ProTalk®

PTQ-LNG

Quantum / Unity Platform

Landis & Gyr Telegyr Interface Module

User Manual

May 29, 2007

ProSoft® TECHNOLOGY

Please Read This Notice

Successful application of this module requires a reasonable working knowledge of the Schneider Electric Quantum / Unity hardware, the PTQ-LNG 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 assure that the information provided is accurate and a true reflection of the product's installation requirements. In order to assure 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.

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-LNG User Manual
May 29, 2007
PSFT.LNG.PTQ.UM.07.05.29

Contents

1 Start Here

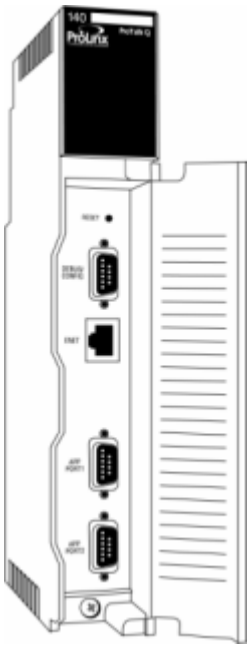
In This Chapter

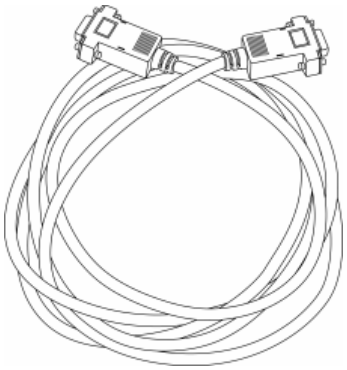
This guide is intended to guide you through the ProTalk module setup process, from removing the module from the box to exchanging data with the processor. In doing this, you will learn how to:

- 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

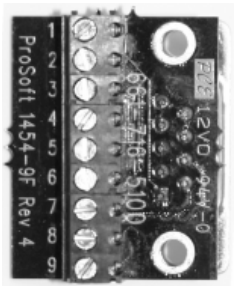
1.1 Hardware and Software Requirements

1.1.1 ProTalk Module Carton Contents





ProTalk Module	Null Modem Serial Cable



1454-9F DB-9 Female to 5 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.

Quantum / Unity Hardware

This guide assumes that you are familiar with the installation and setup of the Quantum / Unity hardware. The following should 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

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 packages, including Concept and UnityPro XL. For all other programming packages, please contact technical support.

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.

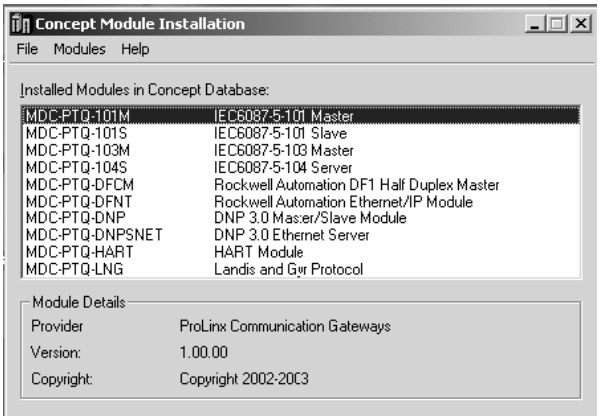
2.1 Information for Concept Version 2.6 Users

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.

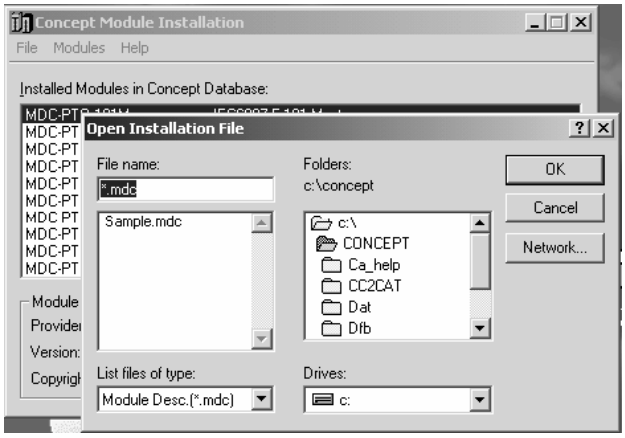
2.1.1 Installing MDC Configuration Files

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:

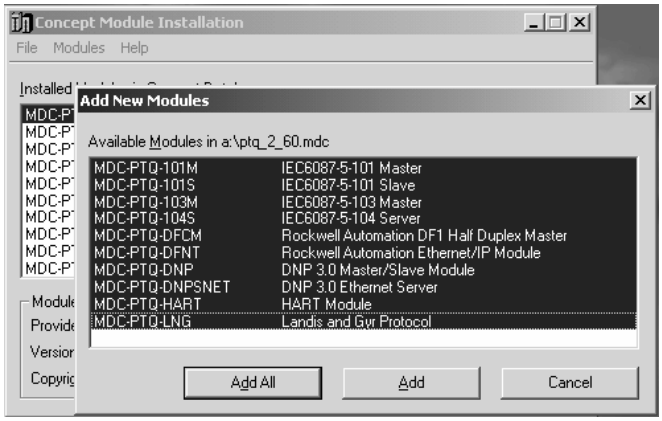


3 If you are using a Quantum processor, you will need the MDC files. In the File/Open dialog box, navigate to the **MDC Files** directory on the ProTalk CD.

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.

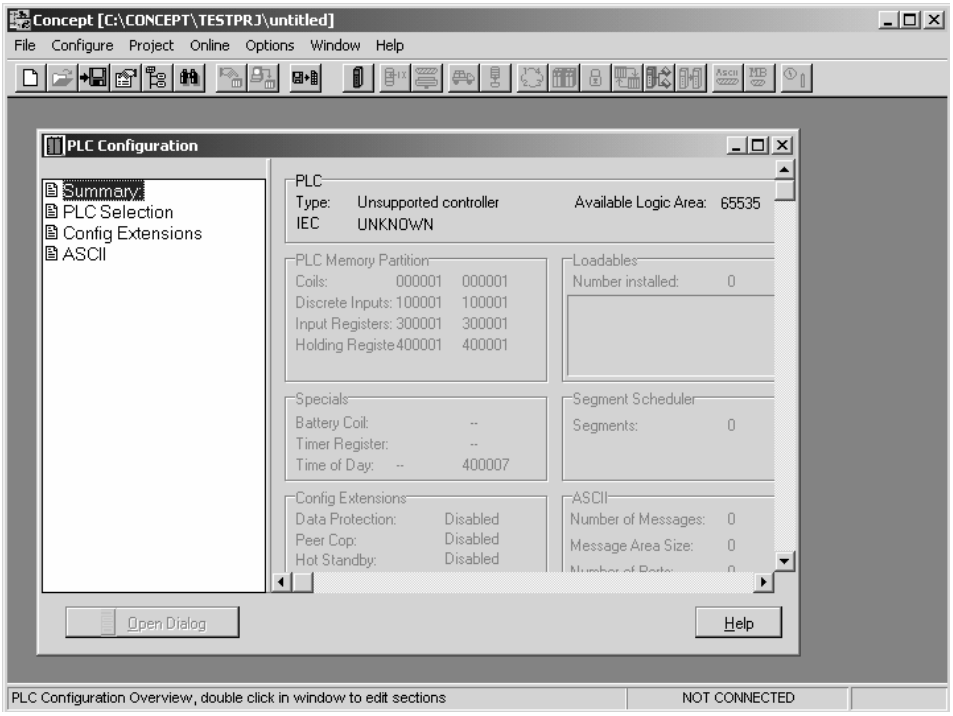
6 When the process is complete, open the File menu and choose Exit to save your changes.

2.2 Create a New Project

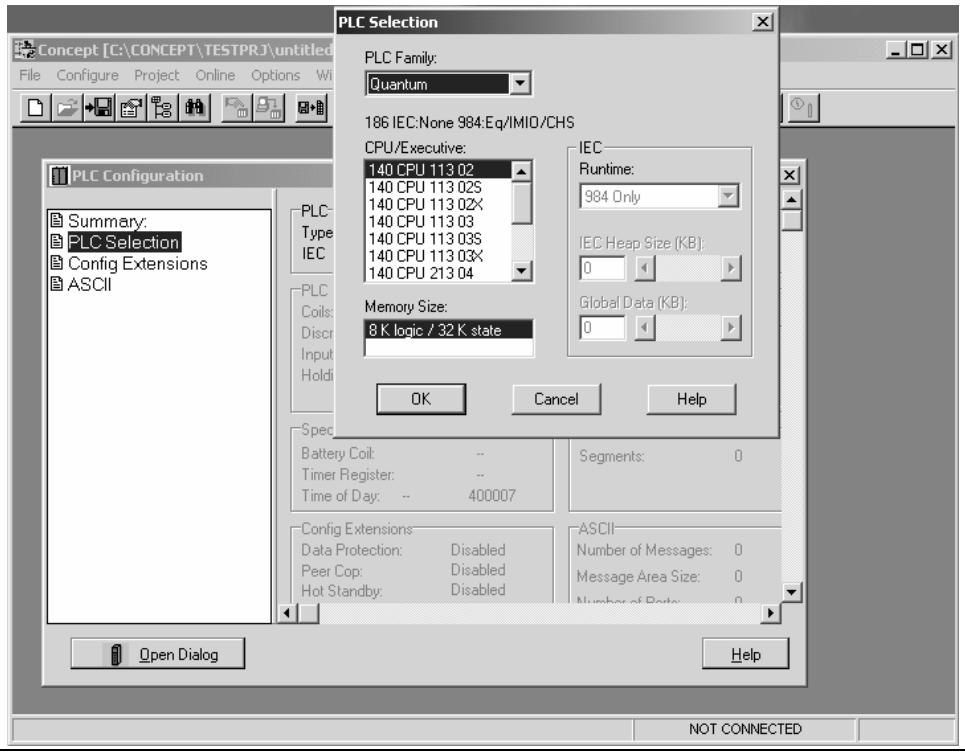
This phase of the setup procedure must be performed on a computer that has the Concept configuration software installed.

1 From your computer, choose **Start** → **Programs** → **Concept V2.6 XL.EN** → **Concept**. This action opens the Concept window.

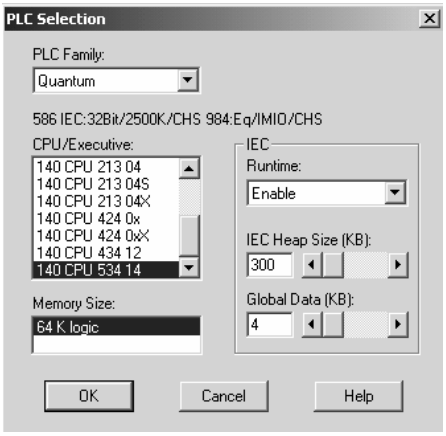
2 Open the File menu, and then choose **New Project**. This action opens the PLC Configuration dialog box.



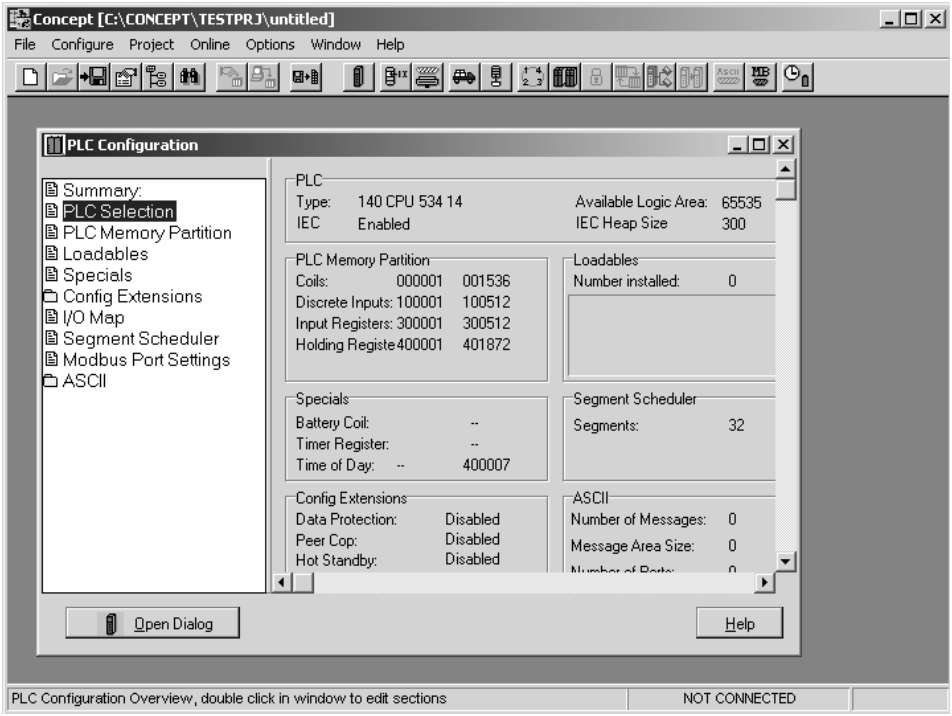
3 In tIn the list of options on the left side of this dialog box, double-click the *PLC Selection* folder. This action opens the PLC Selection dialog box.



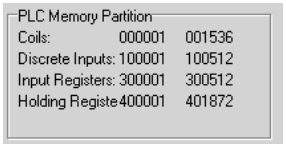
4 In the *CPU/Executive* pane, use the scroll bar to locate and select the PLC to configure.



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



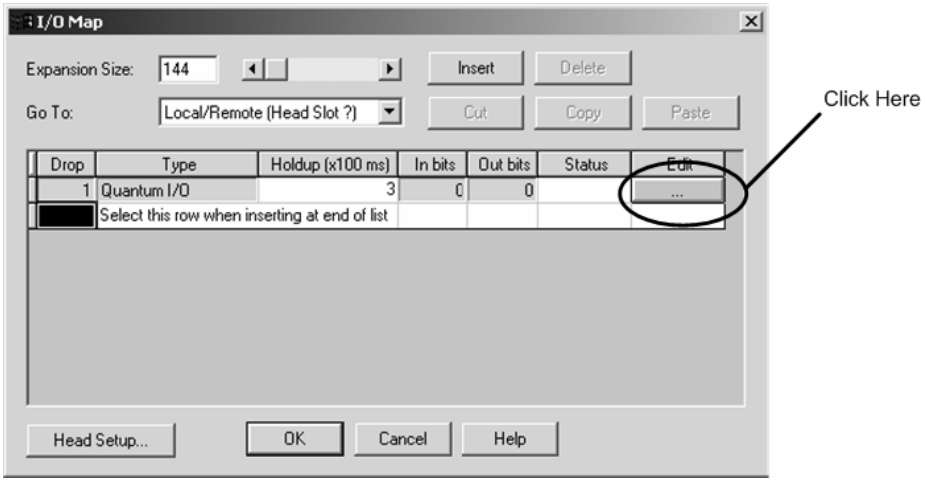
6 Make a note of the holding registers for the module. You will need this information when you modify your application as outlined in the ProTalk Application Reference Guides. The Holding Registers are displayed in the PLC Memory Partition pane of the PLC Configuration dialog box.



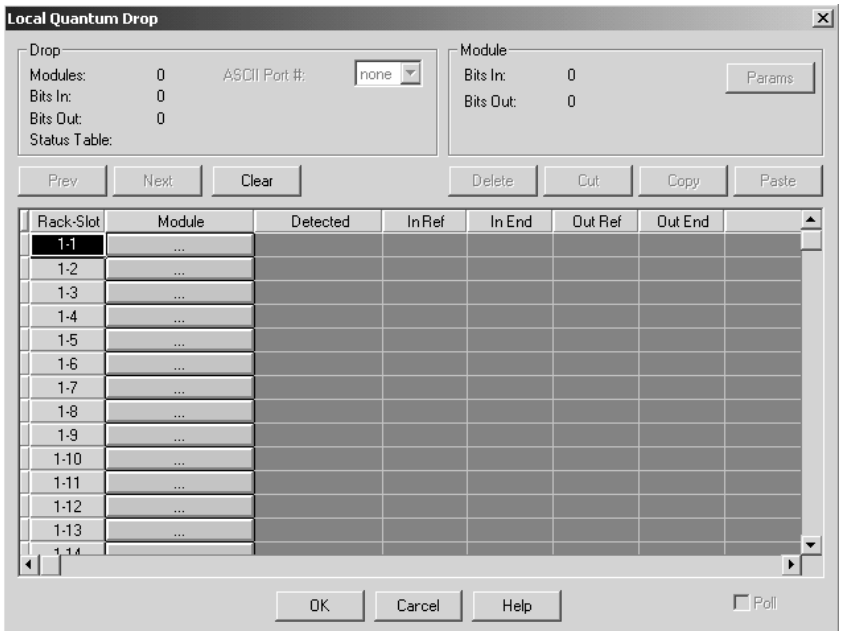
2.3 Add the PTQ Module to the Project

The next step is to add one or more of the PTQ modules to the Project. To add modules:

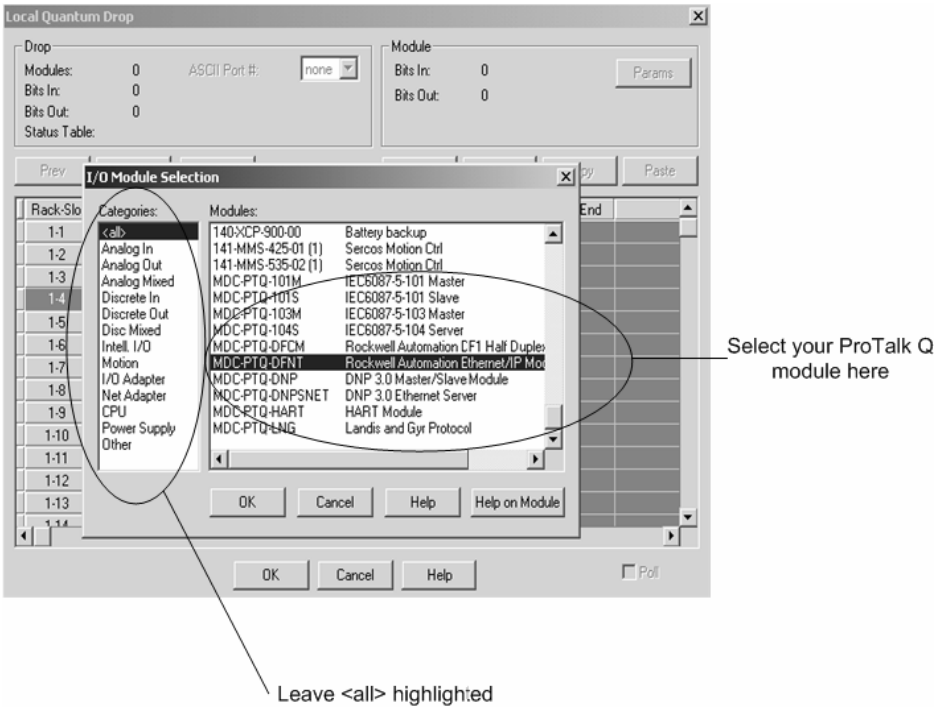
1 In the list of options on the left side of the *PLC Configuration* dialog box, double-click *I/O Map*. This action opens the I/O Map dialog box.



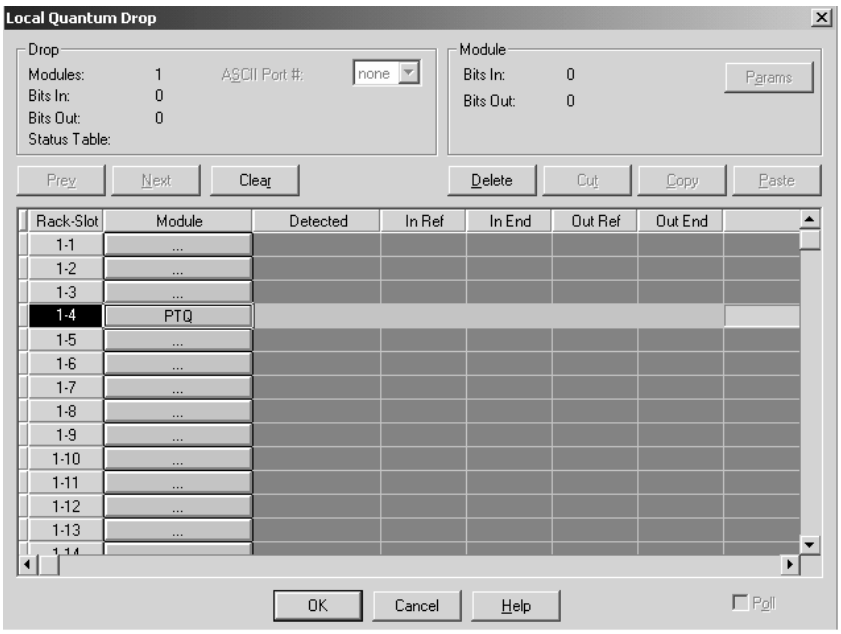
2 Click the **Edit** button to open the *Local Quantum Drop* dialog box. This dialog box is where you identify rack and slot locations.



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.

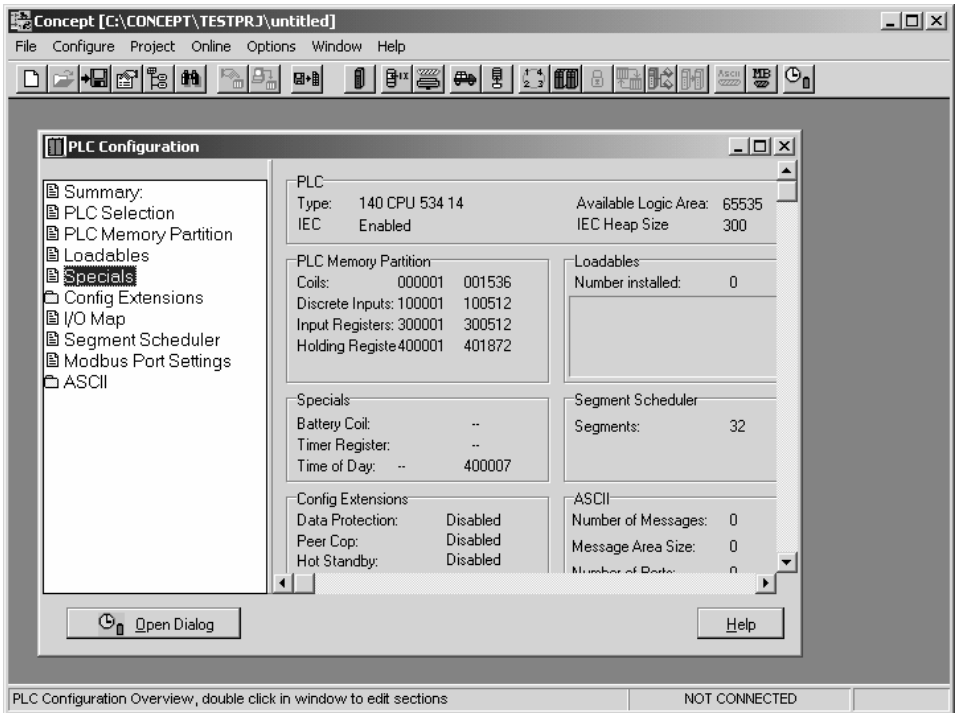


5 Repeat steps 3 through 5 for each ProTalk module you plan to install. When you have finished installing your ProTalk modules, click OK to save your settings. Click Yes to confirm your settings.

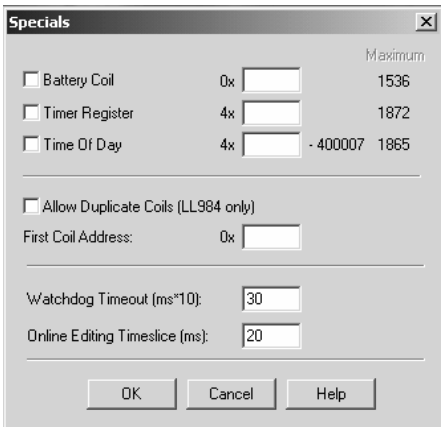
Tip: Select a module, and then click the Help on Module button for help pages.

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.

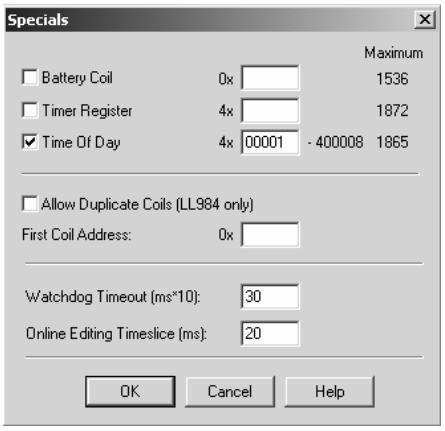


2 This action opens the Specials dialog box.



Selecting the Time of Day

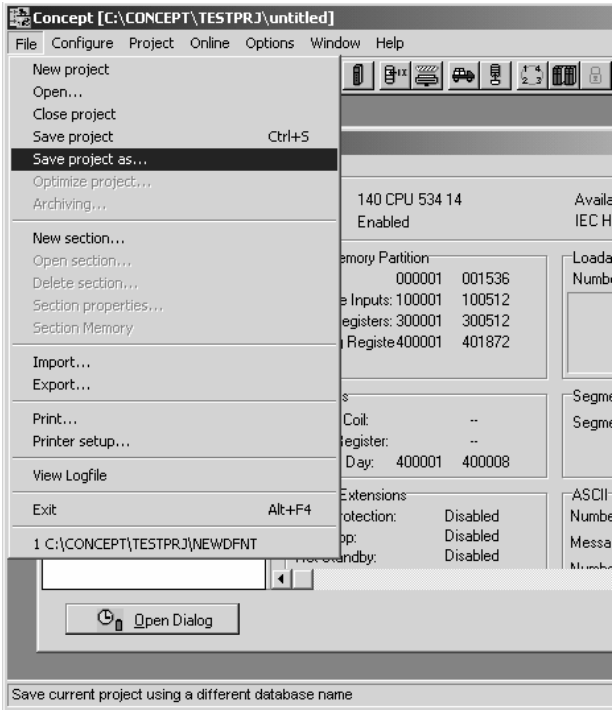
3 Select (check) the Time of Day box, and then enter the value 00001 as shown in the following example. This value sets the first time of day register to 400001.



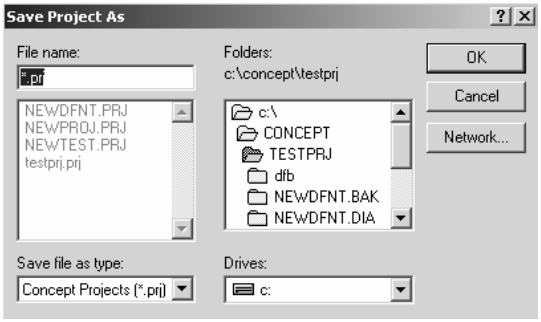
4 Click OK to save your settings and close the Specials dialog box.

Saving your project

5 In the PLC Configuration dialog box, choose File -> Save project as.



6 This action opens the Save Project As dialog box.

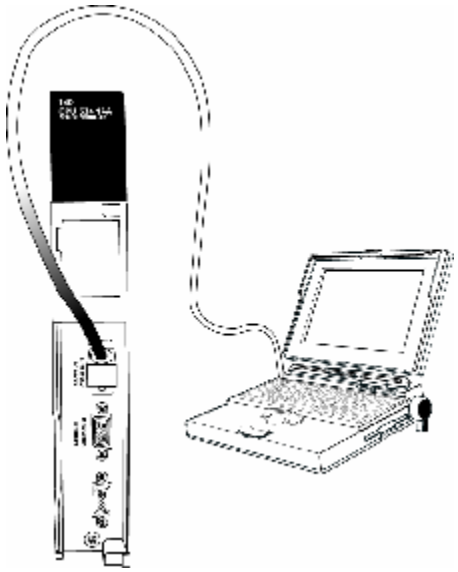


7 Name the project, and then click OK to save the project to a file.

2.5 Download the Project to the Processor

The next step is to download (copy) the project file to the Quantum Processor.

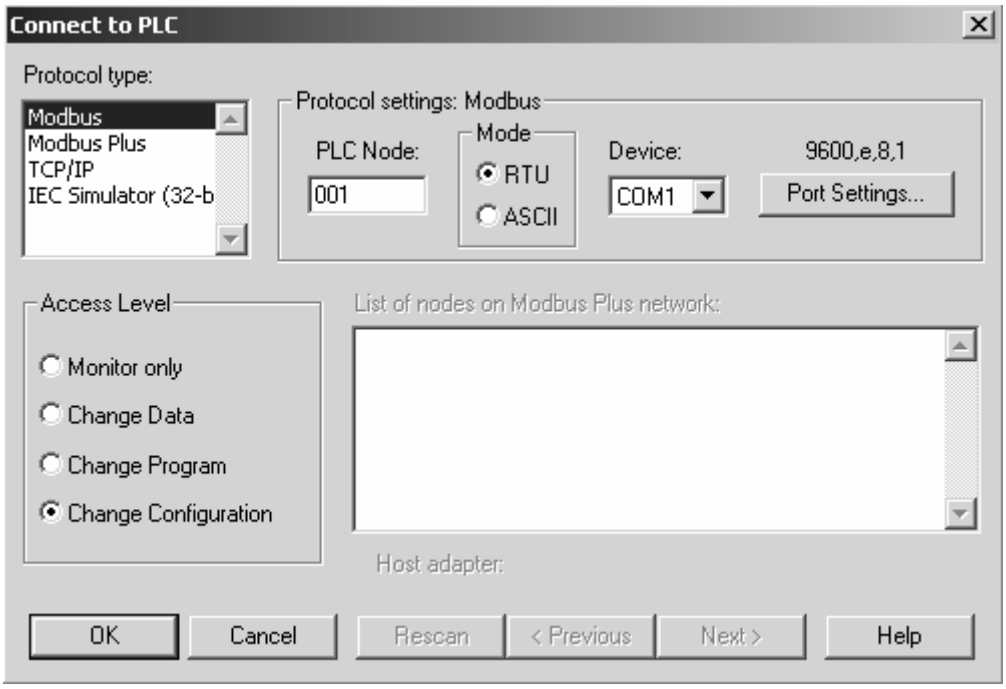
1 Use the null modem cable to connect your PC's serial port to the Quantum processor, as shown in the following illustration.



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

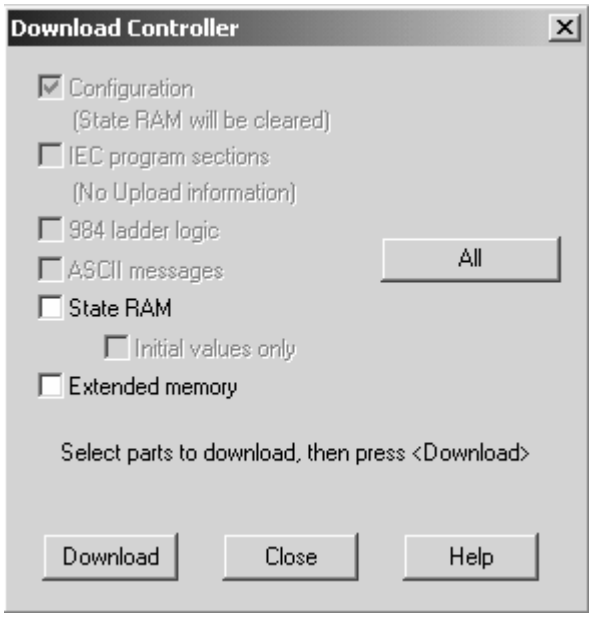
2 Open the PLC menu, and then choose Connect.

3 In the PLC Configuration dialog box, open the Online menu, and then choose Connect. This action opens the Connect to PLC dialog box.

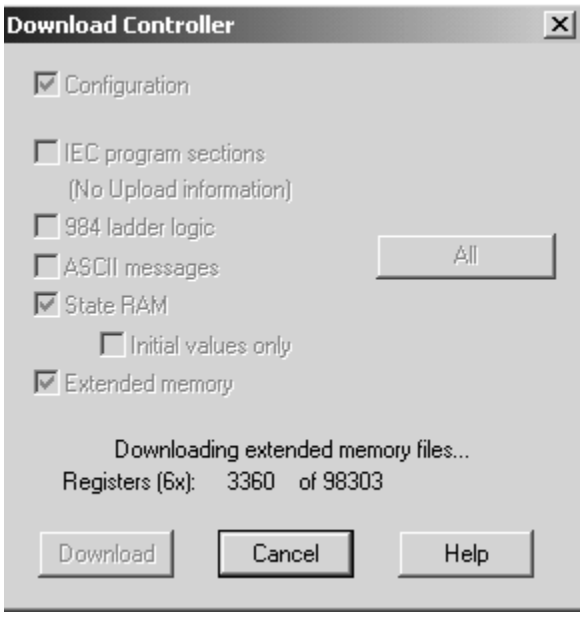


4 Leave the default settings as shown and click OK.
Note: Click OK to dismiss any message boxes that appear during the connection process.

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.

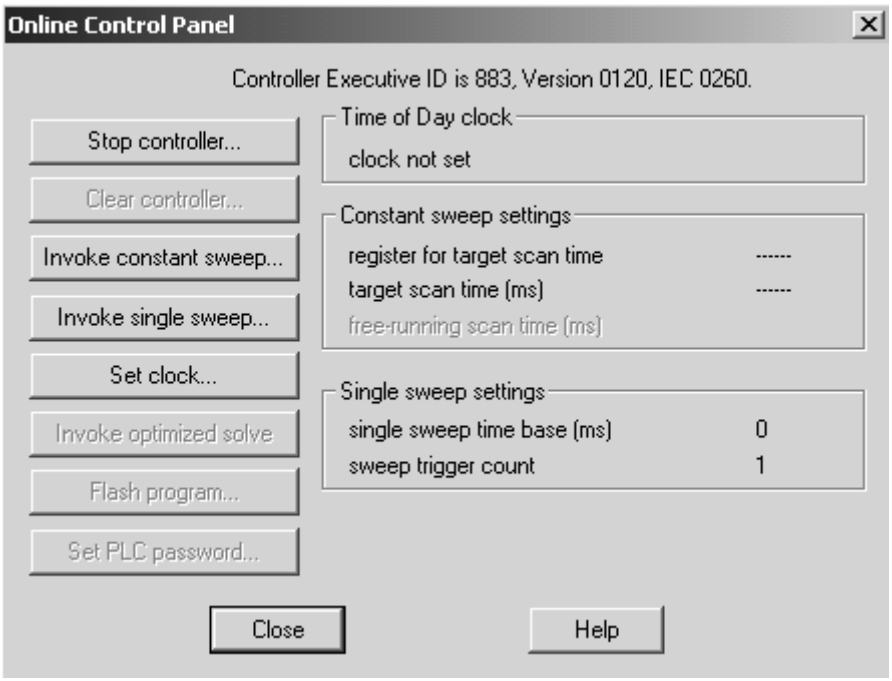


7 When the download is complete, you will be prompted to restart the controller. Click Yes to restart the controller.

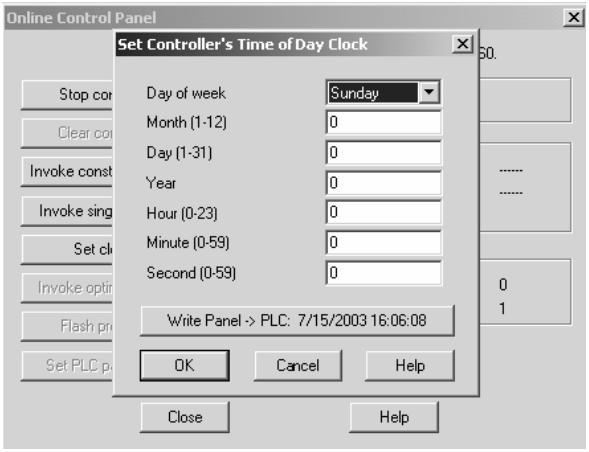
2.6 Verify Successful Download

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.

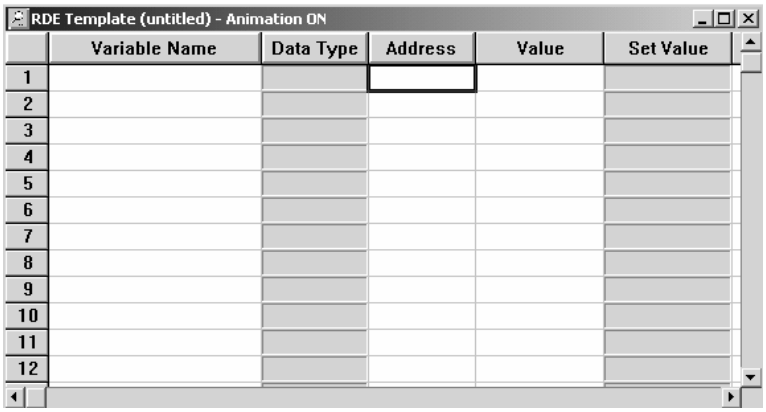


2 Click the Set Clock button to open the Set Controller's Time of Day Clock dialog box.



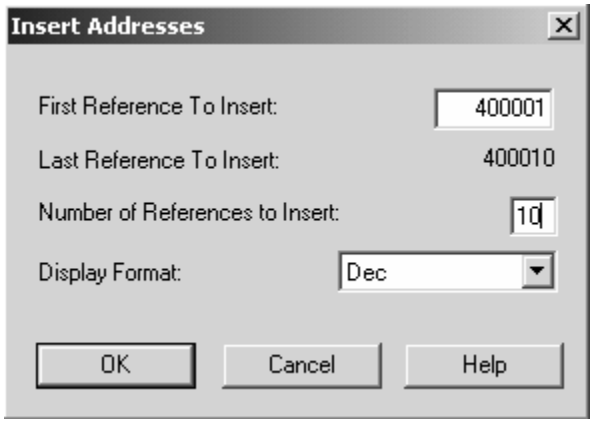
3 Click the Write Panel button. This action updates the date and time fields in this dialog box. Click OK to close this dialog box and return to the previous window.
4 Click Close to close the Online Control Panel dialog box.
5 In the PLC Configuration window, open the Online menu, and then choose Reference 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.

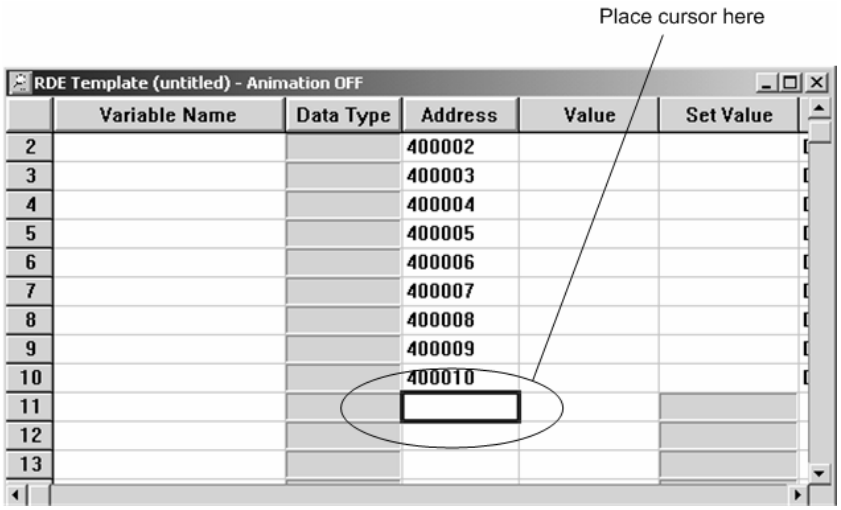


7 In the PLC Configuration window, open the Templates menu, and then choose Insert Addresses. This action opens the Insert Addresses dialog box.

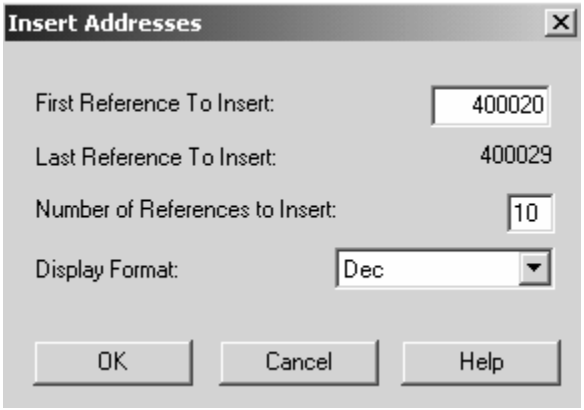
8 On the Insert Addresses dialog box, enter the values shown in the following illustration, and then click OK.



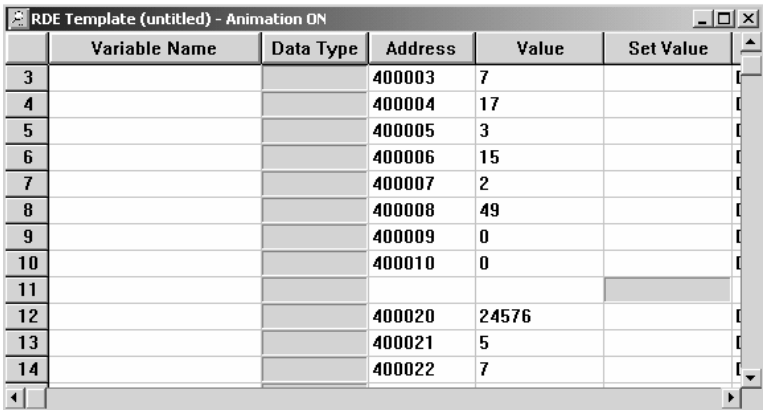
9 Notice that the template populates the address range, as shown in the following illustration. Place your cursor as shown in the first blank address field below the addresses you just entered.



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 400008 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

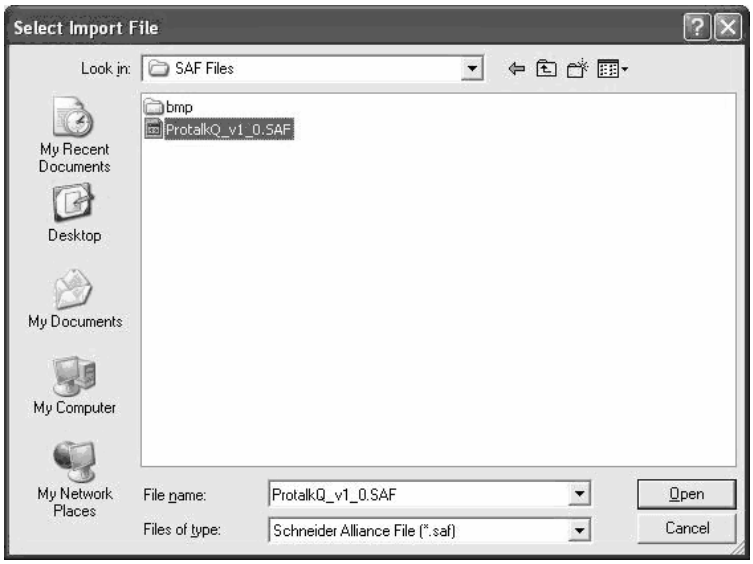
When you use ProWORX 32 software to configure the processor, using the example SAF file provided on the ProTalk Solutions CD-ROM will be required.

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.

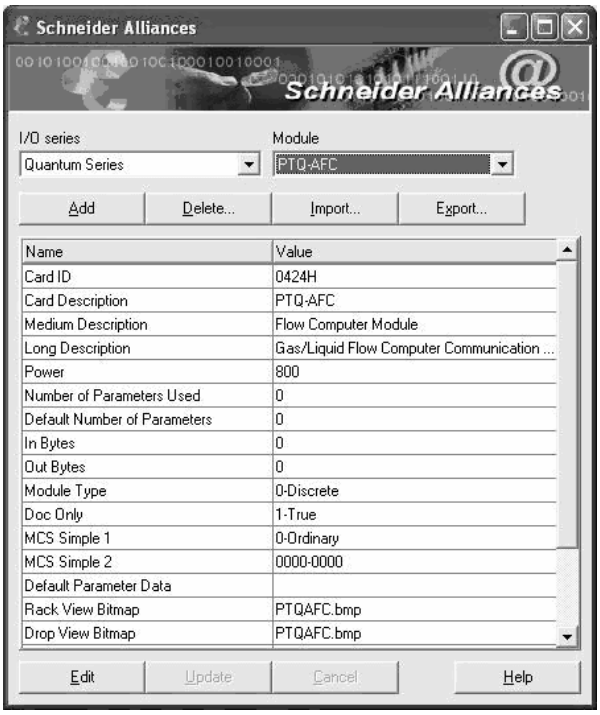
1 Run the Schneider_Alliances.exe application that is installed with the Proworx 32 software:

2 Click on Import…

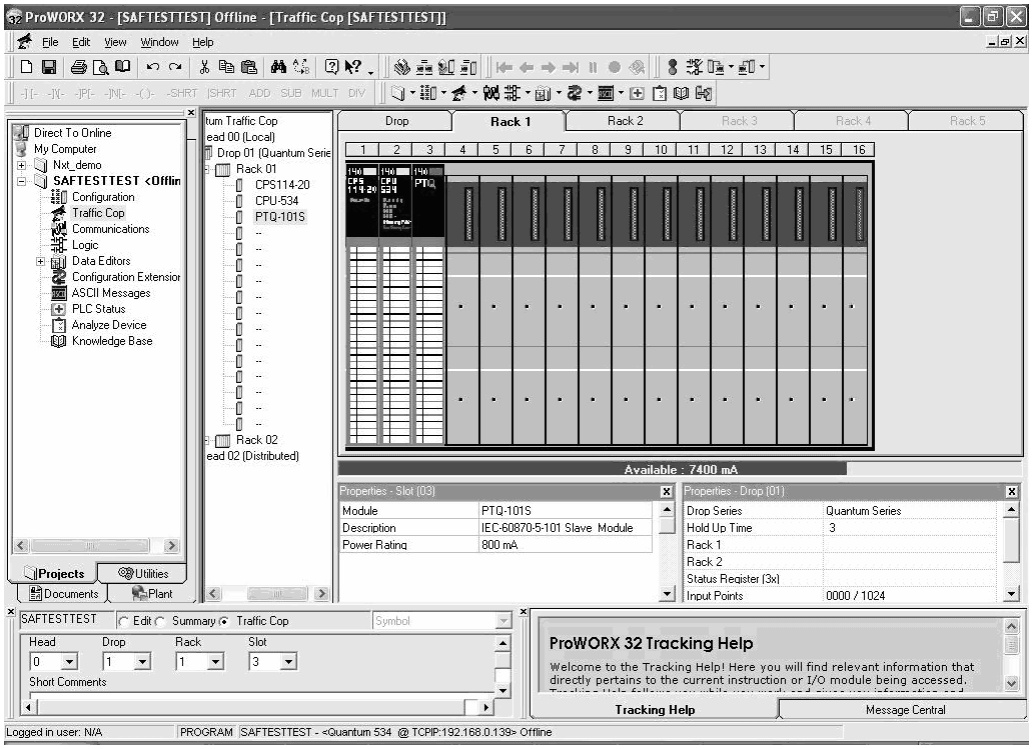
3 Select the .SAF File that is located at the CD-ROM shipped with the PTQ module.



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

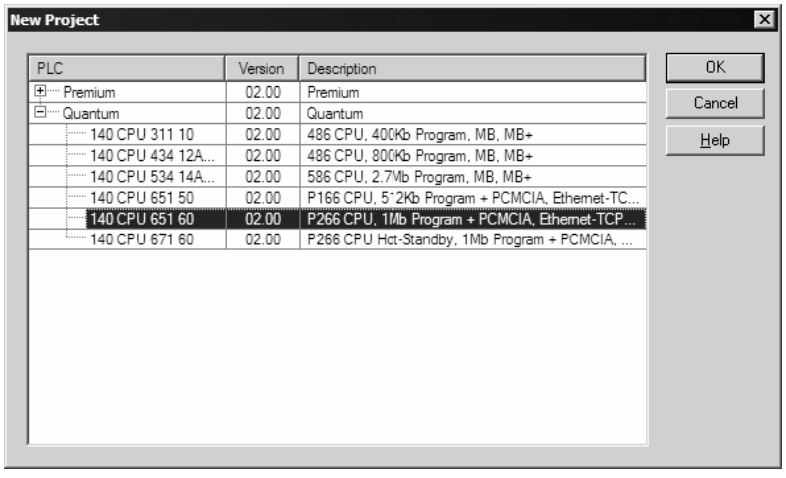
In This Chapter

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.

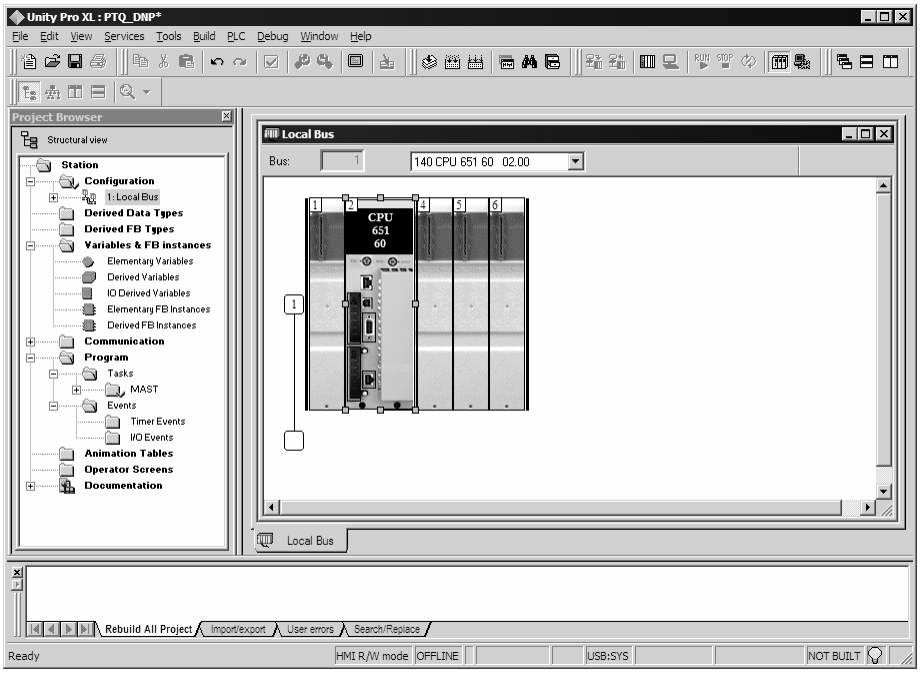
4.1 Create a New Project

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

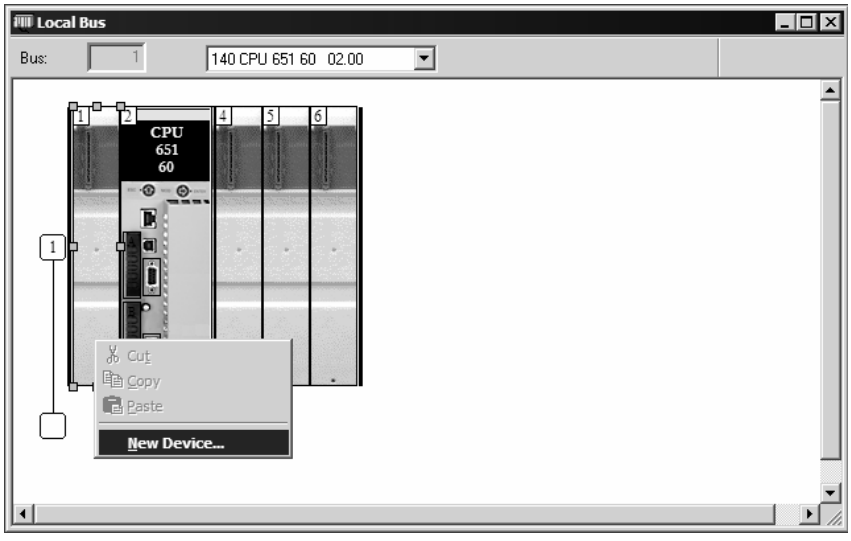
1 In the New Project dialog box, choose the CPU type. In the following illustration, the CPU is 140 CPU 651 60. Choose the processor type that matches your own hardware configuration, if it differs from the example. Click OK to continue.



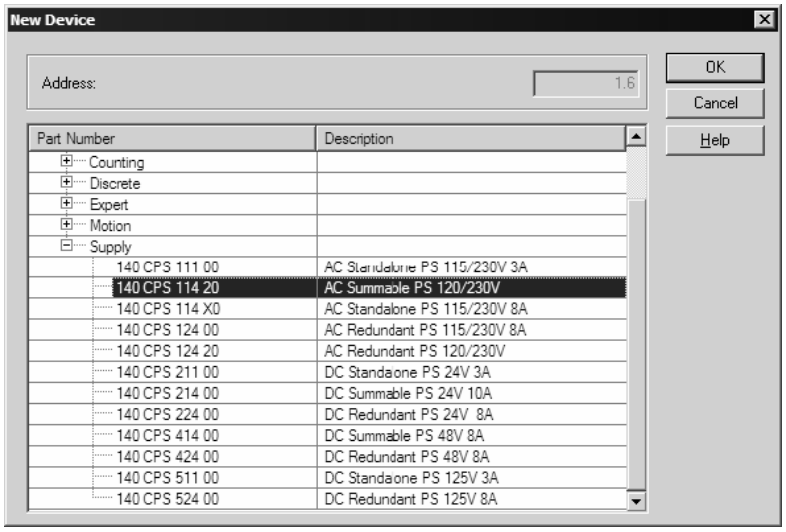
2 The next step is to add a power supply to the project. In the Project Browser, expand the Configuration folder, and then double-click the 1:LocalBus icon. This action opens a graphical window showing the arrangement of devices in your Quantum rack.



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.

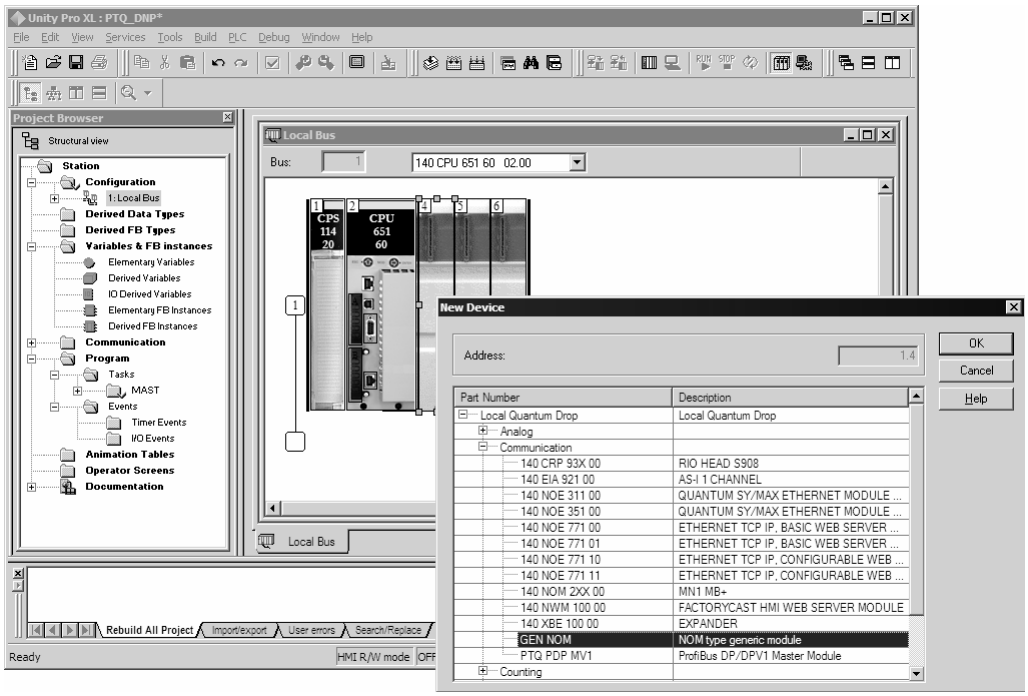


5 Repeat these steps to add any additional devices to your Quantum Rack.

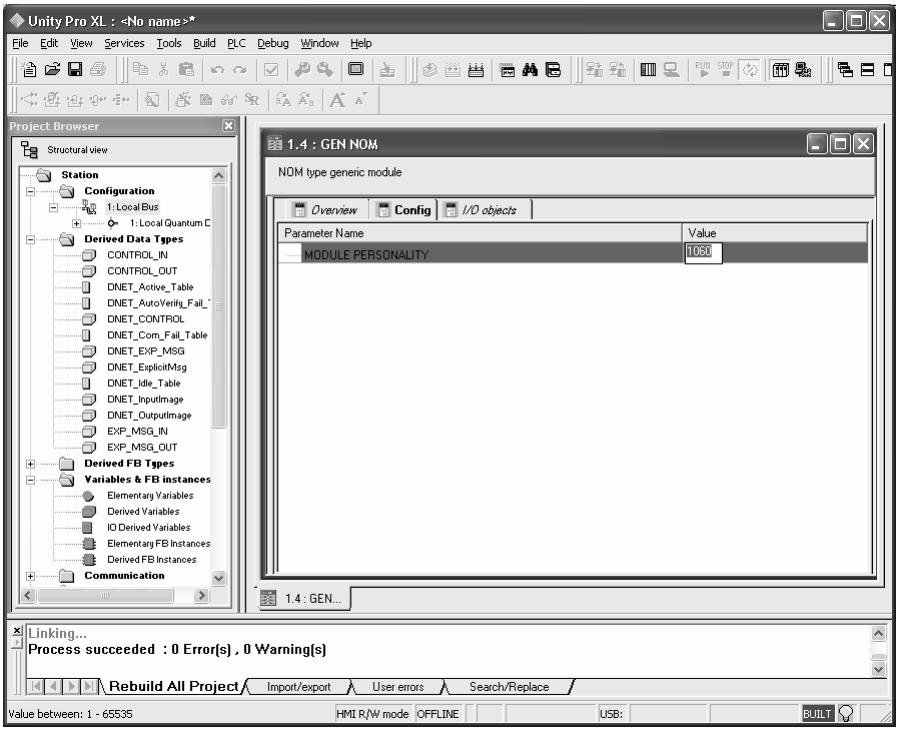
4.2 Add the PTQ Module to the Project

The next step is to add the PTQ module.

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.

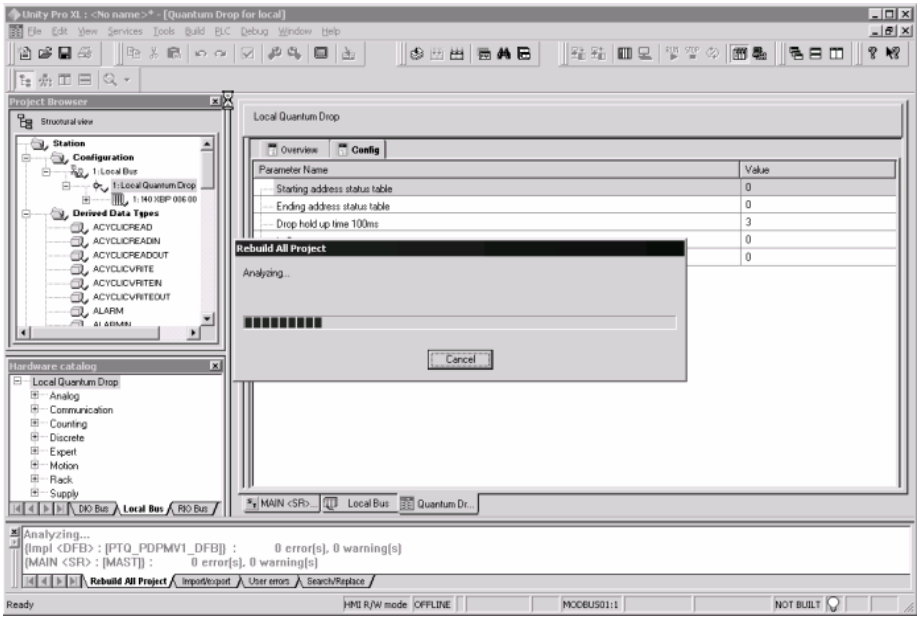
Note: The following steps show you how to build the project in Unity Pro XL. This is not intended to provide detailed information on using Unity Pro XL, or debugging your programs. Refer to the documentation for your processor and for Unity Pro XL for specialized information.

➢ *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.

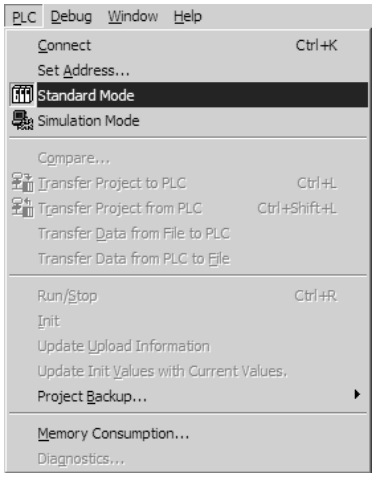
4.4 Connect Your PC to the Processor

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.

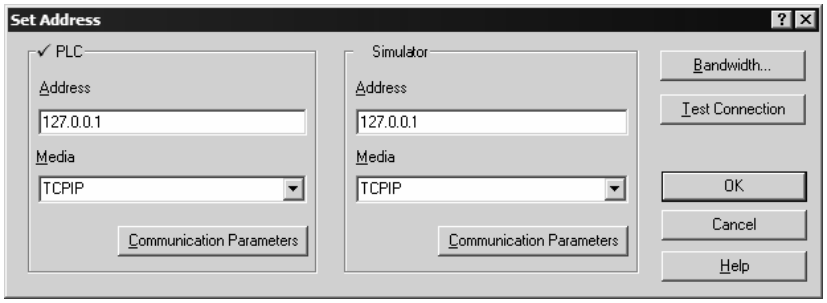
Note: If you have never connected from the PC to your processor before, you must verify that the necessary port drivers are installed and available to UnityPro XL.

➢ *To verify address and driver settings in UnityPro XL:*

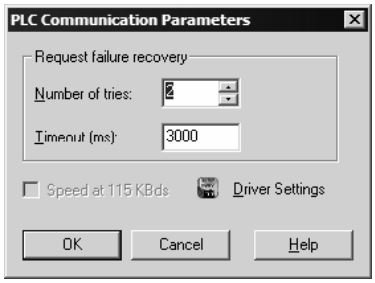
1 Open the PLC menu, and choose Standard Mode. This action turns off the PLC Simulator, and allows you to communicate directly with the Quantum or Unity hardware.



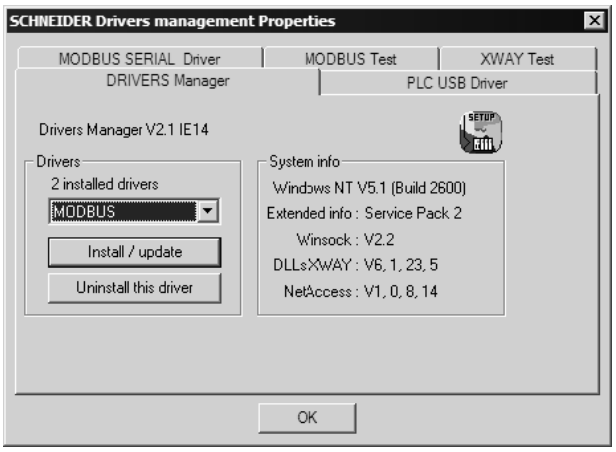
2 Open the PLC menu, and choose Set Address... This action opens the Set Address dialog box. Open the Media dropdown list and choose the connection type to use (TCPIP or USB).



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.

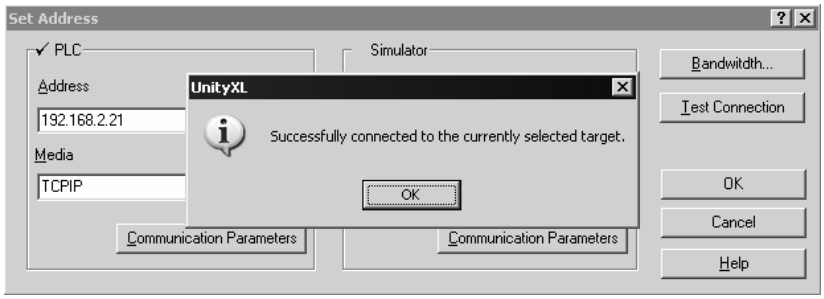
1 If you have not already done so, connect your PC and the processor to an Ethernet hub.

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.

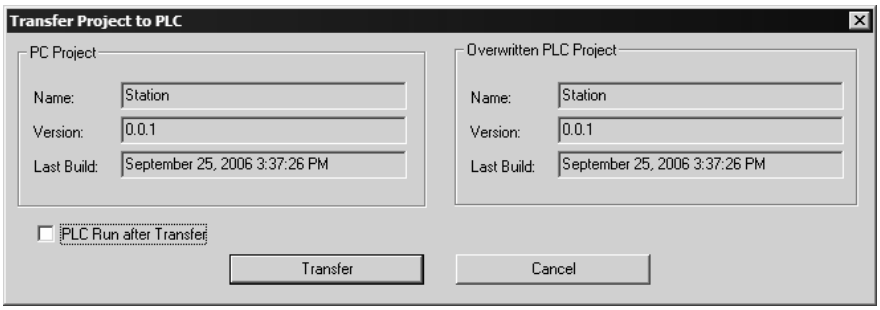


The next step is to download the Project to the Processor.

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

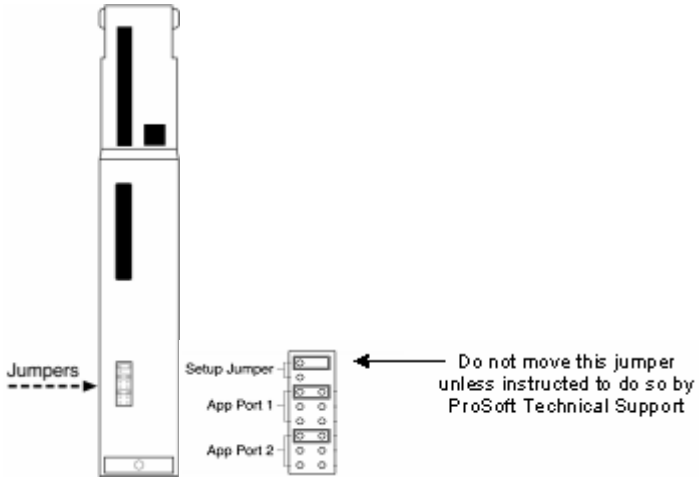
After you complete the following procedures, the ProTalk module will actively be transferring data bi-directionally with the processor.

5.1 Install the ProTalk Module in the Quantum Rack

5.1.1 Verify Jumper Settings

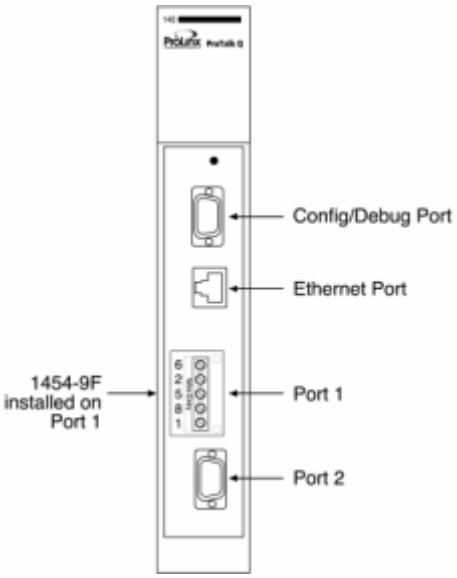
ProTalk modules are configured for RS-232 serial communications by default. To use RS-422 or RS-485, you must change the jumpers.

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



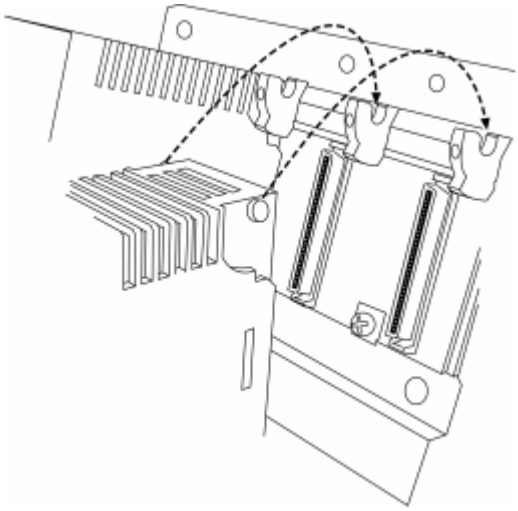
5.1.2 Inserting the 1454-9F connector

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

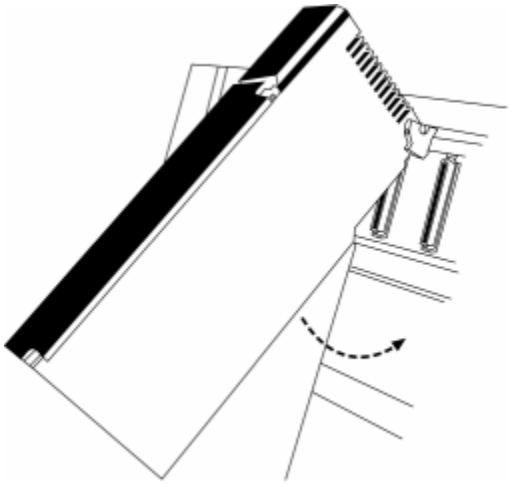


5.1.3 Install the ProTalk Module in the Quantum Rack

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.

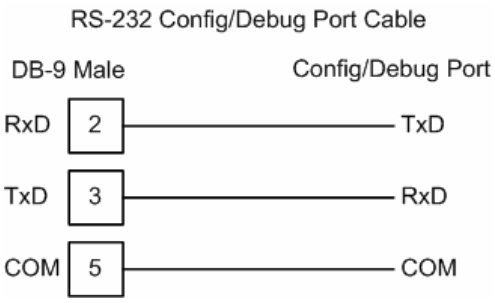
5.1.4 Cable Connections

The application ports on the PTQ-LNG module support RS-232, RS-422, and RS-485 interfaces. Please look at 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.

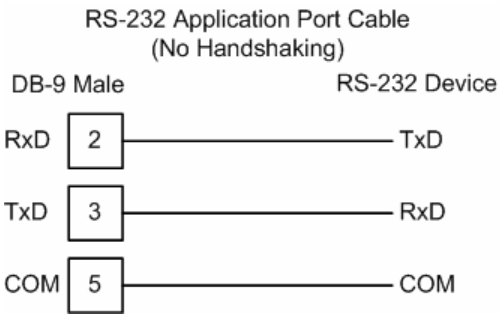
RS-232 Configuration/Debug Port

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:



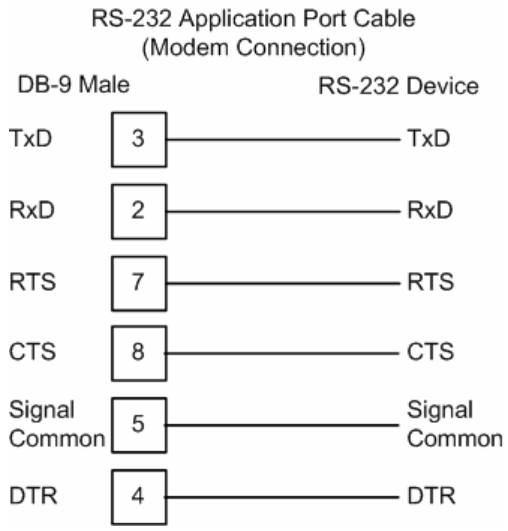
RS-232

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

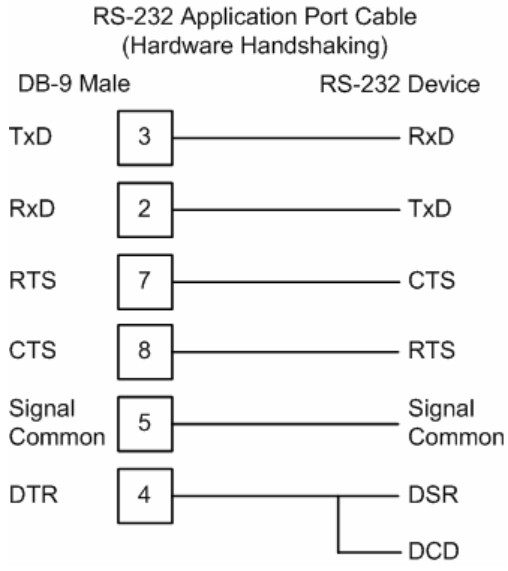
This type of connection is required between the module and a modem or other communication device.



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

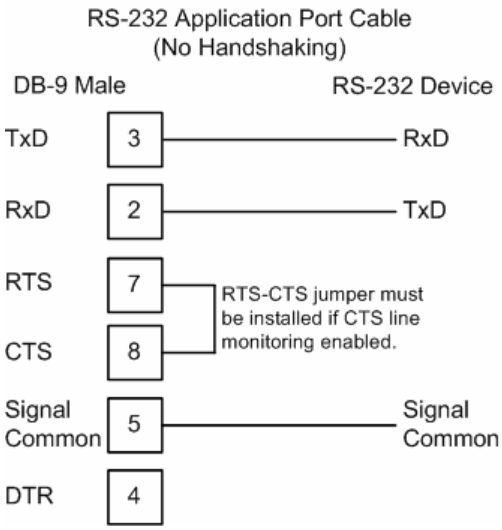
RS-232 -- Null Modem Connection (Hardware Handshaking)

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)

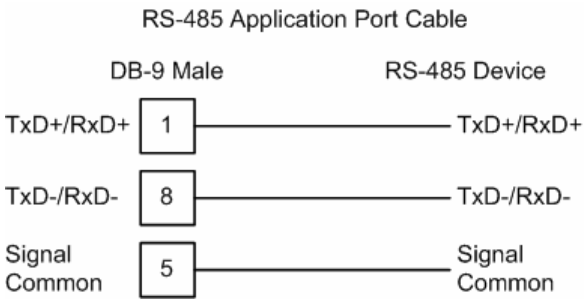
This type of connection can be used to connect the module to a computer or field device communication port.



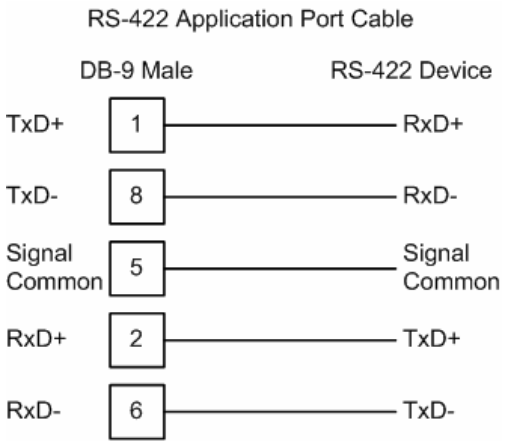
NOTE: If the port is configured with the "Use CTS Line" set to 'Y', then a jumper is required between the RTS and the CTS line on the module connection.

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:



RS-422



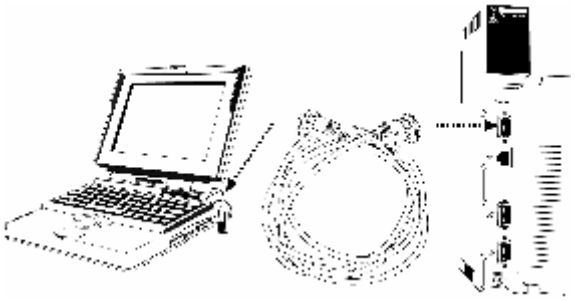
RS-485 and RS-422 Tip

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

5.2 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.

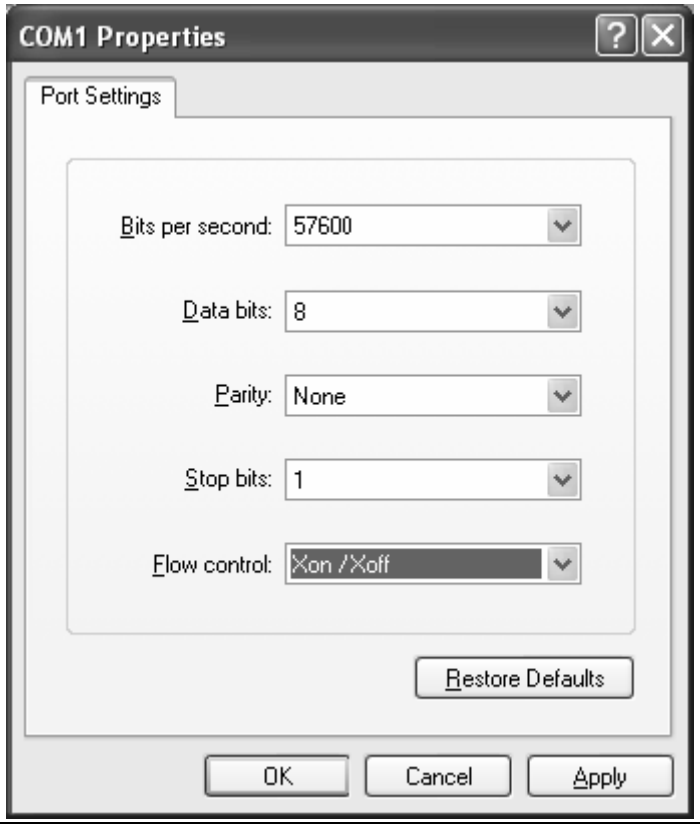
1 Using the supplied Null Modem cable, connect your PC or Laptop to the Configuration/Debug port on the ProTalk module as shown



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.

6 Configuring the Module

In This Chapter

6.1 Before You Begin

This section provides the information required to configure the ProTalk module to work with your application.

Phases I and II described how to set up a Quantum project and then add the ProTalk module to the project.

In this phase, you will:

1 Obtain the Sample Configuration File from the ProTalk CD
2 Edit each of the sections within the LNGS.CFG file
3 Transfer the edited file to the ProTalk module
4 Obtain the Sample Configuration Files

Get the sample configuration files from the ProTalk CD.

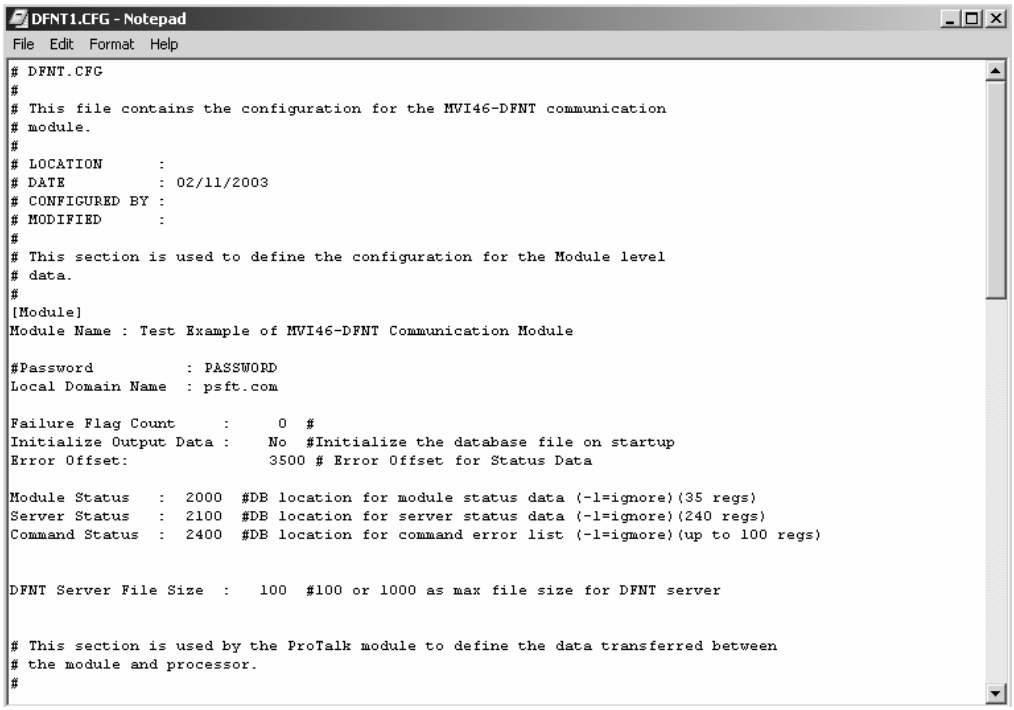
Important: This module supports a maximum configuration file size of 128 kilobytes (131072 bytes). If the configuration file is larger than this size, the module will not accept the download. You can reduce the size of the configuration file by opening the file in a text editor and removing comment lines (lines preceded with the # character).

After you move the files, right-click on each of the files and clear the READ ONLY flag.

Move the appropriate configuration files and ladder to a location on your PC's hard drive. We recommend C:\temp.

1 Choose Start → Programs → Accessories → Notepad.
2 In Notepad, choose File → Open.
The *Open* dialog appears.

3 Browse to the configuration file on your PC and select it. Click **Open**. The configuration file appears in Notepad, ready for editing.



4 Edit the Sample Configuration File Sections

Note: It is important that you plan your configuration before modifying the configuration files. The remainder of this step provides the information to make the appropriate modifications to the configuration files.

6.2 Obtain the Sample Configuration Files

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

1 Use Windows Explorer to locate the sample configuration files for your PTQ module on the PTQ CD.
2 When you have located the correct configuration files, use the Copy and Paste commands to move the files to a location on your PC's hard drive. We recommend C:\temp.
3 Files copied from a CD-ROM are read-only. The next step is to make the files writable. Navigate to the directory where you copied the files, then select the files and click the right mouse button to open a shortcut menu. On the shortcut menu, select Properties, and clear (uncheck) the Read Only check box.
4 The next step is to open the configuration files in a text editor such as Notepad, which comes with Windows. To start Notepad, click the Start button, and then choose **Programs** → **Accessories** → **Notepad**.

5 When Notepad starts, open the File menu, and then choose **Open**. Navigate to the folder where you copied the configuration file on your PC and select the file. Click **Open**. The configuration file will open in Notepad, ready for editing.

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

6.3 Edit the Sample Configuration File Sections

Note: It is important that you plan your configuration before modifying the configuration files. The remainder of this step provides the information to make the appropriate modifications to the configuration files.

Important: This module supports a maximum configuration file size of 128 kilobytes (131072 bytes). If the configuration file is larger than this size, the module will not accept the download. You can reduce the size of the configuration file by opening the file in a text editor and removing comment lines (lines preceded with the # character).

The LNG.CFG file has the following main sections:

- [Module]
- [Backplane Configuration]
- [LNG Client 0]
- [LNG Client 1]
- [I/O Chassis]

Important notes to consider when editing the sample configuration file:

- Comments within the file are preceded by the pound (#) sign. Any text on a line that occurs after the # character will be ignored.
- Do not use tabs or other non-printing characters instead of spaces to separate parameters (spacebar).
- Parameter names must begin in the first column of a line, and may not be preceded with a space (spacebar) or other non-printing character.

6.3.1 [Module]

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

The following example shows a sample [Module] section:

```
[Module]
Module Type : ProTalk-LNG
Module Name : Test Example of ProTalk-LNG Communication Module
```

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

Module Type Parameter

```
Module Type          : PTQ-LNG
```

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

Module Name Parameter

```
Module Name : PTQ-LNG COMMUNICATION MODULE
```

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

6.3.2 [Backplane Configuration]

```
[Backplane Configuration]
#These values are required to define the data area to transfer between the module
#and the processor.
Read Register Start  :   600  #Database start register to move to processor.
#(LNG fix at 600)
Read Register Count  :   400  #Number of words moved from module to processor.
#(LNG fix at 400)
Write Register Start :     0  #Database start register where data placed from
#processor. (LNG fix at 0)
Write Register Count :   600  #Number of words moved from processor to module.
#(LNG fix at 600)

#Used to define the area in the Processor for the module to interface with
3x Register Start :      100     #3x start register where data moved from module
#to processor (1-n)
4x Register Start :     1000     #4x start register where data moved from
#processor to module (1-n)

Initialize Output Data        :   No  #Initialize the database file on startup
Failure Flag Count            :    0  #Determines module operation if BP fails
#0=continue,>0=number of retries before comm shutdown
Error/Status Block Pointer    : 6000  #Number of register to store error and
#status
```

Read Register Start

```
Read Register Start  :  600 #Starting DB address where read data stored
```

The Read Register Start parameter assigns the starting address for data to send to the processor.

Read Register Count

```
Read Register Count       :   600  #Number of registers to write to processor
```

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

Write Register Start

```
Write Register Start :    0 #Database start register where data placed from
                            #processor
```

The Write Register Start parameter assigns the starting address for data to retrieve from the processor.

Write Register Count

```
Write Register Count      :   600  #Number of registers to read from processor
```

This parameter specifies the number of registers to be transferred from the module to the processor. Valid entry for this parameter is 0 to 3999

3x Register Start

1 to n

This parameter sets the first register in the processor where the data transferred from the module to the processor will be placed.

4x Register Start

1 to n

This parameter sets the first register in the processor where the data transferred from the processor to the module is present.

Initialize Output Data

Yes or No

This parameter determines if the output data for the module should be initialized with values from the processor. If the value is set to 0, the output data will be initialized to 0. If the value is set to 1, the data will be initialized with data from the processor. Use of this option requires associated ladder logic to pass the data from the processor to the module.

Failure Flag Count

```
Failure Flag Count     :     0  #Determines module operation if BP fails
                                #0=continue,>0=number of retries before comm
                                #shutdown
```

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, communications will cease if the specified number of failures occur. Valid values are 0 to 65535.

Error/Status Block Pointer

```
Error/Status Block Pointer  : 3500 # Number of register to
                                   # store error and status
```

The Error/Status Block Pointer parameter is used to specify which database register the error and status data will be stored in. Refer to the chapter on Status Data Definition for more information.

6.3.3 [LNG Config]

This section is used to define the LNG's configuration data.

```
[LNG Config]
Analog Input Count     :   300  #Determines the number of Analog Inputs available
                                #(0 - 300)
Digital Input Count    :    50  #Determines the number of Digital Inputs
                                #available (0 - 50)
Accumulator Count      :    50  #Determines the number of Accumulators available
                                #(0 - 50)
Indication Point       :    50  #Determines the number of Indication Points
                                #available (0 - 50)
SBO Timer Base         :     3  #SBO Duration Timer Base (0 = 10ms, 1 = 100ms,
                                #2 = 1s, 3 = 10s)
SBO Select Time        :     0  #SBO Select Time (0-32767)
```

Analog Input Count

```
Analog Input Count        : 300 #Determines the number of Analog Inputs available
                                #(0 - 300)
```

This value represents the number of Analog input data points that will be transferred to the LNG module. Valid values are 0 to 300 points.

Digital Input Count

```
Digital Input Count       : 50 #Determines the number of Digital Inputs
                                #available (0 - 50)
```

This value represents the number of 16-bit digital input words that will be transferred to the LNG module. Valid values are 0 to 50 words.

Accumulator Count

```
Accumulator Count         : 50 #Determines the number of Accumulators available
                                #(0 - 50)
```

This value represents the number of accumulator registers that will be transferred to the LNG module from the Quantum hardware. Values are 0 to 50.

Indication Point

```
Indication Point          : 50 #Determines the number of Indication Points
                                #available (0 - 50)
```

This value represents the number of Contact Status indication words that will be transferred to the LNG module. It is recommended that this be set no higher than necessary to improve processing time. Valid values are 0 to 50.

SBO Timer Base

```
SBO Timer Base             : 3 #SBO Duration Timer Base (0 = 10ms, 1 = 100ms,
                                #2 = 1s, 3 = 10s)
```

SBO Timer Duration is multiplied by the Timer Base

0 = 10 milliseconds
1 = 100 milliseconds
2 = 1 second
3 = 10 seconds

```
SBO Select Time            : 0 #SBO Select Time (0-32767)
```

SBO Time Select

This is the time in milliseconds that a select is valid for an SBO port. This value is used for all SBO ports.

6.3.4 [LNG Port #]

The following shows a sample LNG Port 1 section of the configuration file.

```
[LNG Port 1]
Enable                 :   Yes  #No=Port Disabled,Yes=Port Enabled
Baudrate               :  1200  #Baudrate for port (110 to 115.2K)
Parity                 :  None  #N=None,O=Odd,E=Even,M=Mark,S=Space
Data Bits              :     8  #5 to 8 data bits for messages
Stop Bits              :     1  #1 or 2 stop bits for messages
RTS On                 :    50  #Delay after RTS set before message sent (mSec)
RTS Off                :     2  #Delay after message before RTS dropped(mSec)
Minimum Command Delay  :     1  #Minimum number of msec's between commands
Use CTS Line           :    No  #Monitor CTS modem line (Y/N)
Slave ID               :     1  #1-255 Modbus Node Address (Slave)
```

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

Enable

```
Enable                 :   Yes  #No=Port Disabled,Yes=Port Enabled
```

This parameter defines if the port will be used. If the parameter is set to No, the port is disabled. A value of Yes enables the port.

Baud Rate

```
Baudrate               :  1200  #Baudrate for port (110 to 115.2K)
```

This is the baud rate to be used for the port. Enter the baud rate as a value. Baud rate entries are 110, 150, 300, 600, 1200, 2400, 4800, 9600, 19200, 28800, 57600, 115200.

Parity

```
Parity                 :  None  #N=None,O=Odd,E=Even,M=Mark,S=Space
```

This is the parity code to be used for the port. Values are None, Odd, or Even.

Data Bits

```
Data Bits              :     8  #5 to 8 data bits for messages
```

This parameter sets the number of bits for each word used by the protocol. Valid values are 5 to 8.

Stop Bits

```
Stop Bits               :     1  #1 or 2 stop bits for messages
```

This parameter sets the number of stop bits to be used with each data value sent. Valid values are 1 or 2.

RTS On

```
RTS On                  :    50  #Delay after RTS set before message sent (mSec)
```

This parameter sets the number of milliseconds to delay after RTS is asserted before the data will be transmitted. Valid values are 0 to 65535 milliseconds.

RTS Off

```
RTS Off                 :     2  #Delay after message before RTS dropped(mSec)
```

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. Valid values are 0 to 65535 milliseconds.

Minimum Command Delay

```
Minimum Command Delay   :     1  #Minimum number of msec's between commands
```

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. Valid values are 0 to 65535 milliseconds.

Use CTS Line

```
Use CTS Line            :    No  #Monitor CTS modem line (Y/N)
```

This parameter specifies if the CTS modem control line is to be used. If the parameter is set to N, the CTS line is not monitored. If the parameter is set to Y, the CTS line will be monitored and must be high before the module will send data. Normally, this parameter is required when half-duplex modems are used for communication (2-wire).

Slave ID

```
Slave ID                :     1  #1-255 Modbus Node Address (Slave)
```

Slave address used by the host to access this port. Valid values are 0 to 255.

6.3.5 [I/O Chassis]

This section defines chassis. [I/O Chassis] indicates how many chassis have been used from [Chassis Number 0] to [Chassis Number 6]. Chassis numbers 1 through 6 follow the same format as the [Chassis Number 0] section.

```
# This section defines chassis on LNG
[IO Chassis]
Chassis Count           :     3  #Determine how many chassis is used (1 to 7)

[Chassis Number 0]
Chassis Number 0 Card Code 0  :   0  #Chassis Number 0 Card Code 0
Chassis Number 0 Card Code 1  :   1  #Chassis Number 0 Card Code 1
Chassis Number 0 Card Code 2  :   2  #Chassis Number 0 Card Code 2
Chassis Number 0 Card Code 3  :   3  #Chassis Number 0 Card Code 3
Chassis Number 0 Card Code 4  :   4  #Chassis Number 0 Card Code 4
Chassis Number 0 Card Code 5  :   5  #Chassis Number 0 Card Code 5
Chassis Number 0 Card Code 6  :   7  #Chassis Number 0 Card Code 6
Chassis Number 0 Card Code 7  :   8  #Chassis Number 0 Card Code 7
Chassis Number 0 Card Code 8  :  11  #Chassis Number 0 Card Code 8
Chassis Number 0 Card Code 9  :  12  #Chassis Number 0 Card Code 9
Chassis Number 0 Card Code 10 :  15  #Chassis Number 0 Card Code 10
Chassis Number 0 Card Code 11 :  28  #Chassis Number 0 Card Code 11
Chassis Number 0 Card Code 12 :  29  #Chassis Number 0 Card Code 12
Chassis Number 0 Card Code 13 :  30  #Chassis Number 0 Card Code 13
Chassis Number 0 Card Code 14 :  31  #Chassis Number 0 Card Code 14
Chassis Number 0 Card Code 15 :   0  #Chassis Number 0 Card Code 15
```

[Section]/Item	Range	Description
[IO Chassis]	1 to 7	Determine how many chassis is used (1 to 7)
[Chassis Number 0]		Indicating Chassis Number 0
Chassis Number 0 Card Code 0		Indicate first card for Chassis Number 0
Chassis Number 0 Card Code 1		Indicate second card for Chassis Number 0
Chassis Number 0 Card Code 2		Indicate third card for Chassis Number 0
Chassis Number 0 Card Code 3		Indicate fourth card for Chassis Number 0
Chassis Number 0 Card Code 4		Indicate fifth card for Chassis Number 0
Chassis Number 0 Card Code 5		Indicate sixth card for Chassis Number 0
Chassis Number 0 Card Code 6		Indicate seventh card for Chassis Number 0
Chassis Number 0 Card Code 7		Indicate eighth card for Chassis Number 0
Chassis Number 0 Card Code 8		Indicate ninth card for Chassis Number 0
Chassis Number 0 Card Code 9		Indicate tenth card for Chassis Number 0
Chassis Number 0 Card Code 10		Indicate eleventh card for Chassis Number 0
Chassis Number 0 Card Code 11		Indicate twelve card for Chassis Number 0
Chassis Number 0 Card Code 12		Indicate thirteenth card for Chassis Number 0
Chassis Number 0 Card Code 13		Indicate fourteenth card for Chassis Number 0
Chassis Number 0 Card Code 14		Indicate fifteenth card for Chassis Number 0
Chassis Number 0 Card Code 15		Indicate sixteenth card for Chassis Number 0

6.3.6 Special Functions

Block 9992 – Select Before Operate

If the module retrieves a BLOCK ID of 9992 from the PLC after it is issued the Command Function 22 by a master, it will place the SBO input data contained within the block into the event buffer and alter the data values for the points in the database. Using the example data buffer of 300101 to 300164 described above the contents of the block would look as follows:

Word Offset in Block	Example Address	Data	Description
0	300100	0	If it completed successfully. Counter increments.
1	300101	Block ID	9992
2	300102	SBO Command	This filed contains the number command selected.
3	300103	SBO Bit	This field contains the point for operation.
4	300104	SBO Operate	This field indicated either the bit is trip or close
5	300105	SBO Duration	This field is used as a multiplier to determine how long to activate the selected point.

In your table, word zero will contain a value of zero, word one will contain the BLOCK ID code, and word two and so on will contain the data.

Block 9993 – Pulse Output

If the module retrieves a BLOCK ID of 9993 from the PLC after it is issued the Command Function 25 by a master, it will place the pulse output data contained within the block into the event buffer and alter the data values for the points in the database. Using the example data buffer of 300100 to 300199 described above the contents of the block would look as follows:

Word Offset in Block	Example Address	Data	Description
0	300100	0	If it completed successfully. A counter will increment.
1	300101	Block ID	9993
2	300102	Pulse Raise 10ms Control	Control point Raise at 10ms
3	300103	Pulse Raise 10ms	Pulse Raise at 10ms
4	300104	Pulse Lower 10ms Control	Control point Lower at 10ms
5	300105	Pulse Lower 10ms	Pulse Lower at 10ms
6	300106	Pulse Raise 100ms Control	Control point Raise at 100ms
7	300107	Pulse Raise 100ms	Pulse Raise at 100ms

Word Offset in Block	Example Address	Data	Description
8	300108	Pulse Lower 100ms Control	Control point Lower at 100ms
9	300109	Pulse Lower 100ms	Pulse Lower at 100ms
10	300110	Pulse Raise 1s Control	Control point Raise at 1s
11	300111	Pulse Raise 1s	Pulse Raise at 1s
12	300112	Pulse Lower 1s Control	Control point Lower at 1s
13	300113	Pulse Lower 1s	Pulse Lower at 1s
14	300114	Pulse Raise 10s Control	Control point Raise at 10s
15	300115	Pulse Raise 10s	Pulse Raise at 10s
16	300116	Pulse Lower 10s Control	Control point Lower at 10s
17	300117	Pulse Lower 10s	Pulse Lower at 10s
18	300118	Pulse Point	This field contain number of point selected.

In your table, word zero will contain a value of zero, word one will contain the BLOCK ID code, and word two and so on will contain the data.

Block 9994 – Pulse Train Output

If the module retrieves a BLOCK ID of 9994 from the PLC after it is issued the Command Function 26 by a master, it will place the pulse train output data contained within the block into the event buffer and alter the data values for the points in the database. Using the example data buffer of 300100 to 300199 described above the contents of the block would look as follows:

Word Offset in Block	Example Address	Data	Description
0	300100	0	If it completed successfully. A counter will increment.
1	300101	Block ID	9994
2	300102	Pulse Train Operate	This field indicated that the point number shown was in operation.
3	300103	Pulse Train Point	This field contains the value of point controlled.
4	300104	Pulse Train Command	This field indicated either the bit is trip or close.
5	300105	Pulse Train On Duration	Number of on duration in ms.
6	300106	Pulse Train Off Duration	Number of off duration in ms

In your table, word zero will contain a value of zero, word one will contain the BLOCK ID code, and word two and so on will contain the data.

Block 9995 – Time Synchronization

If the module retrieves a BLOCK ID of 9995 from the PLC after it is issued the Command Function 32 by a master, it will place the date and time data contained within the block into the event buffer and alter the data values for the points in the database. Using the example data buffer of 300100 to 300199 described above the contents of the block would look as follows:

Word Offset in Block	Example Address	Data	Description
0	300100	0	If it completed successfully. A counter will increment.
1	300101	Block ID	9995
2	300102	Month	This field contains the month value that is set.
3	300103	Day	This field contains the day value is set.
4	300104	Hour	This field contains the hour value is set.
5	300105	Minute	This field contains the minute value is set.
6	300106	Second	This field contains the second value is set.
7	300107	MSec	This field contains the millisecond value is set.

In your table, word zero will contain a value of zero, word one will contain the BLOCK ID code, and word two and so on will contain the data.

6.3.7 Uploading and Downloading the Configuration File

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

This section describes these procedures.

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

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 serial communications port available
- A null modem serial cable.

Required Software

In order to send and receive data over the serial port (COM port) on your computer to the module, you must use a communication program (terminal emulator).

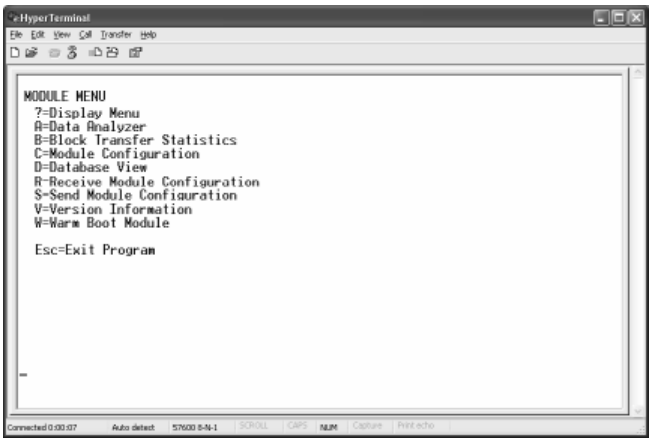
A simple communication program called HyperTerminal is pre-installed with recent versions of Microsoft Windows operating systems. If you are connecting from a machine running DOS, you must obtain and install a compatible communication program. The following table lists communication programs that have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

DOS	ProComm, as well as several other terminal emulation programs
Windows 3.1	Terminal
Windows 95/98	HyperTerminal
Windows NT/2000/XP	HyperTerminal

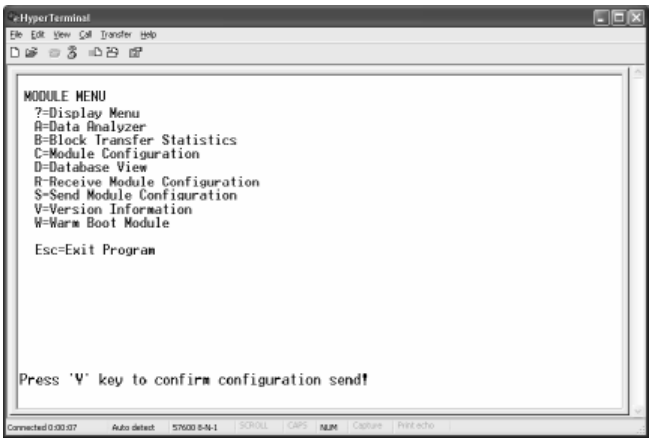
The module uses the Ymodem file transfer protocol to send (upload) and receive (download) configuration files from your module. If you use a communication program that is not on the list above, please be sure that it supports Ymodem file transfers.

Transferring the Configuration File to Your PC

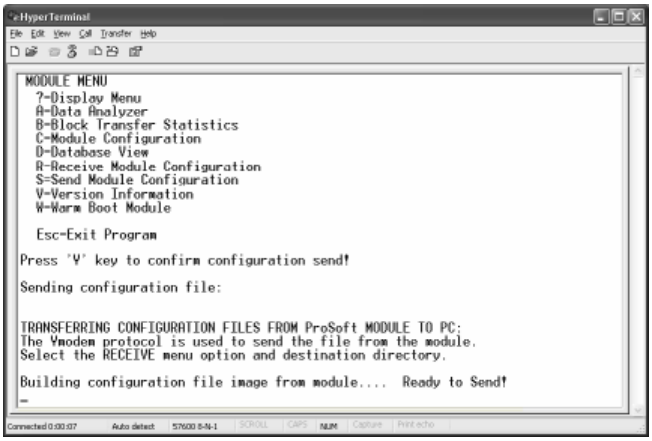
1 Connect your PC to the Configuration/Debug port of the module using a terminal program such as HyperTerminal. Press **[?]** to display the main menu.



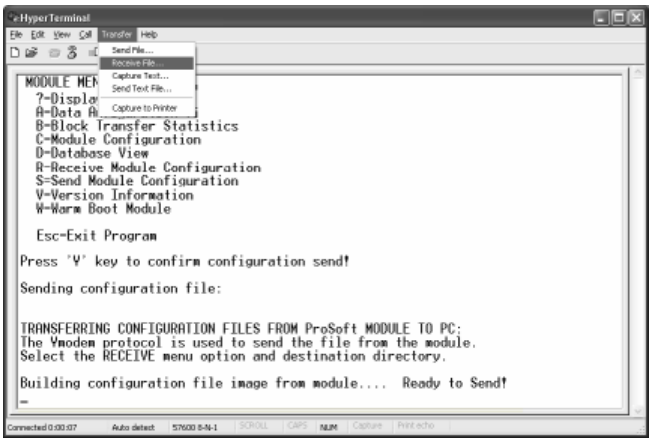
2 Press **[S]** (Send Module Configuration). The message "Press Y key to confirm configuration send!" is displayed at the bottom of the screen.



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



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

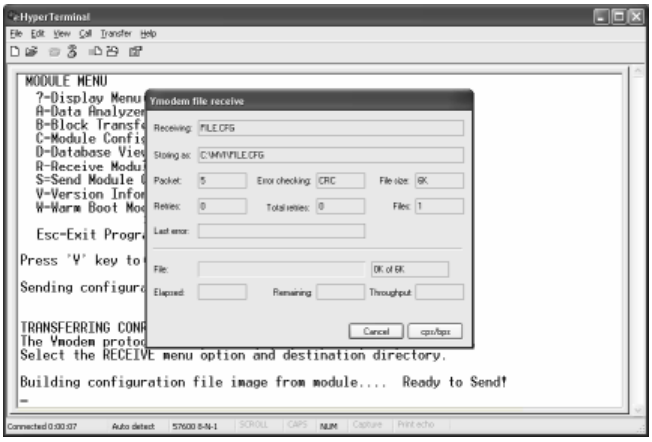


5 Use the Browse button to choose a folder on your computer to save the file, and then click Receive.

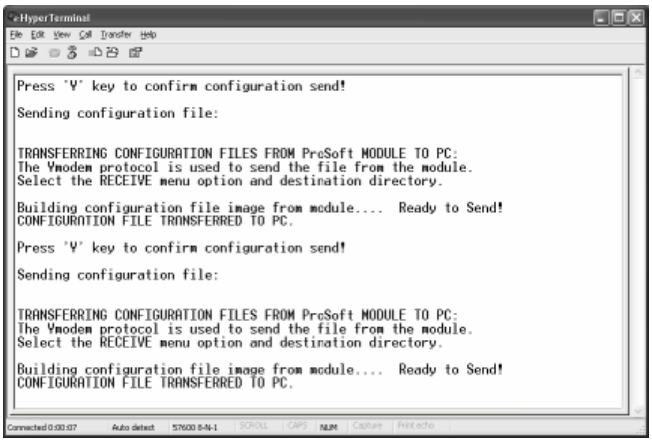
- **Note:** ProSoft Technology suggests that you upload the configuration file pre-loaded on your module. However, configuration files are also available on the ProSoft CD as well as the ProSoft Technology web site at http://www.prosoft-technology.com.

6 Select Ymodem as the receiving protocol.

7 Click the Receive button. This action opens the Ymodem File Receive dialog box, showing the progress of your file transfer.



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



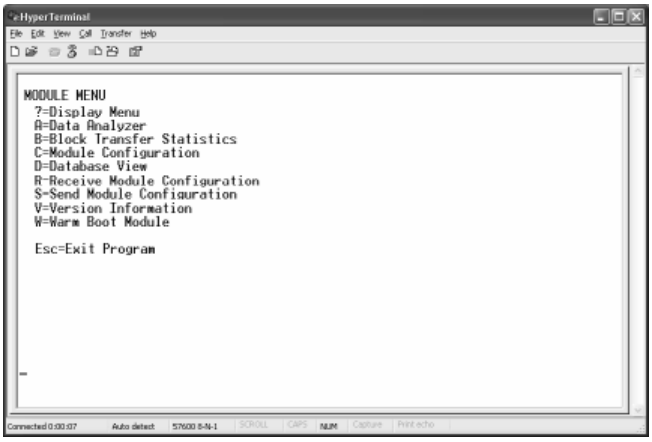
The configuration file is now on your PC at the location you specified.

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

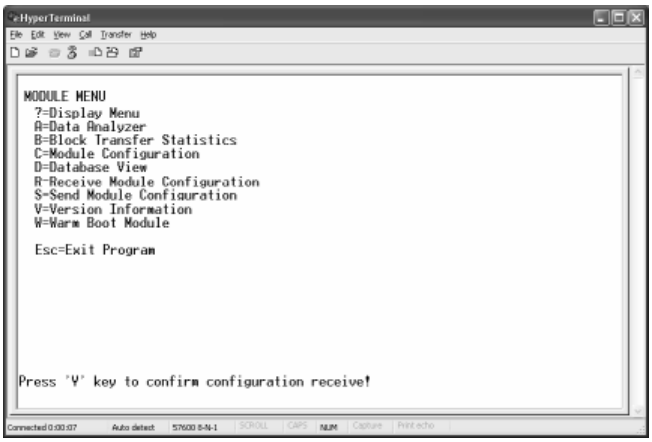
Transferring the Configuration File to the Module

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

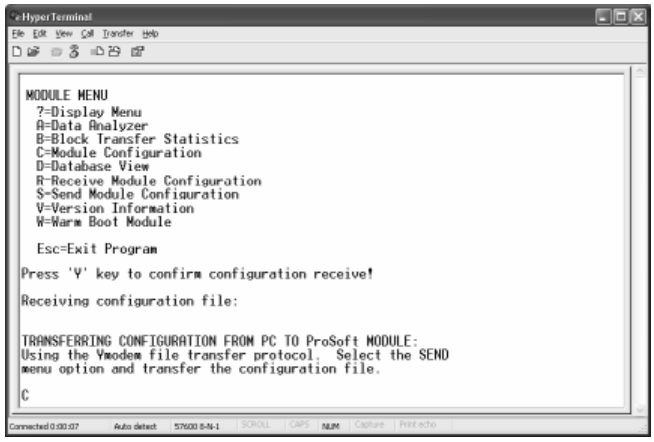
1 Connect your PC to the Configuration/Debug port of the module using a terminal program such as HyperTerminal. Press **[?]** to display the main menu.



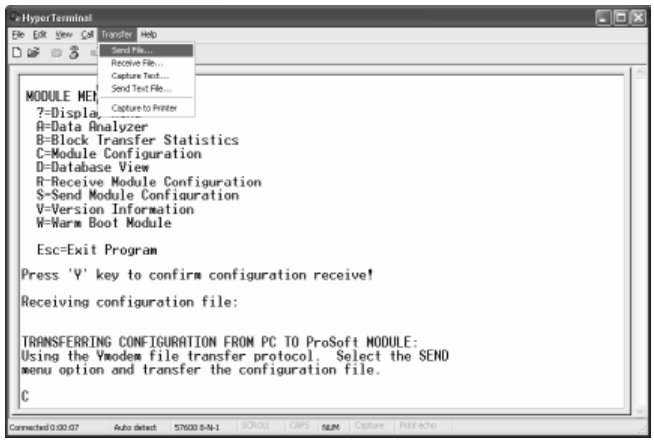
2 Press **[R]** (Receive Module Configuration). The message "Press Y key to confirm configuration receive!" is displayed at the bottom of the screen.



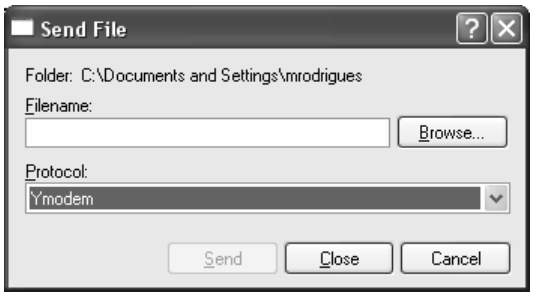
3 Press **[Y]**. The screen now indicates that the PC is ready to send.



4 From the **Transfer** menu in HyperTerminal, select **Send File**.



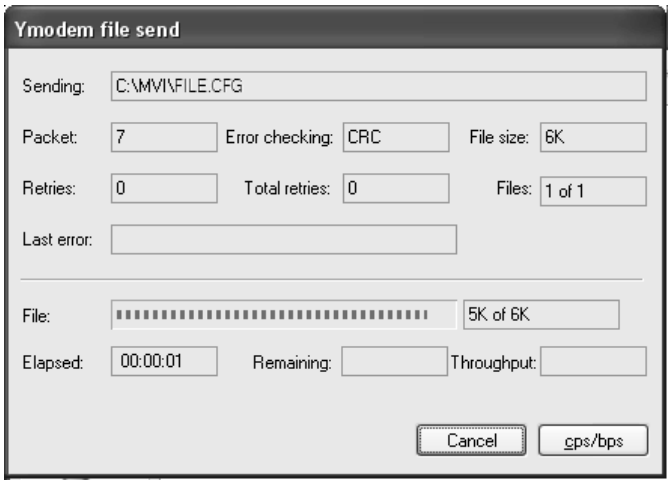
The Send File dialog appears.



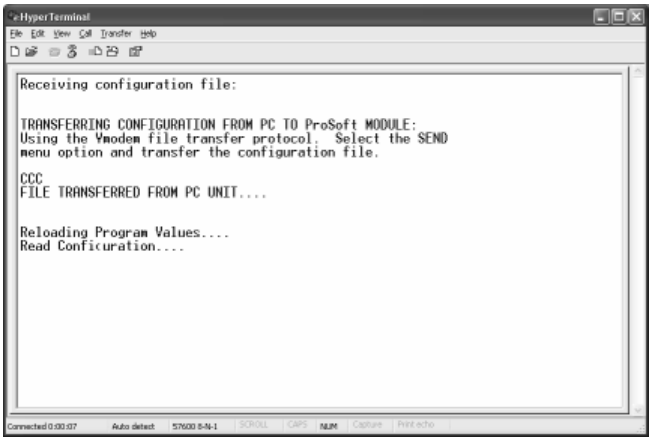
5 Use the Browse button to locate the configuration file your computer.

Note: This procedure assumes that you are uploading a newly edited configuration file from your PC to the module. However, configuration files are also available on the ProSoft CD as well as the ProSoft Technology web site at http://www.prosoft-technology.com.

6 Select **Ymodem** as the protocol.
7 Click the Send button. This action opens the Ymodem File Send dialog box.



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



8 Your module now contains the new configuration.

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

7.1.1 The Configuration/Debug Menu

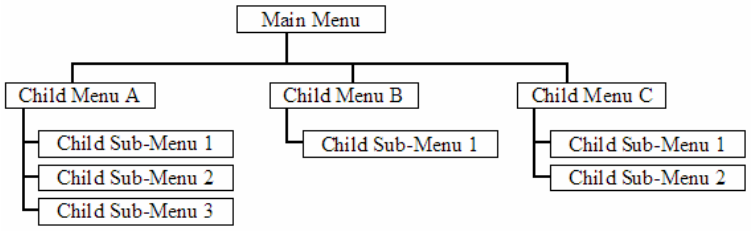
The Configuration and Debug menu for this module is arranged as a tree structure, with the Main Menu at the top of the tree, and one or more sub-menus for each menu command. The first menu you see when you connect to the module is the Main menu.

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

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.

7.1.2 Required Hardware

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

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

- 80486 based processor (Pentium preferred)
- 1 megabyte of memory
- At least one serial communications port available
- A null modem serial cable.

7.1.3 Required Software

In order to send and receive data over the serial port (COM port) on your computer to the module, you must use a communication program (terminal emulator).

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

communication program. The following table lists communication programs that have been tested by ProSoft Technology.

DOS	ProComm, as well as several other terminal emulation programs
Windows 3.1	Terminal
Windows 95/98	HyperTerminal
Windows NT/2000/XP	HyperTerminal

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

7.1.4 Using the Configuration/Debug Port

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

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

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

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

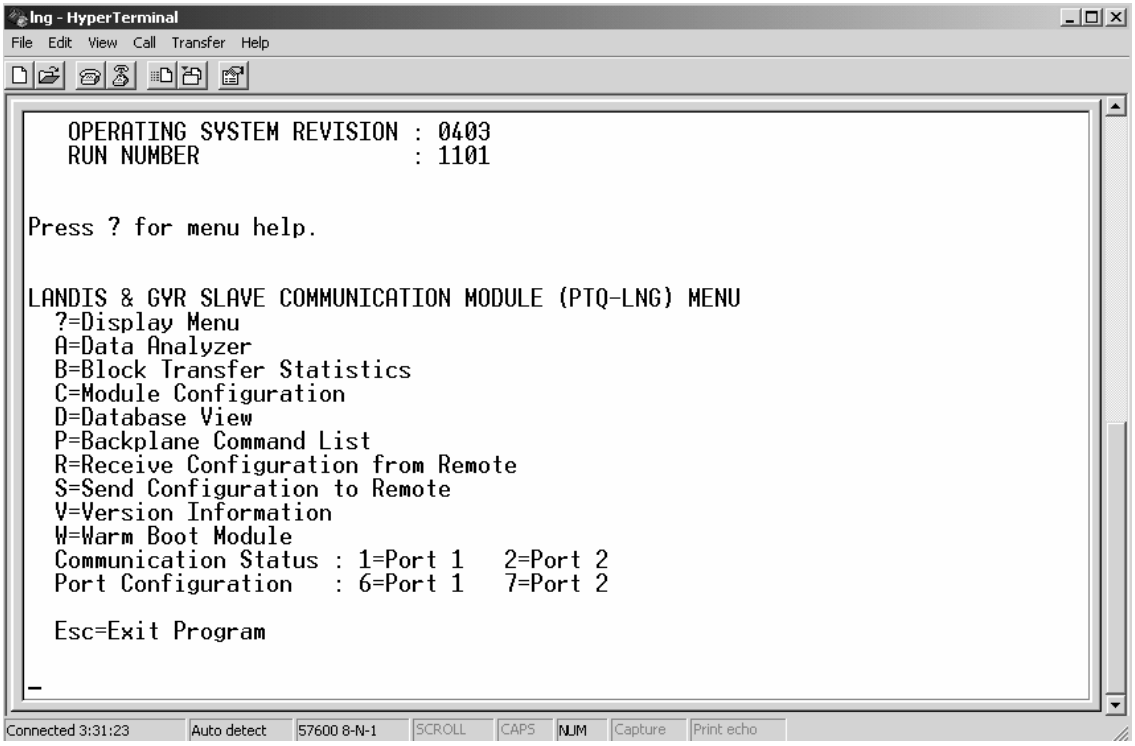
If there is no response from the module, follow these steps:

1. Verify that the null modem cable is connected properly between your computer's serial port and the module. A regular serial cable will not work.
2. Verify that RSLinx is not controlling the COM port. Refer to Disabling the RSLinx Driver for the Com Port on the PC.
3. Verify that your communication software is using the correct settings for baud rate, parity and handshaking.
4. On computers with more than one serial port, verify that your communication program is connected to the same port that is connected to the module.

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

7.1.5 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. The Data Analyzer section has more information about this menu.

Important: When in analyzer mode, program execution will slow down. Only use this tool during a trouble-shooting session. Before disconnecting from the Config/Debug port, please be sure to press **[M]** to return to the main menu and

disable the data analyzer. This action will allow the module to resume its normal 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: Repeat this command at one-second intervals to determine the number of blocks transferred each second.

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 Backplane Menu

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

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 be certain 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 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 trouble-shooting 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.

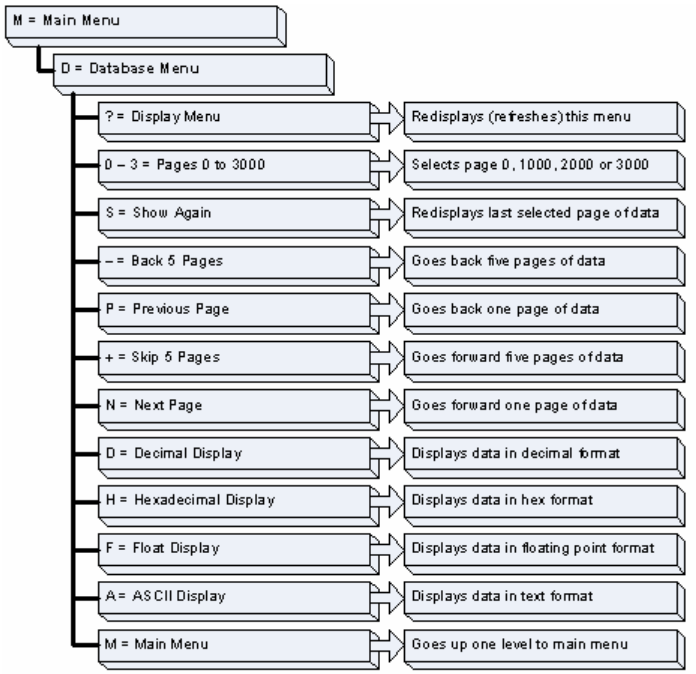
Exiting the Program

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.

7.1.6 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.



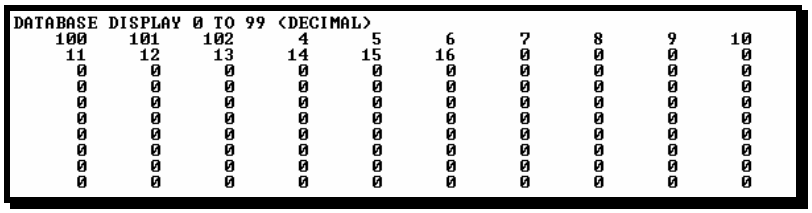
Viewing Register Pages

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



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

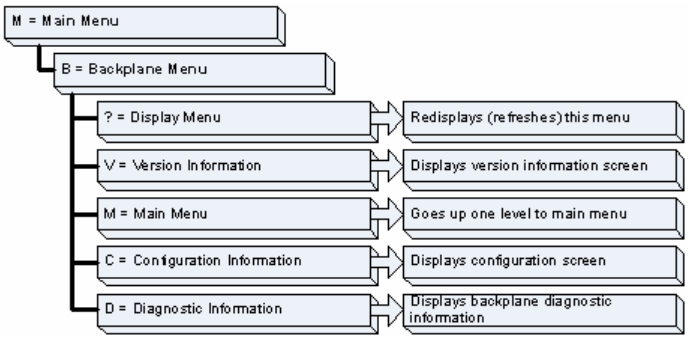
Press **[A]** to display the data on the current page in ASCII format. This is useful for regions of the database that contain ASCII data.

Returning to the Main Menu

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

7.1.7 Backplane Menu

Press **[B]** 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. 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.

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.

Returning to the Main Menu

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

Viewing Configuration Information

Press **[C]** to view configuration information for the selected port, protocol, driver or device.

Viewing Backplane Diagnostic Information

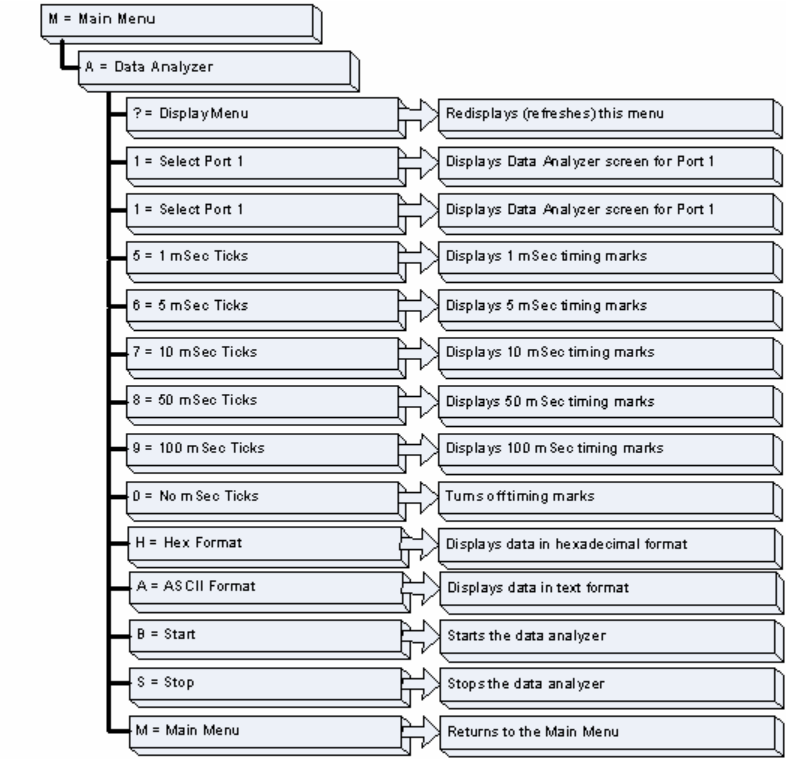
Press **[D]** to view Backplane Diagnostic information.

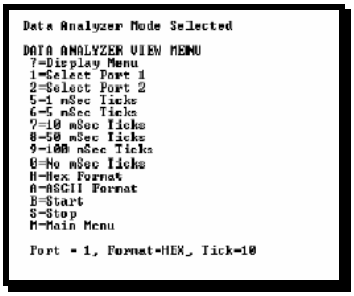
Use this command to display the configuration and statistics of the backplane data transfer operations between the module and the processor. The information on this screen can help determine if there are communication problems between the processor and the module.

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

7.1.8 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.

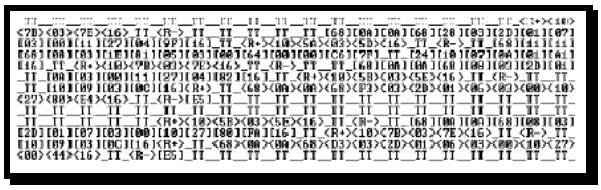




Important: When in analyzer mode, program execution will slow down. Only use this tool during a trouble-shooting session. Before disconnecting from the Config/Debug port, please be sure to press **[M]** to return to the main menu and disable the data analyzer. This action will allow the module to resume its normal operating mode.

Analyzing Data for Port 1

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



Analyzing Data for Port 2

Press **[2]** to display I/O data for Port 2 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 mSec ticks
[6]	5 mSec ticks
[7]	10 mSec ticks
[8]	50 mSec ticks
[9]	100 mSec ticks

Removing Timing Marks in the Data Analyzer

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

Viewing Data in Hexadecimal Format

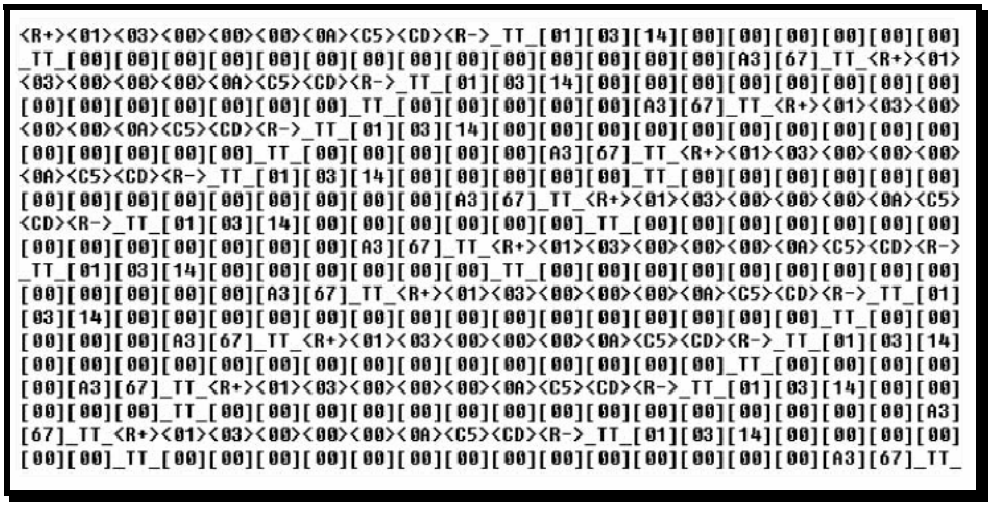
Press **[H]** to display the data on the current page in hexadecimal format.

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. An example display is shown below:



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 trouble-shooting session. Before disconnecting from the Config/Debug port, please be sure to press **[S]** to stop the Data Analyzer

before returning to the main menu or disconnecting from the port. This action will allow the module to resume its normal operating mode.

Returning to the Main Menu

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

7.1.9 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:

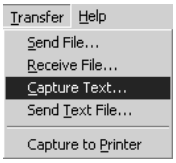
```
DATA ANALYZER VIEW MENU
 ?=Display Menu
 1=Select Port 1
 2=Select Port 2
 5=1 mSec Ticks
 6=5 mSec Ticks
 7=10 mSec Ticks
 8=50 mSec Ticks
 9=100 mSec Ticks
 0=No mSec Ticks
 H=Hex Format
 A=ASCII Format
 B=Start
 S=Stop
 M=Main Menu

 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.

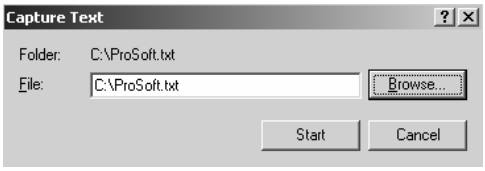
For most applications, HEX is the best format to view the data, and this does include ASCII based messages (because some characters will not display on HyperTerminal and by capturing the data in HEX, we can figure out what the corresponding ASCII characters are supposed to be).

The Tick value is a timing mark. The module will print a _TT for every xx mSec of no data on the line. Usually 10msec 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.

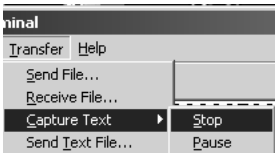
All characters shown in <> brackets are characters being sent out by the module.

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.

7.2 LED Status Indicators

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

ProTalk Module	Color	Status	Indication
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.
PRT1	Green	On	Port not used in application
		Off	Port not used in application
PRT2	Green	On	Port not used in application
		Off	Port not used in application
ERR1 CFG Fail	Red	Off	The PTQ-LNG is working normally.
		On	The PTQ-LNG module program has recognized an application error. This LED will also be turned on if any command presents an error.
ERR2 P1 Fail	N/A		Not used in application
ERR3 P2 Fail	Red	On	Configuration Error

ProTalk Module	Color	Status	Indication
Active	Green	On	The LED is on when the module is 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	Red	Off	The battery voltage is OK and functioning.
Low		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 the factory.

If your module is not operating, and the status LEDs are not illustrated in the table above, please call 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 ***PTQ-LNG Status Data Definition*** (page 87) for data block structure.

8 Reference

In This Chapter

8.1 Product Specifications

The Landis & Gyr Telegyr Slave Communication Module is a Quantum backplane compatible module that allows Quantum processors to interface easily with Telegyr Master host devices. SCADA systems supporting this application are commonly found in the power utility industry.

8.1.1 Features and Benefits

The PTQ-LNG module acts as a communication gateway between the Telegyr 8979 Rev. F version of the protocol and the Quantum backplane. The module functions as a Telegyr slave, receiving commands from the host. Data transfer between the module and the Quantum processor is asynchronous to the Telegyr network, with the module's internal database being used to exchange data between the Quantum processor and the Telegyr network.

The PTQ-LNG 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.

8.1.2 General Specifications

- Single Slot – Quantum backplane compatible
- The module is recognized as an Options module and has access to PLC memory for data transfer
- Configuration data is stored in non-volatile memory in the ProTalk module
- Configuration software for Microsoft Windows XP, 2000 and NT is included with the 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.
  - UnityPro XL
  - Concept
  - ProWORX
- Quantum data types supported: 3x, 4x
- High speed data transfer across backplane provides quick data update times.
- Sample ladder file available.

8.1.3 Hardware Specifications

Specification	Description
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)
LED Indicators	Module Status Backplane Transfer Status Serial Port Activity LED Serial Activity and Error LED Status
Configuration Serial Port (PRT1)	DB-9M PC Compatible RS-232 only No hardware handshaking
Application Serial Ports	(PRT2, PRT3) 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

8.1.4 Functional Specifications

The PTQ-LNG module supports the Landis & Gyr Telegyr 8979 Rev F slave protocol to the following specifications:

- Supports two serial ports emulating the protocol, each individually configurable for:
  - Slave Address
  - Communication parameters
  - Timing
- The module supports a database common to both serial ports. The supported point types and their maximum point counts are:
  - Binary Input: 800 points
  - Binary Output: 800 points
  - Analog Input: 300 points
  - Analog Output: 50 points

- Accumulators: 50 points
- Indication Points: 800 points

Supported Function Codes

Code	Description
0	Analog Change Report
1	Analog Force Report
2	Analog Group Change Report
3	Analog Group Force Report
5	ADC Reference Force Report
6	Indication Change Report
7	Indication Force Report
11	Digital Input Force Report
12	Accumulator Change Report
13	Accumulator Force Report
20	Analog Report
21	SBO Select
22	SBO Operate
23	Digital Output
24	Accumulator Freeze
25	Pulse Output
26	Pulse Train Output
30	Restart RTU
31	RTU Configuration
32	Time Synchronization
34	Analog Deadbands
35	Analog Group Define
36	Accumulator Preset
37	Continuation Request
38	Repeat Last Message
39	Firmware Configuration

8.2 PTQ-LNG Functionality Overview

This section gives the reader a functional overview of the PTQ-LNG module.

A thorough understanding of the information contained in this document is required for successful implementation of the module in a user application. If you already understand the content of this section, refer to the **Module Configuration** section to install and configure the module. If you are not familiar with the data transfer method used by the module, read this section before installing and configuring the module.

8.2.1 General Concepts

The following discussion explains several concepts that are important for understanding the operation of the PTQ-LNG module.

Module Power Up

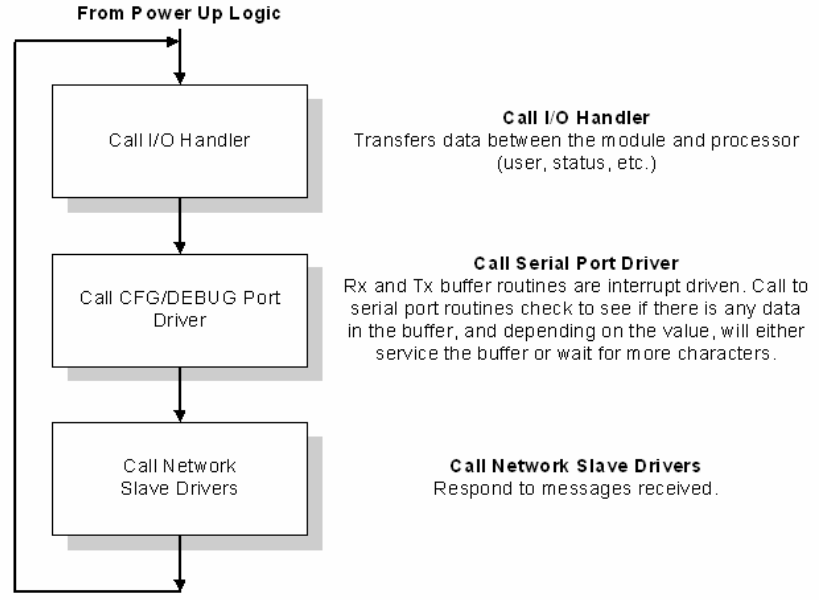
On power up, the module begins performing the following logical functions:

1 Read configuration for module from LNGS.CFG file from internal EPROM
2 Initialize module register space
3 Enable slave driver
4 Initialize all serial communication ports

After the module receives the configuration, it begins communicating with Telegyr devices on the network.

Main Logic Loop

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



Backplane Data Transfer

The PTQ-LNG module is unique in the way that the Quantum backplane is used. All data for the module is contained in the module's internal database. Data is moved between the module and the Quantum processor across the backplane using the 3x Register Start and 4x Register Start parameters in the module configuration file.

PTQ-LNG Database Map

The following table shows the layout of the database.

Start Address	End Address	Data	Quantum Address
0	299	Analog Inputs	401064 to 401363
300	349	Digital Inputs	401364 to 401413
350	399	Accumulators	401414 to 401463
400	449	Indication Points	401464 to 401513
600	649	Analog Outputs	300164 to 300213
650	699	Digital Outputs	300214 to 300263
700	749	Accumulator Presets	300264 to 300313
750		Pulse Raise 10ms Control	300314
751	759	Pulse Raise 10ms	300315 to 300323
760		Pulse Lower 10ms Control	300324
761	769	Pulse Lower 10ms	300325 to 300333
770		Pulse Raise 100ms Control	300334
771	779	Pulse Raise 100ms	300335 to 300343
780		Pulse Lower 100ms Control	300344
781	789	Pulse Lower 100ms	300345 to 300353
790		Pulse Raise 1s Control	300354
791	799	Pulse Raise 1s	300355 to 300363
800		Pulse Lower 1s Control	300364
801	809	Pulse Lower 1s	300365 to 300373
810		Pulse Raise 10s Control	300374
811	819	Pulse Raise 10s	300375 to 300383
820		Pulse Lower 10s Control	300384
821	829	Pulse Lower 10s	300385 to 300393
850		SBO Command	300414
851		SBO Bit	300415
852		SBO Operate	300416
853		SBO Duration	300417
860		PTO Active	300424
861		PTO Point	300425
862		PTO Command	300426
863		PTO Count	300427
864		PTO On Duration	300428
865		PTO Off Duration	300429
2000		RTU Config I/O Chassis	Internal Only
2001		Chassis Number	Internal Only
2002	2017	Card Code	Internal Only
2018		Chassis Number 1	Internal Only
2019	2034	Card Code	Internal Only

Start Address	End Address	Data	Quantum Address
2035		Chassis Number 2	Internal Only
2036	2051	Card Code	Internal Only
2052		Chassis Number 3	Internal Only
2053	2068	Card Code	Internal Only
2069		Chassis Number 4	Internal Only
2070	2085	Card Code	Internal Only
2086		Chassis Number 5	Internal Only
2087	2102	Card Code	Internal Only
2103		Chassis Number 6	Internal Only
2104	2119	Card Code	Internal Only
5000		Configuration	Internal Only
5010	5021	Port 1 Configuration	Internal Only
5030	5041	Port 2 Configuration	Internal Only
6000	6021	Error / Status Table	Internal Only

Note: According to the protocol specification, the accumulators may assume values of up to 65535. Because the PTQ-LNG module displays the database values as a signed integer (-32767 to 32767), the accumulator values might not be displayed as expected by the user. The user should display the accumulator values in the database in hex format.

8.2.2 Module Control Blocks

As discussed in the previous section, range 4000 to 4999 in the internal database controls the module in order to perform specific tasks. These tasks are described in the following topics. Word 4000 contains the block ID that identifies the block to the PTQ-LNG module. The block structure is different for each block.

8.3 Configuration Data Definition

This section contains a listing of the parameters and addresses that are available in the PTQ-LNG database.

Register	Range	Parameter
5000	0 to 300	Number of Analog Inputs
5001	0 to 50	Number of Digital Input Words (16 points per word)
5002	0 to 50	Number of Accumulators
5003	0 to 50	Number if Indication Point Words
5004	0 to 65535	Backplane Fail Counter
5005	0 to 3	SBO Timer Base
5006	0 to 32767	SBO Select Time
5010	0, 1	Port 1 Enable
5011		Port 1 Baud Rate
5012	0 to 2	Port 1 Parity

Register	Range	Parameter
5013	5 to 8	Port 1 Data Bits
5014	1, 2	Port 1 Stop Bits
5015	0 to 65535	Port 1 RTS On (mSec)
5016	0 to 65535	Port 1 RTS Off (mSec)
5017	0 to 65535	Port 1 Minimum Response Delay (mSec)
5018	0, 1	Port 1 Use CTS (Y, N)
5019	0 to 255	Port 1 Slave ID
5030	0, 1	Port 2 Enable
5031		Port 2 Baud Rate
5032	0 to 4	Port 2 Parity
5033	5 to 8	Port 2 Data Bits
5034	1, 2	Port 2 Stop Bits
5035	0 to 65535	Port 2 RTS On (mSec)
5036	0 to 65535	Port 2 RTS Off (mSec)
5037	0 to 65535	Port 2 Minimum Response Delay (mSec)
5038	0, 1	Port 2 Use CTS (Y, N)
5039	0 to 255	Port 2 Slave ID

8.4 PTQ-LNG Status Data Definition

This section contains a listing of the data contained in the PTQ-LNG status data object.

Register	Content	Description
6000	Program cycle counter	This value is incremented each time a complete program cycle occurs in the module.
6001 to 6002	Product name as ASCII string	This register contains the product code of "LNG"
6003 to 6004	Revision level as ASCII string	This register contains the product version for the current software.
6005 to 6006	Operating system level as ASCII string	This register contains the month and year values for the program operating system.
6007 to 6008	Run number as ASCII string	This register contains the run number value for the current software.
6009	Number of requests - Port 1	Contains the number of messages received on the port.
6010	Number of responses - Port 1	Contains the total number of messages sent out of the port.
6011	Number of errors sent - Port 1	Contains the total number of message errors sent out of the port.
6012	Number of errors received - Port 1	Contains the total number of message errors received on the port.
6013	Number of requests - Port 2	Contains the number of messages received on the port.

Register	Content	Description
6014	Number of responses - Port 2	Contains the total number of messages sent out of the port.
6015	Number of errors sent - Port 2	Contains the total number of message errors sent out the port.
6016	Number of errors received - Port 2	Contains the total number of message errors received on the port.
6017	Backplane Transfer Writes	Contains the total number of write blocks transferred from the processor to the module.
6018	Backplane Transfer Reads	Contains the total number of read blocks transferred from the module to the processor.
6019	N/A	
6020	N/A	
6021	N/A	
6022	Backplane Transfer Errors	Contains the total number of block errors recognized by the module.
6023	Current Error - Port 1	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.
6024	Last Error - Port 1	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 the error.
6025	Current Error - Port 2	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 current executing command.
6026	Last Error - Port 2	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.

A data file should be allocated in the ladder logic to hold this block of data. The size of the block of data is 33 words..

8.5 PTQ-LNG Command Descriptions

The following table provides information on supported commands.

Command	Command Description
Analog Change Report - 0	This command is implemented in the module per the protocol specification. Up to 300 analog values are supported by the module.
Analog Force Report - 1	This command is implemented in the module per the protocol specification. Up to 300 analog values are supported by the module.
Analog Group Change Report - 2	This command is implemented in the module per the protocol specification. Up to 300 analog values are supported by the module.
Analog Group Force Report - 3	This command is implemented in the module per the protocol specification. Up to 300 analog values are supported by the module.

Command	Command Description
ADC Reference Force Report - 5	This command returns hardcoded values to the host for one ADC point. The following values are returned for the -90%, 0% and 90% respectively, 205, 2048, and 3890.
Indication Change Report - 6	This command returns the status and memory bit values for the indication points that have had a change since the last host request. This data is derived from the data placed in the module when the indication point section of the ControlLogix processor is read. The actual logic to set and clear the indication bits must be performed by the ladder logic. Indication point data must be written to the ControlLogix when the ladder program detects an indication change. The data block in the ControlLogix processor is read in segments controlled by an internal block counter. The module will cycle through the block reading the entire data area. The indication point section of the database is read each time in addition to the normal data block. The indication block is tested for changes immediately after being read. When a change is found, the status bit corresponding to the indication point is updated and the memory bit is set if there has been more than one change since the last host query. This update method provides for an indication point update time of approximately 5 - 10ms depending on how many indication words are defined. The update time is lower with fewer points defined. Therefore the number of words of indication points should not be set any higher than necessary. When the host requests the indication point data, the change data is cleared in the module. At this time, the functionality to switch from the SOE mode to the Time based mode is not implemented. **Note:** If both ports are receiving indication change report requests, the first master to send a request will see change. The change is cleared after either master performs a read. In order for both masters to see the new data, both masters should use the indication force report.
Indication Force Report - 7	This command returns the status and memory bit values for the indication points that are requested by the host. This data is derived from the data placed in the module when the indication point section is transferred to the ControlLogix. The actual logic to set and clear the indication bits must be performed by the ladder logic. Indication point data must transferred to the ControlLogix when the ladder program detects an indication change. At this time, the functionality to switch from the SOE mode to the Time based mode is not implemented.
Digital Input Force Report - 11	This command is implemented in the module per the protocol specification. 800 digital input values are supported by the module. The module returns values for the specific range of digital input blocks. Each block contains 16 binary inputs. If the stop point number is larger that the largest actual block number in the module, only values for actual blocks are reported. If the stop point number is less than the start point number, an error message is returned.

Command	Command Description
Accumulator Change Report - 12	This command is implemented in the module per the protocol specification. Up to 50 accumulators are supported by the module. This command returns the last frozen value for each accumulator if the freeze flag is set. Use Function 26 to set the freeze flag. The freeze flag is reset by this function.
Accumulator Force Report - 13	This command is implemented in the module per the protocol specification. Up to 50 accumulators are supported by the module. The module reports the last frozen value for each accumulator in the specified range. If the stop point number is greater than the largest actual point number, only values for the actual points are returned. If the stop point number is less than the start point number, an error message is returned. The freeze flag is reset by this function.
Analog Output - 20	This command is implemented in the module per the protocol specification. Up to 50 analog values are supported by the module.
SBO Select - 21	When an SBO select command is received from the host, the point is selected and the timer is started. The point will remain selected until the point is operated upon, the timer expires, or another point is selected. The select time is determined by the SBO Duration value and the SBO Base value. SBO Time = SBO Base * SBO Duration.
SBO Operate - 22	The SBO Operate command will cause a previously selected point to be operated. An error response will be generated if the point has not been selected. There is no real limit on the number of SBO points that can be addressed by the host. The ladder logic implemented as part of the application will determine if a write command will move into the processor data table.
Digital Output - 23	This command is implemented in the module per the protocol specification, except that the upper 8 bits of the 24 bit write are disregarded. This allows 50 words of 16 bit digital output data to be addressed.
Accumulator Freeze - 24	This command is implemented in the module per the protocol specification. Up to 50 analog values are supported by the module.
Pulse Output - 25	This command is implemented in the module per the protocol specification. The module decodes the write command and moves the data to the processor in a series of data structures depending on the type and time base of the command. Through these data structures, support is provided to address 9 pulse points.
Pulse Train Output - 26 (Hold)	This command is not implemented at this time.
Restart RTU - 30	This command is implemented in the module per the protocol specification, performing a cold boot.
RTU Configuration - 31	This command is implemented in the module to support one rack of configuration data. The configuration data is moved to the module with a write to the M1 file.
Analog Deadbands - 34	This command is implemented in the module per the protocol specification. Up to 300 analog values are supported by the module.

Command	Command Description
Analog Group Defines - 35	This command is implemented per the protocol specification. Up to 300 analog values are supported by the module.
Accumulator Preset - 36	This command is implemented in the module per the protocol specification. Up to 50 preset values are supported by the module. The RTU presets the running values of selected accumulators to the specified 16 bit values.
Continuation Request - 37	This command is implemented in the module per the protocol specification. It only pertains to functions 0, 1, 2, 3, and 7 as the other commands cannot return a message long enough to require a continuation.
Repeat Last Message - 38	This command is implemented in the module per the protocol specification.
Firmware Configuration - 39	This command is implemented in the module per the protocol specification. The value returned is the version level of the ProLinx firmware.
Exception Report - 63	This command is implemented in the module per the protocol specification.

8.6 LNG Message Format

Basic format of LNG Master or Remote Terminal Units (RTU) message format (request or response) has the following description: a message header, one optional message Data Block and a CRC code.

Message Header (2 CHAR)
Data Block(s) (Variable Length)
CRC (2 CHAR)

8.6.1 Master Message Header

7	6	5	4	3	2	1	0

SHR	MFC	ACK	0	0
RTU ADDRESS				

The Master Message Header consists of two characters. The first character contains the Acknowledge flag (ACK), an optional multi function code (MFC), and Short Response flag (SHR). The second character contains the RTU address.

ACK: Acknowledges flag is used by the Master to acknowledge RTU reply data.

MFC: When Multi Function Code is set, the short response flag will be set.

SHR: Short Response flag is set if the message does not contain any data blocks.

RTU Address: The RTU address specifies the receiving RTU for the message. There is one byte for the RTU address allowing the master station to address 255 distinctive RTU's. An address of 0 is used for a broadcast message.

8.6.2 Master Data Block

7	6	5	4	3	2	1	0

L	FUNCTION CODE
0	LENGTH
DATA FIELD	

A master originated message can contain only one data block. The data block consists of four fields:

L: The Last Block Mark is always set to one in master originated messages.

Function Code: The function code is a 7 bit field that defines the function to be performed by the RTU.

Length: The Length is a 7 bit field specifying the number of bytes in the Data Field. If it is 0, then the Data Block does not contain a Data Field.

Data Field: The Data Field contain Function Code specific operational information (that is, point numbers, parameters, etc.)

8.6.3 Master CRC

The CRC code included in the message is the standard CRC16 based on the polynomial. For a correctly received message the remainder will be zero after passing the entire message, including the CRC word, through the CRC computation. The bits of each message byte are passed through the CRC computation in the same order that they are transmitted, least significant bit first.

8.6.4 RTU Message Header

7	6	5	4	3	2	1	0
SHR	CON	FRZ	IND	SCH	SLG	0	0
RTU ADDRESS							

The RTU Message Header consisted of two characters. The first character contains several RTU flags. The second character contains the address of the responding RTU.

SHR: Short Response flag is set if the Message does not contain any Data Block.

CON: Continuation flag is used by the RTU when the size of a response to a data request exceeds the maximum message length. The Master can collect the remainder of the requested data by sending a continuation request message to the RTU.

FRZ: Freeze flag indicates the state of the accumulators. The RTU sets the flag when the accumulators are frozen and clears it when the frozen counts are read.

IND: Indication Change flag is set when the RTU has Indication Changes to report. The flag remains set until the Master has read and acknowledge the indication data change.

SCH: The SOE Change flag is set when the RTU has Sequence of Event Change data to report. This module is not support SOE function.

SLG: The SOE Log flag is set when the RTU has Sequence of Event Logs to report. The flag remain set until the Master has read and acknowledges the data. This module is not support SOE function.

8.6.5 RTU Data Block

7	6	5	4	3	2	1	0
L	FUNCTION CODE						
0	LENGTH						
DATA FIELD							

An RTU originated message can contain more than one data block. The data block consists of four fields:

L: The Last Block Mark is always set to one in the last Data Block of the messages, else it is reset to 0.

Function Code: The function code is a 7 bit field that defines the purpose of the message. The RTU generally echoes the function code sent by the Master, except when the RTU cannot perform the requested function due to an error, timeout, bad parameter, or some other condition that is not logical to the RTU.

Length: The Length is a 7 bit field specifying the number of bytes in the Data Field. If it is 0, then the Data Block does not contain a Data Field.

Data Field: The Data Field contain Function Code specific operational information (that is, point numbers, parameters, etc.)

8.6.6 RTU CRC

The CRC code included in the message is the standard CRC16 based on the polynomial. For a correctly received message the remainder will be zero after passing the entire message, including the CRC word, through the CRC computation. The bits of each message byte are passed through the CRC computation in the same order that they are transmitted, least significant bit first.

8.7 Frequently Asked Questions

8.7.1 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.

8.7.2 What software package 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.

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

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

8.7.4 Is a .MDC available for configuration of the Module?

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

8.7.5 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.

8.7.6 Can I use the 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 ProSoft communication products. Please call our technical support technicians when considering this application.

Support, Service & Warranty

ProSoft Technology, Inc. survives on its ability to provide meaningful support to its customers. Should any questions or problems arise, please feel free to contact us at:

Internet	Web Site: http://www.prosoft-technology.com/support E-mail address: support@prosoft-technology.com
Phone	+1 (661) 716-5100 +1 (661) 716-5101 (Fax)
Postal Mail	ProSoft Technology, Inc. 1675 Chester Avenue, Fourth Floor Bakersfield, CA 93301

Before calling for support, please prepare yourself for the call. In order to provide the best and quickest support possible, we will most likely ask for the following information:

1. Product Version Number
2. System architecture
3. Module configuration and contents of configuration file, if the module requires one.
4. Module Operation
   - Configuration/Debug status information
   - LED patterns
5. Information about the processor and user data files as viewed through the processor configuration software and LED patterns on the processor
6. Details about the serial devices interfaced

An after-hours answering system allows pager access to one of our qualified technical and/or application support engineers at any time to answer the questions that are important to you.

Module Service and Repair

The PTQ-LNG device is an electronic product, designed and manufactured to function under somewhat adverse conditions. As with any product, through age, misapplication, or any one of many possible problems the device may require repair.

When purchased from ProSoft Technology, Inc., the device has a 1 year parts and labor warranty (3 years for RadioLinx) according to the limits specified in the warranty. Replacement and/or returns should be directed to the distributor from whom the product was purchased. If you must return the device for repair, obtain an RMA (Returned Material Authorization) number from ProSoft Technology, Inc. Please call the factory for this number, and print the number prominently on the outside of the shipping carton used to return the device.

General Warranty Policy – Terms and Conditions

ProSoft Technology, Inc. (hereinafter referred to as ProSoft) warrants that the Product shall conform to and perform in accordance with published technical specifications and the accompanying written materials, and shall be free of defects in materials and workmanship, for the period of time herein indicated, such warranty period commencing upon receipt of the Product. Limited warranty service may be obtained by delivering the Product to ProSoft in accordance with our product return procedures and providing proof of purchase and receipt date. Customer agrees to insure the Product or assume the risk of loss or damage in transit, to prepay shipping charges to ProSoft, and to use the original shipping container or equivalent. Contact ProSoft Customer Service for more information.

This warranty is limited to the repair and/or replacement, at ProSoft's election, of defective or non-conforming Product, and ProSoft shall not be responsible for the failure of the Product to perform specified functions, or any other non-conformance caused by or attributable to: (a) any misuse, misapplication, accidental damage, abnormal or unusually heavy use, neglect, abuse, alteration (b) failure of Customer to adhere to ProSoft's specifications or instructions, (c) any associated or complementary equipment, software, or user-created programming including, but not limited to, programs developed with any IEC1131-3 programming languages, 'C' for example, and not furnished by ProSoft, (d) improper installation, unauthorized repair or modification (e) improper testing, or causes external to the product such as, but not limited to, excessive heat or humidity, power failure, power surges or natural disaster, compatibility with other hardware and software products introduced after the time of purchase, or products or accessories not manufactured by ProSoft; all of which components, software and products are provided as-is. In no event will ProSoft be held liable for any direct or indirect, incidental consequential damage, loss of data, or other malady arising from the purchase or use of ProSoft products.

ProSoft's software or electronic products are designed and manufactured to function under adverse environmental conditions as described in the hardware specifications for this product. As with any product, however, through age, misapplication, or any one of many possible problems, the device may require repair.

ProSoft warrants its products to be free from defects in material and workmanship and shall conform to and perform in accordance with published technical specifications and the accompanying written materials for up to one year (12 months) from the date of original purchase (3 years for RadioLinx products) from ProSoft. If you need to return the device for repair, obtain an RMA (Returned Material Authorization) number from ProSoft Technology, Inc. in accordance with the RMA instructions below. Please call the factory for this number, and print the number prominently on the outside of the shipping carton used to return the device.

If the product is received within the warranty period ProSoft will repair or replace the defective product at our option and cost.

Warranty Procedure: Upon return of the hardware product ProSoft will, at its option, repair or replace the product at no additional charge, freight prepaid, except as set forth below. Repair parts and replacement product will be furnished on an exchange basis and will be either reconditioned or new. All replaced product and parts become the property of ProSoft. If ProSoft determines that the Product is not under warranty, it will, at the Customer's option, repair the Product using then current ProSoft standard rates for parts and labor, and return the product freight collect.

Limitation of Liability

EXCEPT AS EXPRESSLY PROVIDED HEREIN, PROSOFT MAKES NO WARRANT OF ANY KIND, EXPRESSED OR IMPLIED, WITH RESPECT TO ANY EQUIPMENT, PARTS OR SERVICES PROVIDED PURSUANT TO THIS AGREEMENT, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. NEITHER PROSOFT OR ITS DEALER SHALL BE LIABLE FOR ANY OTHER DAMAGES, INCLUDING BUT NOT LIMITED TO DIRECT, INDIRECT, INCIDENTAL, SPECIAL OR CONSEQUENTIAL DAMAGES, WHETHER IN AN ACTION IN CONTRACT OR TORT (INCLUDING NEGLIGENCE AND STRICT LIABILITY), SUCH AS, BUT NOT LIMITED TO, LOSS OF ANTICIPATED PROFITS OR BENEFITS RESULTING FROM, OR ARISING OUT OF, OR IN CONNECTION WITH THE USE OR FURNISHING OF EQUIPMENT, PARTS OR SERVICES HEREUNDER OR THE PERFORMANCE, USE OR INABILITY TO USE THE SAME, EVEN IF ProSoft OR ITS DEALER'S TOTAL LIABILITY EXCEED THE PRICE PAID FOR THE PRODUCT.

Where directed by State Law, some of the above exclusions or limitations may not be applicable in some states. This warranty provides specific legal rights; other rights that vary from state to state may also exist. This warranty shall not be applicable to the extent that any provisions of this warranty are prohibited by any Federal, State or Municipal Law that cannot be preempted. Contact ProSoft Customer Service at +1 (661) 716-5100 for more information.

RMA Procedures

In the event that repairs are required for any reason, contact ProSoft Technical Support at +1 661.716.5100. A Technical Support Engineer will ask you to perform several tests in an attempt to diagnose the problem. Simply calling and asking for a RMA without following our diagnostic instructions or suggestions will lead to the return request being denied. If, after these tests are completed, the module is found to be defective, we will provide the necessary RMA number with instructions on returning the module for repair.

Index

K

L

M

N

O

P

Q

R

S

T

U

V

W

Y