ProTalk CD.

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.



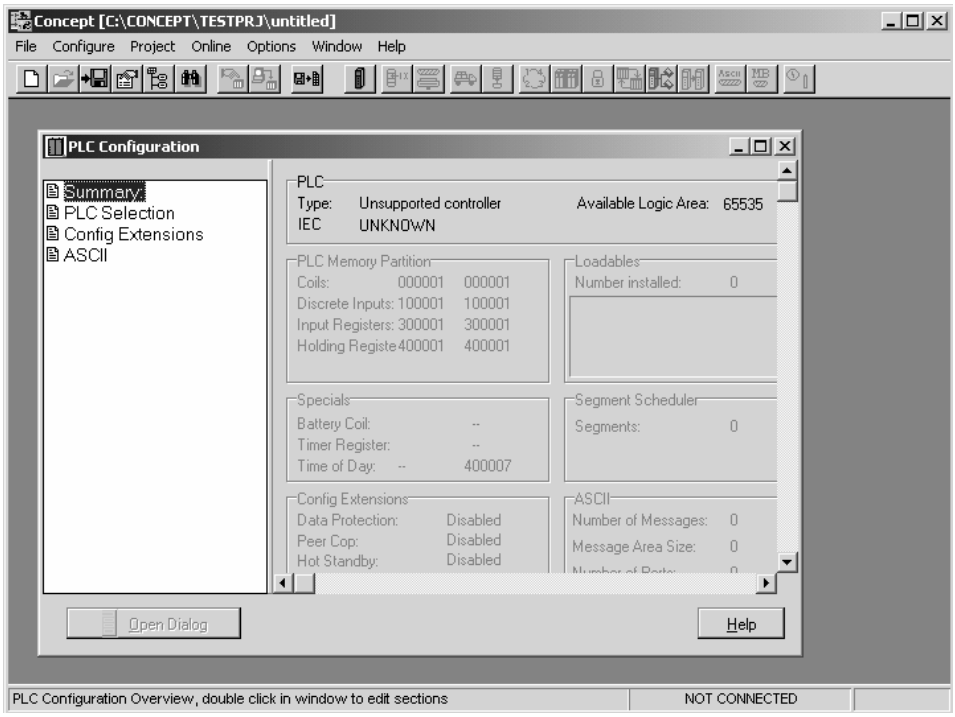
5 Click the **add all** button. A series of message boxes may appear during this process. Click **Yes** or **OK** for each message that appears.

6 When the process is complete, open the File menu and choose Exit to save your changes.

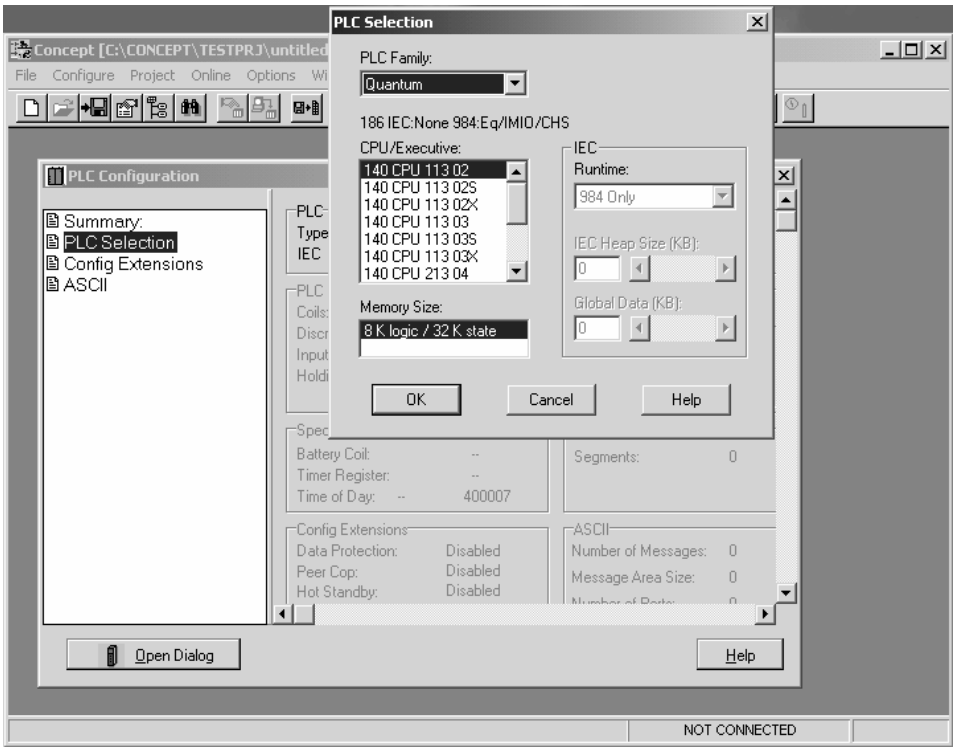
2.2 Create a New Project

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

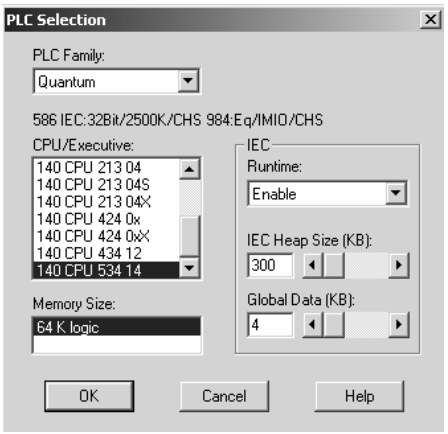
1. From your computer, choose **Start / Programs / Concept V2.6 XL.EN / Concept**. This action opens the Concept window.
2. Open the File menu, and then choose **New Project**. This action opens the PLC Configuration dialog box.



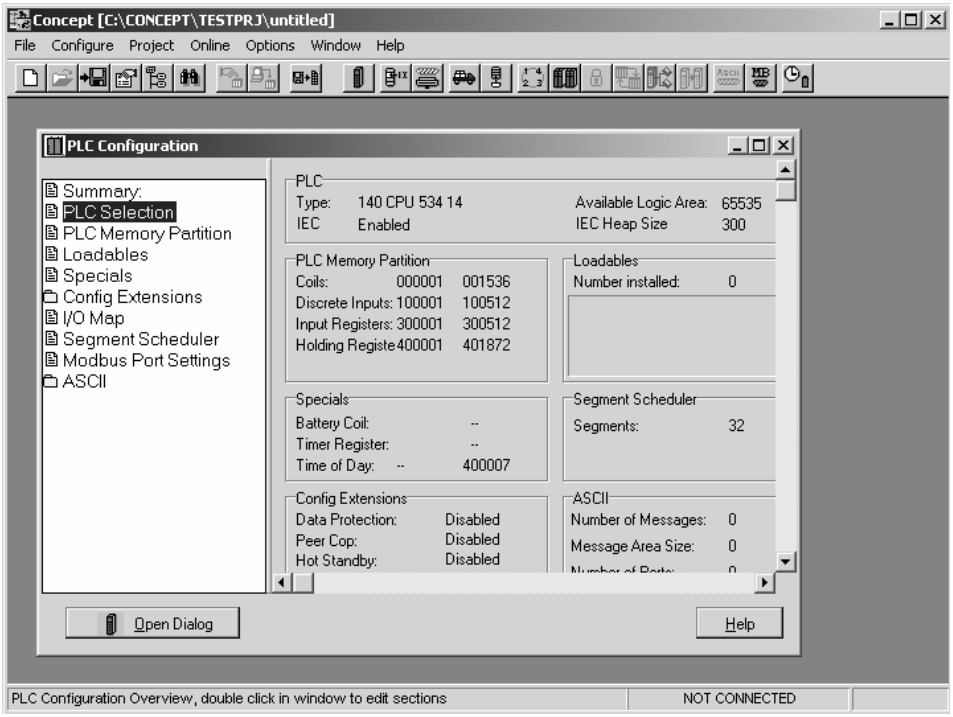
3 In the list of options on the left side of this dialog box, double-click the *PLC Selection* folder. This action opens the PLC Selection dialog box.



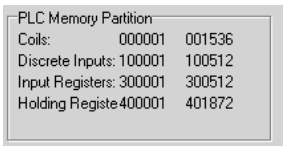
4 In the *CPU/Executive* pane, use the scroll bar to locate and select the PLC to configure.



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

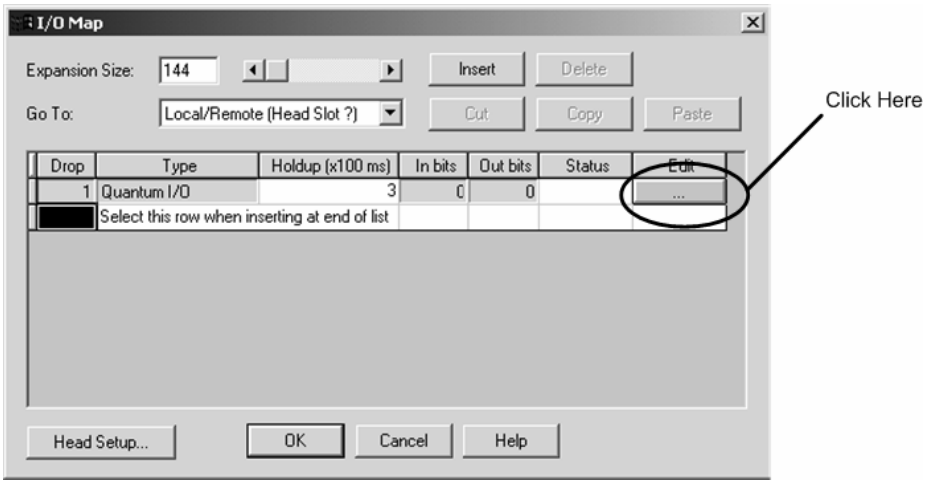


6 Make a note of the holding registers for the module. You will need this information when you modify your application as outlined in the ProTalk application Reference Guides. The Holding Registers are displayed in the PLC Memory Partition pane of the PLC Configuration dialog box.

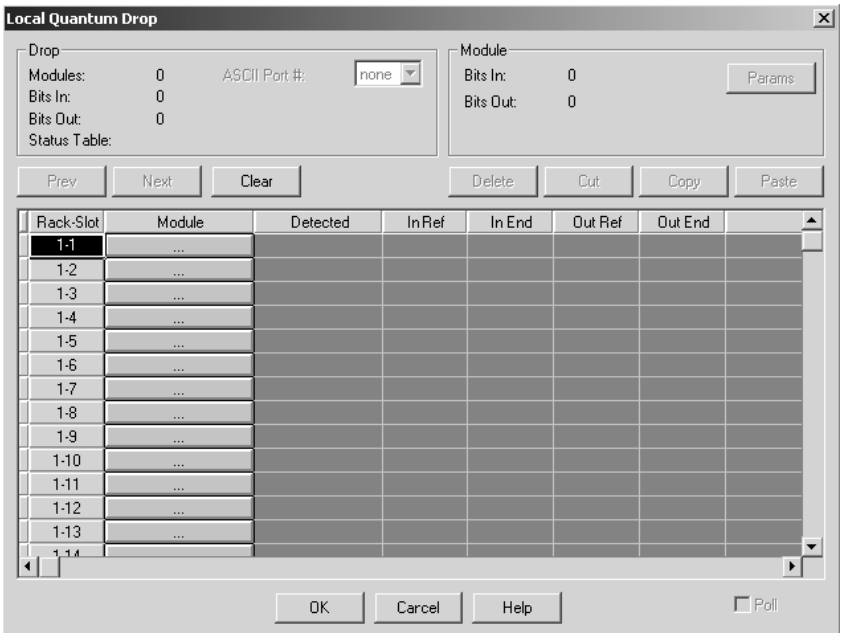


2.3 Add the PTQ Module to the Project

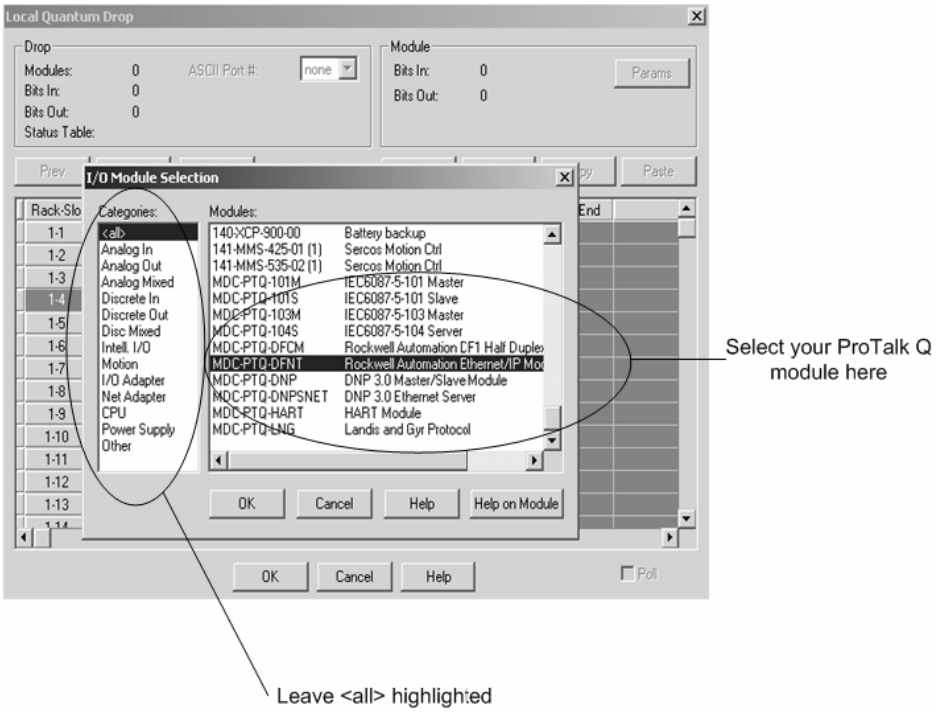
1 In the list of options on the left side of the *PLC Configuration* dialog box, double-click *I/O Map*. This action opens the I/O Map dialog box.



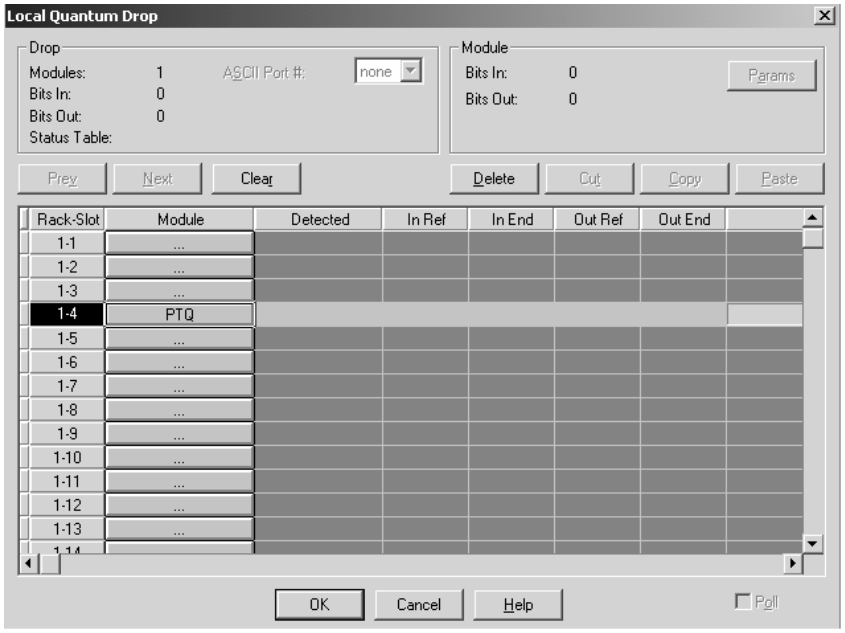
2 Click the **Edit** button to open the *Local Quantum Drop* dialog box. This dialog box is where you identify rack and slot locations.



3 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.



4 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.

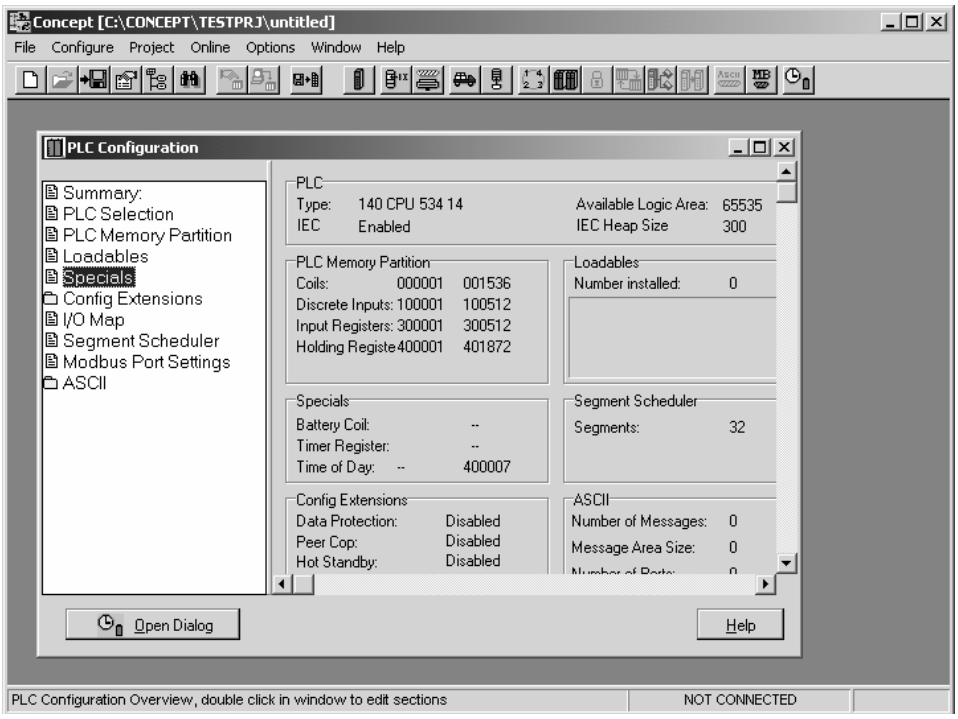


5 Repeat steps 3 through 5 for each ProTalk module you plan to install. When you have finished installing your ProTalk modules, click OK to save your settings. Click Yes to confirm your settings.

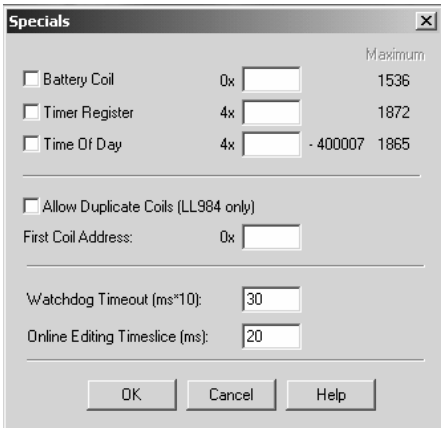
Tip: Select a module, and then click the Help on Module button for help pages.

2.4 Set up Data Memory in Project

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

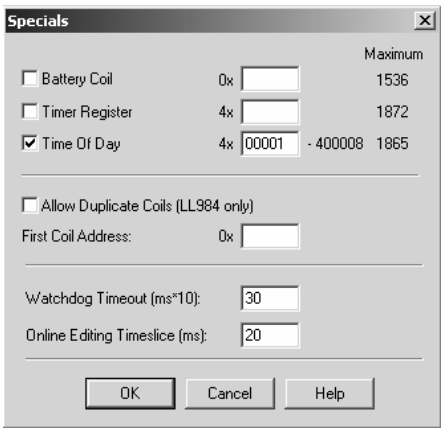


2 This action opens the Specials dialog box.



Selecting the Time of Day

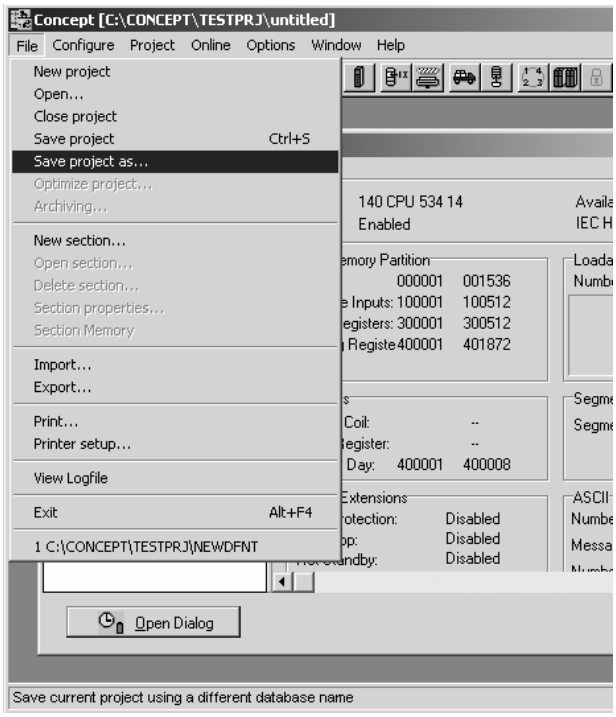
1 Select (check) the Time of Day box, and then enter the value 00001 as shown in the following example. This value sets the first time of day register to 400001.



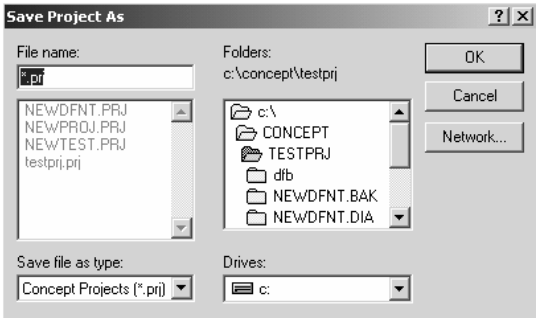
2 Click OK to save your settings and close the Specials dialog box.

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.

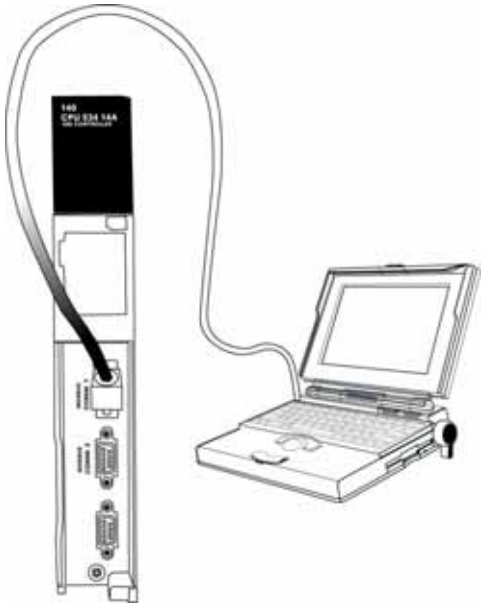


3 Name the project, and then click OK to save the project to a file.

2.5 Download the Project to the Processor

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

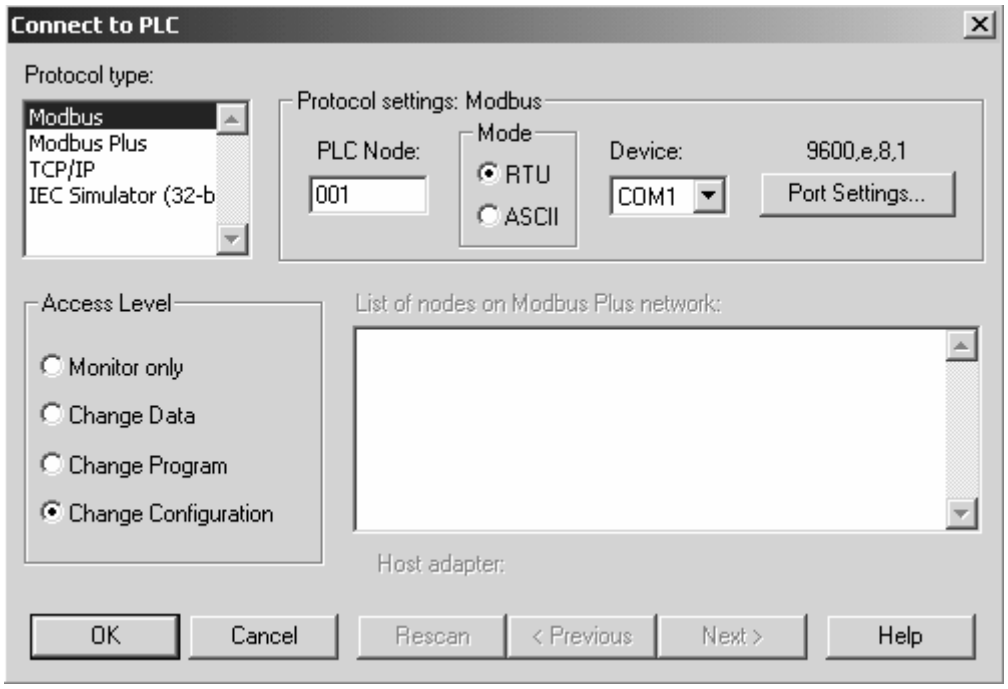
1 Use the null modem cable to connect your PC's serial port to the Quantum processor, as shown in the following illustration.



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.

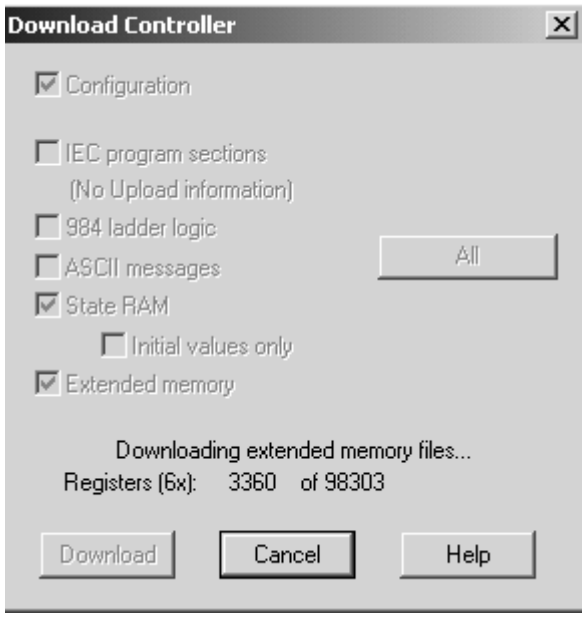


4 Leave the default settings as shown and click OK.

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.

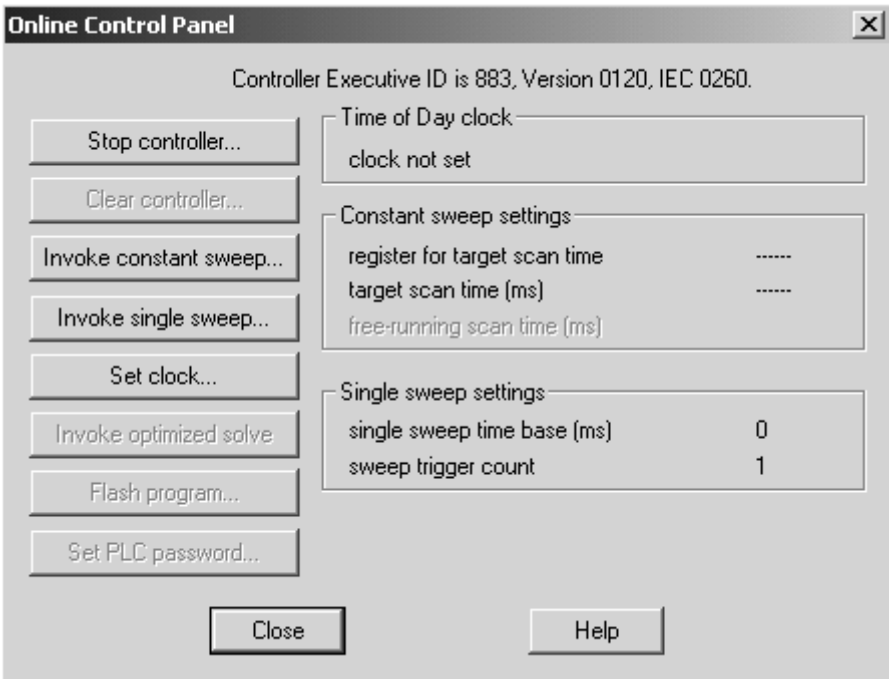


7 When the download is complete, you will be prompted to restart the controller. Click Yes to restart the controller.

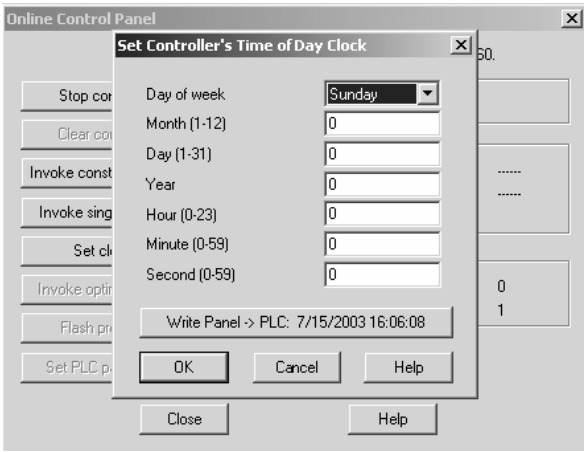
2.6 Verify Successful Download

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.

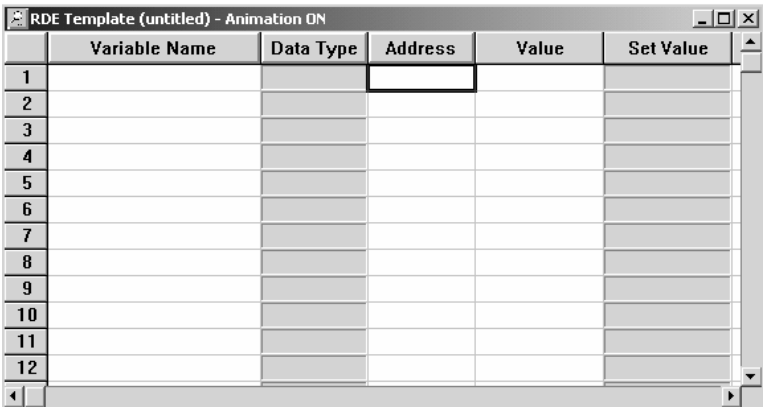


2 Click the Set Clock button to open the Set Controller's Time of Day Clock dialog box.

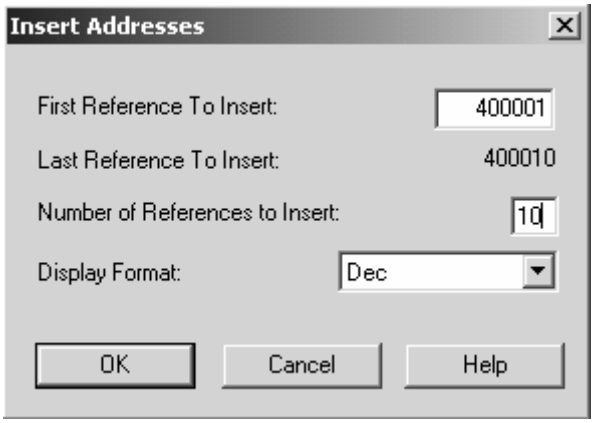


3 Click the Write Panel button. This action updates the date and time fields in this dialog box. Click OK to close this dialog box and return to the previous window.

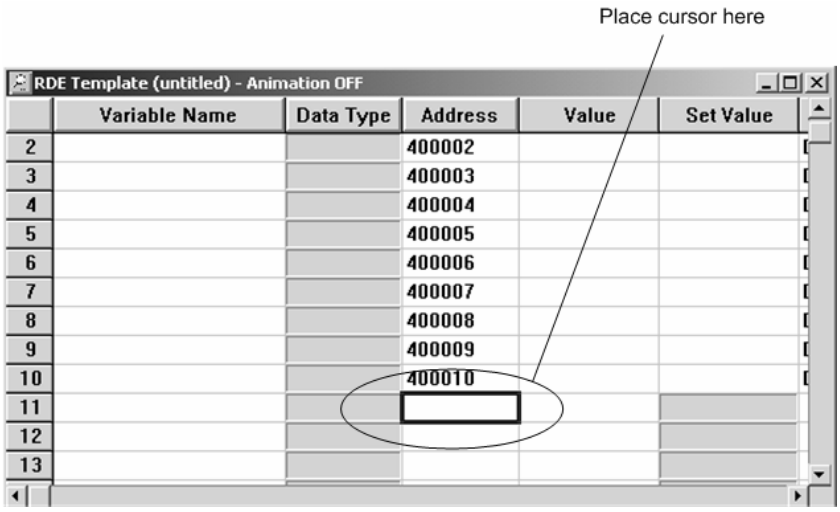
4 Click Close to close the Online Control Panel dialog box.
5 In the PLC Configuration window, open the Online menu, and then choose Reference Data Editor. This action opens the Reference Data Editor dialog box. On this dialog box, you will add preset values to data registers that will later be monitored in the ProTalk module.
6 Place the cursor over the first address field, as shown in the following illustration.



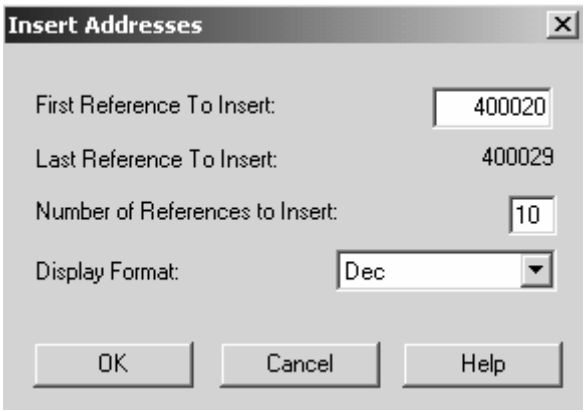
7 In the PLC Configuration window, open the Templates menu, and then choose Insert addresses. This action opens the Insert addresses dialog box.
8 On the Insert addresses dialog box, enter the values shown in the following illustration, and then click OK.



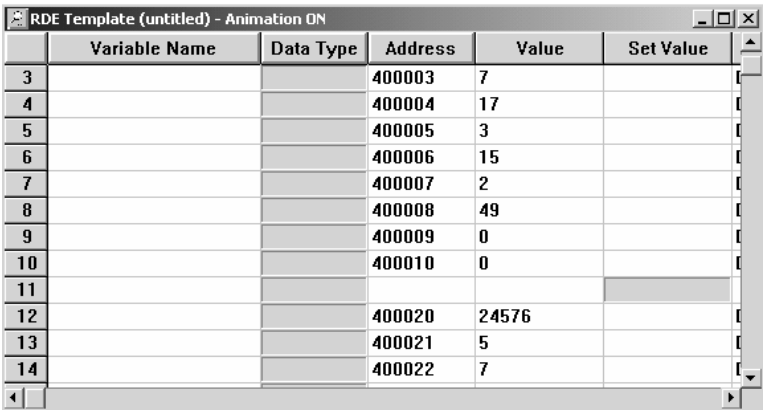
9 Notice that the template populates the address range, as shown in the following illustration. Place your cursor as shown in the first blank address field below the addresses you just entered.



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 on 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.

3 Configuring the Processor with ProWORX

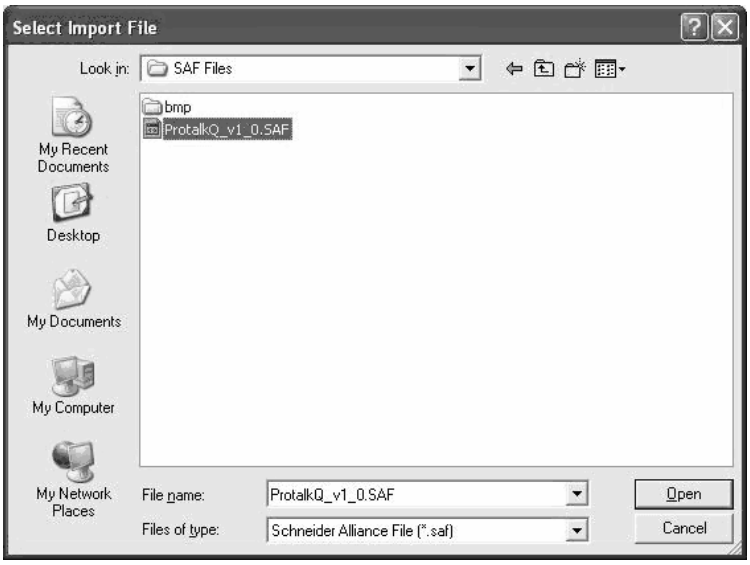
When you use ProWORX 32 software to configure the processor, use the example SaF file provided on the ProTalk Solutions CD-ROM.

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 take this into account when monitoring the status of the PTQ module.

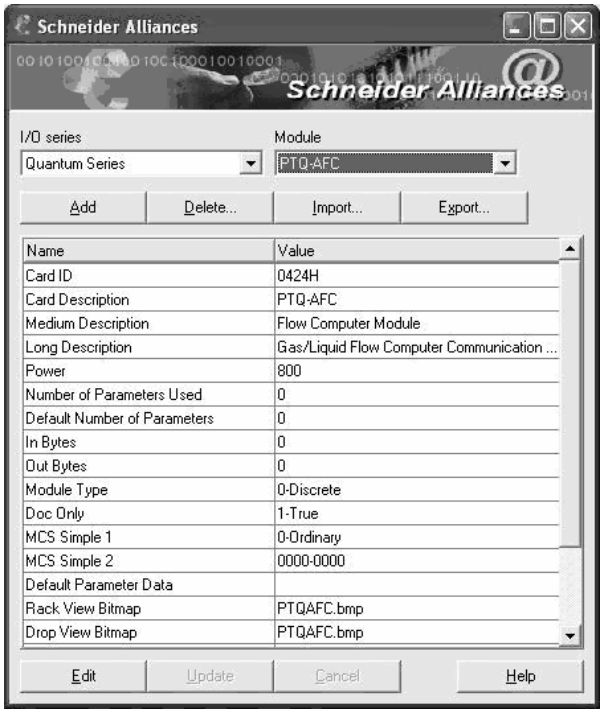
1 Run the Schneider_alliances.exe application that is installed with the Proworx 32 software:

2 Click on Import…

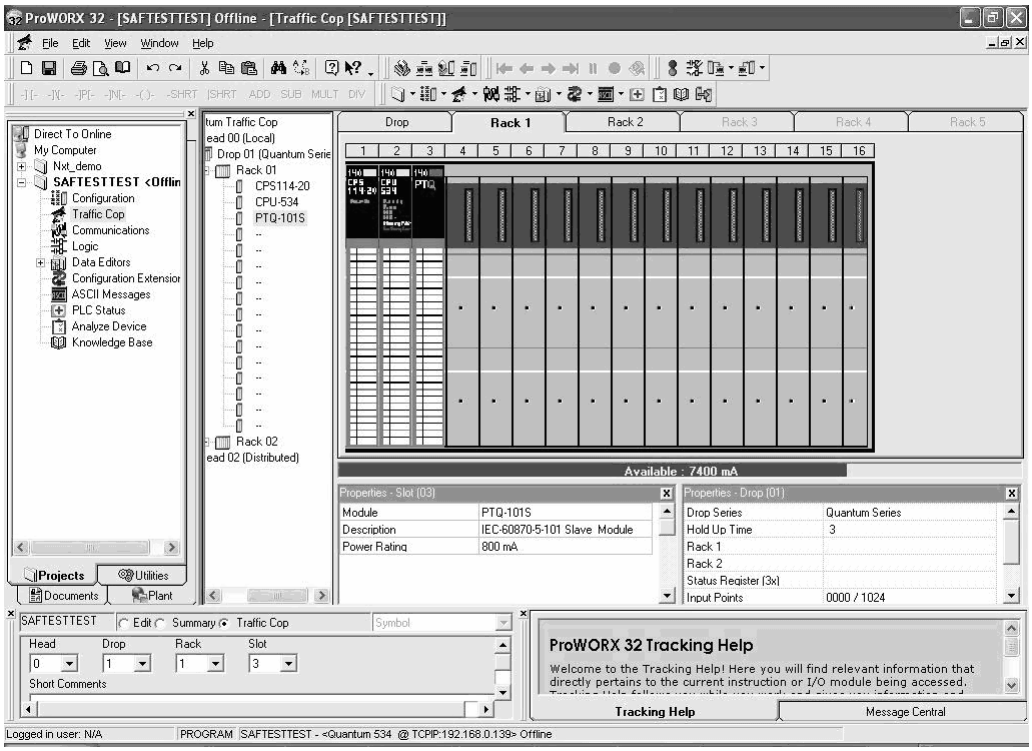
3 Select the .SaF File that is located at the CD-ROM shipped with the PTQ module.



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:



4 Configuring the Processor with UnityPro XL

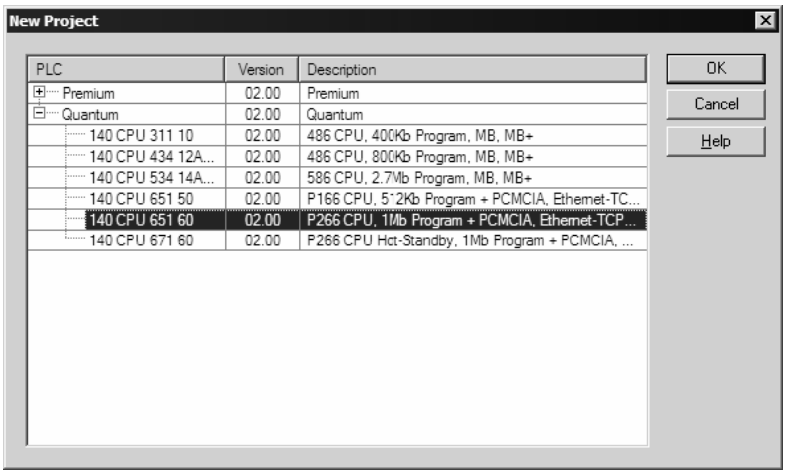
In This Chapter

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 UnityPro XL 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.

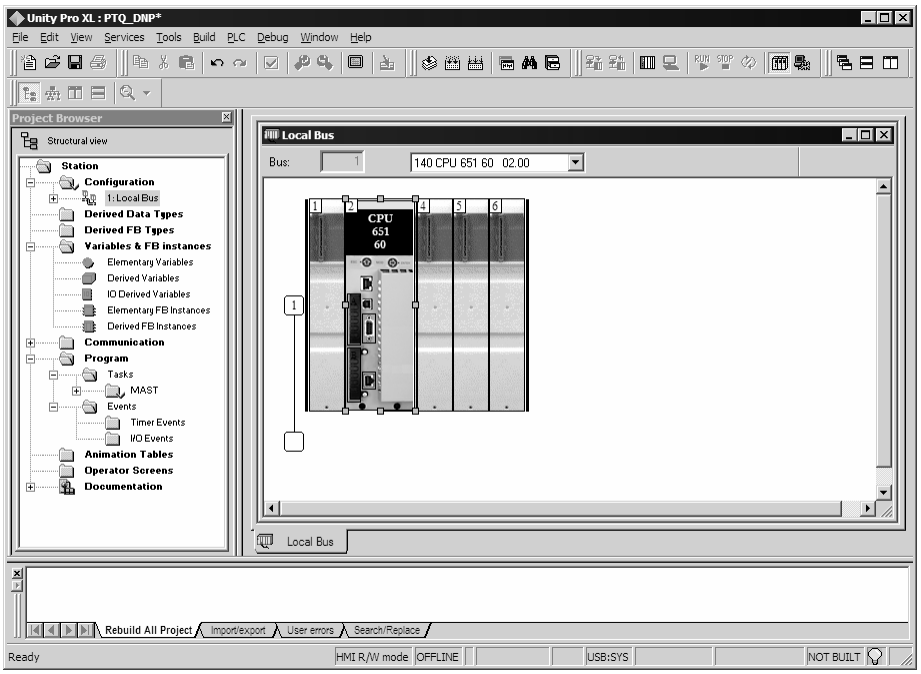
4.1 Create a New Project

The first step is to open UnityPro XL and create a new project.

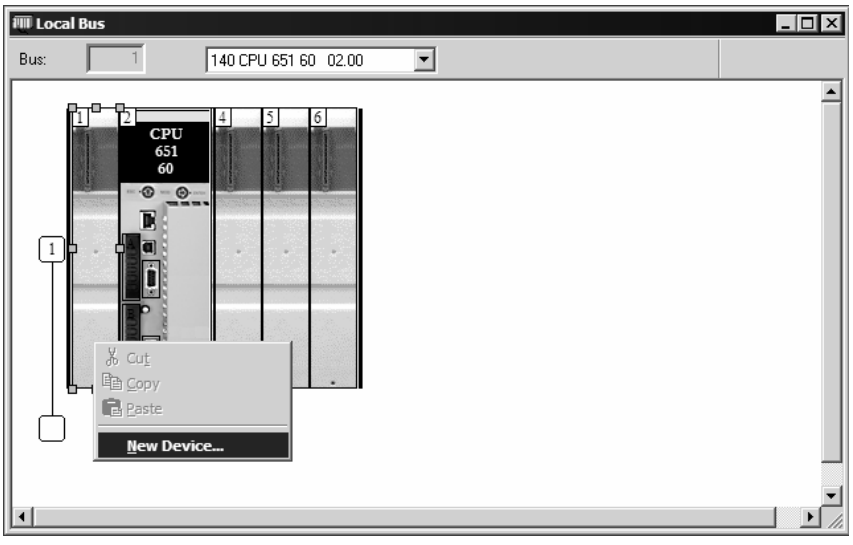
1 In the New Project dialog box, choose the CPU type. In the following illustration, the CPU is 140 CPU 651 60. Choose the processor type that matches your own hardware configuration, if it differs from the example. Click OK to continue.



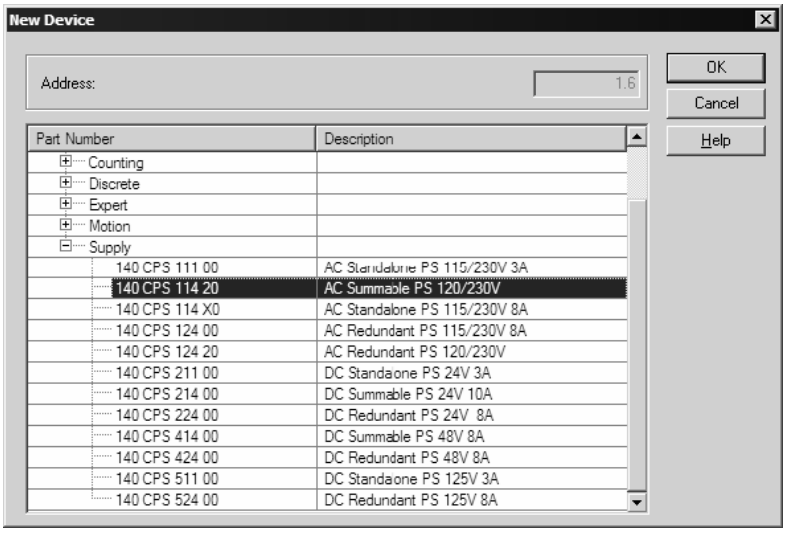
2 Next, add a power supply to the project. In the Project Browser, expand the Configuration folder, and then double-click the 1:LocalBus icon. This action opens a graphical window showing the arrangement of devices in your Quantum rack.



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..



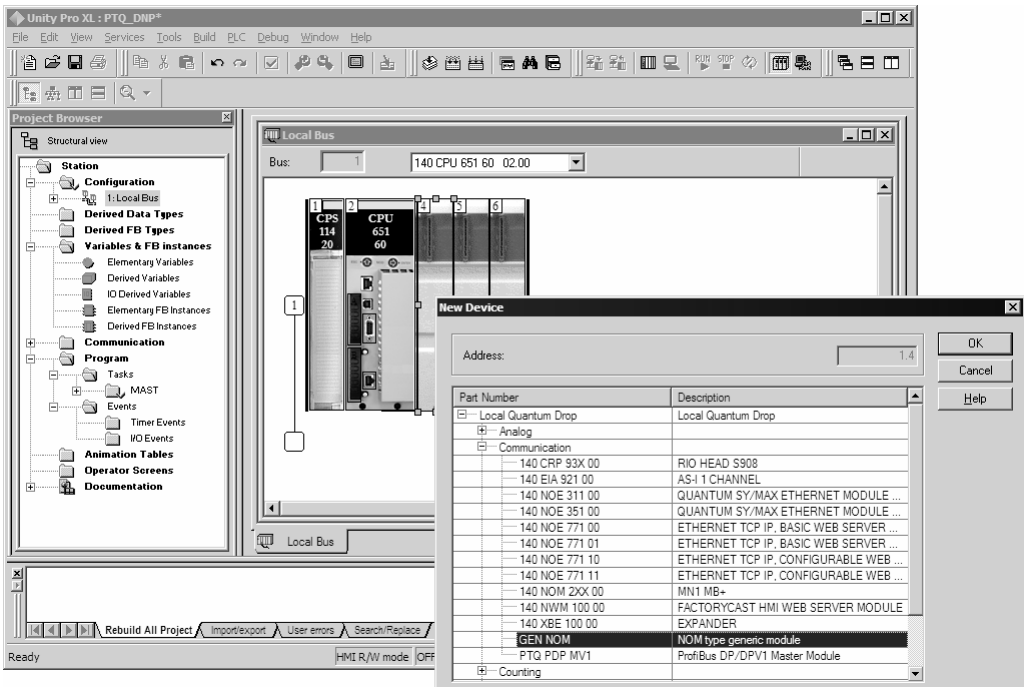
4 Expand the Supply folder, and then select your power supply from the list. Click OK to continue.



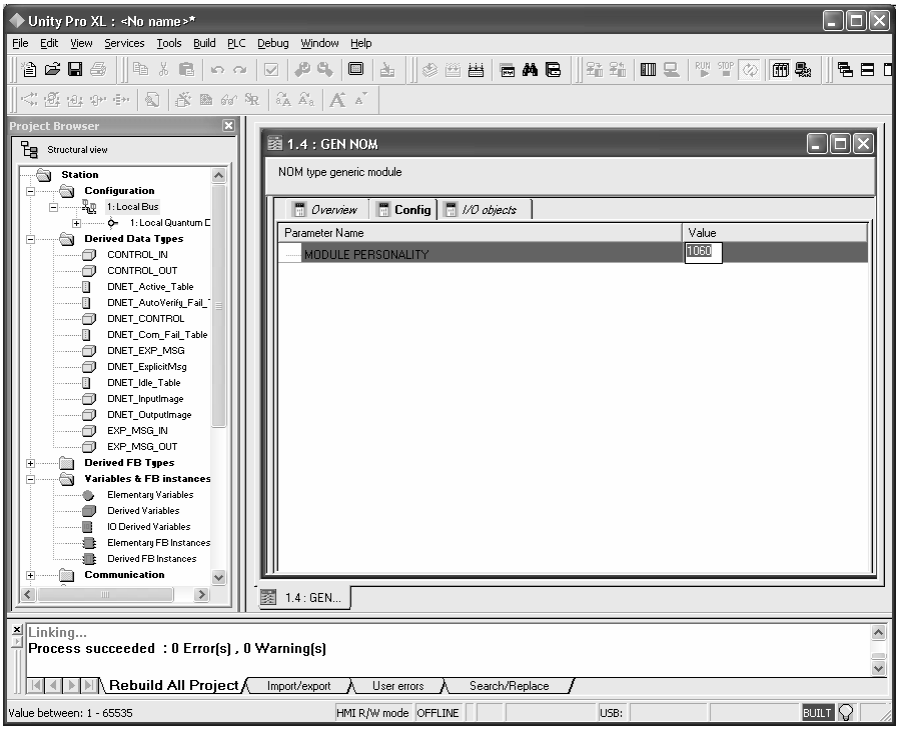
5 Repeat these steps to add any additional devices to your Quantum Rack.

4.2 Add 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 UnityProXL, 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.

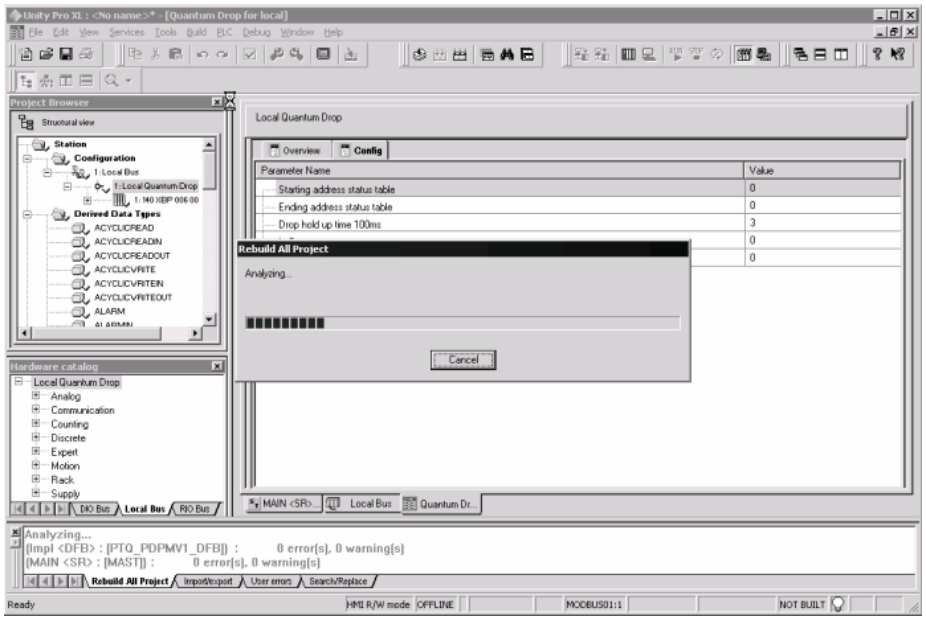
4.3 Build the Project

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 XL. This is not intended to provide detailed information on using Unity Pro XL, or debugging your programs. Refer to the documentation for your processor and for Unity Pro XL 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 XL 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.

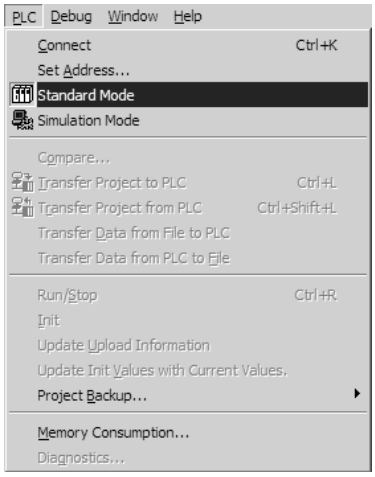
4.4 Connect Your PC to the Processor

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.

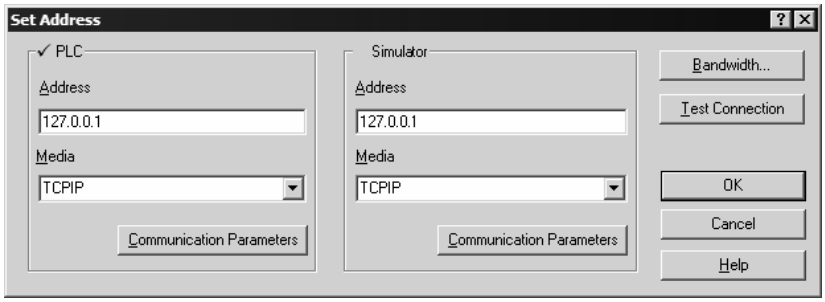
Note: If you have never connected from the PC to your processor before, you must verify that the necessary port drivers are installed and available to UnityPro XL.

To verify address and driver settings in UnityPro XL:

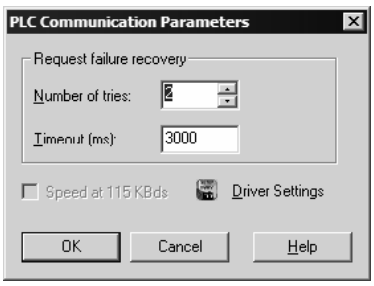
1 Open the PLC menu, and choose Standard Mode. This action turns off the PLC Simulator, and allows you to communicate directly with the Quantum or Unity hardware.



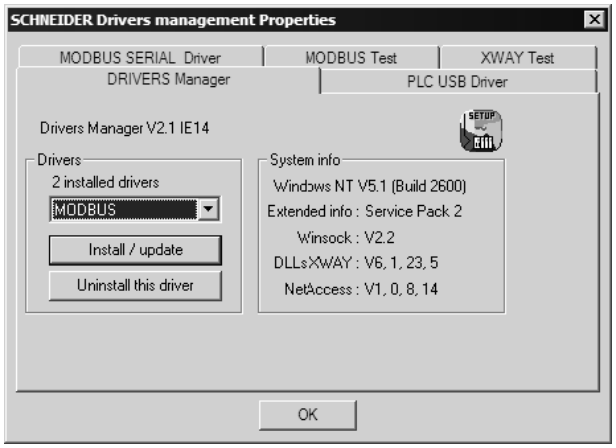
2 Open the PLC menu, and choose Set address... This action opens the Set address dialog box. Open the Media dropdown list and choose the connection type to use (TCPIP or USB).



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 UnityPro XL 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 UnityPro XL.

4.4.1 Connecting to the Processor with TCPIP

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 UnityPro XL.

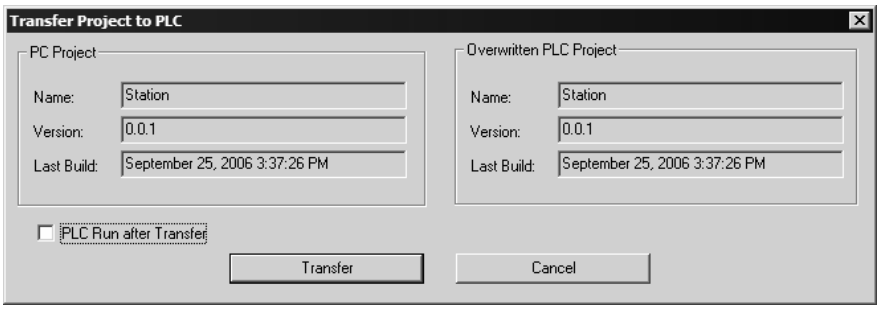
1. If you have not already done so, connect your PC and the processor to an Ethernet hub.
2. Open the PLC menu, and then choose Set address.

- 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 TCPIP.
4. Click the Test Connection button to verify that your settings are correct.

4.5 Download the Project to the Processor

1. Open the PLC menu and then choose Connect. This action opens a connection between the Unity Pro XL 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 XL 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.

5 Setting Up the ProTalk Module

In This Chapter

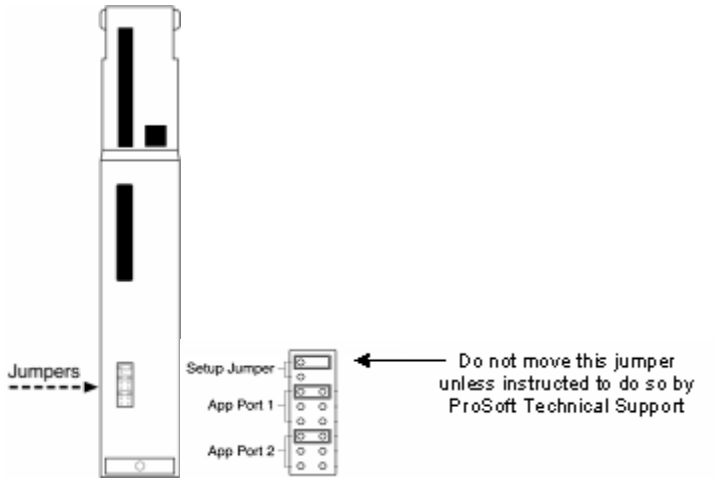
After you complete the following procedures, the ProTalk module will actively be transferring data bi-directionally with the processor.

5.1 Install the ProTalk Module in the Quantum Rack

5.1.1 Verify Jumper Settings

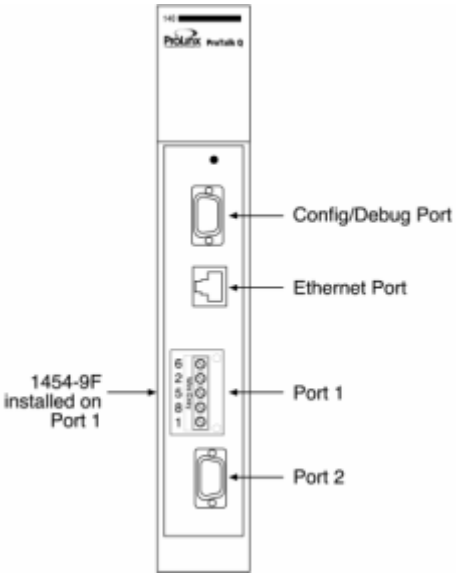
ProTalk modules are configured for RS-232 serial communications by default. To use RS-422 or RS-485, you must change the jumpers.

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



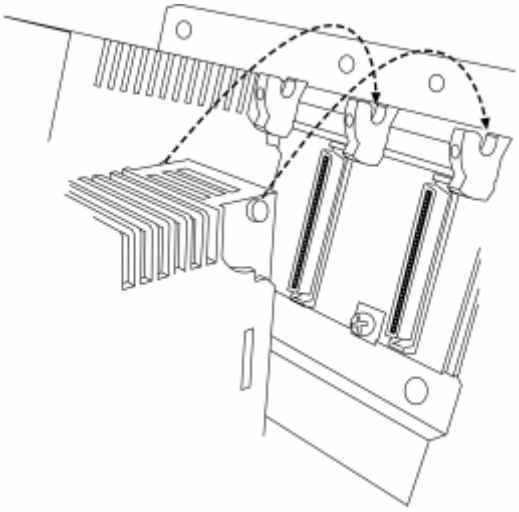
5.1.2 Inserting the 1454-9F connector

Insert the 1454-9F connector as shown. Wiring locations are shown in the table:

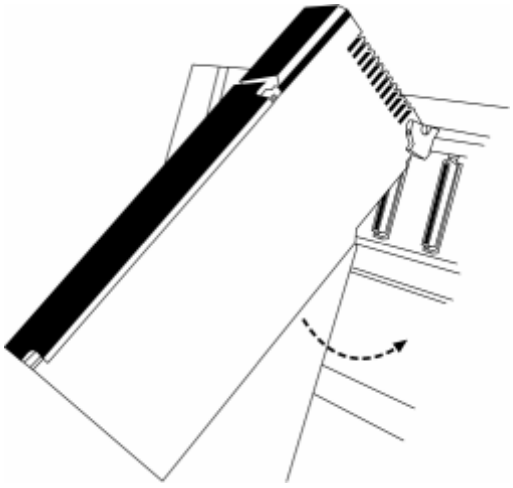


5.1.3 Install the ProTalk Module in the Quantum Rack

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.

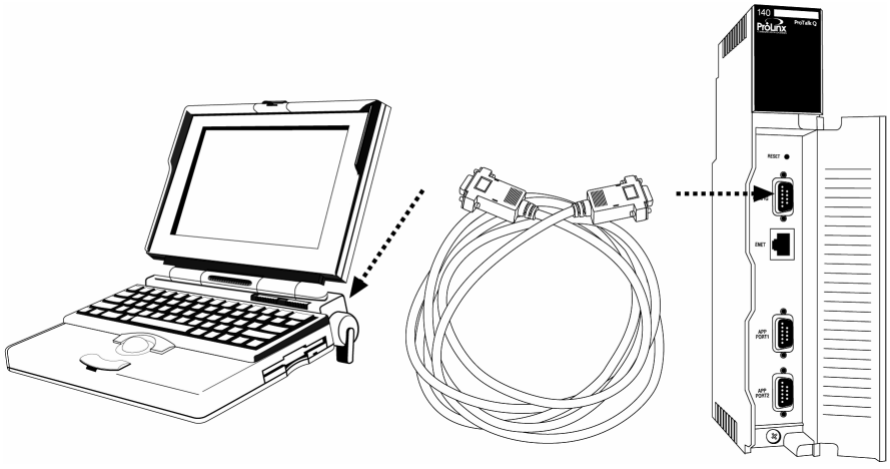


Caution: The PTQ module is hot-swappable, meaning that you can install and remove it while the rack is powered up. You should not assume that this is the case for all types of modules unless the user manual for the product explicitly states that the module is hot-swappable. Failure to observe this precaution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

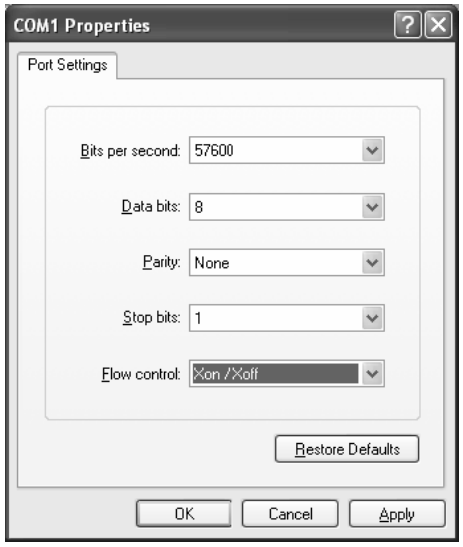
1 Using the supplied Null Modem cable, connect your PC or Laptop to the Configuration/Debug port on the ProTalk module as shown



2 Click the Windows Start button, then choose Programs / accessories / Communications / HyperTerminal.

3 In the HyperTerminal window, enter a connection name, for example **Test,** 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.

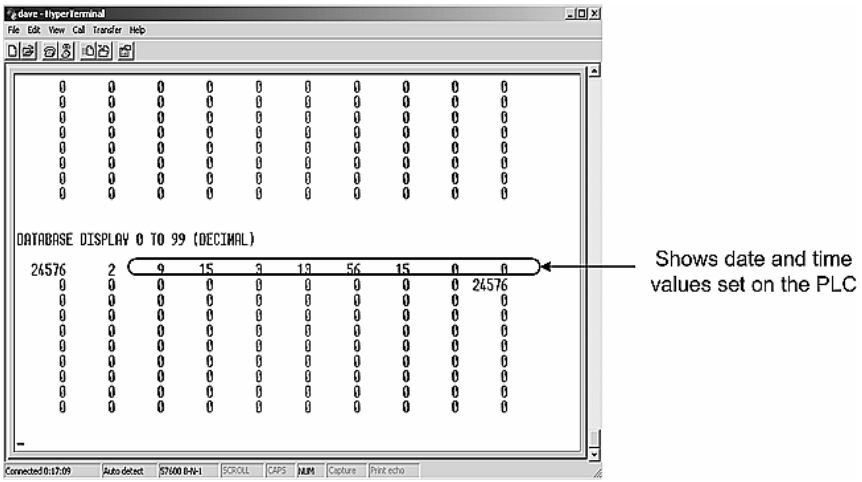
5.3 Verify Communication Between the Processor and the Module

This procedure will verify that the clock values we entered in the processor's data memory (page 21) can be read into the ProTalk module.

1 From the *Configuration/Debug Menu*, type **[D]**, then press **[?]**. This action opens the *Database View Menu*.

```
DATABASE VIEW MENU
 ?=Display Menu
 0-3=Pages 0 to 3000
 S=Show Again
 -=Back 5 Pages
 P=Previous Page
 +=Skip 5 Pages
 N=Next Page
 D=Decimal Display
 H=Hexadecimal Display
 F=Float Display
 A=ASCII Display
 M=Main Menu
```

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



Value	Description
9	Month (September)
15	Day of the Month
3	Year (2003)
13	Hour (13:00 or 1:00 P.M.)
56	Minutes
15	Seconds

In this example, the register values read from the PLC indicate that the date and time returned is September, 15, 2003, 1:56:15 p.m.

3 Type [0] again. The values should be different than 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 Modifying the Configuration File

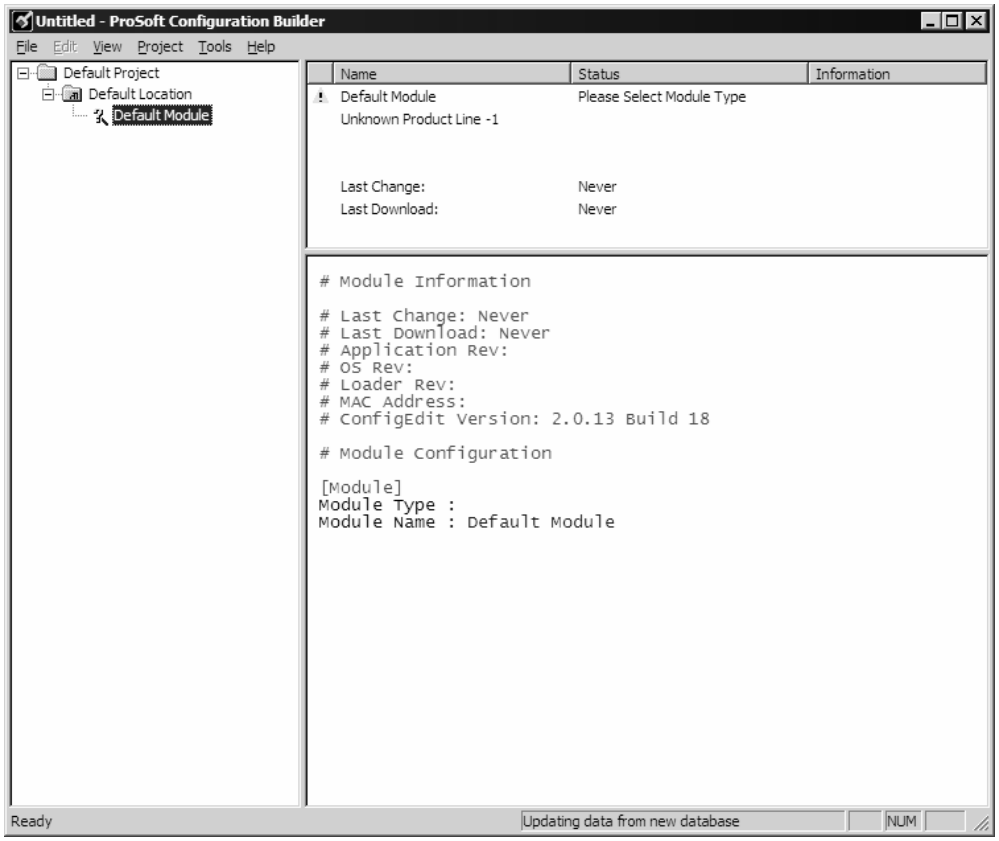
In This Chapter

6.1 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

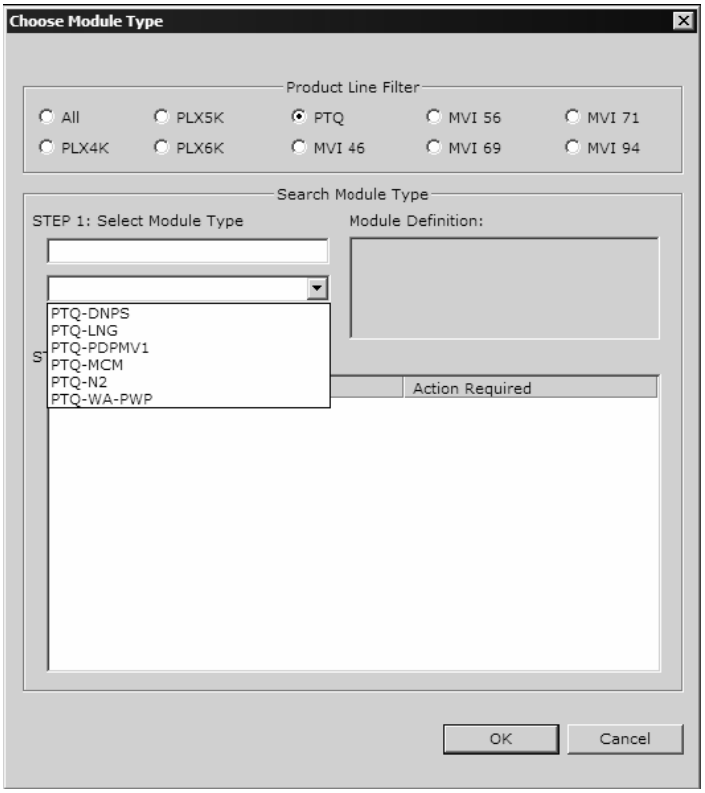
To begin, start ProSoft Configuration Builder. If you have used other Windows configuration tools before, you will find the screen layout familiar. ProSoft Configuration Builder's window consists of a tree view on the left, an information pane and a configuration pane on the right side of the window.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

Adding a Project

To add a project to an existing project file:

1 Select the Default Project icon.
2 Choose Project from the Project menu, then choose Add Project. A new project folder appears.

Adding a Module

To add a module to your project:

1 Double-click the Default Module icon to open the Choose Module Type dialog box.
2 On the Choose Module Type dialog box, select the module type.

Or

1 Open the Project menu and choose Location.
2 On the Location menu, choose Add Module.

To add a module to a different location:

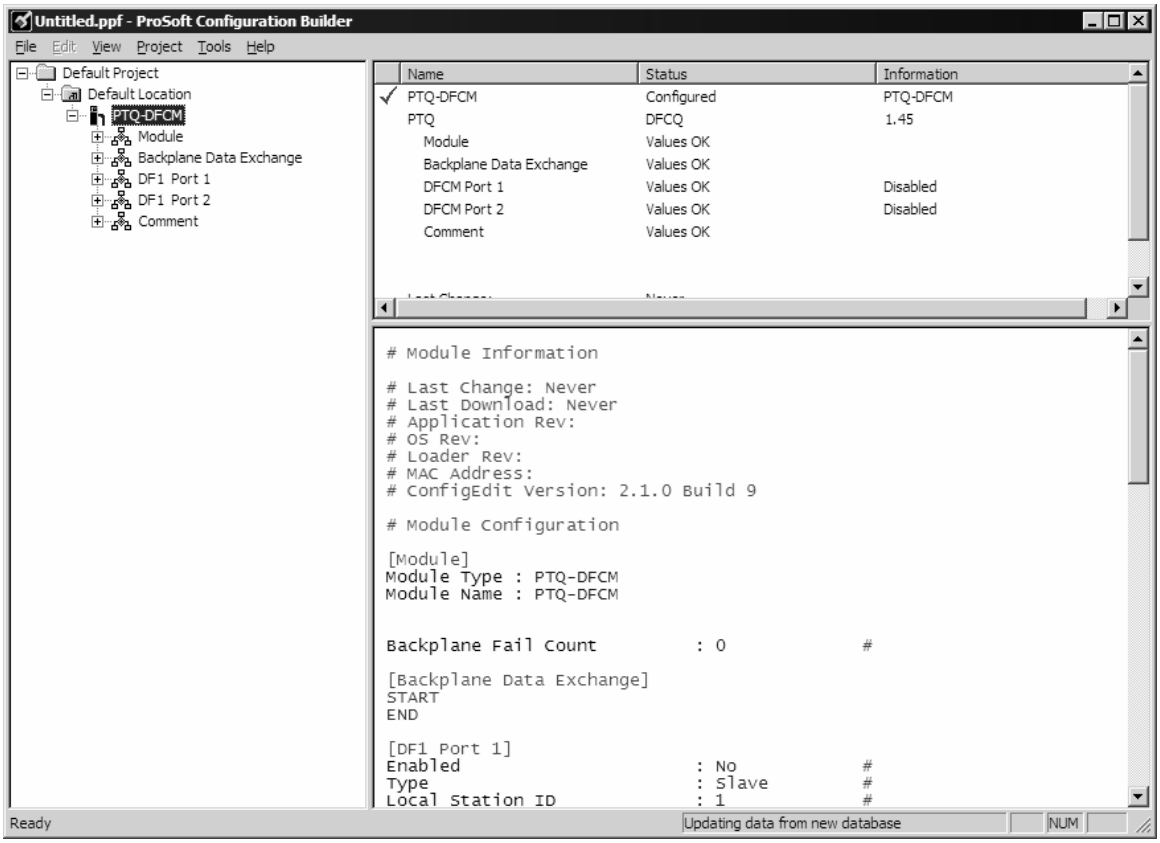
1 Right-click the Location folder and choose Add Module. A new module icon appears.

Or

1 Select the Location icon.
2 From the Project menu, select Location, then select Add Module.

6.1.2 Set Module Parameters

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



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

To rename an object:

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

Module Entries

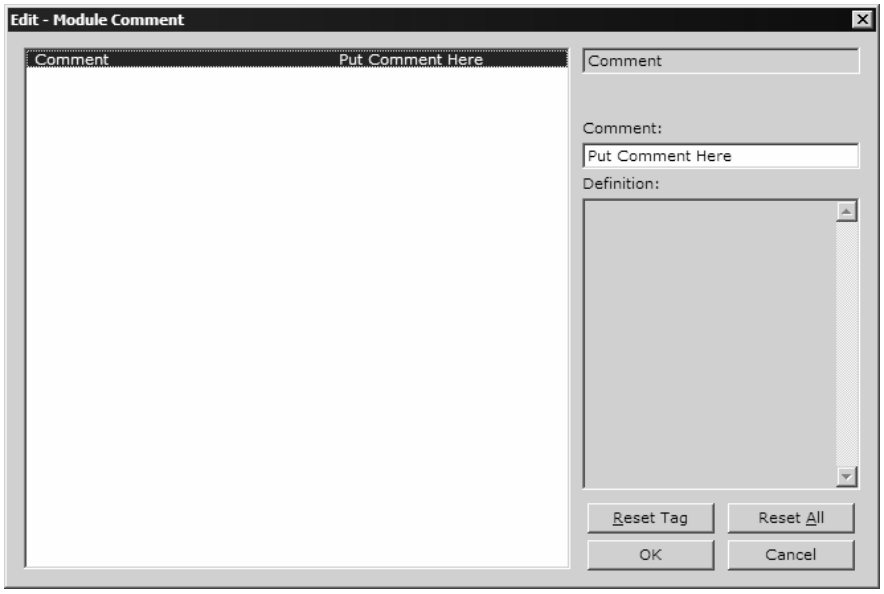
To configure module parameters

1. Click on the plus sign next to the icon Comment to expand module information.
2. Double-click the Module Comment icon to open the Edit dialog box.
3. To edit a parameter, select the parameter in the left pane and make your changes in the right pane.
4. Click OK to save your changes.

Comment Entries

To add comments to your configuration file:

1. Click the plus sign to the left of the Comment icon to expand the Module Comments.
2. Double-click the Module Comment icon. The Edit - Module Comment dialog appears.



3. Enter your comment and click OK to save your changes.

Printing a Configuration File

To print 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. On the View Configuration window, open the File menu, and choose Print. This action opens the Print dialog box.
4. On the Print dialog box, choose the printer to use from the dropdown 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 Quantum / Unity is not in run, and describes how to initialize the module upon startup.

The following example shows a sample [Module] section:

```
[Module]
Module Name: Test Example of DFCM Communication Module
```

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

6.2.1 Module Name

0 to 80 characters

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

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 PRQ (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 Set 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 Set 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 Set 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

Block ID 1000/2000 - Event Port

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

Offset	Description	Length
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

The block IDs 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

The following table shows the block request:

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

The block IDs are used as follows:

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

Block IDs 9940 and 9941

Block IDs 9940 and 9941 are practical to the user when, for example, a SCADA is able to write to the Module only a momentary logical 1 (or logical 0) bit. The Module will read and hold that value until the user's PLC logic sets or clears that bit in the Module's database using block 9940 or 9941. This clearing or setting of the bit by the PLC logic in the Module's database will enable the PLC logic to respond to the SCADA's next setting of the bit.

Block ID 9940 - Clears/Sets bits in the database (mask in bits)

Sending this block will mask the bits in the specific words in the specified locations of the database table in the module.

Clear or Set bits Request (Write Block)

Word Offset in Block	Value	Descriptions
0	9940	Activate the clearing or setting of the bits in the database.
1	0 to 30	Number of words whose bits to set.
2	0 to 9999	Database word offset where write begins.
3	xxxx	Enter value as the AND mask for the 1st word in the database.
4	xxxx	Enter value as the OR mask for the 1st word in the database.
5	xxxx	Enter value as the AND mask for the 2d word in the database.
6	xxxx	Enter value as the OR mask for the 2st word in the database.
.		
.		
63	xxxx	Enter value as the OR mask for the 30th word in the database.

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

Clear or Set bits Response (Read Block)

Word Offset in Block	Value	Descriptions
0	9940	Module's response
1	9940	Module's response

Block ID 9941- Forces new values in database.

Sending this block will force specific words in the specified locations of the database table in the module.

Clear or Set bits Request (Write Block)

Word Offset in Block	Value	Descriptions
0	9941	Activate the clearing or setting of the bits in the database.
1	0 to 60	Number of words to set.
2	0 to 9999	Database word offset where write begins.
3	xxxx	Enter value for the 1st word in the database.
4	xxxx	Enter value for the 2d word in the database.
5	xxxx	Enter value for the 3d word in the database.
.		
.		
63	xxxx	Enter value for the 60th word in the database.

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

Clear or Set bits Response (Read Block)

Word Offset in Block	Value	Descriptions
0	9941	Module's response
1	9941	Module's response

Block ID 9998 or 9999 - Reboot Module

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 [DF1 Port x]

The following shows a sample DF1 Port x section of the configuration file.

```
[DF1 Port x]
Enabled               : Yes    # Y=Use port, N=Do not use port
RS Interface          : 0      # 0=RS-232, 1=RS-485, 2=RS-422
Type                  : Master # M=Master, S=Slave
Local Station ID      : 0      # DF1 node address
Protocol              : F      # F=Full-Duplex, H=Half-Duplex
Termination Type      : CRC    # B=BCC, C=CRC
Baud Rate             : 19200  # Baud rate for Port 110-38400
Parity                : None   # N=None,O=Odd,E=Even,M=Mark,S=Space
Data Bits             : 8      # 5, 6, 7 or 8
Stop Bits             : 1      # 1 or 2
Min Response Delay    : 0      # 0-65535 mSec before sending response msg
RTS On                : 0      # 0-65536 mSec before message
RTS Off               : 1      # 0-65536 mSec after message
Use CTS Line          : No     # Use CTS modem control line (Y/N)
Response Timeout      : 1000   # Response message timeout (0-65535 mSec)
Retry Count           : 3      # Response failure retry count
ENQ Delay             : 10     # 0-65535 mSec before DLE-ENQ sent
Minimum Command Delay : 10     # Minimum number of msec's between commands
Error Delay Counter   : 100    # 0-65535 Command cycle count if error
Command Control Reg   : -1     # Cmd control start DB Reg (-1=ignore)
First File            : 7      # First file number for SLC simulation
File Size             : 200    # Number of elements in each file
File Offset           : 0      # Database offset for first file element
```

Modify each of the parameters as follows based on the needs of your application:

6.4.1 Enabled

This parameter specifies if the port will be used. If the parameter is set to No, the port is disabled. If the parameter is set to Yes, the port is enabled.

6.4.2 Type

```
Type                  : Master #M=Master, S=Slave
```

This parameter defines if the port will emulate a master or a slave device. Enter Master if the port is to emulate a master device or Slave if the port is to emulate a slave device.

6.4.3 Local Station ID

This parameter specifies the local station ID for all DF1 messages sent from this master port. A value of 255 is not permitted as this is the broadcast address. Enter a value in the range of 0 to 254.

6.4.4 Protocol

```
Protocol              : Full   #F=Full-Duplex, H=Half-Duplex
```

This parameter specifies the DF1 protocol to be used on the port. Enter Full for full-duplex communications or Half for half-duplex communications.

6.4.5 Termination Type

```
Termination Type       : CRC    #B=BCC, C=CRC
```

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

6.4.6 Baud Rate

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

6.4.7 Parity

None, Odd, Even

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

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

6.4.8 Data Bits

5, 6, 7 or 8

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

6.4.9 Stop Bits

1 or 2

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

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

6.4.10 Minimum Response Delay

0 to 65535

This parameter sets the number of milliseconds to wait to respond to a request on the port. This is required for slow reacting devices.

6.4.11 RTS On

0 to 65535 milliseconds

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

6.4.12 RTS Off

0 to 65535 milliseconds

This parameter sets the number of milliseconds to delay after the last byte of data is sent before the RTS modem signal will be set low.

6.4.13 Use CTS Line

Yes or No

This parameter specifies if the CTS modem control line is to be used. If the parameter is set to No, the CTS line will not be monitored. If the parameter is set to Yes, the CTS line will be monitored and must be high before the module will send data. Normally, this parameter is required when half-duplex modems are used for communication (2-wire).

6.4.14 Response Timeout

Number of milliseconds to wait for response to command. The value is set depending upon the communication network used and the expected response time of the slowest device on the network. Valid values are 0 to 5000 milliseconds.

6.4.15 Retry Count

0 to 10

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

6.4.16 ENQ Delay

```
ENQ Delay              : 0      #0-65535 milliseconds before DLE-ENQ sent
```

This parameter specifies the number of milliseconds to wait after a DLE-ACK is received from a slave using half-duplex mode before the DLE-ENQ request is made for data. Enter a value in the range of 0 to 65535 milliseconds.

6.4.17 Minimum Command Delay

This parameter specifies the number of milliseconds to wait between issuing each command. This delay value is not applied to retries.

6.4.18 Error Delay Count

```
Error Delay Count      : 100    #0-65535 Command cycle count if error
```

This parameter specifies the number of polls to be skipped on the slave before trying to re-establish communications. After the slave fails to respond, the master will skip commands to be sent to the slave the number of times entered in this parameter. Enter a value in the range of 0 to 65535.

6.4.19 Command Error Pointer

```
Command Error Pointer  : 3000   #Cmd Error list data (-1=ignore)
```

This parameter sets the address in the internal database where the command error data will be placed. If the value is set to -1, the data will not be transferred to the database. Enter a value from 0 to 4999.

6.4.20 Slave List Pointer

```
Slave List Pointer     : 3100   #Slave status list data (-1=ignore)
```

This parameter specifies the starting address in the virtual database where the 256 slave status values will be written. If the parameter is set to -1, the slave data will not be placed in the database. Enter a value in the range of -1 to 4743.

6.4.21 First File

```
First File             : 7      #First file number for SLC simulation
```

This parameter is used when a request for a file is received on the communication port. This field is required when responding to PLC5 and SLC DF1 commands. Use this parameter to define the virtual file(s) to be simulated on the module. Enter a value in the range of 0 to 100.

6.4.22 File Size

Range 1 to 1000

This parameter specifies the size of each file to be simulated on the module. All files simulated are defined to have the same assigned size.

6.4.23 File Offset

This parameter sets the database register location of the first element in the first file simulated in the module. All offsets in the first file and subsequent files will be computed using the address specified. Enter a value in the range of 0 to 4999.

6.5 [DFCM Port x Commands]

This section defines the 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 network.

```
[DF1 Port 1 Commands]
# The file contains examples for a SLC 5/03 processor.
#
# LOCATION     :
# DATE         : 06/24/99
# CONFIGURED BY: RAR
# MODIFIED     :
#   07/23/99 -- Set to read more data file types.
#
START
#      1       2       3      4    5        6    7    8    9   10   11
#        Internal     Poll       Swap     Node Func File File  Elm  Sub
# Enable  Address Interval Count Code  Address Code Type    #    #  Elm
       1       10        0    10    0       10  501    N    7    0
       1     1500        0    10    0       10  502    N    7    0    0
       1       10        0    10    0       10  509    N    7   10
END
```

6.5.1 Command List Overview

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

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

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

The commands take the following parameters:

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

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

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

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

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

The following defines the values and operations:

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

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

Node Address: The address of the device being addressed by the command.

Function Code: These parameters specify the function to be executed by the command. Refer to DF1 Command Set For ProSoft Technology Communication Modules (page 117) for more information on constructing commands for 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 slave device.

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

File	→	Database Register
N7:0	→	0
N8:0	→	200
N9:0	→	400
N10:0	→	600
N11:0	→	800
N12:0	→	1000
N13:0	→	1200
N14:0	→	1400
N15:0	→	1600
N16:0	→	1800
N17:0	→	2000
N18:0	→	2200
N18:0	→	2400
N20:0	→	2600
N21:0	→	2800
N22:0	→	3000
N23:0	→	3200
N24:0	→	3400
N25:0	→	3600
N26:0	→	3800

File	→	Database Register
N27:0	→	4000
N28:0	→	4200
N29:0	→	4400
N30:0	→	4600
N31:0	→	4800
N32:0	→	5000

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

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

Register Range	File Start	File End	Content	Size
0 to 4999	N7:0	N31:199	User Data	5000
5000 to 5009	N32:0	N32:9	Backplane Configuration	10
5010 to 5039	N32:10	N32:39	Port 1 Setup	30
5040 to 5069	N32:40	N32:69	Port 2 Setup	30
5070 to 5199	N32:70	N32:199	Reserved	130
5200 to 6399	N33:0	N38:199	Port 1 Commands	1200
6400 to 7599	N39:0	N44:199	Port 2 Commands	1200
7600 to 7700	N45:0	N45:199	Misc. Status Data	200
7800 to 7999	N46:0	N46:199	Command Control	200
8000 to 9999	N47:0	N56:199	Reserved	2000

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

6.6 [DF1 Port x Override Data File Maps]

```
[DF1 Port 1 OVERRIDE DATA FILE MAPS]
#      DB     File     First     Word
# Address   Number   Element   Length
START
END
```

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

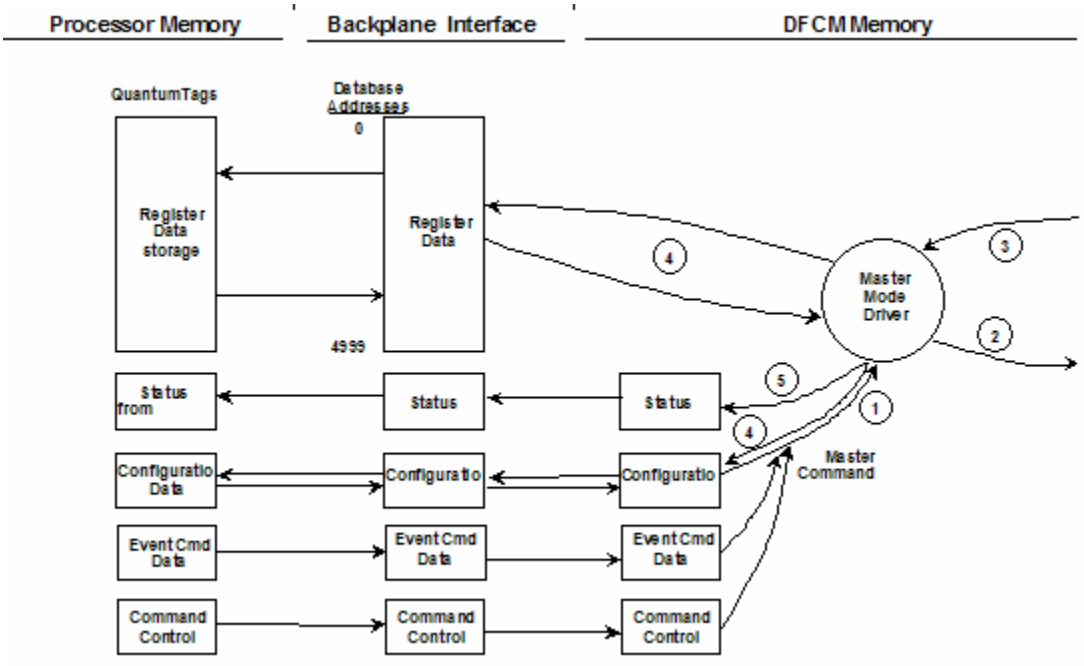
6.6.1 P1 and P2 Override File Mappings

This feature allows a write command to be re-directed to the module's ReadData area. If the DF1 master has a fixed address to write to, the module's WriteData area with the delivered data would be overwritten on the next scan. This feature requires that the port be configured as a slave.

Note: A Slave ID setting of 255 will respond to all commands sent on the network. If this setting is used in a multipoint network, only writes should be used on the network from the DF1 master.

6.7 Master Driver Mode

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



Step	Description
1	The Master driver obtains configuration data from the Quantum processor. The configuration data obtained includes the number of commands and the Master Command List. These values are used by the Master driver to determine the type of commands to be issued to the other nodes on the DF1 network (Refer to the Module Configuration section).

Step	Description
2	After configuration, the Master driver begins transmitting read and/or write commands to the other nodes on the network. If writing data to another node, the data for the write command is obtained from the module's internal database to build the command.
3	Presuming successful processing by the node specified in the command, a response message is received into the Master driver for processing.
4	Data received from the node on the network is passed into the module's internal database, assuming a read command.
5	Status is returned to the Quantum processor for each command in the Master Command List.

Care must be taken in constructing each command in the list for predictable operation of the module. If two commands write to the same internal database address of the module, the results will not be as desired. All commands containing invalid data will be ignored by the module. The following table describes the functions supported by the module and the format of each command:

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

6.8 Slave Driver Mode

The slave driver supports the following DF1 command set:

6.8.1 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
0x06	0x00	Echo Request	X
0x06	0x03	Status Request	X
0x08	N/A	Unprotected Write	X

6.8.2 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	0x26	Read-Modify-Write (Binary Address)	X
0x0F	0x00	Word Range Write (ASCII Address)	
0x0F	0x01	Word Range Read (ASCII Address)	
0x0F	0x26	Read-Modify-Write (ASCII Address)	

6.8.3 SLC-500 Command Set Functions

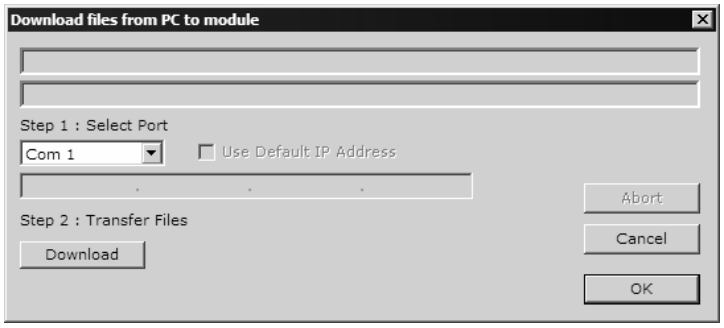
Command	Function	Definition	Supported in Slave
0x0F	0xA1	Protected Typed Logical Read With Two Address Fields	X
0x0F	0XA2	Protected Typed Logical Read With Three Address Fields	X
0x0F	0XA9	Protected Typed Logical Write With Two Address Fields	X
0x0F	0XAA	Protected Typed Logical Write With Three Address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

6.9 Download the Project to the Module

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

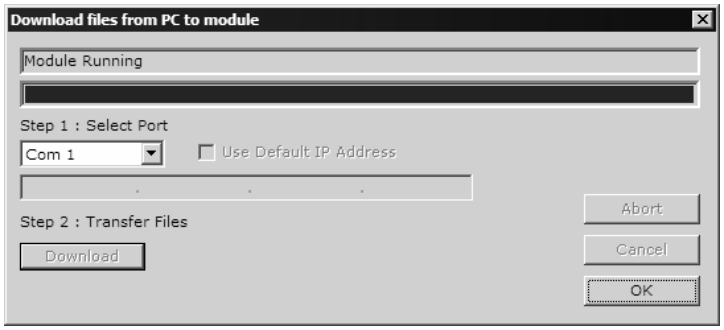
To Download the Project File

1. In the tree view in ProSoft Configuration Builder, click once to select the PTQ-DFCM 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 following 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 ProSoft Configuration Builder will be updated with the message *"Module Running*".



7 Diagnostics and Troubleshooting

In This Chapter

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

- Status data values are transferred from the module to the processor.
- Data contained in the module can be viewed through the Configuration/Debug port attached to a terminal emulator.
- LED status indicators on the front of the module provide information on the module's status.

7.1 Reading Status Data from the Module

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

7.1.1 Required Hardware

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

ProSoft Technology recommends the following minimum hardware to connect your computer to the module:

- 80486 based processor (Pentium preferred)
- 1 megabyte of memory
- At least one UART hardware-based serial communications port available. USB-based virtual UART systems (USB to serial port adapters) often do not function reliably, especially during binary file transfers, such as when uploading/downloading configuration files or module firmware upgrades.
- A null modem serial cable.

7.1.2 The Configuration/Debug Menu

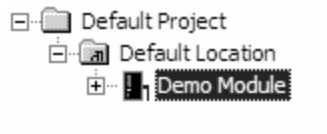
The Configuration and Debug menu for this module is arranged as a tree structure, with the Main Menu at the top of the tree, and one or more sub-menus for each menu command. The first menu you see when you connect to the module is the Main menu.

Because this is a text-based menu system, you enter commands by typing the command letter from your computer keyboard in 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.

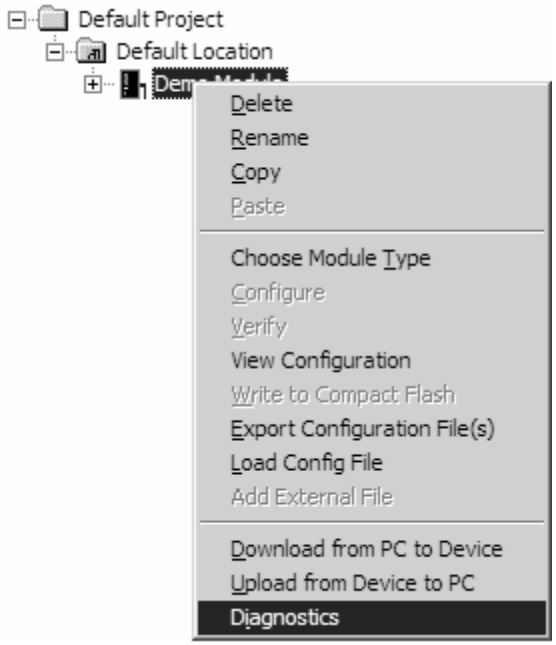
Using the Diagnostic Window in ProSoft Configuration Builder

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

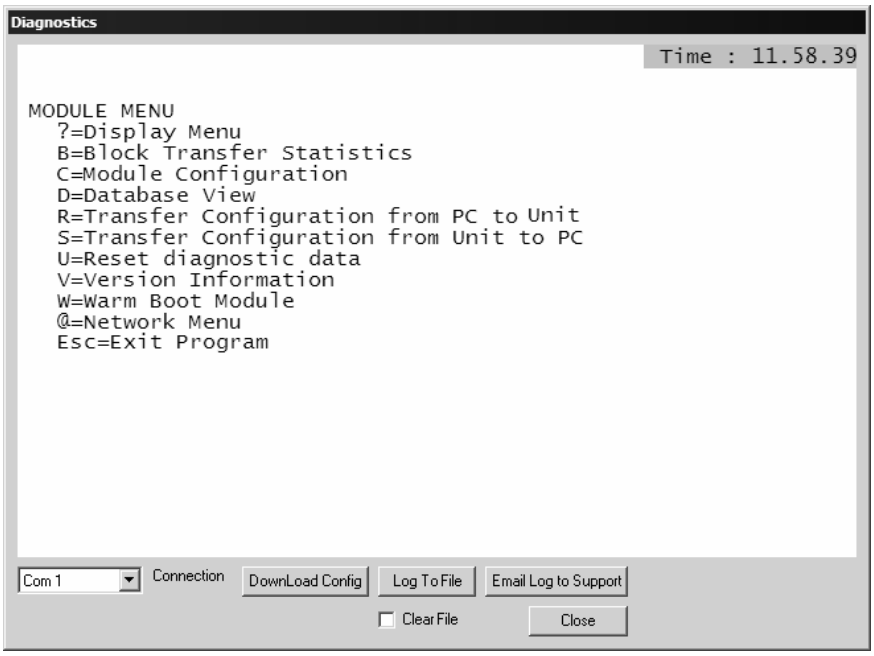
1 Start PCB program with the application file to be tested. Right click over the module icon.



2 On the shortcut menu, choose Diagnostics.



3 This action opens the Diagnostics dialog box. Press "?" to display the Main Menu.



Important: The illustrations of configuration/debug menus in this section are intended as a general guide, and may not exactly match the configuration/debug menus in your own module.

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

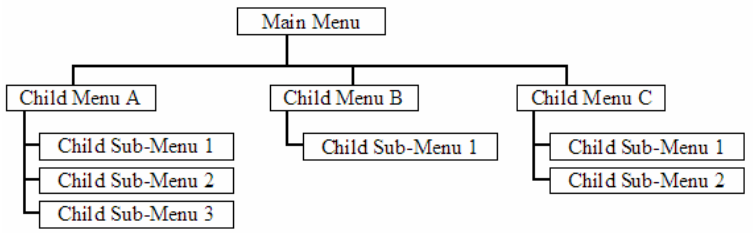
1 Verify that the null modem cable is connected properly between your computer's serial port and the module. A regular serial cable will not work.
2 On computers with more than one serial port, verify that your communication program is connected to the same port that is connected to the module.

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

Navigation

All of the sub-menus for this module contain commands to redisplay the menu or return to the previous menu. You can always return from a sub-menu to the next higher menu by pressing **[M]** on your keyboard.

The organization of the menu structure is represented in simplified form in the following illustration:



The remainder of this section shows you the menus available for this module, and briefly discusses the commands available to you.

Keystrokes

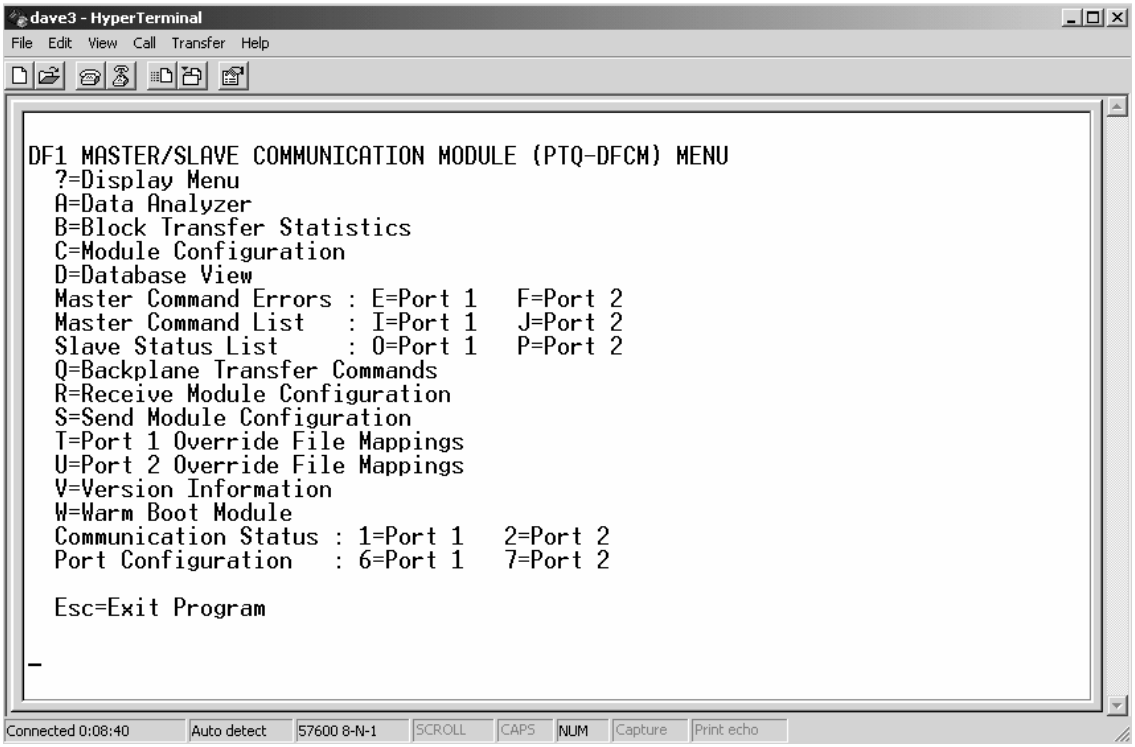
The keyboard commands on these menus are almost always non-case sensitive. You can enter most commands in lower case or capital letters.

The menus use a few special characters (**[?]**, **[-]**, **[+]**, **[@]**) that must be entered exactly as shown. Some of these characters will require you to use the **[Shift]**, **[Ctrl]** or **[Alt]** keys to enter them correctly. For example, on US English keyboards, enter the **[?]** command as **[Shift][/]**.

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

7.1.3 Main Menu

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



Caution: Some of the commands available to you from this menu are designed for advanced debugging and system testing only, and can cause the module to stop communicating with the processor or with other devices, resulting in potential data loss or other failures. Only use these commands if you are specifically directed to do so by ProSoft Technology Technical Support staff. Some of these command keys are not listed on the menu, but are active nevertheless. Please be careful when pressing keys so that you do not accidentally execute an unwanted command.

Opening the Data Analyzer Menu

Press **[A]** to open the Data Analyzer Menu. Use this command to view all bytes of data transferred on each port. Both the transmitted and received data bytes are displayed. Refer to Data Analyzer for more information about this menu.

Important: When in analyzer mode, program execution will slow down. Only use this tool during a troubleshooting session. Before disconnecting from the Config/Debug port, please press [S] to stop the data analyzer, and then press [M] to return to the main menu. This action will allow the module to resume its normal high speed operating mode.

Viewing Block Transfer Statistics

Press **[B]** from the Main Menu to view the Block Transfer Statistics screen.

Use this command to display the configuration and statistics of the backplane data transfer operations between the module and the processor. The information on this screen can help determine if there are communication problems between the processor and the module.

Tip: To determine the number of blocks transferred each second, mark the numbers displayed at a specific time. Then some seconds later activate the command again. Subtract the previous numbers from the current numbers and divide by the quantity of seconds passed between the two readings.

Viewing Module Configuration

Press **[C]** to view the Module Configuration screen.

Use this command to display the current configuration and statistics for the module.

Opening the Database Menu

Press **[D]** to open the Database View menu. Use this menu command to view the current contents of the module's database.

Opening the Command Error List Menu

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

Opening the Command List Menu

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

Viewing the Slave Status List (Port 1 and 2)

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

0 = slave is not used,

1 = slave being actively polled,

2 = slave suspended and

3 = slave disabled.

Viewing the Backplane Command List

Press **[Q]** 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 PC to PTQ module

Press **[R]** to send (upload) the configuration file from your PC to the module and store the file on the module's Compact Flash Disk.

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 PTQ module to PC

Press **[S]** to receive (download) 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 downloaded, you can open and edit the file to change the module's configuration.

Viewing Override File Mappings

Press **[T]** or **[U]** for ports 1 and 2, respectively, to view override file mappings.

```
DF1 OVERRIDE FILE MAP LIST FOR PORT 0, (0 TO 9 OF 0)

 DB ADDR   FILE#   ELEMENT   LENGTH
       0       0         0        0
       0       0         0        0
       0       0         0        0
       0       0         0        0
       0       0         0        0
       0       0         0        0
       0       0         0        0
       0       0         0        0
       0       0         0        0
       0       0         0        0
```

Viewing Version Information

Press **[V]** to view Version information for the module.

Use this command to view the current version of the software for the module, as well as other important values. You may be asked to provide this information when calling for technical support on the product.

Values at the bottom of the display are important in determining module operation. The Program Scan Counter value is incremented each time a module's program cycle is complete.

Tip: Repeat this command at one-second intervals to determine the frequency of program execution.

Warm Booting the Module

Caution: Some of the commands available to you from this menu are designed for advanced debugging and system testing only, and can cause the module to stop communicating with the processor or with other devices, resulting in potential data loss or other failures. Only use these commands if you are specifically directed to do so by ProSoft Technology Technical Support staff. Some of these command keys are not listed on the menu, but are active nevertheless. Please be careful when pressing keys so that you do not accidentally execute an unwanted command.

Press **[W]** from the Main Menu to warm boot (restart) the module. This command will cause the program to exit and reload, refreshing configuration parameters that must be set on program initialization. Only use this command if you must force the module to re-boot.

Viewing Port Communication Status

Press **[1]** or **[2]** from the Main Menu to view the port communication status for Ports 1 and 2.

Use this command to view communication status and statistics for the selected port. This information can be informative when troubleshooting communication problems.

Viewing Port Configuration

Press **[6]** or **[7]** from the Main Menu to view configuration information for ports 1 and 2.

Use this command to display detailed configuration information for the selected port.

Exiting the Program

Caution: Some of the commands available to you from this menu are designed for advanced debugging and system testing only, and can cause the module to stop communicating with the processor or with other devices, resulting in potential data loss or other failures. Only use these commands if you are specifically directed to do so by ProSoft Technology Technical Support staff. Some of these command keys are not listed on the menu, but are active nevertheless. Please be careful when pressing keys so that you do not accidentally execute an unwanted command.

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

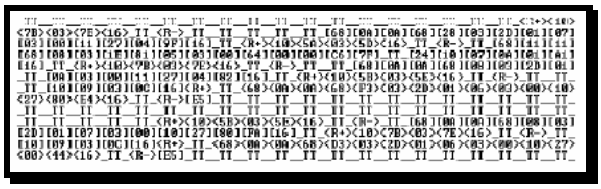
7.1.4 Data Analyzer

The data analyzer mode allows you to view all bytes of data transferred on each port. Both the transmitted and received data bytes are displayed. Use of this feature is limited without a thorough understanding of the protocol.

Note: The Port selection commands on the Data Analyzer menu differs very slightly in different modules, but the functionality is basically the same. Use the illustration above as a general guide only. Refer to the actual data analyzer menu on your module for the specific port commands to use.
Important: When in analyzer mode, program execution will slow down. Only use this tool during a troubleshooting session. Before disconnecting from the Config/Debug port, please press [S] to stop the data analyzer, and then press [M] to return to the main menu. This action will allow the module to resume its normal high speed operating mode.

Analyzing Data for the first application port

Press **[1]** to display I/O data for the first application port in the Data Analyzer. The following illustration shows an example of the Data Analyzer output.



Analyzing Data for the second application port

Press **[2]** to display I/O data for the second application port in the Data Analyzer.

Displaying Timing Marks in the Data Analyzer

You can display timing marks for a variety of intervals in the data analyzer screen. These timing marks can help you determine communication-timing characteristics.

Key	Interval
[5]	1 milliseconds ticks
[6]	5 milliseconds ticks
[7]	10 milliseconds ticks
[8]	50 milliseconds ticks
[9]	100 milliseconds ticks
[0]	Turn off timing marks

Removing Timing Marks in the Data Analyzer

Press **[0]** to turn off timing marks in the Data Analyzer screen.

Viewing Data in Hexadecimal Format

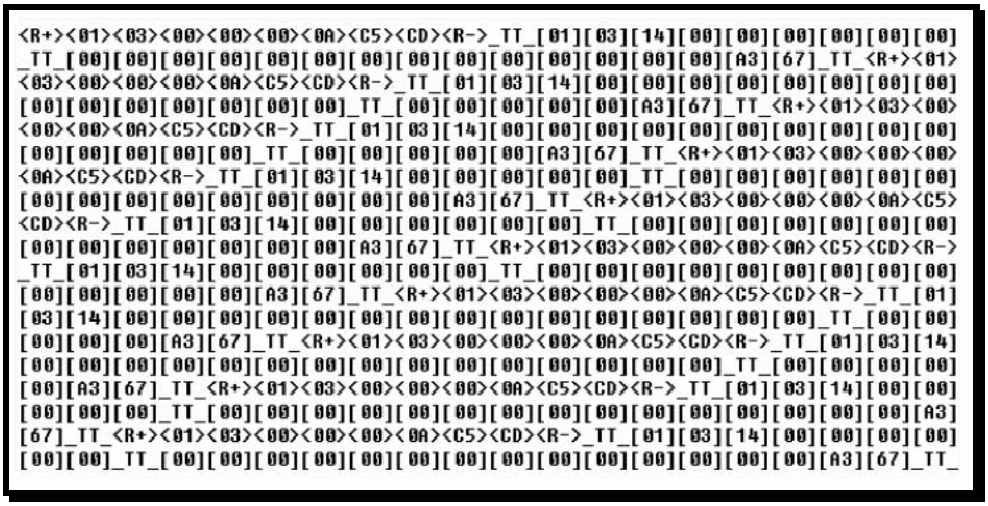
Press **[H]** to display the data on the current page in hexadecimal format.

Viewing Data in ASCII (Text) Format

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

Starting the Data Analyzer

Press **[B]** to start the data analyzer. After the key is pressed, all data transmitted and received on the currently selected port will be displayed. The following illustration shows an example.



The Data Analyzer displays the following special characters:

Character	Definition
[ ]	Data enclosed in these characters represent data received on the port.
< >	Data enclosed in these characters represent data transmitted on the port.
<R+>	These characters are inserted when the RTS line is driven high on the port.
<R->	These characters are inserted when the RTS line is dropped low on the port.
<CS>	These characters are displayed when the CTS line is recognized high.
_TT_	These characters are displayed when the timing mark interval has been reached. This parameter is user defined.

Stopping the Data Analyzer

Press **[S]** to stop the data analyzer. Use this option to freeze the display so the data can be analyzed. To restart the analyzer, press **[B]**.

Important: When in analyzer mode, program execution will slow down. Only use this tool during a troubleshooting session. Before disconnecting from the Config/Debug port, please press [S] to stop the data analyzer, and then press [M] to return to the main menu. This action will allow the module to resume its normal high speed operating mode.

Returning to the Main Menu

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

7.1.5 Data Analyzer Tips

From the main menu, press **[A]** for the "Data Analyzer". You should see the following text appear on the screen:

```
Data Analyzer Mode Selected
```

After the "Data Analyzer" mode has been selected, press **[?]** to view the Data Analyzer menu. You will see the following menu:

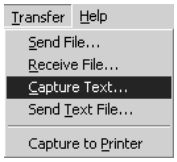
```
DATA ANALYZER VIEW MENU
 ?=Display Menu
 1=Select Port 1
 2=Select Port 2
 5=1 mSec Ticks
 6=5 mSec Ticks
 7=10 mSec Ticks
 8=50 mSec Ticks
 9=100 mSec Ticks
 0=No mSec Ticks
 H=Hex Format
 A=ASCII Format
 B=Start
 S=Stop
 M=Main Menu

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

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

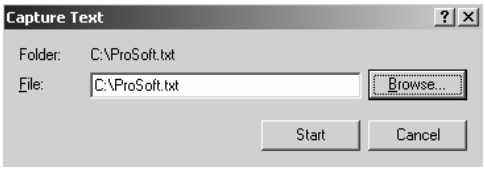
For most applications, HEX is the best format to view the data, and this does include ASCII based messages (because some characters will not display on HyperTerminal and by capturing the data in HEX, we can figure out what the corresponding ASCII characters are supposed to be).

The Tick value is a timing mark. The module will print a _TT for every xx milliseconds of no data on the line. Usually 10milliseconds is the best value to start with.

After you have selected the Port, Format, and Tick, we are now ready to start a capture of this data. The easiest way to do so is to go up to the top of you HyperTerminal window, and do a **Transfer / Capture Text** as shown below:



After selecting the above option, the following window will appear:



Next name the file, and select a directory to store the file in. In this example, we are creating a file ProSoft.txt and storing this file on our root C: drive. After you have done this, press the Start button.

Now you have everything that shows up on the HyperTerminal screen being logged to a file called ProSoft.txt. This is the file that you will then be able to email to ProSoft Technical Support to assist with issues on the communications network.

To begin the display of the communications data, you will then want to press 'B' to tell the module to start printing the communications traffic out on the debug port of the module. After you have pressed 'B', you should see something like the following:

```
[03][00][04][00][05][00][06][00][07][00][08][00][09][FB][B7]_TT__TT_<R+><01><02>
<00><00><00><0A><F8><0D><R->_TT__TT__TT_[01][02][02][00][00][B9][B8]_TT__TT_<R+>
<01><03><00><00><00><0A><C5><CD><R->_TT__TT_[01][03][14][00][00][00][01][00]_TT_
[02][00][03][00][04][00][05][00][06][00][07][00][08][00][09][CD][51]_TT__TT_<R+>
<01><01><00><00><00><A0><3C><72><R->_TT__TT_[01][01][14][00][00][01][00][02]_TT_
[00][03][00][04][00][05][00][06][00][07][00][08][00][09][00][B7][52]_TT__TT_<R+>
<01><04><00><00><00><0A><70><0D><R->_TT__TT_[01][04][14][00][00][00][01][00]_TT_
[02][00][03][00][04][00][05][00][06][00][07][00][08][00][09][FB][B7]_TT__TT_<R+>
<01><02><00><00><00><0A><F8><0D><R->_TT__TT__TT_[01][02][02][00][00][B9][B8]_TT_
_TT_<R+><01><03><00><00><00><0A><C5><CD><R->_TT__TT_[01][03][14][00][00][00][01]
[00]_TT_[02][00][03][00][04][00][05][00][06][00][07][00][08][00][09][CD][51]_TT_
_TT_<R+><01><01><00><00><00><A0><3C><72><R->_TT__TT__TT_[01][01][14][00][00][01]
[00][02]_TT_[00][03][00][04][00][05][00][06][00][07][00][08][00][09][00][B7][52]
_TT__TT_<R+><01><04><00><00><00><0A><70><0D><R->_TT__TT_[01][04][14][00][00][00]
[01][00]_TT_[02][00][03][00][04][00][05][00][06][00][07][00][08][00][09][FB][B7]
_TT__TT_<R+><01><02><00><00><00><0A><F8><0D><R->_TT__TT_[01][02][02][00][00][B9]
[B8]_TT__TT_<R+><01><03><00><00><00><0A><C5><CD><R->_TT__TT_[01][03][14][00][00]
[00][01][00]_TT_[02][00][03][00][04][00][05][00][06][00][07][00][08][00][09][CD]
[51]_TT__TT_<R+><01><01><00><00><00><A0><3C><72><R->_TT__TT__TT_[01][01][14][00]
[00][01][00][02]_TT_[00][03][00][04][00][05][00][06][00][07][00][08][00][09][00]
[B7][52]_TT__TT_<R+><01><04><00><00><00><0A><70><0D><R->_TT__TT_[01][04][14][00]
[00][00][01][00]_TT_[02][00][03][00][04][00][05][00][06][00][07][00][08][00][09]
[FB][B7]_TT__TT_<R+><01><02><00><00><00><0A><F8><0D><R->_TT__TT__TT_[01][02][02]
[00][00][B9][B8]_TT__TT_<R+><01><03><00><00><00><0A><C5><CD><R->_TT__TT_
```

The <R+> means that the module is transitioning the communications line to a transmit state.

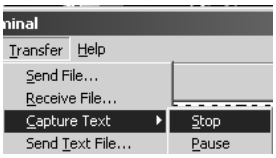
All characters shown in <> brackets are characters being sent out by the module.

The <R-> shows when the module is done transmitting data, and is now ready to receive information back.

And finally, all characters shown in the [ ] brackets is information being received from another device by the module.

After taking a minute or two of traffic capture, you will now want to stop the "Data Analyzer". To do so, press the 'S' key, and you will then see the scrolling of the data stop.

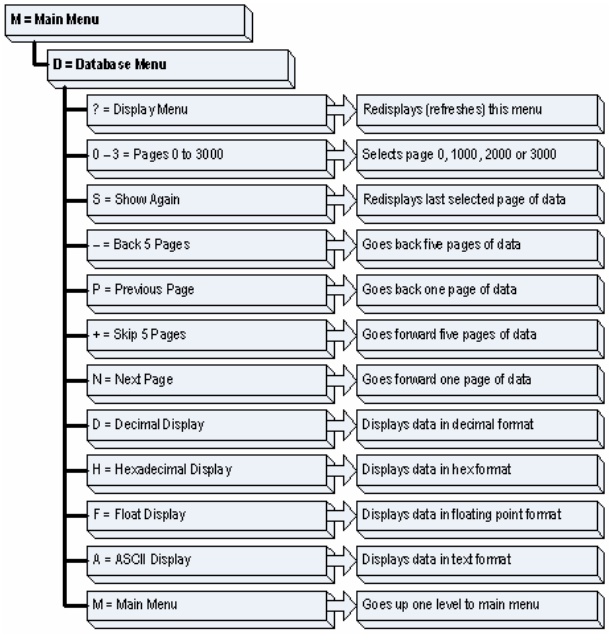
When you have captured the data you want to save, open the Transfer menu and choose Capture Text. On the secondary menu, choose Stop.



You have now captured, and saved the file to your PC. This file can now be used in analyzing the communications traffic on the line, and assist in determining communication errors.

7.1.6 Database View Menu

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



Viewing Register Pages

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

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

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

Displaying the Current Page of Registers Again

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

This screen displays the current page of 100 registers in the database.

Moving Back Through 5 Pages of Registers

Press **[-]** from the Database View menu to skip back to the previous 500 registers of data.

Viewing the Previous 100 Registers of Data

Press **[P]** from the Database View menu to display the previous 100 registers of data.

Skipping 500 Registers of Data

Hold down **[Shift]** and press **[=]** to skip forward to the next 500 registers of data.

Viewing the Next 100 Registers of Data

Press **[N]** from the Database View menu to select and display the next 100 registers of data.

Viewing Data in Decimal Format

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

Viewing Data in Hexadecimal Format

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

Viewing Data in Floating Point Format

Press **[F]** from the Database View menu. Use this command to display the data on the current page in floating point format. The program assumes that the values are aligned on even register boundaries. If floating-point values are not aligned as such, they are not displayed properly.

Viewing Data in ASCII (Text) Format

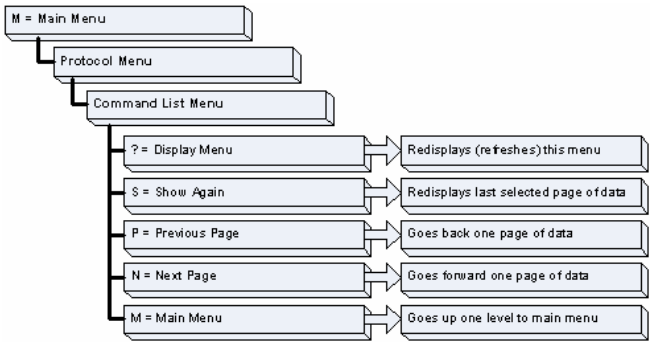
Press **[A]** to display the data on the current page in ASCII format. This is useful for regions of the database that contain ASCII data.

Returning to the Main Menu

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

7.1.7 Master Command Error List Menu

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



Redisplaying the Current Page

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

Viewing the Previous 20 Commands

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

Viewing the Previous Page of Commands

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

Viewing the Next 20 Commands

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

Viewing the Next Page of Commands

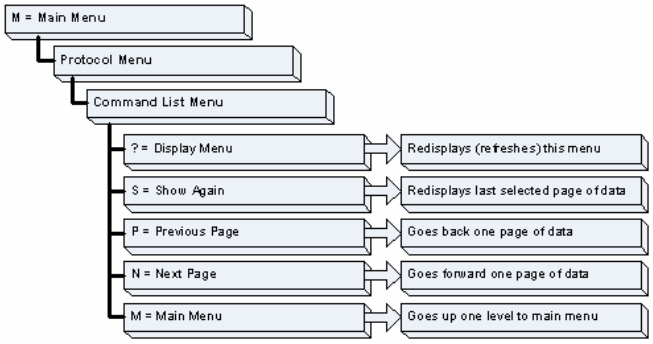
Press **[N]** to display the next page of commands.

Returning to the Main Menu

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

7.1.8 Command List Menu

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



Redisplaying the Menu

Press **[?]** to display the current menu. Use this command when you are looking at a screen of data, and want to view the menu choices available to you.

Redisplaying the Current Page

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

Use this command to display the current page of commands. Ten commands are displayed on each page.

If an enabled command has an error, the EN field will contain a value of -1. This indicates that the command will be re-issued every 30 seconds.

Viewing the Previous Page of Commands

Press **[P]** to display the previous 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 LED Status Indicators

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

ProLinx 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.
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 is communicating
		Off	Port is not communicating
PRT2	Green	On	Port is communicating
		Off	Port is not communicating
ERR1	Red	Off	The PTQ-DFCM is working normally.
		On	The PTQ-DFCM module program has recognized an application error. This LED will also be turned on if any command presents an error.
ERR2	Red	Off	The PTQ-DFCM is working normally.
		On	The PTQ-DFCM module program has recognized an application error. This LED will also be turned on if any command presents an error.

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

7.2.1 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 the Reference chapter for status data tables.

8 Reference

In This Chapter

8.1 Product Specifications

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

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

The PTQ-DFCM module is a powerful communication interface for Quantum platform processors. Developed under license from Schneider Electric, the module incorporates proprietary backplane technology that enables powerful data access to the Quantum processor.

8.1.1 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.

- Compatible with common Quantum / Unity programming tools.
- Quantum data types supported: 0x, 1x, 3x, 4x
- High speed data transfer across backplane provides quick data update times.

8.1.2 Hardware Specifications

Specification	Value
Backplane Current Load	800 mA @ 5 V
Operating Temperature	0 to 60°C (32 to 140°F)
Storage Temperature	-40 to 85°C (-40 to 185°F)
Relative Humidity	5% to 95% (non-condensing)
Vibration	Sine vibration 4-100 Hz in each of the 3 orthogonal axes
Shock	30G, 11 mSec. in each of the 3 orthogonal axes
LED Indicators	Module Status Backplane Transfer Status Serial Port Activity Serial Activity and Error Status
Configuration Serial Port (Debug)	DB-9M PC Compatible RS-232 only No hardware handshaking
Application Serial Ports (PRT1, PRT2)	DB-9M PC Compatible RS-232/422/485 jumper selectable RS-422/485 screw termination included RS-232 handshaking configurable 500V Optical isolation from backplane
Certifications	cULus, ATEX, CE

8.1.3 Functional Specifications

DF1 ports

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

DF1 Master Protocol Specifications

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

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

DF1 Slave Protocol Specifications

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

Tested Hardware Connections

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

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

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 54, page 98). 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
…		…

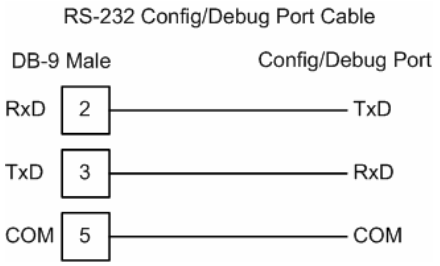
8.3 Cable Connections

The application ports on the PTQ-DFCM module support RS-232, RS-422, and RS-485 interfaces. Please inspect the module to ensure that the jumpers are set correctly to correspond with the type of interface you are using.

Note: When using RS-232 with radio modem applications, some radios or modems require hardware handshaking (control and monitoring of modem signal lines). Enable this in the configuration of the module by setting the UseCTS parameter to 1.

8.3.1 RS-232 Configuration/Debug Port

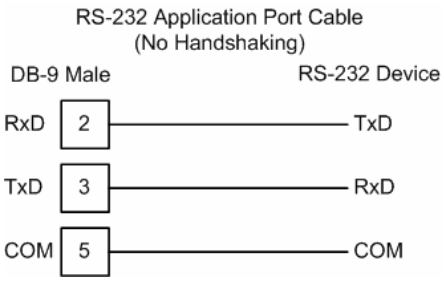
This port is physically a DB-9 connection. This port permits a PC based terminal emulation program to view configuration and status data in the module and to control the module. The cable for communications on this port is shown in the following diagram:



The Ethernet port on this module (if present) is inactive.

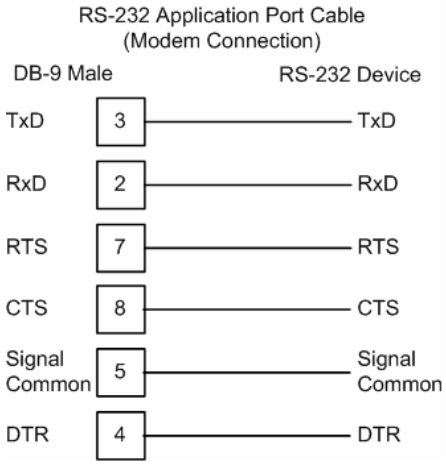
8.3.2 RS-232

When the RS-232 interface is selected, the use of hardware handshaking (control and monitoring of modem signal lines) is user definable. If no hardware handshaking will be used, the cable to connect to the port is as shown below:



RS-232: Modem Connection

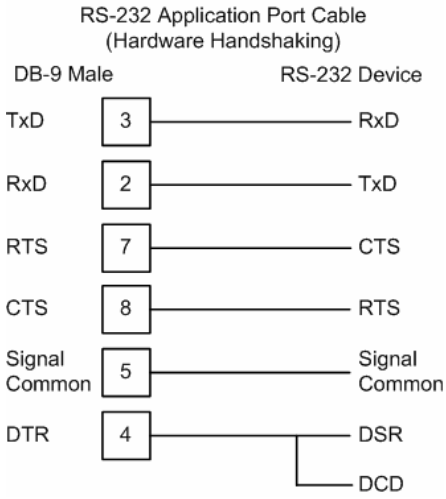
This type of connection is required between the module and a modem or other communication device.



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

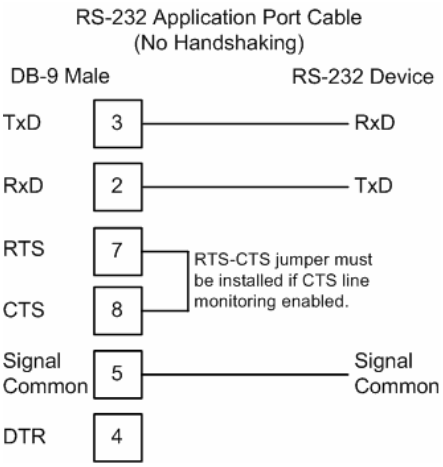
RS-232: Null Modem Connection (Hardware Handshaking)

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



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

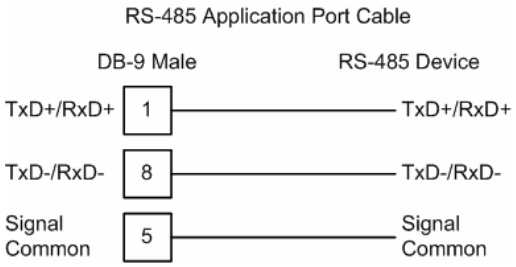
This type of connection can be used to connect the module to a computer or field device communication port.



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

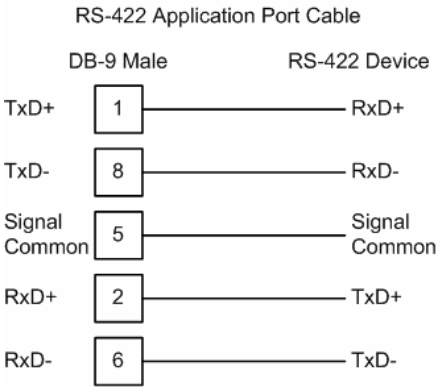
8.3.3 RS-485

The RS-485 interface requires a single two or three wire cable. The Common connection is optional and dependent on the RS-485 network. The cable required for this interface is shown below:



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

8.3.4 RS-422



RS-485 and RS-422 Tip

If communication in the RS-422/RS-485 mode does not work at first, despite all attempts, try switching termination polarities. Some manufacturers interpret +/- and A/B polarities differently.

8.4 PTQ-DFCM Database Definition

The database is used as follows:

Register Range	Content	Size
0 to 4999	User Data	5000
5000 to 5009	Backplane Configuration	10
5010 to 5039	Port 1 Setup	30
5040 to 5069	Port 2 Setup	30
5070 to 5199	Reserved	130
5200 to 6399	Port 1 Commands	1200
6400 to 7599	Port 2 Commands	1200
7600 to 7632	Misc. Status Data	33
7800 to 7999	Reserved	200
8000 to 8199	Port 1 Override File Maps	200
8200 to 8399	Port 2 Override File Maps	200
8400 to 8499	Port 1 Command Errors	100
8500 to 8599	Port 2 Command Errors	100
8600 to 8855	Port 1 Slave Status	256
8856 to 8899	Reserved	
8900 to 9155	Port 2 Slave Status	
9156 to 9999	Reserved	

8.5 PTQ-DFCM Status Data

The status information starts at address 7600 in the PTQ-DFCM database. The status information is divided as follows:

Register Range	Content	Size
7600 to 7632	Misc. Status Data	33
8400 to 8499	Port 1 Command Errors	100
8500 to 8599	Port 2 Command Errors	100
8600 to 8855	Port 1 Slave Status	256
8900 to 9155	Port 2 Slave Status	156

The status information can be copied into the Quantum database using a backplane command type 2.

8.6 Miscellaneous Status

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

Register	Content	Description
7619	Port 2 Requests	This field contains the total number of messages sent out the port.
7620	Port 2 Responses	This field contains the total number of messages received on the port.
7621	Port 2 Errors Sent	This field contains the total number of message errors sent out the port.
7622	Port 2 Errors Received	This field contains the total number of message errors received on the port.
7623	Read Block Count	This field contains the total number of read blocks transferred from the module to the processor.
7624	Write Block Count	This field contains the total number of write blocks transferred from the processor to the module.
7625	Parse Block Count	This field contains the total number of blocks successfully parsed that were received from the processor.
7626	Command Event Block Count	This field contains the total number of command event blocks received from the processor.
7627	Command Block Count	This field contains the total number of command blocks received from the processor.
7628	Error Block Count	This field contains the total number of block errors recognized by the module.
7629	Port 1 Current Error/Index	For a slave port, this field contains the value of the current error code returned. For a master port, this field contains the index of the currently executing command.
7630	Port 1 Last Error/Index	For a slave port, this field contains the value of the last error code returned. For a master port, this field contains the index of the command with an error.
7631	Port 2 Current Error/Index	For a slave port, this field contains the value of the current error code returned. For a master port, this field contains the index of the currently executing command.
7632	Port 2 Last Error/Index	For a slave port, this field contains the value of the last error code returned. For a master port, this field contains the index of the command with an error.

8.7 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:

8.7.1 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

8.7.2 Remote STS Error Codes

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

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

8.7.3 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

8.7.4 Module Specific Error (not DFCM Compliant)

Code (Int)	Code (Hex)	Description
-1	0xFFFF	CTS modem control line not set before transmit
-2	0xFFFE	Timeout while transmitting message
-10	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

8.8 Configuration Data

The configuration values in the configuration file are copied to the PTQ-DFCM database. Refer to the registers described in this section to read the configuration data at any time.

IMPORTANT: You cannot configure the ProTalk module by simply modifying these parameters. In order to accomplish this, you must refer to the configuration file as described in the manual.

Register Range	Content	Size
5000 to 5009	Backplane Configuration	10
5010 to 5039	Port 1 Setup	30
5040 to 5069	Port 2 Setup	30
5070 to 5199	Reserved	130
5200 to 6399	Port 1 Commands	1200
6400 to 7599	Port 2 Commands	1200
7800 to 7999	Reserved	200
8000 to 8199	Port 1 Override File Maps	200
8200 to 8399	Port 2 Override File Maps	200

8.8.1 Backplane Setup

Register	Content	Description
5000	Reserved	Reserved
5001	Reserved	Reserved
5002	Reserved	Reserved
5003	Reserved	Reserved
5004	Backplane Fail	This parameter specifies the number of successive transfer errors that must occur before the communication ports are shut down. If the parameter is set to 0, the communication ports will continue to operate under all conditions. If the value is set larger than 0 (1 to 65535), communications will cease if the specified number of failures occur.

Register	Content	Description
5005	Error Status Pointer	This parameter specifies the register location in the module's database where module status data will be stored. If a value less than 0 is entered, the data will not be stored in the database. If the value specified is in the range of 0 to 4940, the data will be placed in the user data area.
5006 to 5009	Spare	

8.8.2 Port 1 Setup

Register	Content	Description
5010	Enable	This parameter is used to define if this port will be utilized. If the parameter is set to 0, the port is disabled. A value of 1 will enable the port.
5011	Type	This parameter defines if the port will emulate a master or slave device. Enter 0 to emulate a master device and 1 to emulate a slave device.
5012	Local Station ID	This parameter specifies the local station ID for all DF1 messages sent from this master port. A value of 255 is not permitted as this is the broadcast address. Enter a value in the range of 0 to 254.
5013	Protocol	0=full duplex, 1=half-duplex
5014	Termination Type	This parameter specifies the error checking for all DF1 messages. 0=BCC, 1=CRC
5015	Baud Rate	This is the baud rate to be used on the port. Enter the baud rate as a value. For example, to select 19K baud, enter 8.
5016	Parity	This is the Parity code to be used for the port. The coded values are as follows: 0=None, 1=Odd, 2=Even.
5017	Data Bits	This parameter sets the number of data bits for each word used by the protocol. Enter a value in the range of 5 to 8.
5018	Stop Bits	This parameter sets the number of stop bits to be used with each data value sent. Enter a value of 1 or 2.

Baud Rate	Parameter Value
110	1
300	2
600	3
1200	4
2400	5
4800	6
9600	7
19,200	8
38,400	9

Register	Content	Description
5019	Minimum Response Delay	This parameter sets the number of milliseconds to wait before a response message is sent out of the port. This parameter is required when interfacing to a slow responding device. Enter a value in the range of 0 to 65535.
5020	RTS On Delay	This parameter sets the number of milliseconds to delay after RTS is asserted before the data will be transmitted. Enter a value in the range of 0 to 65535.
5021	RTS Off Delay	This parameter sets the number of milliseconds to delay after the last byte of data is sent before the RTS modem signal will be set low. Enter a value in the range of 0 to 65535.
5022	Use CTS Line	This parameter specifies if the CTS modem control line is to be used. If the parameter is set to 0, the CTS line will not be monitored. If the parameter is set to 1, the CTS line will be monitored and must be high before the module will send data. Normally, this parameter is required when half-duplex modems are used for communication (2-wire).
5023	ENQ Delay	This parameter specifies the number of milliseconds to wait after a DLE-ACK is received from a slave using half-duplex mode before the DLE-ENQ request is made for data. Enter a value in the range of 0 to 65535.
5024	Command Count	This parameter specifies the number of commands to be processed for the port. Enter a value of 0 to 100.
5025	Minimum Command Delay	This parameter specifies the number of milliseconds to wait between the initial issuance of a command. This parameter can be used to delay all commands sent to slaves to avoid "flooding" commands on the network. This parameter does not affect retries of a command as they will be issued when failure is recognized. Enter a value in the range of 0 to 65535.
5026	Command Error Pointer	This parameter sets the address in the internal Modbus database where the command error data will be placed. If the value is set to -1, the data will not be transferred to the database. Enter a value of 0 to 4999.
5027	Response Timeout	This parameter represents the message response timeout period in 1-ms increments. This is the time that a port configured as a master will wait before re-transmitting a command if no response is received from the addressed slave. The value is set depending upon the communication network used and the expected response time of the slowest device on the network.
5028	Retry Count	This parameter specifies the number of times a command will be retried if it fails. Enter a value in the range of 0 to 10.
5029	Error Delay Count	This parameter specifies the number of polls to be skipped on the slave before trying to re-establish communications. After the slave fails to respond, the master will skip commands to be sent to the slave the number of times entered in this parameter. Enter a value in the range of 0 to 65535.

Register	Content	Description
5030	Slave List Pointer	This parameter specifies the starting address in the virtual database where the 256 slave status values will be written. If the parameter is set to -1, the slave data will not be placed in the database. Enter a value in the range of -1 to 4743.
5031	Slave List Frequency	Not used in this version of the software
5032	First File	This parameter is used when a request for a file is received on the communication port. This field is required when responding to PLC5 and SLC DF1 commands. Use this parameter to define the virtual file(s) to be simulated on the module. This value is fixed at 7 for N7.
5033	File Size	This parameter is used to specify the size of each file to be simulated on the module. All files simulated are defined to have the same assigned size. This value is fixed at 200.
5034	File Offset	This parameter sets the database register location of the first element in the first file simulated in the module. All offsets in the first file and subsequent files will be computed using the address specified. This value is fixed at 0. Therefore, the module assumes N7:0 starts at the database offset of 0 and each file is 200 words in size.
5035	Map Count	Number of file map overrides.
5036 to 5039	Spare	

8.8.3 Port 2 Setup

Register	Content	Description
5040	Enable	This parameter is used to define if this port will be utilized. If the parameter is set to 0, the port is disabled. A value of 1 will enable the port.
5041	Type	This parameter defines if the port will emulate a master or slave device. Enter 0 to emulate a master device and 1 to emulate a slave device.
5042	Local Station ID	This parameter specifies the local station ID for all DF1 messages sent from this master port. A value of 255 is not permitted as this is the broadcast address. Enter a value in the range of 0 to 254.
5043	Protocol	0=full duplex, 1=half-duplex
5044	Termination Type	This parameter specifies the error checking for all DF1 messages. 0=BCC, 1=CRC

<table>
<tr><th>Register</th><th>Content</th><th>Description</th></tr>
<tr><td>5045</td><td>Baud Rate</td><td>This is the baud rate to be used on the port. Enter the baud rate as a value. For example, to select 19K baud, enter 8.
<table>
<tr><th>Baud Rate</th><th>Parameter Value</th></tr>
<tr><td>110</td><td>1</td></tr>
<tr><td>300</td><td>2</td></tr>
<tr><td>600</td><td>3</td></tr>
<tr><td>1200</td><td>4</td></tr>
<tr><td>2400</td><td>5</td></tr>
<tr><td>4800</td><td>6</td></tr>
<tr><td>9600</td><td>7</td></tr>
<tr><td>19,200</td><td>8</td></tr>
<tr><td>38,400</td><td>9</td></tr>
</table></td></tr>
<tr><td>5046</td><td>Parity</td><td>This is the Parity code to be used for the port. The coded values are as follows: 0=None, 1=Odd, 2=Even.</td></tr>
<tr><td>5047</td><td>Data Bits</td><td>This parameter sets the number of data bits for each word used by the protocol. Enter a value in the range of 5 to 8.</td></tr>
<tr><td>5048</td><td>Stop Bits</td><td>This parameter sets the number of stop bits to be used with each data value sent. Enter a value of 1 or 2.</td></tr>
<tr><td>5049</td><td>Minimum Response Delay</td><td>This parameter sets the number of milliseconds to wait before a response message is sent out of the port. This parameter is required when interfacing to a slow responding device. Enter a value in the range of 0 to 65535.</td></tr>
<tr><td>5050</td><td>RTS On Delay</td><td>This parameter sets the number of milliseconds to delay after RTS is asserted before the data will be transmitted. Enter a value in the range of 0 to 65535.</td></tr>
<tr><td>5051</td><td>RTS Off Delay</td><td>This parameter sets the number of milliseconds to delay after the last byte of data is sent before the RTS modem signal will be set low. Enter a value in the range of 0 to 65535.</td></tr>
<tr><td>5052</td><td>Use CTS Line</td><td>This parameter specifies if the CTS modem control line is to be used. If the parameter is set to 0, the CTS line will not be monitored. If the parameter is set to 1, the CTS line will be monitored and must be high before the module will send data. Normally, this parameter is required when half-duplex modems are used for communication (2-wire).</td></tr>
<tr><td>5053</td><td>ENQ Delay</td><td>This parameter specifies the number of milliseconds to wait after a DLE-ACK is received from a slave using half-duplex mode before the DLE-ENQ request is made for data. Enter a value in the range of 0 to 65535.</td></tr>
<tr><td>5054</td><td>Command Count</td><td>This parameter specifies the number of commands to be processed for the port. Enter a value of 0 to 100.</td></tr>
</table>

Register	Content	Description
5055	Minimum Command Delay	This parameter specifies the number of milliseconds to wait between the initial issuance of a command. This parameter can be used to delay all commands sent to slaves to avoid "flooding" commands on the network. This parameter does not affect retries of a command as they will be issued when failure is recognized. Enter a value in the range of 0 to 65535.
5056	Command Error Pointer	This parameter sets the address in the internal Modbus database where the command error data will be placed. If the value is set to -1, the data will not be transferred to the database. Enter a value of 0 to 4999.
5057	Response Timeout	This parameter represents the message response timeout period in 1-ms increments. This is the time that a port configured as a master will wait before re-transmitting a command if no response is received from the addressed slave. The value is set depending upon the communication network used and the expected response time of the slowest device on the network.
5058	Retry Count	This parameter specifies the number of times a command will be retried if it fails. Enter a value in the range of 0 to 10.
5059	Error Delay Count	This parameter specifies the number of polls to be skipped on the slave before trying to re-establish communications. After the slave fails to respond, the master will skip commands to be sent to the slave the number of times entered in this parameter. Enter a value in the range of 0 to 65535.
5060	Slave List Pointer	This parameter specifies the starting address in the virtual database where the 256 slave status values will be written. If the parameter is set to -1, the slave data will not be placed in the database. Enter a value in the range of -1 to 4743.
5061	Slave List Frequency	Not used in this version of the software
5062	First File	This parameter is used when a request for a file is received on the communication port. This field is required when responding to PLC5 and SLC DF1 commands. Use this parameter to define the virtual file(s) to be simulated on the module. This value is fixed at 7 for N7.
5063	File Size	This parameter is used to specify the size of each file to be simulated on the module. All files simulated are defined to have the same assigned size. This value is fixed at 200.
5064	File Offset	This parameter sets the database register location of the first element in the first file simulated in the module. All offsets in the first file and subsequent files will be computed using the address specified. This value is fixed at 0. Therefore, the module assumes N7:0 starts at the database offset of 0 and each file is 200 words in size.
5065	Map Count	Number of file map overrides.
5066 to 5069	Spare	

8.8.4 Port 1 Commands

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

8.8.5 Port 2 Commands

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

8.8.6 Port 1 Override File Maps

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

8.8.7 Port 2 Override File Maps

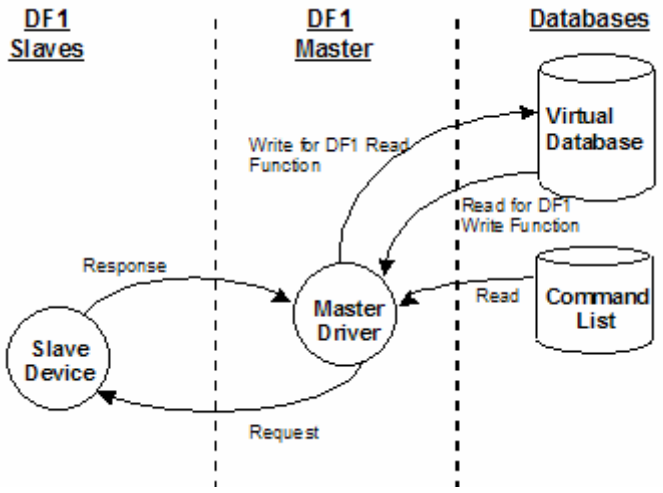
Register	Content	Description
8200 to 8203	File Map #1	This set of registers contains the first override file map for the slave port.
8204 to 8207	File Map #2	This set of registers contains the second override file map for the slave port.
-		
8396 to 8399	Command # 50	This set of registers contains the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

8.9 DF1 Command Set For ProSoft Technology Communication Modules

8.9.1 Introduction

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

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



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

8.9.2 Command Function Codes

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

Column	Parameter	Description	Parameter
1	Enable/Type Word	0=Disabled, 1=Continuous and 2=Conditional.	
2	Virtual Database Address	This parameter defines the database address of the first data point to be associated with the command.	
3	Poll Interval	Minimum time in tenths of a second to wait before polling with this command.	
4	Count	Number of data word values to be considered by the function.	
5	Swap Type Code	Swap type code for command: 0=None, 1=Swap words, 2=Swap words & bytes and 3=swap bytes in each word.	
6	Node Address	Address of unit to reach on the data highway.	
7	Function Code = 100	Word Range Write Command.	
8	File Number	PLC-5 file number to be associated with the command. If a value of -1 is entered for the parameter, the field will not be used in the command, and the default file will be used.	P1
9	Element Number	The parameter defines the element in the file where write operation will start. If a value of -1 is entered for the parameter, the field will not be used in the command, and the default element will be used.	P2
10	Sub-Element Number	This parameter defines the sub-element to be used with the command. Refer to the AB documentation for a list of valid sub- element codes. If the value is set to -1, the default sub-element number will be used.	P3
11	Not Used	This field is not used by the command. Values entered in this column will be ignored.	P4

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

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

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

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

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

Column	Parameter	Description	Parameter
1	Enable/Type Word	0=Disabled, 1=Continuous and 2=Conditional.	
2	Virtual Database Address	This parameter defines the database address for the data to be associated with the command.	
3	Poll Interval	Minimum number of seconds to wait before polling with this command.	
4	Count	Number of data word values to be considered by the function.	
5	Swap Type Code	Swap type code for command: Always zero (0).	
6	Node Address	Address of unit to reach on the data highway.	
7	Function Code = 102	Read-Modify-Write Command.	
8	File Number	PLC-5 file number to be associated with the command. If a value of -1 is entered for the parameter, the field will not be used in the command, and the default file will be used.	P1

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

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

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

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

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

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

Column	Parameter	Description	Parameter
1	Enable/Type Word	0=Disabled and 1=Continuous.	
2	Virtual Database Address	This parameter defines the database address of the first data point to be associated with the command.	
3	Poll Interval	Minimum time in tenths of a second to wait before polling with this command.	
4	Count	Number of data word values to be considered by the function.	

Column	Parameter	Description	Parameter
5	Swap Type Code	Swap type code for command: 0=None, 1=Swap words, 2=Swap words & bytes and 3=swap bytes in each word.	
6	Node Address	Address of unit to reach on the data highway.	
7	Function Code = 151	Word Range Read Command.	
8	File String	PLC-5 address as specified as an ASCII string. For example, N10:300.	P1
9 to 11	Not Used	These fields are not used by the command. Values entered in these columns will be ignored.	P2 to P4

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

8.9.3 PLC-5 Processor Specifics

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

PLC-5 Sub-Element Codes

Timer / Counter

Code	Description
0	Control
1	Preset
2	Accumulated

Control

Code	Description
0	Control
1	Length
2	Position

PD*

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

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

BT

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

MG

Code	Description
0	Control
1	Error
2	RLEN
3	DLEN

8.9.4 SLC Processor Specifics

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

SLC File Types

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

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

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

8.9.5 MicroLogix Processor Specifics

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

SLC File Types

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

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

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

8.9.6 ControlLogix Processor Specifics

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

8.10 DF1 Command List Form

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

8.11 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 PLC. What this means is that if you had a list of 10 commands at the end of every PLC scan one command would execute. This would mean that it would take 10 PLC 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 PLC 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 ProTalk requires a simple text based configuration file to make it operational. For a really quick tutorial on the modules communications with the PLC 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.

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 PLC 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 instance 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 ProTalk 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

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

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

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

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

9.1 How to Contact Us: Technical Support

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

Asia Pacific

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

Languages spoken include: Chinese, English

Europe (location in Toulouse, France)

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

Languages spoken include: French, English

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

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

Languages spoken include: English, Spanish

For technical support calls within the United States, an after-hours *answering system allows pager access to one of our qualified technical and/or application support engineers at any time to answer your questions.*

Brasil (location in Sao Paulo)

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

Languages spoken include: Portuguese, English

9.2 Return Material Authorization (RMA) Policies and Conditions

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

9.2.1 All Product Returns:

a) In order to return a Product for repair, exchange or otherwise, the Customer must obtain a Returned Material Authorization (RMA) number from ProSoft and comply with ProSoft shipping instructions.
b) In the event that the Customer experiences a problem with the Product for any reason, Customer should contact ProSoft Technical Support at one of the telephone numbers listed above (page 133). 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.

9.2.2 Procedures for Return of Units Under Warranty:

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

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

9.2.3 Procedures for Return of Units Out of Warranty:

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

The following is a list of non-repairable units:

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

9.2.4 Purchasing Warranty Extension:

a) ProSoft's standard warranty period is three (3) years from the date of shipment as detailed in "Limited Warranty (page 136)". The Warranty Period may be extended at the time of equipment purchase for an additional charge, as follows:

- Additional 1 year = 10% of list price
- Additional 2 years = 20% of list price
- Additional 3 years = 30% of list price

9.3 LIMITED WARRANTY

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

9.3.1 What Is Covered By This Warranty

a) *Warranty On New Products*: ProSoft warrants, to the original purchaser, that the Product that is the subject of the sale will (1) conform to and perform in accordance with published specifications prepared, approved and issued by ProSoft, and (2) will be free from defects in material or workmanship; provided these warranties only cover Product that is sold as new. This Warranty expires three years from the date of shipment (the "Warranty Period"). If the Customer discovers within the Warranty Period a failure of the Product to conform to specifications, or a defect in material or workmanship of the Product, the Customer must promptly notify ProSoft by fax, email or telephone. In no event may that notification be received by ProSoft later than 39 months. Within a reasonable time after notification, ProSoft will correct any failure of the Product to conform to specifications or any defect in material or workmanship of the Product, with either new or used replacement parts. Such repair, including both parts and labor, will be performed at ProSoft's expense. All warranty service will be performed at service centers designated by ProSoft.
b) *Warranty On Services*: Materials and labor performed by ProSoft to repair a verified malfunction or defect are warranteed in the terms specified above for new Product, provided said warranty will be for the period remaining on the original new equipment warranty or, if the original warranty is no longer in effect, for a period of 90 days from the date of repair.

9.3.2 What Is Not Covered By This Warranty

a) ProSoft makes no representation or warranty, expressed or implied, that the operation of software purchased from ProSoft will be uninterrupted or error free or that the functions contained in the software will meet or satisfy the purchaser's intended use or requirements; the Customer assumes complete responsibility for decisions made or actions taken based on information obtained using ProSoft software.

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

9.3.3 Disclaimer Regarding High Risk Activities

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

9.3.4 Intellectual Property Indemnity

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

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

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

9.3.5 Disclaimer of all Other Warranties

The Warranty set forth in What Is Covered By This Warranty (page 136) 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.

*9.3.6 Limitation of Remedies ***

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

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

9.3.7 Time Limit for Bringing Suit

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

9.3.8 No Other Warranties

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

9.3.9 Allocation of Risks

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

9.3.10 Controlling Law and Severability

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

Index

S

T

U

V

W

Y