ProSoft® TECHNOLOGY

Where Automation Connects.

ProTalk®

PTQ-HART

Quantum Platform

HART Multi-drop Network Interface Module for Quantum

September 13, 2010

USER MANUAL

Your Feedback Please

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

How to Contact Us

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

PTQ-HART User Manual
September 13, 2010

ProSoft Technology® Product Documentation

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

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

North America: +1.661.716.5100

Asia Pacific: +603.7724.2080

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

Latin America: +1.281.298.9109

Information for ProTalk® Product Users

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

The following or equivalent warnings shall be included:

A Warning - Explosion Hazard - Substitution of components may Impair Suitability for Class I, Division 2;
B Warning - Explosion Hazard - When in Hazardous Locations, Turn off Power before replacing Wiring Modules, and
C Warning - Explosion Hazard - Do not Disconnect Equipment unless Power has been switched Off or the Area is known to be Nonhazardous.
D Caution: The Cell used in this Device may Present a Fire or Chemical Burn Hazard if Mistreated. Do not Disassemble, Heat above 100°C (212°F) or Incinerate.

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

AVERTISSEMENT - RISQUE D'EXPLOSION - AVANT DE DÉCONNECTER L'ÉQUIPEMENT, COUPER LE COURANT OU S'ASSURER QUE L'EMPLACEMENT EST DÉSIGNÉ NON DANGEREUX.

Warnings

North America Warnings

A Warning - Explosion Hazard - Substitution of components may impair suitability for Class I, Division 2.
B Warning - Explosion Hazard - When in hazardous locations, turn off power before replacing or rewiring modules.
Warning - Explosion Hazard - Do not disconnect equipment unless power has been switched off or the area is known to be non-hazardous.
C Suitable for use in Class I, Division 2 Groups A, B, C and D Hazardous Locations or Non-Hazardous Locations.

ATEX Warnings and Conditions of Safe Usage:

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

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

Electrical Ratings

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

Markings:

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

243333

ME06

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.

Contents

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

Guide to the PTQ-HART User Manual 9

1 Start Here 11

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

2 Configuring the Processor with Unity Pro 15

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

3 Configuring the Processor with Concept 25

3.1 Information for Concept Version 2.6 Users 26
3.1.1 Installing MDC Configuration Files 26
3.2 Creating a New Project 28
3.3 Adding the PTQ Module to the Project 30
3.4 Setting up Data Memory in Project 32
3.5 Downloading the Project to the Processor 35
3.6 Verifying Successful Download 37

4 Configuring the Processor with ProWORX 41

5 Setting Up the ProTalk Module 45

5.1 Installing the ProTalk Module in the Quantum Rack 46
5.2 Connect the PC to the ProTalk Configuration/Debug Port 47
5.3 Verifying Communication Between the Processor and the Module 49

6 Module Configuration 51

6.1 Using ProSoft Configuration Builder ... 52
6.1.1 Set Up the Project ... 52
6.1.2 Renaming PCB Objects ... 54
6.2 [Module] ... 55
6.2.1 Module Name ... 55
6.2.2 Error/Status Offset ... 55
6.2.3 Failure Flag Count ... 55
6.2.4 Initializing Output Data ... 55
6.2.5 Read Register Start ... 56
6.2.6 Write Register Start ... 56
6.2.7 Read Register Count ... 56
6.2.8 Write Register Count ... 56
6.2.9 3x Register Start ... 57
6.2.10 4x Register Start ... 57
6.3 [HART PORT x] ... 58
6.3.1 Enabled ... 58
6.3.2 Preambles ... 58
6.3.3 Primary Master ... 58
6.3.4 Retry Count ... 58
6.3.5 DB Address Status ... 59
6.3.6 Command Count ... 59
6.3.7 Auto-Poll Code ... 59
6.3.8 Auto-Poll DB Address ... 60
6.3.9 Auto-Poll Swap Float ... 60
6.3.10 Max Device Count ... 60
6.3.11 Error/Status Offset ... 60
6.4 [HART PORT x COMMANDS] ... 61
6.4.1 Command List Overview ... 61
6.4.2 Commands Supported by the Module ... 62
6.4.3 HART Command Entry Formats ... 64
6.4.4 Enable ... 65
6.4.5 Float DB Address ... 65
6.4.6 Poll Interval ... 65
6.4.7 FP Word Count ... 66
6.4.8 Swap Code ... 66
6.4.9 Short Address ... 66
6.4.10 Function Code ... 66
6.4.11 Int. DB Address ... 67
6.4.12 Int Word Count ... 67
6.4.13 Swap Code ... 67
6.4.14 Use Long ... 67
6.4.15 Enable DB Address ... 67
6.4.16 Done DB Address ... 68
6.4.17 Write DB Address ... 68
6.4.18 Swap Code ... 68
6.4.19 Byte Count ... 68
6.4.20 Fixed Data ... 69
6.5 Hart Command Examples ... 70
6.5.1 Example of HART Command Function 3 ... 70
6.5.2 Example of HART Command Function 34 ... 72
6.6 Downloading the Project to the Module Using a Serial COM port ... 74

7 Diagnostics and Troubleshooting 75

7.1 LED Status Indicators ... 76
7.2 Using ProSoft Configuration Builder (PCB) for Diagnostics ... 77
7.2.1 Using the Diagnostic Window in ProSoft Configuration Builder ... 77
7.2.2 Navigation ... 79
7.2.3 Main Menu ... 80
7.2.4 HART Master Menu ... 83
7.2.5 Data Analyzer ... 85
7.2.6 Master Command Error List Menu ... 87
7.2.7 Master Command List Menu ... 88
7.2.8 Database View Menu ... 89
7.3 Reading Status Data from the Module ... 91

8 Reference 93

8.1 Product Specifications ... 94
8.1.1 General Specifications ... 94
8.1.2 Hardware Specifications ... 95
8.1.3 Functional Specifications ... 96
8.2 Functional Overview ... 97
8.2.1 General Concepts ... 97
8.2.2 Backplane Data Transfer ... 98
8.2.3 Command Control ... 101
8.2.4 HART Channels ... 105
8.2.5 HART Command List ... 107
8.2.6 Auto-Polling ... 108
8.3 RS-232 Configuration/Debug Port ... 115
8.4 Application Port Connection ... 116
8.5 Example Field Terminations ... 117
8.5.1 Notes ... 117
8.6 Thermal Connector Port Details ... 124
8.7 PTQ-HART Error Status Table ... 125
8.7.1 Error/Status Data Block ... 125
8.7.2 Protocol Error Codes ... 129
8.8 HART Universal Commands ... 132
COMMAND 00 - Read Unique Identifier ... 132
COMMAND 01 - Read Primary Variable ... 133
COMMAND 02 - Read Current And Percent Of Range ... 134
COMMAND 03 - Read Dynamic Variables ... 135
COMMAND 06 - Write Polling Address ... 136
COMMAND 12 - Read Message ... 137
COMMAND 13 - Read Tag, Descriptor and Date ... 138
COMMAND 14 - Read PV Sensor Info ... 139
COMMAND 15 - Read Output Information ... 140
COMMAND 16 - Read Final Assembly Number ... 141
COMMAND 17 - Write Message ... 142
COMMAND 18 - Write Tag, Descriptor and Date ... 143
COMMAND 19 - Write Final Assembly Number ... 144
8.9 HART Common Practice Commands ... 145
COMMAND 33 - Read Transmitter Variables ... 145
COMMAND 34 - Write Damping Value ... 146
COMMAND 35 - Write Range Values ... 147

COMMAND 36 - Set Upper Range Value 148
COMMAND 37 - Set Lower Range Value 149
COMMAND 38 - Reset Configuration Changed Flag 150
COMMAND 39 - EEPROM Control 151
COMMAND 40 - Enter Exit Fixed Current Mode 152
COMMAND 41 - Perform Transmitter Self Test 153
COMMAND 42 - Perform Master Reset 154
COMMAND 43 - Set PV Zero 155
COMMAND 44 - Write PV Units 156
COMMAND 45 - Trim DAC Zero 157
COMMAND 46 - Trim DAC Gain 158
COMMAND 47 - Write Transfer Function 159
COMMAND 48 - Read Additional Transmitter Status 160
COMMAND 49 - Write PV Sensor Serial Number 161
COMMAND 50 - Read Dynamic Variable Assignments 162
COMMAND 51 - Write Dynamic Variable Assignments 163
COMMAND 52 - Set Transmitter Variable Zero 164
COMMAND 53 - Write Transmitter Variable Units 165
COMMAND 54 - Read Transmitter Variable Information 166
COMMAND 55 - Write Transmitter Variable Damping Value 167
COMMAND 56 - Write Transmitter Variable Sensor Serial Number 168
COMMAND 57 - Read Unit Tag Descriptor Date 169
COMMAND 58 - Write Unit Tag Descriptor Date 170
COMMAND 59 - Write Number Of Response Preambles 171
COMMAND 108 - Write Burst Mode Command Number 172
COMMAND 109 - Burst Mode Control 173
COMMAND 110 - Read All Dynamic Variables 174
8.10 Frequently Asked Questions 175

9 Support, Service & Warranty 177

Contacting Technical Support 177
9.1 Return Material Authorization (RMA) Policies and Conditions 179
9.1.1 Returning Any Product 179
9.1.2 Returning Units Under Warranty 180
9.1.3 Returning Units Out of Warranty 180
9.2 LIMITED WARRANTY 181
9.2.1 What Is Covered By This Warranty 181
9.2.2 What Is Not Covered By This Warranty 182
9.2.3 Disclaimer Regarding High Risk Activities 182
9.2.4 Intellectual Property Indemnity 183
9.2.5 Disclaimer of all Other Warranties 183
9.2.6 Limitation of Remedies ** 184
9.2.7 Time Limit for Bringing Suit 184
9.2.8 No Other Warranties 184
9.2.9 Allocation of Risks 184
9.2.10 Controlling Law and Severability 185

Index 187

Guide to the PTQ-HART User Manual

Function		Section to Read	Details
Introduction (Must Do)	→	Start Here (page 11)	This section introduces the customer to the module. Included are: package contents, system requirements, hardware installation, and basic configuration.
Diagnostic and Troubleshooting	→	Diagnostics and Troubleshooting (page 75)	This section describes Diagnostic and Troubleshooting procedures.
Reference Product Specifications Functional Overview	→	Reference (page 93) Product Specifications (page 94) Functional Overview (page 97, page 90)	These sections contain general references associated with this product, Specifications, and the Functional Overview.
Support, Service, and Warranty Index	→	Support, Service and Warranty (page 177) Index	This section contains Support, Service and Warranty information. Index of chapters.

1 Start Here

In This Chapter

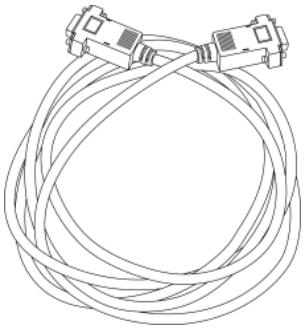
❖ Hardware and Software Requirements 12
❖ Installing ProSoft Configuration Builder Software 14

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

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

1.1 Hardware and Software Requirements

1.1.1 Package Contents

	
ProTalk Module	Null Modem Serial Cable
	
1454-9F DB-9 Female to 9 Pos Screw Terminal adapter (Serial protocol modules only)	ProSoft Solutions CD

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

1.1.2 Quantum Hardware

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

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

1.1.3 PC and PC Software

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

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

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

1.2 Installing ProSoft Configuration Builder Software

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

Installing ProSoft Configuration Builder from the ProSoft website

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

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

Installing ProSoft Configuration Builder from the Product CD-ROM

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

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

2 Configuring the Processor with Unity Pro

In This Chapter

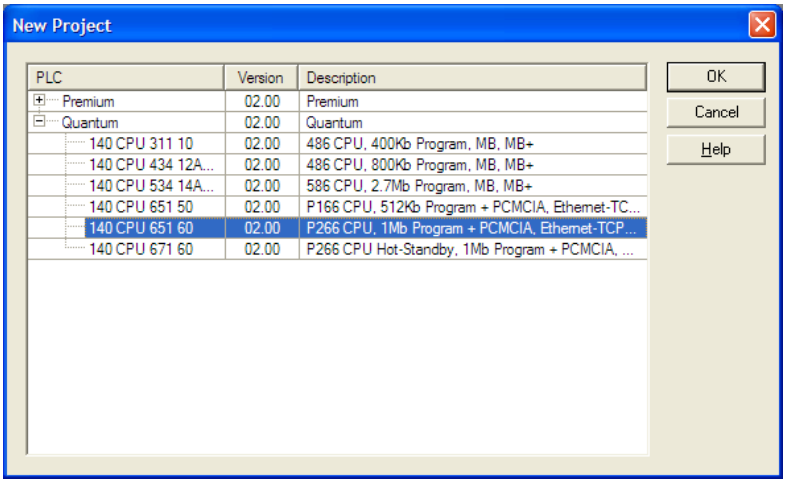
- Creating a New Project 16
- Adding the PTQ Module to the Project 18
- Building the Project 19
- Connect Your PC to the Processor 20
- Downloading the Project to the Processor 23

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

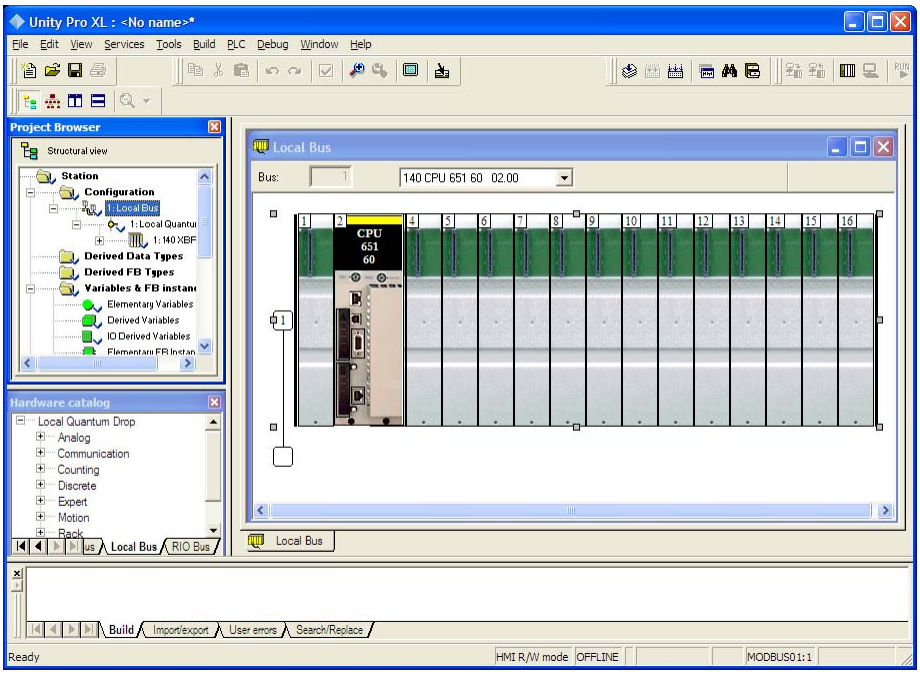
2.1 Creating a New Project

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

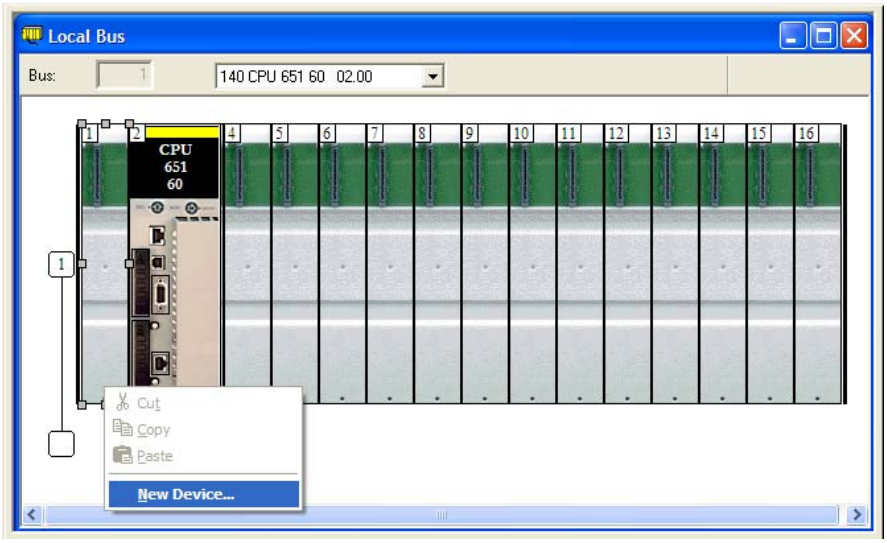
1 In the *New Project* dialog box, choose the CPU type. In the following illustration, the CPU is 140 CPU 651 60. Choose the processor type that matches your own hardware configuration, if it differs from the example. Click **OK** to continue.



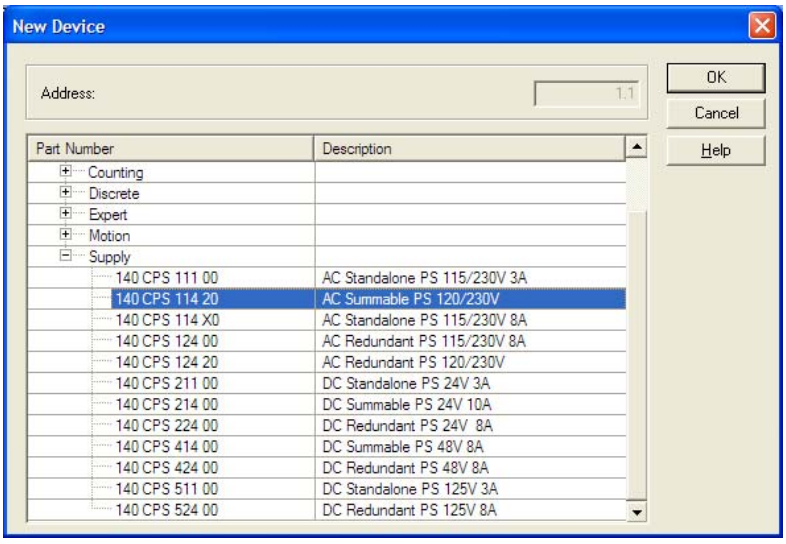
2 Next, add a power supply to the project. In the *Project Browser*, expand the *Configuration* folder, and then double-click the **1:LOCALBUS** icon. This action opens a graphical window showing the arrangement of devices in your Quantum rack.



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



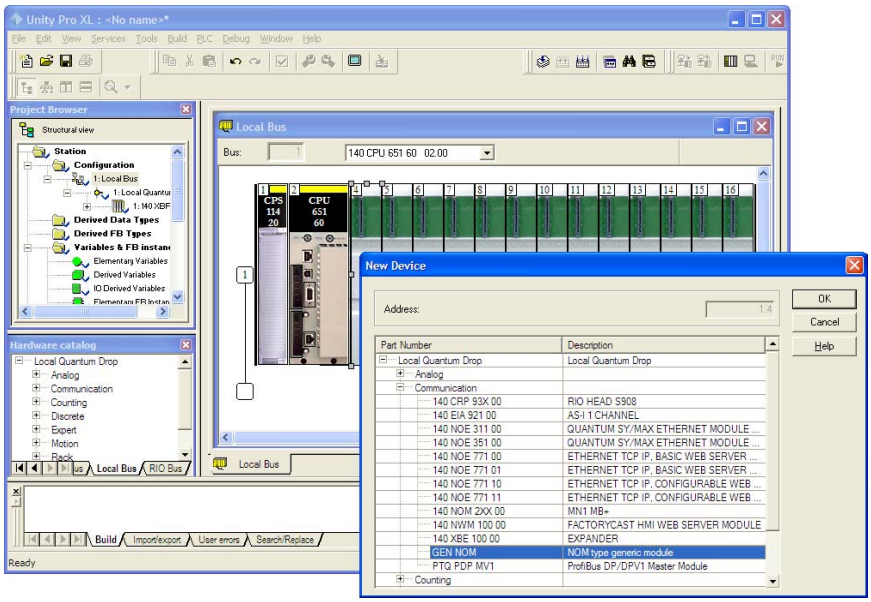
4 Expand the *Supply* folder, and then select your power supply from the list. Click **OK** to continue.



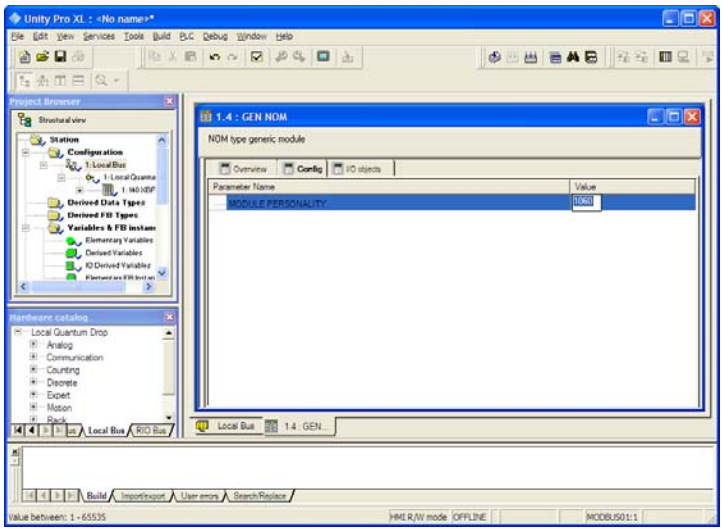
5 Repeat these steps to add any additional devices to your Quantum Rack.

2.2 Adding the PTQ Module to the Project

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



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



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

4 **SAVE** the project.

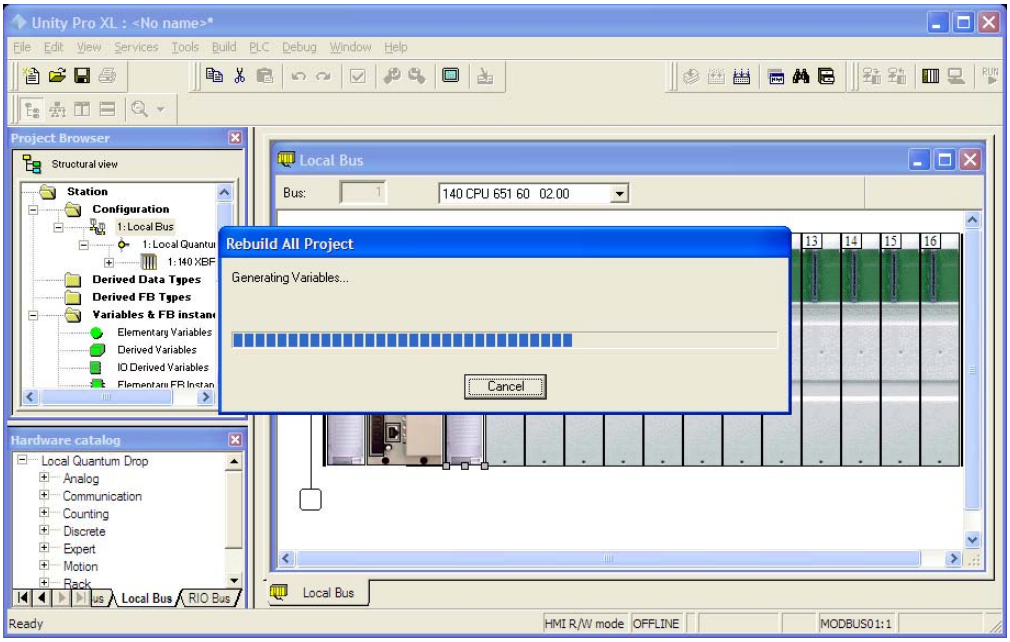
2.3 Building the Project

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

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

To build (compile) the project

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



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

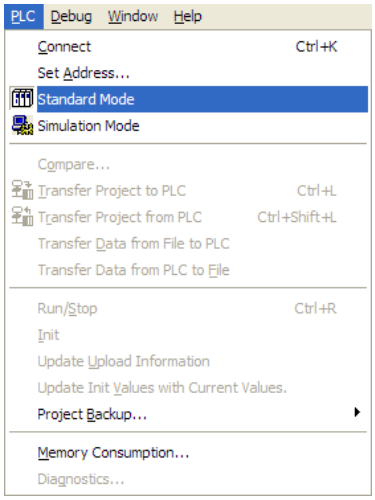
2.4 Connect Your PC to the Processor

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

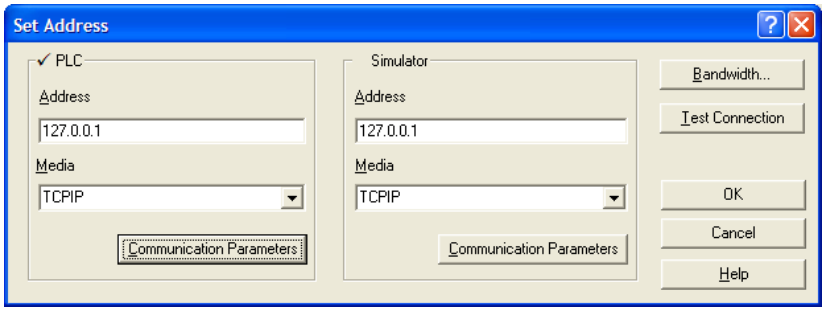
Note: If you have never connected from the PC to your processor before, you must verify that the necessary port drivers are installed and available to Unity Pro.

To verify address and driver settings in Unity Pro

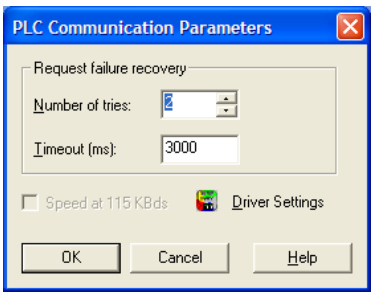
1 Open the **PLC** menu, and choose **STANDARD MODE**. This action turns off the PLC Simulator, and allows you to communicate directly with the Quantum or Unity hardware.



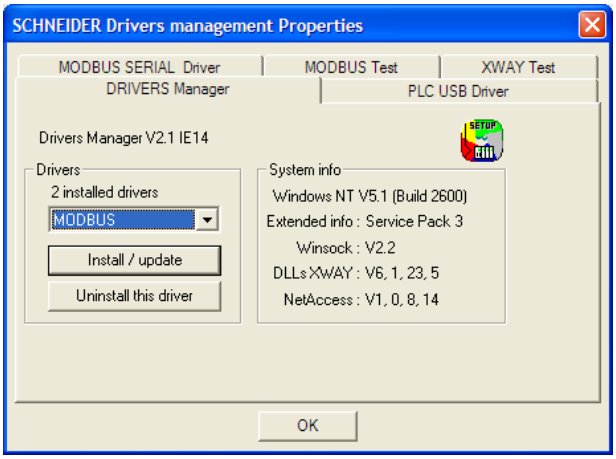
2 Open the **PLC** menu, and choose **SET ADDRESS...** This action opens the *Set Address* dialog box. Open the **MEDIA** dropdown list and choose the connection type to use (*TCPIP or USB*).



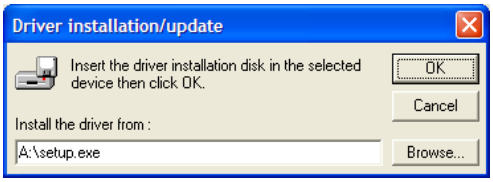
3 If the **MEDIA** dropdown list does not contain the connection method you wish to use, click the **COMMUNICATION PARAMETERS** button in the PLC area of the dialog box. This action opens the *PLC Communication Parameters* dialog box.



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



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



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

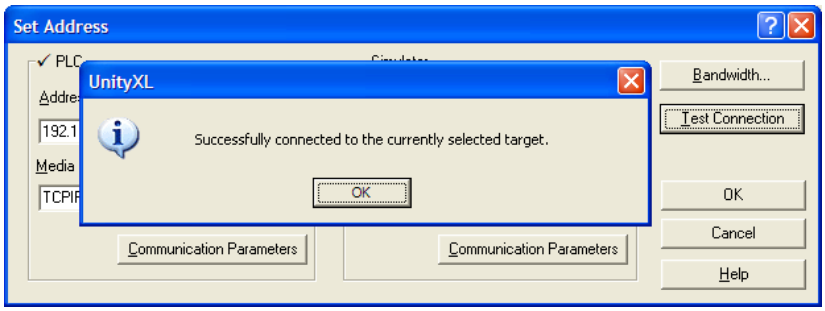
2.4.1 Connecting to the Processor with TCPIP

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 Unity Pro.

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

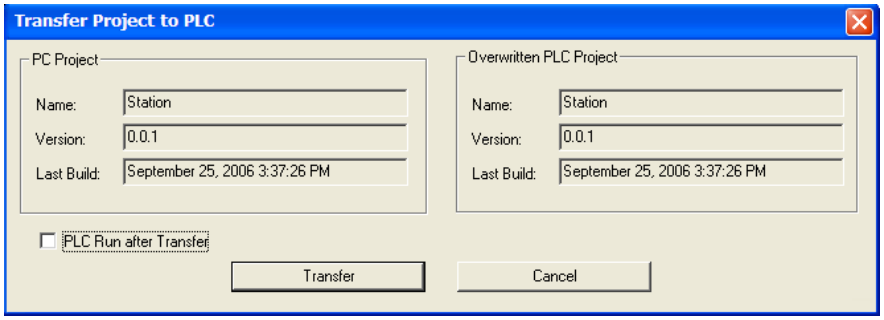
Important: Notice that the *Set Address* dialog box is divided into two areas. Enter the address and media type in the *PLC* area of the dialog box, not the *Simulator* area.

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



2.5 Downloading the Project to the Processor

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



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

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

3 Configuring the Processor with Concept

In This Chapter

- Information for Concept Version 2.6 Users 26
- Creating a New Project 28
- Adding the PTQ Module to the Project 30
- Setting up Data Memory in Project 32
- Downloading the Project to the Processor 35
- Verifying Successful Download 37

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

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

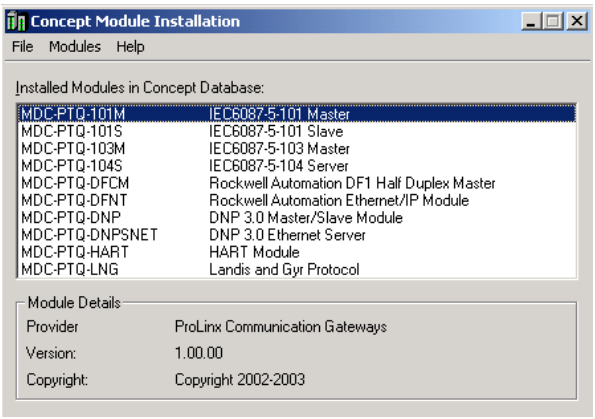
3.1 Information for Concept Version 2.6 Users

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

3.1.1 Installing MDC Configuration Files

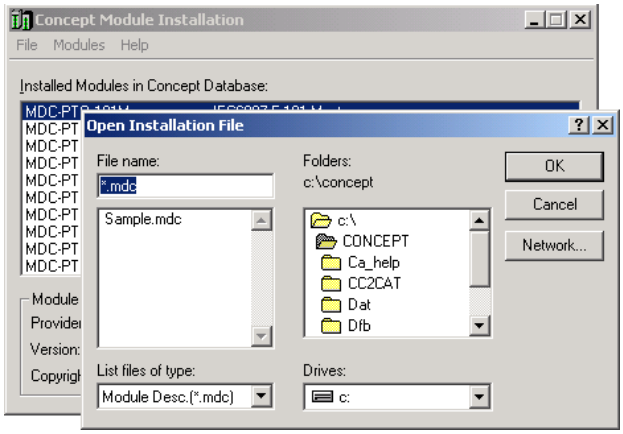
1 From a PC with Concept 2.6 installed, choose **START / PROGRAMS / CONCEPT / MODCONNECT TOOL**.

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



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

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

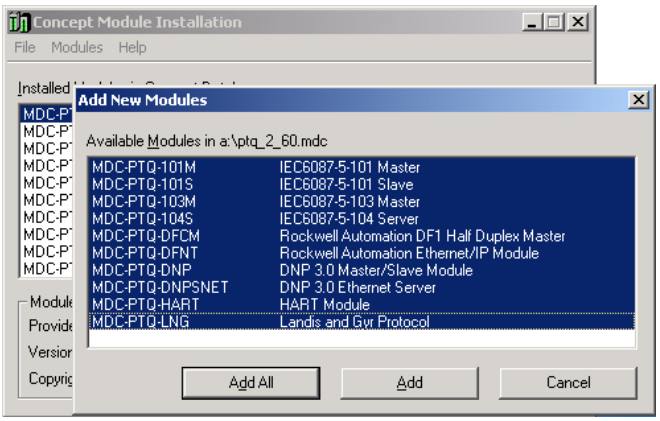


3 If you are using a Quantum processor, you will need the MDC files. In the *Open Installation File* dialog box, navigate to the *MDC Files* directory on the ProTalk CD.

4 Choose the MDC file and help file for your version of Concept:

- Concept 2.6 users: select PTQ_2_60.mdc and PTQMDC.hlp
- Concept 2.5 users: select PTQ_2_50.mdc and PTQMDC.hlp.

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

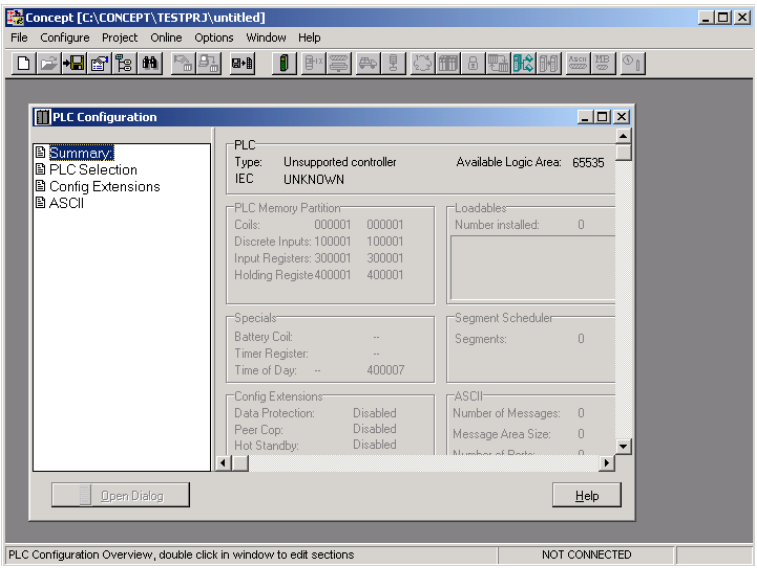


5. Click the **ADD ALL** button. A series of message boxes may appear during this process. Click **YES** or **OK** for each message that appears.
6. When the process is complete, open the **FILE** menu and choose **EXIT** to save your changes.

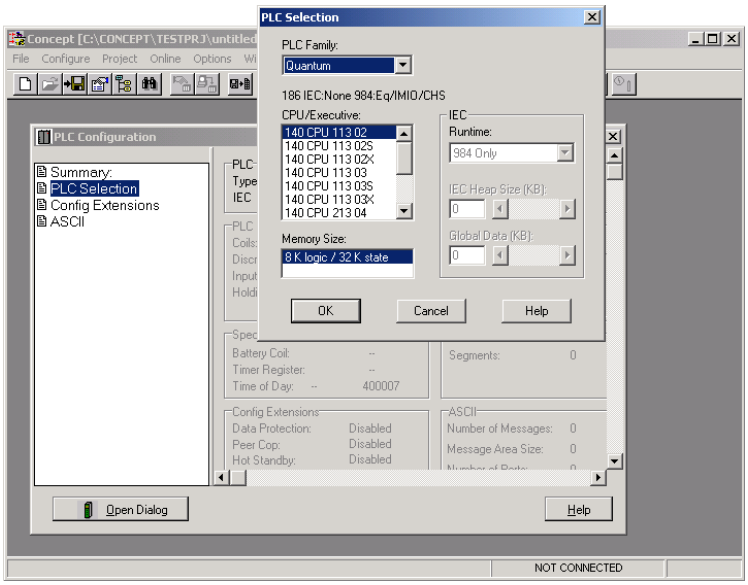
3.2 Creating a New Project

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

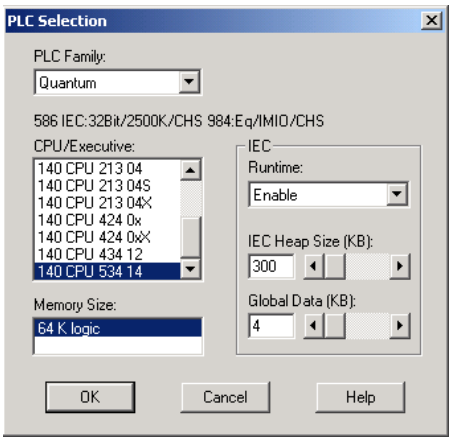
1. From your computer, choose **START / PROGRAMS / CONCEPT V2.6 XL.EN / CONCEPT**. This action opens the *Concept* window.
2. Open the File menu, and then choose **NEW PROJECT**. This action opens the *PLC Configuration* dialog box.



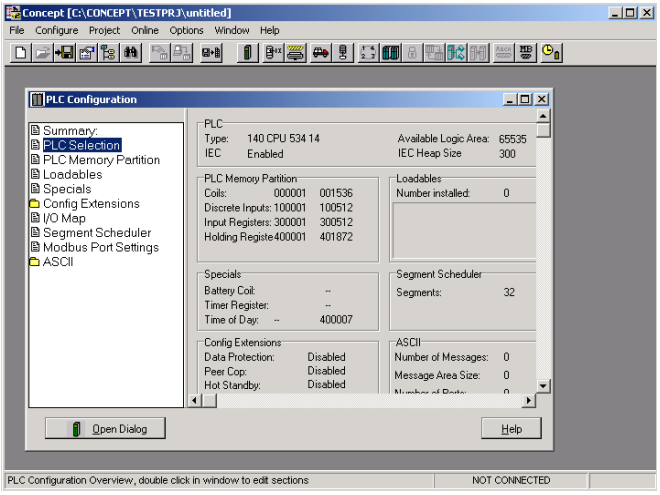
3. In the list of options on the left side of this dialog box, double-click the **PLC SELECTION** folder. This action opens the *PLC Selection* dialog box.



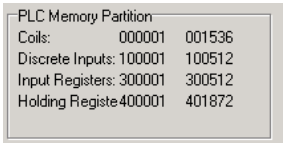
4 In the *CPU/Executive* pane, use the scroll bar to locate and select the **PLC** to configure.



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

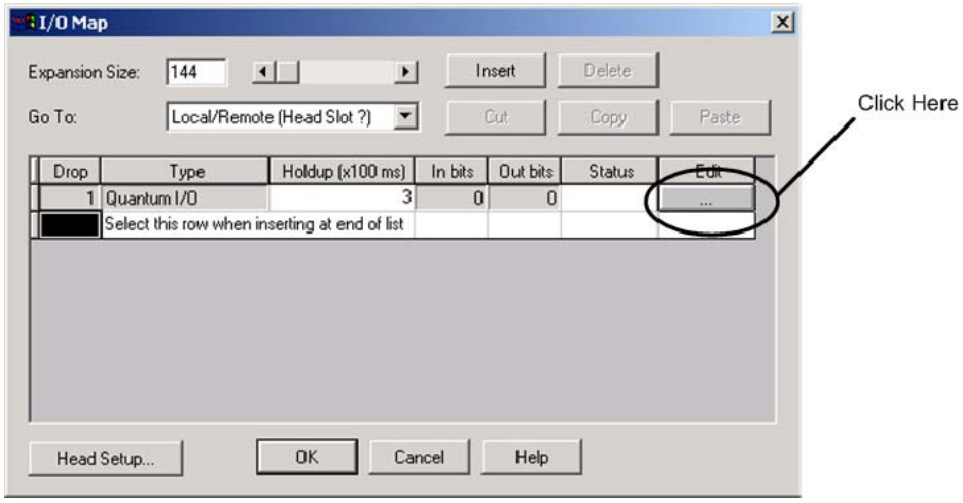


6 Make a note of the holding registers for the module. You will need this information when you modify your application. The Holding Registers are displayed in the *PLC Memory Partition* pane of the *PLC Configuration* dialog box.

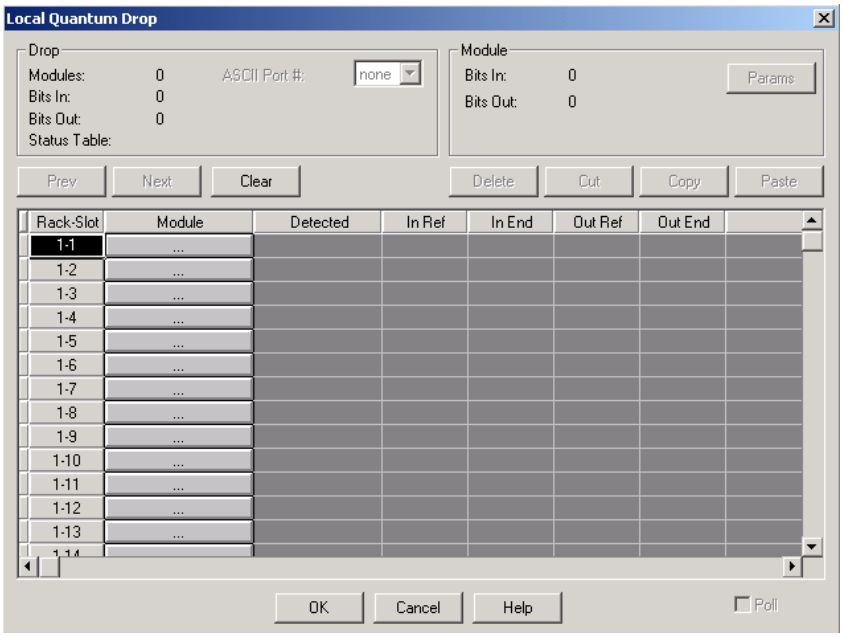


3.3 Adding the PTQ Module to the Project

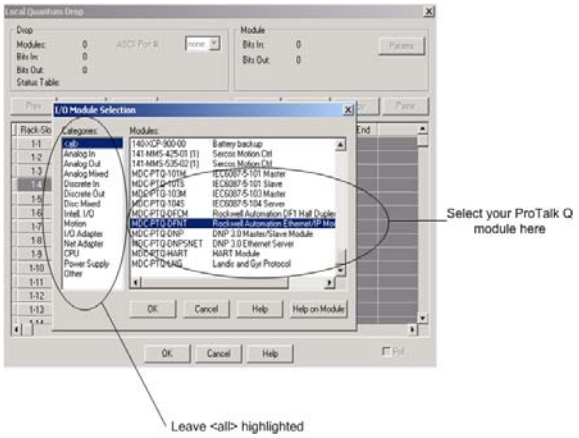
1 In the list of options on the left side of the *PLC Configuration* dialog box, double-click **I/O MAP**. This action opens the *I/O Map* dialog box.



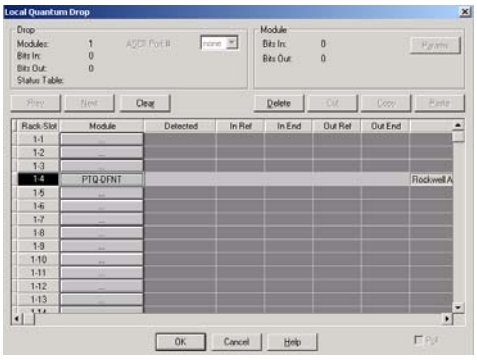
2 Click the **EDIT** button to open the *Local Quantum Drop* dialog box. This dialog box is where you identify rack and slot locations.



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.

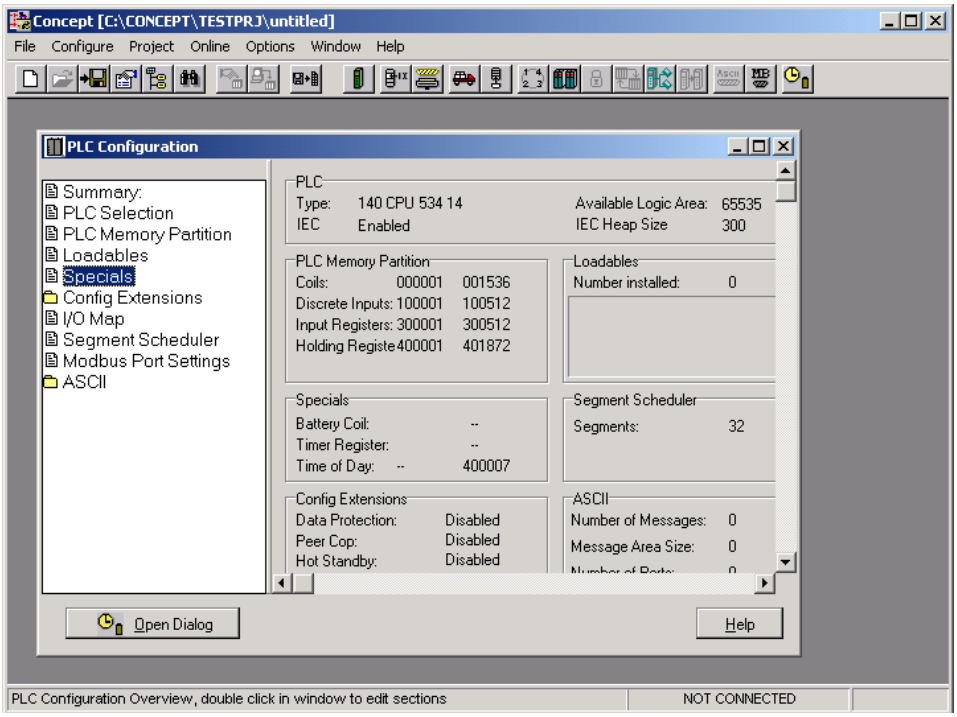


5 Repeat steps 3 through 5 for each ProTalk module you plan to install. When you have finished installing your ProTalk modules, click **OK** to save your settings. Click **YES** to confirm your settings.

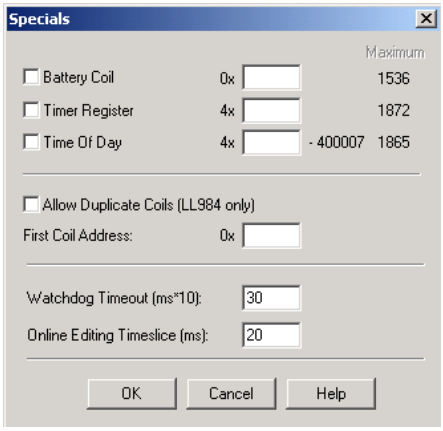
Tip: Select a module, and then click the Help on Module button for help pages.

3.4 Setting up Data Memory in Project

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

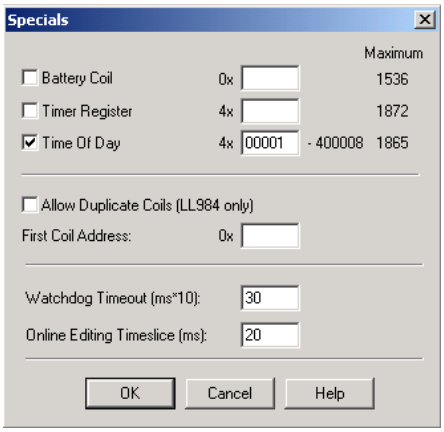


2 This action opens the *Specials* dialog box.



Selecting the Time of Day

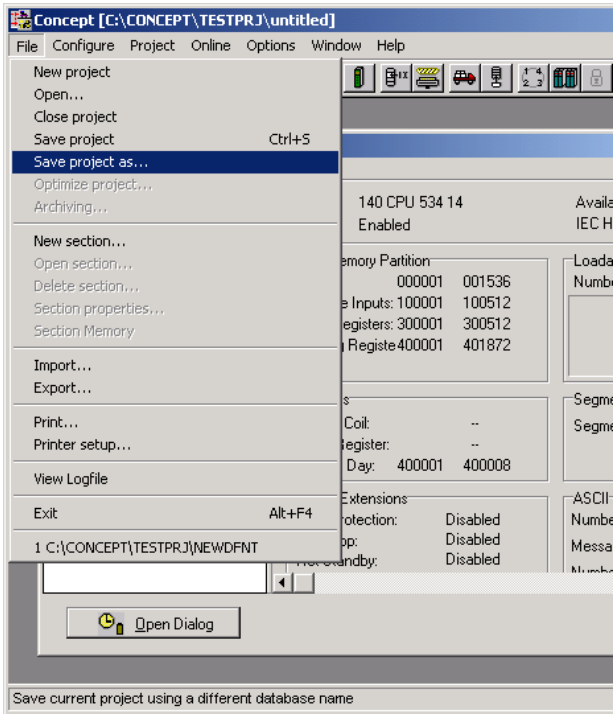
1 Select (check) the *Time of Day* box, and then enter the value 00001 as shown in the following illustration. This value sets the first time of day register to 400001.



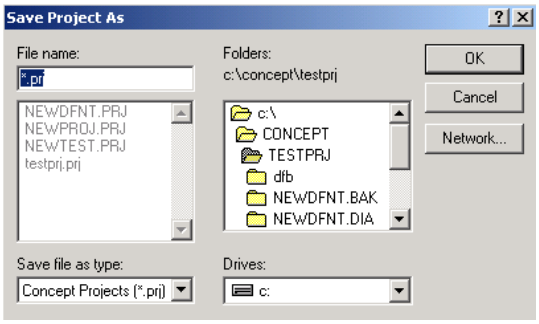
2 Click **OK** to save your settings and close the *Specials* dialog box.

Saving your project

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



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

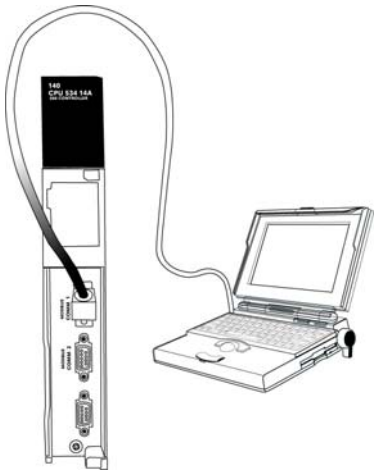


3 Name the project, and then click **OK** to save the project to a file.

3.5 Downloading the Project to the Processor

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

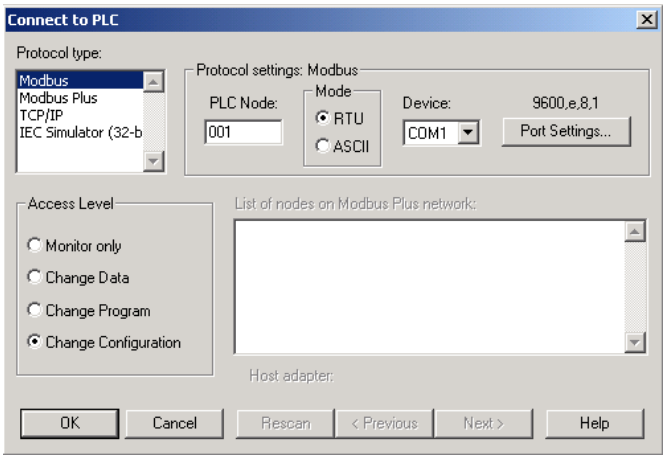
1 Use the null modem cable to connect your PC's serial port to the Quantum processor, as shown in the following illustration.



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

2 Open the **PLC** menu, and then choose **CONNECT.**

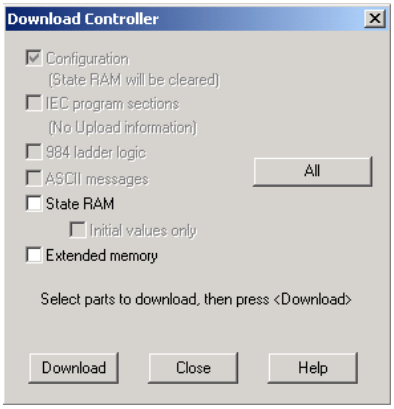
3 In the *PLC Configuration* dialog box, open the **ONLINE** menu, and then choose **CONNECT.** This action opens the *Connect to PLC* dialog box.



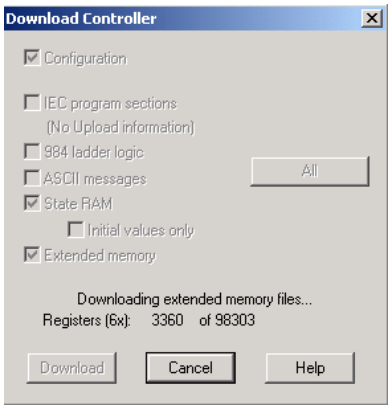
4 Leave the default settings as shown and click **OK.**

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

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



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

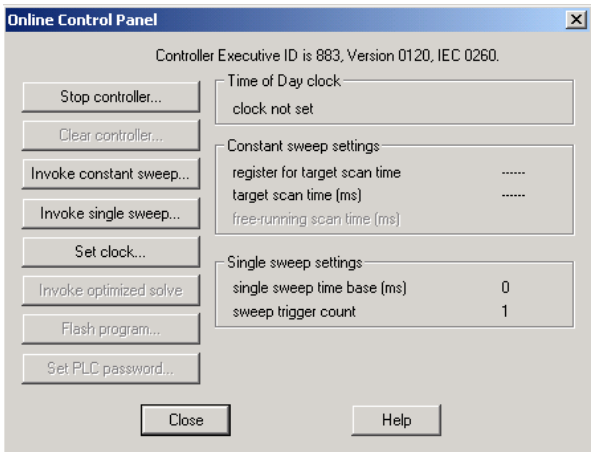


7 When the download is complete, you will be prompted to restart the controller. Click **YES** to restart the controller.

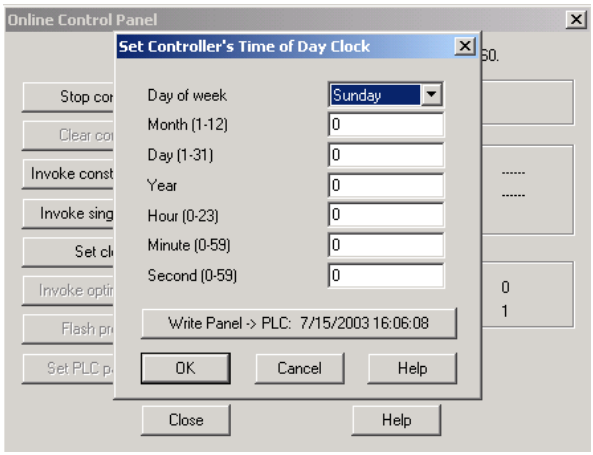
3.6 Verifying Successful Download

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

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

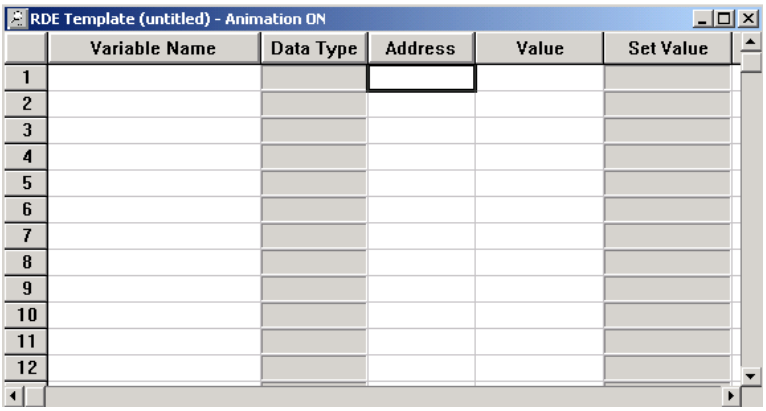


2. Click the **SET CLOCK** button to open the *Set Controller's Time of Day Clock* dialog box.



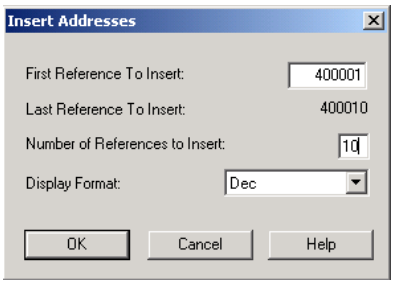
3. Click the **WRITE PANEL** button. This action updates the date and time fields in this dialog box. Click **OK** to close this dialog box and return to the previous window.
4. Click **CLOSE** to close the *Online Control Panel* dialog box.
5. In the *PLC Configuration* window, open the **ONLINE** menu,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

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

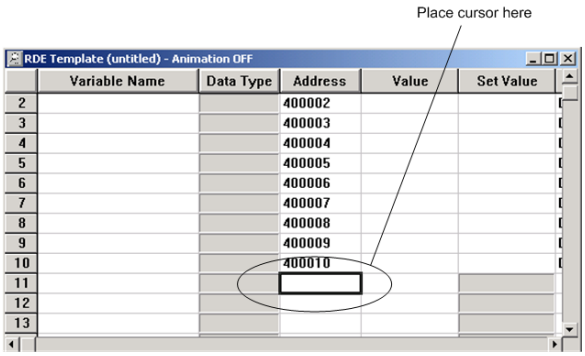


7 In the *PLC Configuration* window, open the **TEMPLATES** menu, and then choose **INSERT ADDRESSES.** This action opens the Insert addresses dialog box.

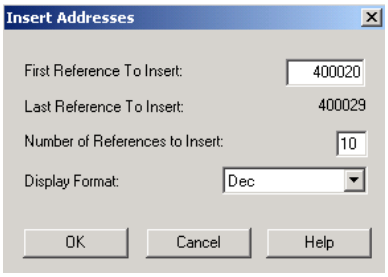
8 On the *Insert Addresses* dialog box, enter the values shown in the following illustration, and then click **OK.**



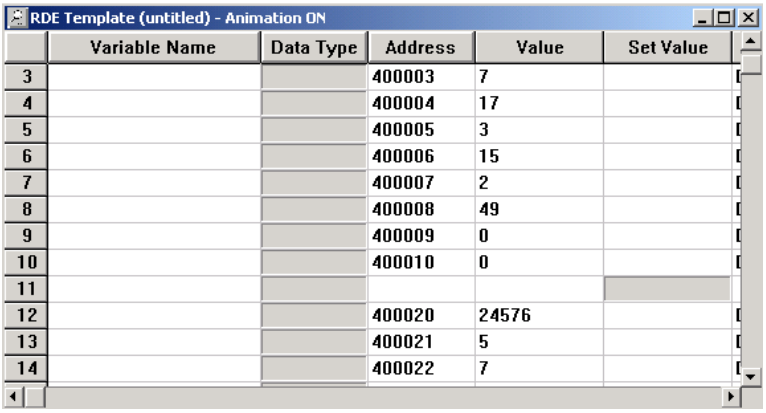
9 Notice that the template populates the address range, as shown in the following illustration. Place your cursor as shown in the first blank address field below the addresses you just entered.



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



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



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

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

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

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

At this point, you have successfully

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

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

4 Configuring the Processor with ProWORX

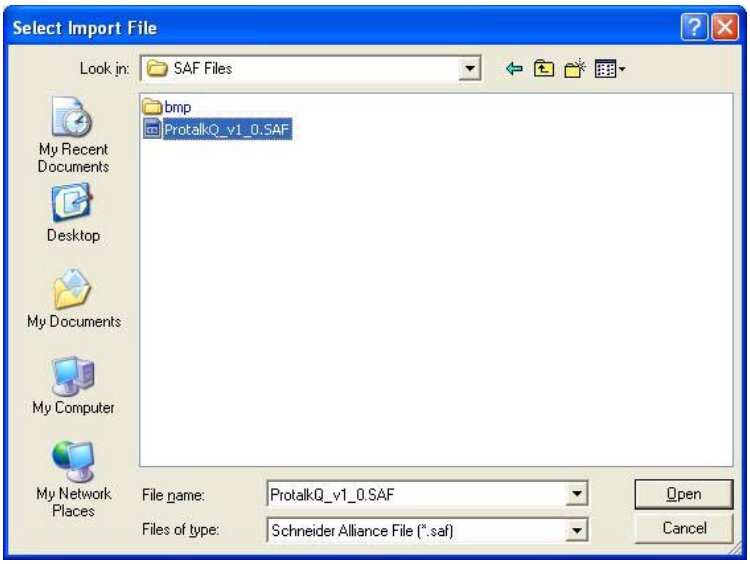
When you use ProWORX 32 software to configure the processor, use the example SAF file provided on the ProTalk Solutions CD-ROM.

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

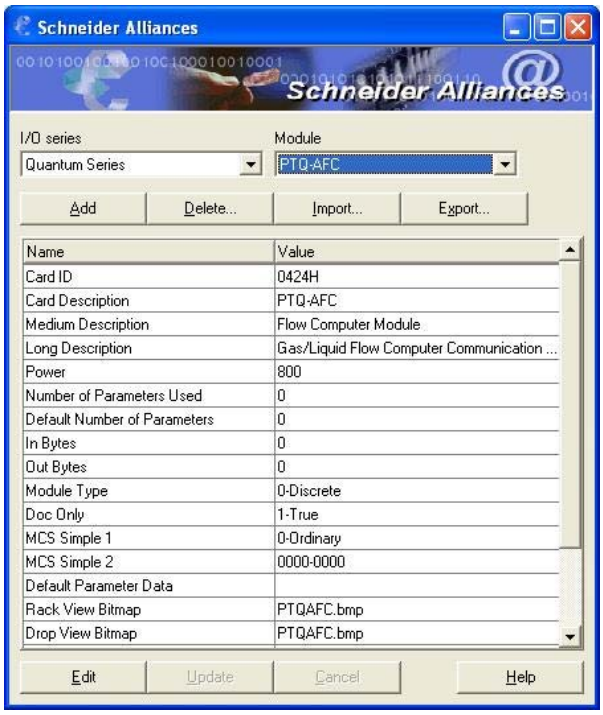
1 Run the **SCHNEIDER_ALLIANCES.EXE** application that is installed with the ProWORX 32 software:

2 Click on **IMPORT…**

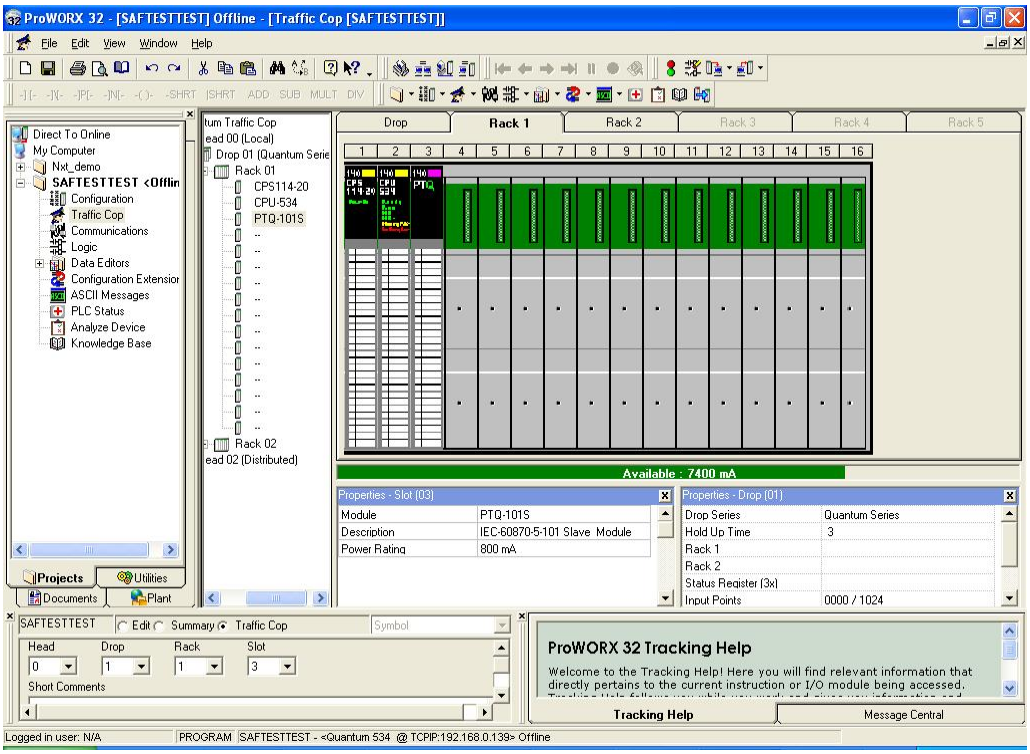
3 Select the *.SAF* File that is located on the CD-ROM shipped with the PTQ module.



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



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



5 Setting Up the ProTalk Module

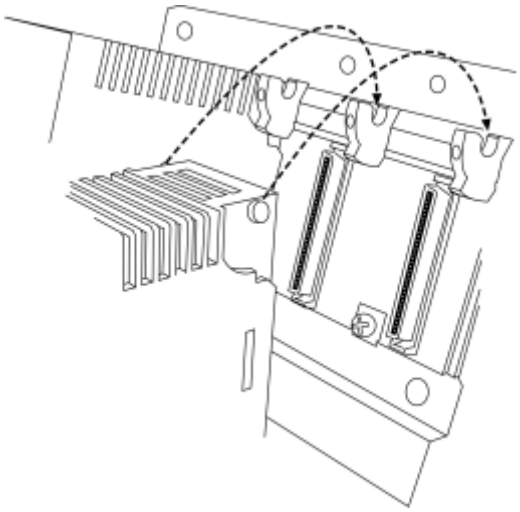
In This Chapter

- Installing the ProTalk Module in the Quantum Rack 46
- Connect the PC to the ProTalk Configuration/Debug Port 47
- Verifying Communication Between the Processor and the Module 49

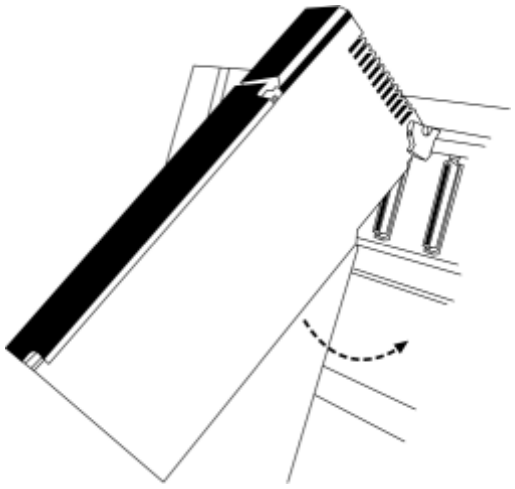
After you complete the following procedures, the ProTalk module will actively be transferring data bi-directionally with the processor.

5.1 Installing the ProTalk Module in the Quantum Rack

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



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

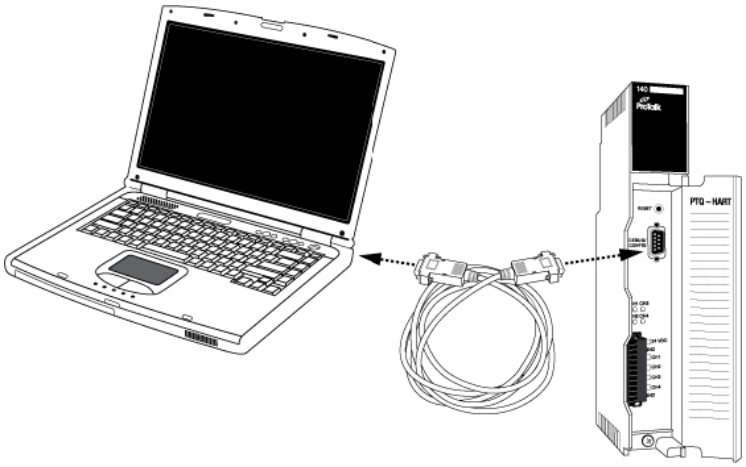


Caution: The PTQ module is hot-swappable, meaning that you can install and remove it while the rack is powered up. You should not assume that this is the case for all types of modules unless the user manual for the product explicitly states that the module is hot-swappable. Failure to observe this precaution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

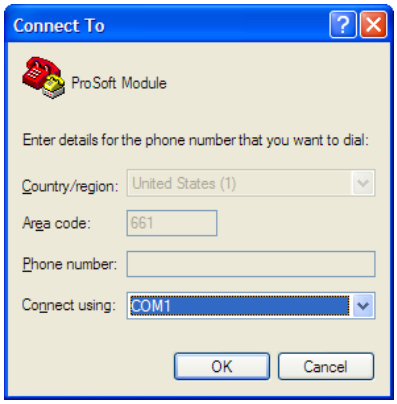
5.2 Connect the PC to the ProTalk Configuration/Debug Port

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

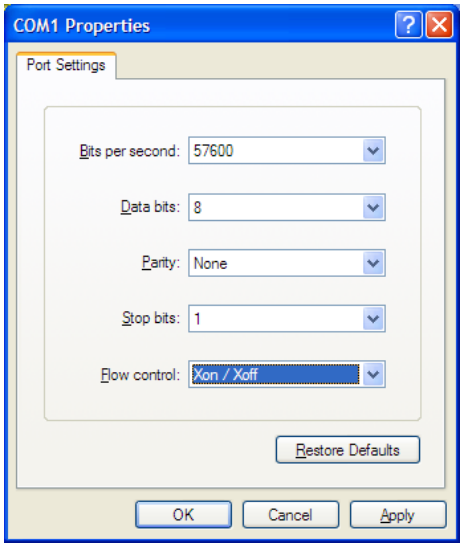
1. Using the supplied Null Modem cable, connect your PC to the Configuration/Debug port on the ProTalk module as shown



2. Click the Windows Start button, then choose Programs / Accessories / Communications / HyperTerminal.
3. In the HyperTerminal window, enter a connection name, for example **Test,** and then click OK. This action opens the Connect To dialog box.



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



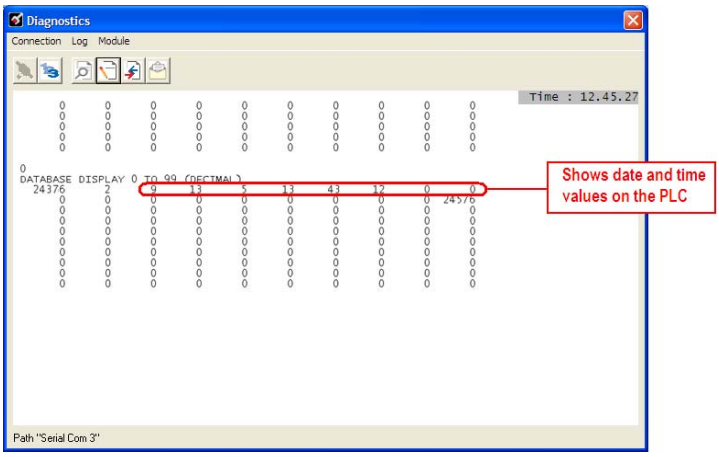
5 Verify that the settings match those shown in the example above, and then click OK. If your port settings are configured correctly, you will return to the HyperTerminal window.

6 In the HyperTerminal window, press **[?]**. This action opens the module's Configuration/Debug menu.

5.3 Verifying Communication Between the Processor and the Module

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

1. From the **CONFIGURATION/DEBUG MENU,** type **[D],** then press **[?].** This action opens the *Database View Menu***.**
2. Type **[0]** (zero). This displays values present in the ProTalk database for 0 to 99.



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

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

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

At this point, you have successfully:

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

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

6 Module Configuration

In This Chapter

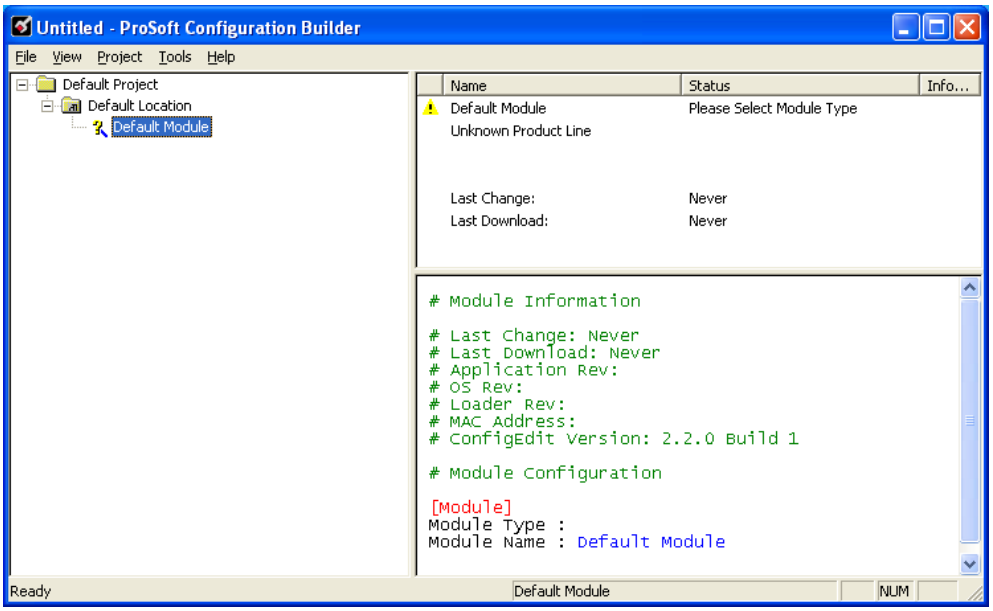
- Using ProSoft Configuration Builder 52
- [Module] 55
- [HART PORT x] 58
- [HART PORT x COMMANDS] 61
- Hart Command Examples 70
- Downloading the Project to the Module Using a Serial COM port 74

6.1 Using ProSoft Configuration Builder

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

6.1.1 Set Up the Project

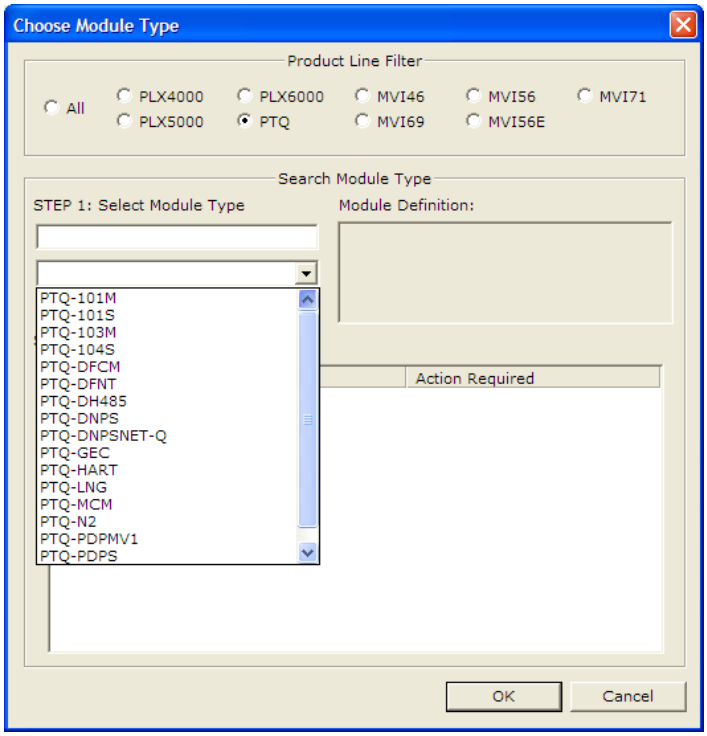
To begin, start *ProSoft Configuration Builder*. If you have used other Windows configuration tools before, you will find the screen layout familiar. *ProSoft Configuration Builder*'s window consists of a tree view on the left, an information pane and a configuration pane on the right side of the window.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



Your first task is to add the PTQ-HART module to the project.

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

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

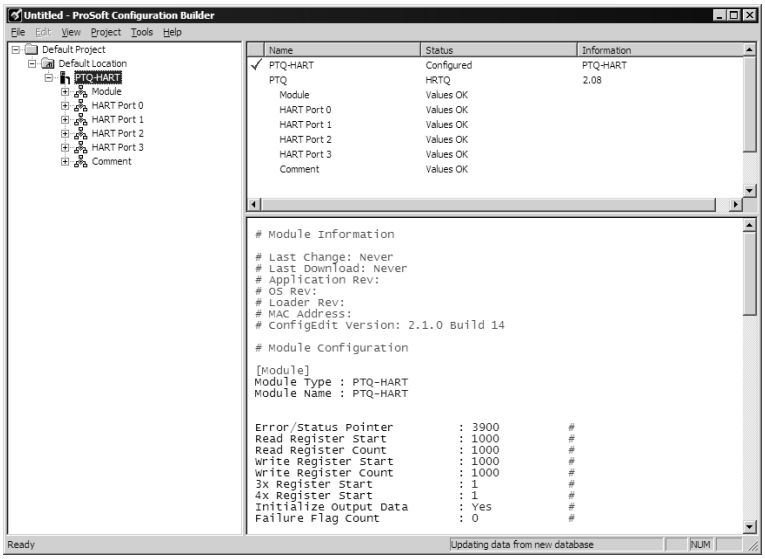


3 In the **PRODUCT LINE FILTER** area of the dialog box, select **PTQ.** In the **SELECT MODULE TYPE** dropdown list, select **PTQ-HART**, and then click **OK** to save your settings and return to the *ProSoft Configuration Builder* window.

The next task is to set the module parameters.

6.1.2 Renaming PCB Objects

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



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

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

Configuring Module Parameters

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

Printing a Configuration File

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

6.2 [Module]

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

6.2.1 Module Name

0 to **80** characters

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

6.2.2 Error/Status Offset

-1 or 0 to 3935

This parameter defines the database location where the module status data will be stored. If set to -1, data not placed in database.

6.2.3 Failure Flag Count

0 through **65535**

This parameter specifies the number of successive transfer errors that must occur before halting communication on the application port(s). If the parameter is set to **0**, the application port(s) will continue to operate under all conditions. If the value is set larger than **0** (**1** to **65535**), communications will cease if the specified number of failures occur.

6.2.4 Initializing Output Data

YES or **NO**

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

6.2.5 Read Register Start

0 to **4999**

The *Read Register Start* parameter specifies the start of the Read Data area in module memory. Data in this area will be transferred from the module to the processor.

Note: Total user database memory space is limited to the first 5000 registers of module memory, addresses 0 through 4999. Therefore, the practical limit for this parameter is 4999 minus the value entered for *Read Register Count*, so that the Read Data Area does not try to extend above address 4999. Read Data and Write Data Areas must be configured to occupy separate address ranges in module memory and should not be allowed to overlap.

6.2.6 Write Register Start

0 to **4999**

The *Write Register Start* parameter specifies the start of the Write Data area in module memory. Data in this area will be transferred in from the processor.

Note: Total user database memory space is limited to the first 5000 registers of module memory, addresses 0 through 4999. Therefore, the practical limit for this parameter is 4999 minus the value entered for *Write Register Count*, so that the Write Data Area does not try to extend above address 4999. Read Data and Write Data Areas must be configured to occupy separate address ranges in module memory and should not be allowed to overlap.

6.2.7 Read Register Count

0 to **4000**

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

6.2.8 Write Register Count

0 to **4000**

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

6.2.9 3x Register Start

1 to **N**

The 3x Register Start parameter defines the starting address in the processor's 3x (Quantum) or %iw (Unity) memory area to use for data being moved from the module. Take care to use a starting address that will accommodate the entire block from the module, but that will not overwrite data that is used for other purposes.

6.2.10 4x Register Start

1 to **N**

The 4x Register Start parameter defines the starting address in the processor's 4x (Quantum) or %iw (Unity) memory area to use for data being moved from the processor to the module. Take care to use a starting address that does not contain data in the processor's registers that is used for other purposes.

6.3 [HART PORT x]

The [HART PORT *x*] sections of the configuration file set the HART channel communication parameters, define the protocol specifics and set the command list parameters. The parameters are the same for all ports. The command list for each HART channel is entered in a different section of the file.

6.3.1 Enabled

Y or N

This parameter enables or disables the specific HART channel. If the parameter is set to "Y", the channel will be utilized. If set to "N", the channel will not be used.

6.3.2 Preambles

5 to 20

This parameter sets the number of preambles to be transmitted before each message is sent from the channel. The value of 5 is normally utilized for the parameter. It can be set to a value from 5 to 20.

6.3.3 Primary Master

Y or N

This parameter determines if the specific HART channel will emulate a primary or secondary master. You can have only one of each type on a HART network. If you plan on using a handheld device (secondary master), you must set the parameter to 'Y'. If the parameter is set to 'Y', the channel will act as the primary master. A value of 'N' will set the channel to act as a secondary master.

6.3.4 Retry Count

0 to 10

This parameter sets the number of retries for a command if the command response is not received from the slave device. This parameter is normally set to a value of 3. The module will accept values of 0 to 10.

6.3.5 DB Address Status

-1, 0 to 3999

This parameter is utilized to set the database address in the module where the status word for the channel will be placed. If the parameter is set to -1, the word value will not be placed in the database. If a value from 0 to 3999 is set for the parameter, the status word for the channel will be placed at the specified database offset. This word is bit mapped with each bit representing a slave device. The bit will be set if slave device has a communication error.

Slave List Status

The configuration parameter "DB Address Status" defines the register address in the virtual database where the status data for each HART channel will be placed. This word has one bit for each HART device and if this bit is in "1" it means that the corresponding HART device is not answering to the poll.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HART Device	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

The bit 0 of the "Slave List Status" is used for a slave connected in a point-to-point configuration. In this mode, the slave's 4 to 20 milliamp signal will also be active and can be utilized by an analog input or out module depending on the signal type.

6.3.6 Command Count

0 to 99

This parameter sets the number of user commands to be utilized. The first command in the list is always reserved for the auto-poll command so the user should configure this value considering one command for the auto-poll. For example, if the user configures two commands, the command count parameter should be set as 3. This parameter can be set from 0 to 99. If the parameter is set to a value other than 0, commands should be present in the [HART PORT x COMMANDS] section.

6.3.7 Auto-Poll Code

P, M or N

This parameter sets the auto-poll mode of the channel. If the parameter is set to P, the module will automatically poll device 0 in point-to-point mode. If the parameter is set to M, the module will automatically poll devices 1 to n (n=value of parameter Max Device Count). If the parameter is set to N, the auto-polling option will be disabled and only commands in the command list will be utilized for the channel. In the auto-poll mode, the module will automatically execute HART commands 0, 3, 13, 14 and 15.

6.3.8 Auto-Poll DB Address

0 to 3700

This parameter sets the starting address for the data obtained by the auto-poll feature. Each device on a channel requires 50 words in the database. The data area selected must not overlap any portion of the database used by other channels or the module.

6.3.9 Auto-Poll Swap Float

0 to 3

This parameter swaps the floating-point data values received by the auto-poll feature.

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

Depending on the host processor using the data, the proper swap code must be utilized to present the data in the correct format.

6.3.10 Max Device Count

1 to 15

This parameter sets the maximum number of slave devices to be utilized for the auto-poll feature. In point-to-point mode, the parameter should be set to a value of 1. In multi-drop mode, the parameter should be set from 1 to 15 to represent the number of slave devices attached to the channel.

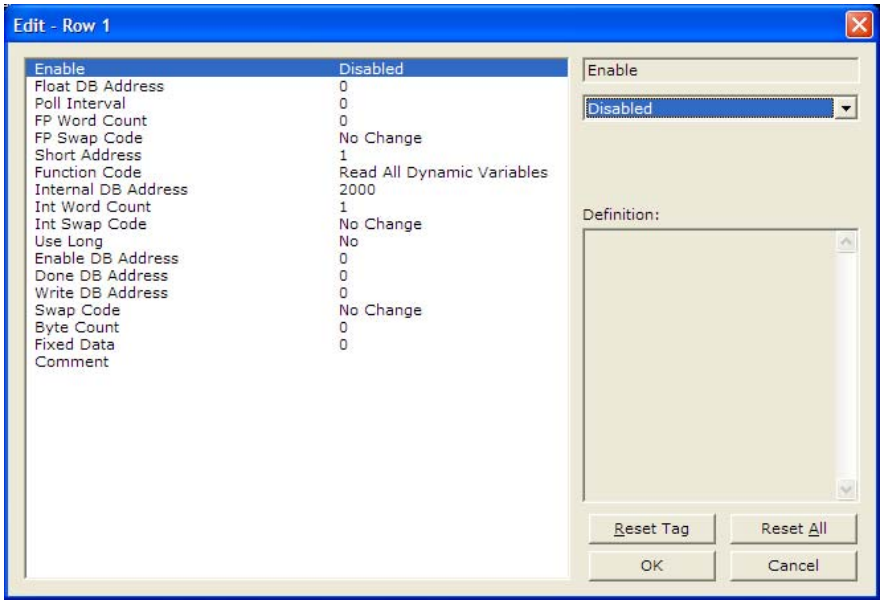
6.3.11 Error/Status Offset

-1 or 0 to 3935

This parameter defines the database location where the module status data will be stored. If set to -1, data not placed in database.

6.4 [HART PORT x COMMANDS]

The [HART PORT *x* COMMANDS] sections of the CFG file set the user defined HART channel command lists. These lists poll slave devices attached to the HART channels. The module supports numerous commands.



6.4.1 Command List Overview

The PTQ-HART module uses a command list to interface with HART slave devices. The commands in the list specify

- the slave device to be addressed
- the function to be performed (read or write)
- the registers in the internal database to be associated with the device data.

There is a separate command list for each HART channel, with up to 99 commands allowed per channel. The command list is processed from top (command #0) to bottom.

A poll interval parameter is associated with each command to specify a minimum delay time in seconds between the issuance of a command. For example, a poll interval of 10 executes the command no more frequently than every 10 seconds.

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

The module supports all the Universal (page 132) and Common Practice (page 145) commands, as well as device specific commands. A Device Specific command is supported without any translation of the data.

6.4.2 Commands Supported by the Module

The format of each command in the list is independent on the function being executed. All parameters in the command table must be entered. Only one parameter is optional, the Fixed Data field, which contains data to be sent to a HART device. The tables below list the functions supported by the module:

HART Universal Commands Set

Command	Definition
00	Read Unique Identifier
01	Read Primary Variable
02	Read Current And Percent Of Range
03	Read Dynamic Variables
06	Write Polling Address
07	Read Loop Configuration
08	Read Dynamic Variable Classifications
09	Read Device Variables with Status
11	Read Unique Identifier Associated With Tag
12	Read Message
13	Read Tag Descriptor Date
14	Read PV Sensor Info
15	Read Output Information
16	Read Final Assembly Number
17	Write Message
18	Write Tag Descriptor Date
19	Write Final Assembly Number
20	Read Long Tag
21	Read Unique Identifier Associated With Long Tag
22	Write Long Tag

HART Common Practice Commands Set

Command	Definition
33	Read Transmitter Variables
34	Write Damping Value
35	Write Range Values
36	Set Upper Range Value
37	Set Lower Range Value
38	Reset Configuration Changed Flag
39	EEPROM Control
40	Enter Exit Fixed Current Mode
41	Perform Transmitter Self Test
42	Perform Master Reset

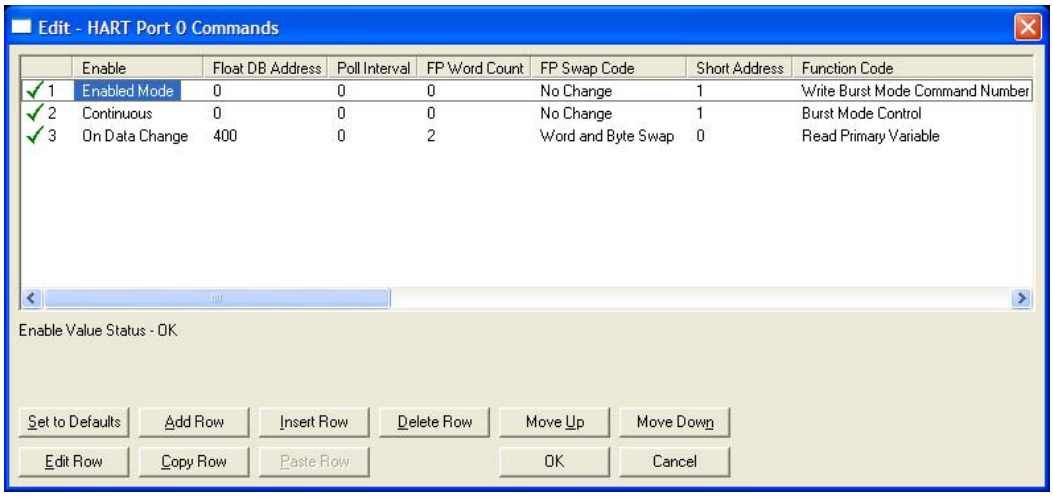
Command	Definition
43	Set PV Zero
44	Write PV Units
45	Trim DAC Zero
46	Trim DAC Gain
47	Write Transfer Function
48	Read Additional Transmitter Status
49	Write PV Sensor Serial Number
50	Read Dynamic Variable Assignments
51	Write Dynamic Variable Assignments
52	Set Transmitter Variable Zero
53	Write Transmitter Variable Units
54	Read Transmitter Variable Information
55	Write Transmitter Variable Damping Value
56	Write Transmitter Variable Sensor Serial Number
57	Read Unit Tag Descriptor Date
58	Write Unit Tag Descriptor Date
59	Write Number Of Response Preambles
60	Read Analog Channel and Percent of Range
61	Read Dynamic Variables and PV Analog Ch
62	Read Analog Channels
63	Read Analog Channel Information
64	Write Analog Channel Additional Damping Value
65	Write Analog Channel Range Values
66	Enter/Exit Fixed Analog Channel Mode
67	Trim Analog Channel Zero
68	Trim Analog Channel Gain
69	Write Analog Channel Transfer Function
70	Read Analog Channel Endpoint Values
71	Lock Device
72	Squawk
73	Find Device
74	Read I/O System Capabilities
75	Poll Sub-Device
76	Read Lock Device State
79	Write Device Variable
80	Read Device Variable Trim Points
81	Read Device Variable Trim Guidelines
82	Write Device Variable Trim Point
83	Reset Device Variable Trim
105	Read Burst Mode Configuration

Command	Definition
106	Flush Delayed Responses
107	Write Burst Device Variables
108	Write Burst Mode Command Number
109	Burst Mode Control
110	Read All Dynamic Variables

6.4.3 HART Command Entry Formats

Refer to HART Command Support (page 62) for a complete discussion of the HART commands supported by the module, and the structure and content of the data returned for each command.

The following illustration shows a command list section of the CFG file:



6.4.4 Enable

0, 1, 2, 3, 4, 5

This field defines whether the command is to be executed and under what conditions.

Code	Description
0	The command is disabled and will not be executed in the normal polling sequence. This can be used to process a command from a bursting slave device.
1	Causes the command to be executed each scan of the command list if the Poll Interval Time is set to zero. If the Poll Interval time is set, the command will be executed, when the interval timer expires.
2	The command will execute only if the internal data associated with the command changes. This value is valid only when there is a specified "Write DB Address" (see below) with a non zero byte count for write commands.
3	The HART module will send the command if either the PTQ-HART module OR the HART device is powered up. This is mainly used for configuration of HART devices on startup.
4	Places the command in enabled mode. This option is valid only if there is a specified "Enabled DB Address" (see below). If the Virtual Database word specified in "Enabled DB Address" has "-1" the command will be executed otherwise it will not.
5	Places the command in one shot enabled mode. This option is valid only if there is a specified "Enabled DB Address" (see below). If the Virtual Database word specified in "Enabled DB Address" has a value of "-1" the command will be executed otherwise it will not. When the command has been successful the Virtual Database word specified in "Enabled DB Address" will be written with "0", so the command will be executed only once.

Refer to Command Enable Control Block (9902) (page 103) and Command Disable Control Block (9903) (page 103) for more information on how to use the enable code.

6.4.5 Float DB Address

0 to 3998

This field specifies the internal database register where the floating point values returned by the command will be placed.

6.4.6 Poll Interval

0 TO 65535

This parameter specifies the minimum interval between executions of a continuous commands (*Enable* code of 1). The value is in seconds. Therefore, if a value of 10 is entered, the command will execute no more frequently than once every 10 seconds.

6.4.7 FP Word Count

-1 to 125

This parameter specifies the number of words from the floating point data returned by a HART command that will be placed on the Virtual Database. If this parameter is 0 no data will be written to the Database. If this parameter is -1 then all the floating point data will be written in the integer block of data.

Special care should be taken with this number, because is a word count and a floating point value is 2 words long. For example if you execute a HART command 3 which takes 5 floating point values from the device, you should place a word count of 10 words.

6.4.8 Swap Code

0, 1, 2, 3

This parameter defines the byte order of each four-byte group of data received. This parameter is helpful when dealing with floating-point or other multi-register values, as there is no standard byte order for storing these data types. The following table describes the values and their associated operations:

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

6.4.9 Short Address

1 to 15

This parameter specifies the HART slave node address on the network to be considered. Values of 1 to 15 are permitted. If the device to be addressed only accepts long address, then the parameter "Use Long" should be selected so the module can ask for the long address with the short one and then execute the command.

6.4.10 Function Code

0 to 255

This parameter specifies the HART function to be executed. Any HART function can be executed, even device specific ones, but only supported commands will return formatted data and classified in floating point data and integer data.

6.4.11 Int. DB Address

0 to 3999

This field specifies the internal database register where the integer or packed ASCII string values returned by the command will be placed.

If the Floating Point Word Count parameter is "-1", then all the data returned by the HART command will be placed in this address without any formatting.

6.4.12 Int Word Count

0 to 125

This parameter specifies the number of words from the integer or packed ASCII string data returned by a HART command that will be placed on the Virtual Database. If this parameter is "0", no data will be written to the Database.

6.4.13 Swap Code

0, 1, 2, 3

This parameter defines the byte order of each four-byte group of data received. This parameter is helpful when dealing with floating-point or other multi-register values, as there is no standard byte order for storing these data types. The following table describes the values and their associated operations:

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

6.4.14 Use Long

0 or 1

This parameter defines if the command will be executed with short or long address. If the value is "0", then the configured command will be executed using the Short Address specified. If the value is "1" then the specified Short Address will be used only to ask for the long address and that will be used to execute the configured command.

6.4.15 Enable DB Address

0 to 3999

This field specifies the internal database register to be used to enable the execution of a command. This parameter is only used if "Enable" is "4" or "5". If the value of this database register is "-1", then the command will be executed, otherwise it will not. If the "Enable" value is "5", then after the successful execution of the command this value will become "0"

6.4.16 Done DB Address

-1 to 3999

This field specifies the internal database register to be used to signal the successful execution of a command. When a command is successfully executed a "-1" is written to this register. This "Done DB Address" can be shared with the "Enable DB Address" of another command to do a chained command execution.

6.4.17 Write DB Address

-1 to 3999

This field specifies the internal database register to be as a source of data for HART command which includes data. It is possible to include data with every HART command, but it depends of the command and of the device if it will accept this data.

If this value is "-1", it indicates that there will not be data with the command or that it will not come from database.

6.4.18 Swap Code

0, 1, 2, 3

This parameter defines the byte order of each four-byte group of data received. This parameter is helpful when dealing with floating-point or other multi-register values, as there is no standard byte order for storing these data types. The following table describes the values and their associated operations:

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

6.4.19 Byte Count

0 to 250

This parameter specifies the number of bytes to be sent to a HART device in the command. If the command has no data then this value should be "0".

If the value of this field is different of "0" and "Write DB Address" is different of "-1" then the data for the command will be taken from the Virtual Database. If the value of this field is different of "0" and "Write DB Address" is "-1" then the data for the command will be from the "Fixed Data" field for the command.

6.4.20 Fixed Data

Up to 250 HEX values separated by space

This parameter is a string of HEX values to be sent with the HART command. There should be at least the number of bytes specified in the "Byte Count" parameter. The bytes should be written in hexadecimal format and separated by a space. This data will be sent if the Parameter "Write DB Address" is "-1" and "Byte Count" is greater than "0", and it will be sent in the same order that they are written.

6.5 Hart Command Examples

This section describes two examples that shows how to configure HART commands. The first example shows a read command (function code 3 - READ DYNAMIC VARIABLES) and the second example shows a write command (function 34 - WRITE DAMPING VALUE).

6.5.1 Example of HART Command Function 3

The following example shows how to configure a command function 3 (READ DYNAMIC VARIABLES) to read the process variables from the HART slave device. According to the HART specification, this command will return four floating-point variables:

Word	High Byte	Low Byte
0	Current (mA)	
1		
2	Primary Variable	
3		
4	Second Variable	
5		
6	Third Variable	
7		
8	Fourth Variable	
9		

The command also returns the following integer data:

Word	High Byte	Low Byte
0	Status Word	
1	Primary Variable Units Code	Second Variable Units Code
2	Third Variable Units Code	Fourth Variable Units Code

So, this command will return the following number of words:

Parameter	Data Type	Direction	Number of Words	Database Address (this example)
Variable Results	Floating Point	Read from slave to MVI	10	240 (word address)
Status/Unit Codes	Integer	Read from slave to MVI	3	290 (word address)

The user can configure the command parameters as described in the following table in order to correctly read the command 3 results to the module database:

Index	Parameter	Value	Observation
1	Enable	1	The command is sent continuously
2	Float DB Address	240	The floating point results will be copied to the internal database starting at word-address 240 (from 240 to 249)
3	Poll Interval	0	The command is sent without any delay
4	Word Count	10	The command returns 10 words of floating point data

Index	Parameter	Value	Observation
5	Swap Code	3	The bytes in each returned floating point word are swapped
6	Short Address	0	This command is sent to the Hart slave device using short address 0
7	Function Code	3	The command function 3 (READ DYNAMIC VARIABLES) is used in this example
8	Int. DB Address	290	The integer data will copied to the database starting at word-address 290 (from 290 to 292)
9	Word Count	3	The command returns 3 words of integer data
10	Swap Code	3	The bytes in each returned integer data will be swapped
11	Use Long	1	This command executes using Long Address
12	Enable Address	-1	This command does not use this feature
13	DB Done Address	-1	This command does not use this feature
14	DB Write Address	-1	This parameter is ignored since no data is sent to the HART device (this is a read command)
15	DB Swap Code	3	This parameter is ignored since no data is sent to the HART device (this is a read command)
16	Byte Count	0	This parameter is ignored since no data is sent to the HART device (this is a read command)
17	Fixed Data		This command does not use this feature

So these values could be entered in the configuration file as described below:

```
START
#     1        2        3     4    5       6    7       8     9   10   11      12      13       14      15    16    17
#       Float DB     Poll  Word Swap   Short Func Int. DB  Word Swap  Use  Enable DB Done DB Write DB Swap  Byte Fixed
#Enable  Address Interval Count Code Address Code Address Count Code Long Address Address  Address    Code  Count Data
      1      240        0    10    3       0    3     290     3    3    1      -1      -1       -1       3      0
END
```

In this example, the user should verify that the command result is located inside the Read Data area, in order to have the data copied to the PLC processor (through the backplane). The following backplane configuration could be used for this example:

```
Read Register Start :    0 #Starting DB address where read by processor
Read Register Count :  300 #Number of regs for processor to read
Write Register Start:  300 #Starting DB address where write data placed
Write Register Count:  300 #Number of regs to write to module from processor
```

6.5.2 Example of HART Command Function 34

The following example shows how to configure a command function 34 (WRITE DAMPING VALUE) to write a damping values (seconds) to the HART slave device.

According to the HART specification, this command will write one floating point variables:

Word	High Byte	Low Byte
0	Floating Point Damping Value (Sec)	
1		

The HART device should also return the floating point value after it processes the request:

Word	High Byte	Low Byte
0	Floating Point Damping Value (Sec)	
1		

The HART device also returns an integer status data:

Word	High Byte	Low Byte
1	STATUS WORD	

This command will return the following number of words:

Parameter	Data Type	Direction	Number of Words	Database Address (this example)
Damping Value	Floating Point	Written from MVI to slave	2 (4 bytes)	400 (word address) or 800 (byte address)
Damping Value	Floating Point	Read from slave to MVI	2	240 (word address)
Status Word	Integer	Read from slave to MVI	1	290 (word address)

The user can configure the command parameters as described in the following table.

Index	Parameter	Value	Observation
1	Enable	1	The command is sent continuously
2	Float DB Address	300	The damping value response will be copied to the internal database starting at word-address 300 (occupies addresses 300 to 301)
3	Poll Interval	0	The command is sent without any delay
4	Word Count	2	The command returns 10 words of floating point data
5	Swap Code	3	The bytes in each returned floating point word are swapped
6	Short Address	0	This command is sent to the Hart slave device using short address 0
7	Function Code	34	The command function 4 (WRITE DAMPING VALUE) is used in this example
8	Int. DB Address	320	The integer data will copied to the database word-address 320

Index	Parameter	Value	Observation
9	Word Count	1	The command returns 1 word of integer data
10	Swap Code	3	The bytes in each returned integer data will be swapped
11	Use Long	1	This command executes using Long Address
12	Enable Address	-1	This command does not use this feature
13	DB Done Address	-1	This command does not use this feature
14	DB Write Address	800	The floating point value located at byte-address 800 (word-address 400) will be used as the damping value to be written to the HART device
15	DB Swap Code	3	The bytes in each returned integer data will be swapped
16	Byte Count	4	The command will write 4 bytes (2 words) to the HART device, since the damping value uses floating point format (2 words)
17	Fixed Data		This command does not use this feature

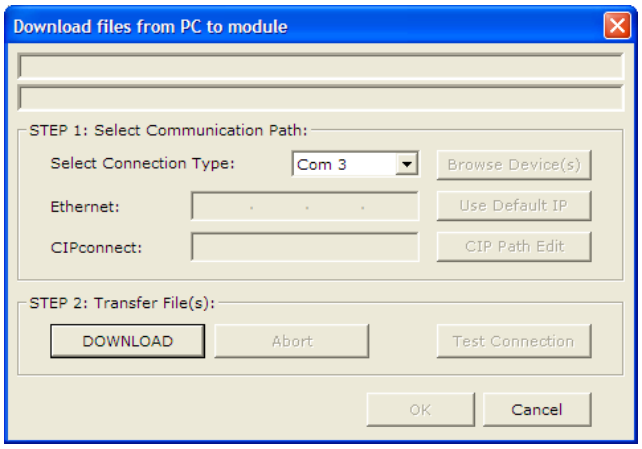
```
START
#      1        2        3     4    5       6    7       8     9   10   11      12      13       14      15    16    17
#        Float DB     Poll  Word Swap   Short Func Int. DB  Word Swap  Use  Enable DB Done DB Write DB Swap  Byte Fixed
# Enable  Address Interval Count Code Address Code Address Count Code Long Address Address  Address    Code Count  Data
       1      300        0     2    3       0   34     320     1    3    1      -1      -1       800       3     4
END
```

The user would have to verify that addresses 300 and 320 are located inside the Read Data area (read from the MVI database to the PLC processor). Address 400 would have to be located inside the Write Data area (written from the PLC processor to the MVI database).

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

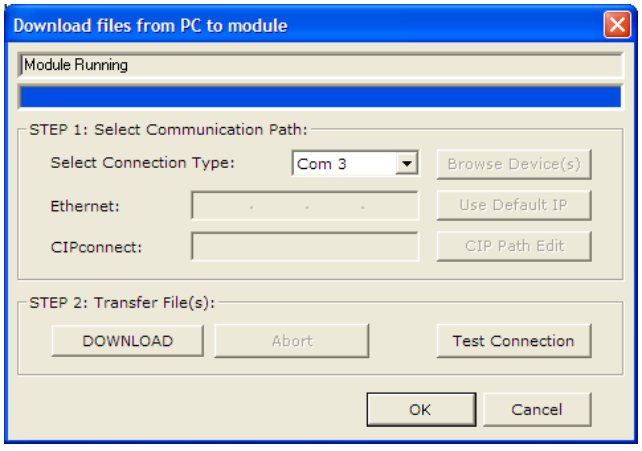
For the module to use the settings you configured, you must download (copy) the updated *Project* file from your PC to the module.

1. In the tree view in *ProSoft Configuration Builder*, click once to select the module.
2. Open the *Project* menu, and then choose **MODULE/DOWNLOAD**. The program will scan your PC for a valid com port (this may take a few seconds). When *PCB* has found a valid COM port, the *Download* dialog box will open.



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

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



7 Diagnostics and Troubleshooting

In This Chapter

- LED Status Indicators 76
- Using ProSoft Configuration Builder (PCB) for Diagnostics 77
- Reading Status Data from the Module 91

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

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

7.1 LED Status Indicators

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

LED	Color	Status	Indication
ACTIVE	Green	On	The LED is set to ON after a power up after the module recognizes the presence of the processor in the rack.
		Off	The LED is set to OFF if the module does not recognize the processor after a power up
BAT LOW	Red	On	The battery voltage is low or the battery is not present. The battery LED will illuminate briefly upon the first installation of the module or if the unit has been un-powered for an extended period of time. This behavior is normal, however should the LED come on in a working installation please contact ProSoft Technology.
		Off	The battery voltage is OK and functioning.
DEBUG	Red	On	ON: Indicates serial activity while the PC is communicating with the DEBUG port of the module through an ASCII terminal. This port is typically used for module configuration, status monitoring and troubleshooting access.
		Off	No serial activity at DEBUG port
CFG ERR	Red	On	Configuration error detected. Check configuration parameter values.
		Off	No configuration error.
ERR 1	Red	On	Application error (includes backplane failure caused by processor in STOP mode)
		Off	No application error
ERR 2	N/A	N/A	Not used

If your module is not operating, and the status LEDs are not illustrated in the table above, please call ProLinx Communications Gateways for technical assistance.

7.2 Using ProSoft Configuration Builder (PCB) for Diagnostics

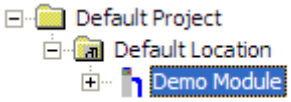
The *Configuration and Debug* menu for this module is arranged as a tree structure, with the *Main* menu at the top of the tree, and one or more submenus for each menu command. The first menu you see when you connect to the module is the *Main* menu.

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

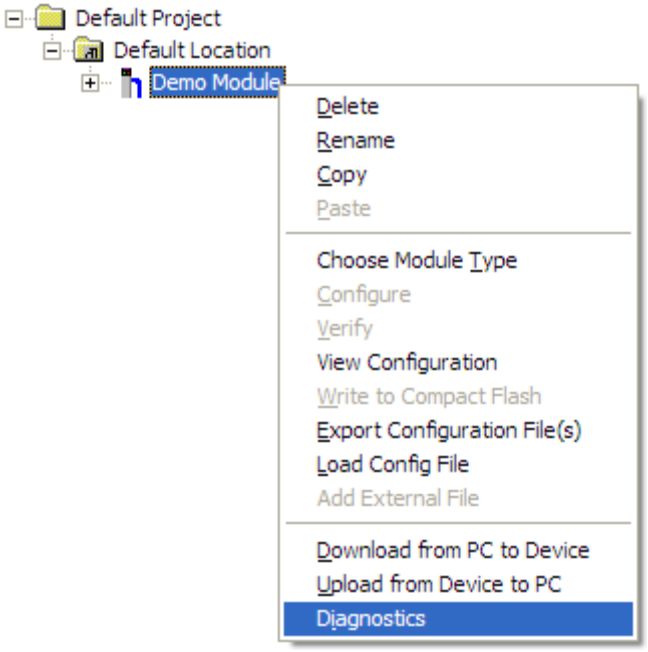
7.2.1 Using the Diagnostic Window in ProSoft Configuration Builder

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

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

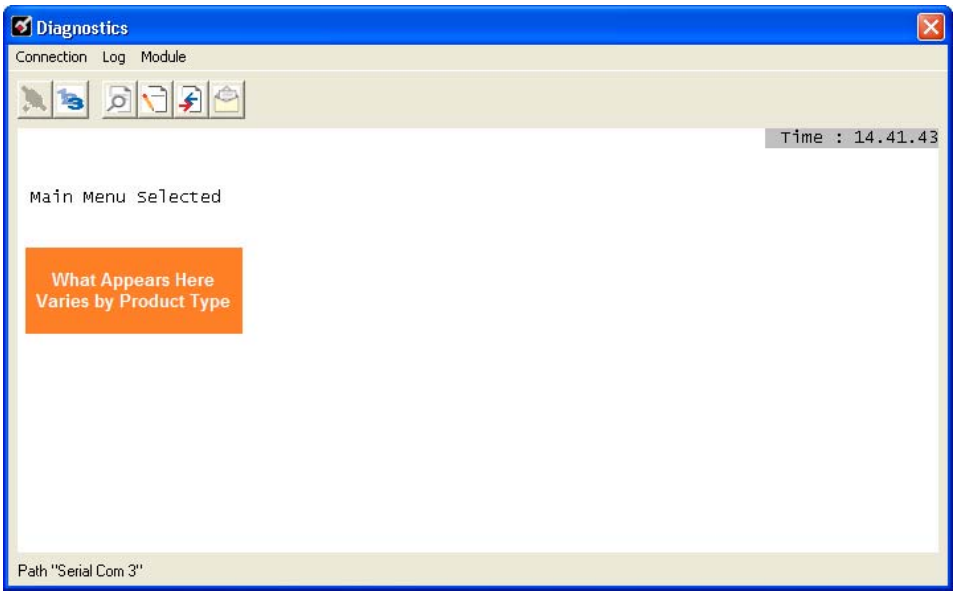


2 On the shortcut menu, choose **DIAGNOSTICS.**



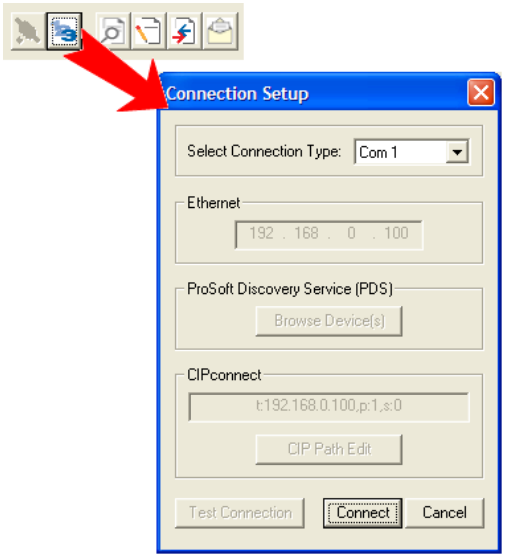
This action opens the *Diagnostics* dialog box.

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



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

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



2 Verify that the null modem cable is connected properly between your computer's serial port and the module. A regular serial cable will not work.

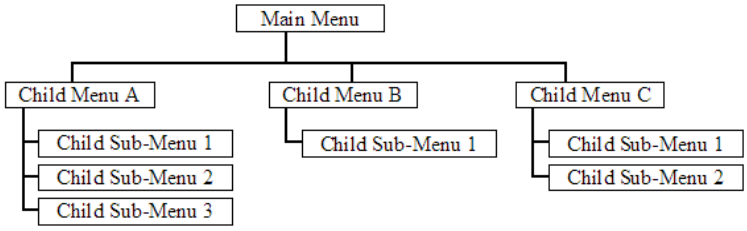
3 On computers with more than one serial port, verify that your communication program is connected to the same port that is connected to the module.

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

7.2.2 Navigation

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

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



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

Keystrokes

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

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

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

7.2.3 Main Menu

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.

```
HART COMMUNICATION MODULE MENU
  ?=Display Menu
  B=Block Transfer Statistics
  C=Module Configuration
  D=Modbus Database View
  H=HART Menu
  R=Transfer Configuration from PC to MVI Unit
  S=Transfer Configuration from MVI Unit to PC
  V=Version Information
  W=Warm Boot Module
  Esc=Exit Program
```

Caution: Some of the commands available to you from this menu are designed for advanced debugging and system testing only, and can cause the module to stop communicating with the processor or with other devices, resulting in potential data loss or other 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.

Viewing Block Transfer Statistics

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

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

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

Viewing Module Configuration

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

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

Opening the Database View Menu

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

Use this menu command to view the current contents of the module's database. For more information about this submenu, see Database View Menu (page 89).

Opening the HART Master Menu

Press **[H]** to open the HART Master Menu. This menu allows you to view information about the protocol driver.

```
HART INTERFACE MENU
  ?=Display Menu
  A=HART Data Analyzer
  V=Slave Status List
  M=Main Menu
  HART Command List Errors:
      E=Port 0   F=Port 1   G=Port 2   H=Port 3
  HART Command List:
      N=Port 0   O=Port 1   P=Port 2   Q=Port 3
  Port Status and Configuration:
      1=Port 0   2=Port 1   3=Port 2   4=Port 3
```

Transferring the Configuration File from the PC to the Module

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

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

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

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

Transferring the Configuration File from The Module to the PC

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

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

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

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

Viewing Version Information

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

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

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

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

Warm Booting the Module

Press **[W]** from the *Main* menu to warm boot (restart) the module.

This command will cause the program to exit and reload, refreshing configuration parameters that must be set on program initialization. Only use this command if you must force the module to reboot.

Exiting the Program

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

7.2.4 HART Master Menu

```
HART INTERFACE MENU
  ?=Display Menu
  A=HART Data Analyzer
  V=Slave Status List
  M=Main Menu
  HART Command List Errors:
      E=Port 0   F=Port 1   G=Port 2   H=Port 3
  HART Command List:
      N=Port 0   O=Port 1   P=Port 2   Q=Port 3
  Port Status and Configuration:
      1=Port 0   2=Port 1   3=Port 2   4=Port 3
```

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.

Opening the Data Analyzer Menu

Press **[A]** to open the Data Analyzer Menu. Use this command to view all bytes of data transferred on each port. Both the transmitted and received data bytes are displayed. Refer to Data Analyzer (page 85) for more information about this menu.

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

Viewing the Slave Status List

Press **[V]** to view the slave status values associated with the ports. The slave status values are defined as follows:

- ERR = Device in Error
- OK = Device OK
- [Blank] = Device Not Polled.

Opening the Command Error List Menu

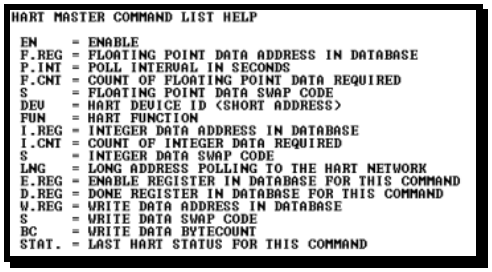
Press **[E], [F], [G]** or **[H]** to open the Command Error List for clients 1 through 4 respectively. This list consists of multiple pages of command list error/status data. Press **[?]** to view a list of commands available on this menu.

Opening the Command List Menu

Press **[N], [O], [P]** or **[Q]** to open the Command List menu for clients 1 through 4 respectively. Use this command to view the configured command list for the module.

Viewing the Master Command List Help

Press **[H]** to view a help screen with explanations of each item that appears on the command list.



Viewing Port Status and Configuration

Press **[1], [2], [3],** or **[4]** to view status and configuration for ports 0 through 3 respectively.

HART Error Descriptions

Error Type	Description
Gap Errors	Increments when a delay of more than 20 milliseconds occurs between characters in a HART message
Overflow	Increments when a received HART message is longer than the internal buffer can hold
SOM Errors	**Start of Message** error - Increments whenever the module does not see at least 3 preambles (FF characters or all bits set ON) at the beginning of a HART message
Retry Count	Increments every time a HART command fails and is retried
Check Byte	Increments when a Checksum error is detected in the received HART packet.
Overrun/Parity/Frame	Increments every time one of three errors occur: **Overrun** - The HART driver wasn't able to read the current data byte before a new one arrived, causing the current character to be lost. **Parity -** HART communications uses EVEN parity. A byte is received with parity error. **Frame** - There is a zero bit where the stop bit should be. The message is not formatted correctly. These errors are typically caused by electrical wiring problems or electrical interference on the network loop.
Response Timeout	Increments when the response to a HART Command is not received within the configured timeout period.

Returning to the Main Menu

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

7.2.5 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.

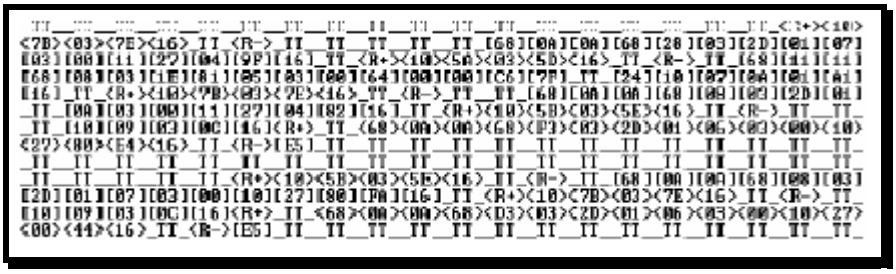
```
HART DATA ANALYZER VIEW MENU
 ?=Display Menu
 1=Select HART Port 0
 2=Select HART Port 1
 3=Select HART Port 2
 4=Select HART Port 3
 C=1 mSec Ticks
 D=5 mSec Ticks
 E=10 mSec Ticks
 F=50 mSec Ticks
 G=100 mSec Ticks
 0=No mSec Ticks
 H=Hex Format
 A=ASCII Format
 B=Start
 S=Stop
 Z=HART Menu

 Port = HART PORT 0, Format=HEX, Tick=0
```

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.



Displaying Timing Marks in the Data Analyzer

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

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

Removing Timing Marks in the Data Analyzer

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

Viewing Data in Hexadecimal Format

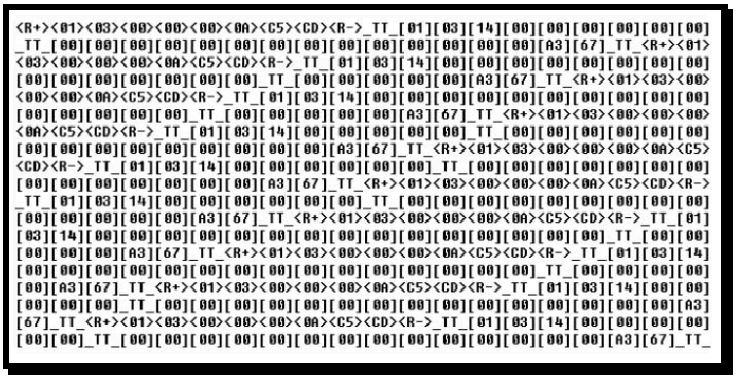
Press **[H]** from the *Database View* menu to display the data on the current page in hexadecimal format.

Viewing Data in ASCII (Text) Format

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

Starting the Data Analyzer

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



The Data Analyzer displays the following special characters:

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

Stopping the Data Analyzer

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

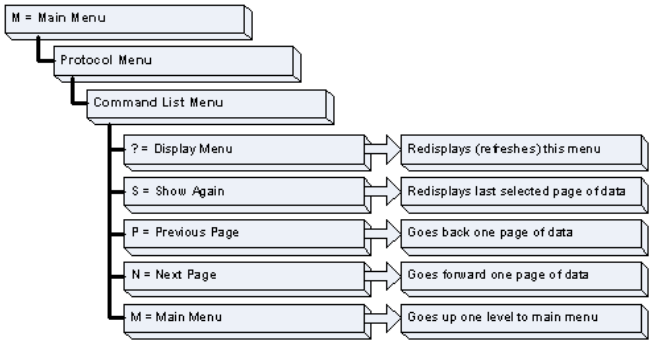
Important: When in analyzer mode, program execution will slow down. Only use this tool during a troubleshooting session. Before disconnecting from the Config/Debug port, please press [S] to stop the data analyzer, and then press [M] to return to the main menu. This action will allow the module to resume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

Returning to the Main Menu

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

7.2.6 Master Command Error List Menu

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



Redisplaying the Current Page

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

Moving Back Through 5 Pages of Commands

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

Viewing the Previous Page of Commands

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

Moving Forward (Skipping) Through 5 Pages of Commands

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

Viewing the Next Page of Commands

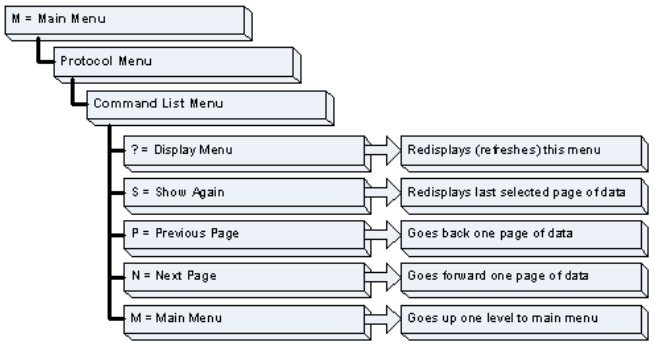
Press **[N]** to display the next page of commands.

Returning to the Main Menu

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

7.2.7 Master Command List Menu

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



Redisplaying the Current Page

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

Viewing the Previous 50 Commands

Press **[-]** to view the previous 50 commands.

Viewing the Previous Page of Commands

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

Viewing the Next 50 Commands

Press **[+]** to view the next 50 commands from the master command list.

Viewing the Next Page of Commands

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

Returning to the Main Menu

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

7.2.8 Database View Menu

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

```
DB Menu Selected

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

Viewing Register Pages

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

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

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

Displaying the Current Page of Registers Again

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

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

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

Moving Back Through 5 Pages of Registers

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

Moving Forward (Skipping) Through 5 Pages of Registers

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

Viewing the Previous Page of Registers

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

Viewing the Next Page of Registers

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

Viewing Data in Decimal Format

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

Viewing Data in Hexadecimal Format

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

Viewing Data in Floating-Point Format

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

Viewing Data in ASCII (Text) Format

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

Returning to the Main Menu

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

7.3 Reading Status Data from the Module

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

8 Reference

In This Chapter

- Product Specifications 94
- Functional Overview 97
- RS-232 Configuration/Debug Port 115
- Application Port Connection 116
- Example Field Terminations 117
- Thermal Connector Port Details 124
- PTQ-HART Error Status Table 125
- HART Universal Commands 132
- HART Common Practice Commands 145
- Frequently Asked Questions 175

8.1 Product Specifications

The PTQ HART Multi-drop Master Communication Module is an Quantum backplane-compatible module that allows Quantum processors to interface easily with HART compatible devices. Devices commonly supporting the protocol include pressure, temperature, flow transmitters, as well as other similar instruments commonly found in the process (and other) industry.

The PTQ-HART Master Communication Module allows Schneider Electric Quantum compatible processors to easily communicate with HART slave devices. The PTQ-HART module interfaces up to 15 devices on each HART channel with the Quantum processor. HART channels on the module support master protocol commands to interface with slave devices on their own networks. Each port is individually configurable. Data is exchanged between the HART network and the Quantum processor backplane using the internal database contained in the module and direct control by the processor's ladder logic and pre-defined data objects (5000 registers maximum).

The PTQ-HART module is the perfect solution for industrial applications in chemical and refining operations, to gas and liquid distribution systems, and remote offshore monitoring stations are addressing virtually all aspects of control, data acquisition, and maintenance.

HART® is a registered trademark of the HART Communication Foundation

8.1.1 General Specifications

- Single Slot - Quantum backplane compatible
- The module is recognized as an Options module and has access to PLC memory for data transfer
- Configuration data is stored in non-volatile memory in the ProTalk module
- Up to six modules can be placed in a rack
- Local rack - The module must be placed in the same rack as processor
- Compatible with all common Quantum programming packages, including Concept (version 2.6 or higher), Unity Pro (version 2.2 or higher), ProWORX (version 2.20 or later), and ModSoft
- Quantum data types supported: 3x, 4x
- High speed data transfer across backplane provides quick data update times
- Sample ladder file available

8.1.2 Hardware Specifications

Specification	Value
Backplane Current Load	1100 mA maximum @ 5 Vdc ± 5%
Operating Temperature	0°C to 60°C (32°F to 140°F)
Storage Temperature	-40°C to 85°C (-40°F to 185°F)
Relative Humidity	5% to 95% (without condensation)
Vibration	Sine vibration 4-100 Hz in each of the 3 orthogonal axes
Shock	30g, 11 msec. in each of the 3 orthogonal axes
Dimensions (HxWxD), Approx.	250 x 103.85 x 40.34 mm 9.84 x 4.09 x 1.59 in
LED Indicators	Module Status Backplane Transfer Status Serial Port Activity Serial Activity and Error Status
Debug/Configuration Port (Debug)	
CFG Port (DEBUG)	DB-9M PC Compatible RS-232 only No hardware handshaking
Application Ports	
Application Serial Ports (PRT1, PRT2)	DB-9M PC Compatible RS-232/422/485 jumper selectable RS-422/485 screw termination included RS-232 handshaking configurable 500V Optical isolation from backplane

8.1.3 Functional Specifications

The PTQ-HART module supports the HART Multi-drop implementation of the protocol. Following are some general specifications for the module:

- Built in accordance to the HART Bell 202 Frequency Shift Keying (FSK) standard to superimpose digital signals at a low level on top of the 4 to 20mA
  - Four independent HART master ports that are completely user-configurable
  - Supports up to 15 devices per port
  - Point-to-point (slave address 0), and Multi-drop (slave address 1 to 15) modes supported
- Supports 99 universal and common practice commands per port to control and monitor devices with integer, IEEE754 floating-point and packed ASCII character string data blocks
- Burst mode can be used for faster update of data from a single slave
- Supports an auto polling feature that will automatically collect data from each HART instrument on the channel and store the data in the module database
- Communication ports can be configured as a secondary master (that is, handheld configuration device)

Protocol Supported: HART protocol uses the Bell 202 standard frequency shift-keying (FSK) digital signal to communicate at 1200 baud, superimposed at a low level on the 4 to 20mA analog measurement signal. The PTQ-HART module supports version 5 of the HART protocol.

Supported Function Codes: HART Universal Commands Set supported are 00 to 03, 06 to 09, and 11 to 22. HART Common Practice Commands Set supported are 33 to 83 and 105 to 110.

HART Network Communications: Supports four master channels. Each channel on the module is configured independently to emulate a HART master. Burst mode can be used for faster update of data from a slave device.

Command polling is also user-configurable, including disabled, continuous, on change of data (write only), and dynamically user or automatic enabled.

Status: Error codes returned by the HART protocol available on an individual command basis. In addition, a slave status list is maintained per active channel.

8.2 Functional Overview

8.2.1 General Concepts

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

Module Power Up

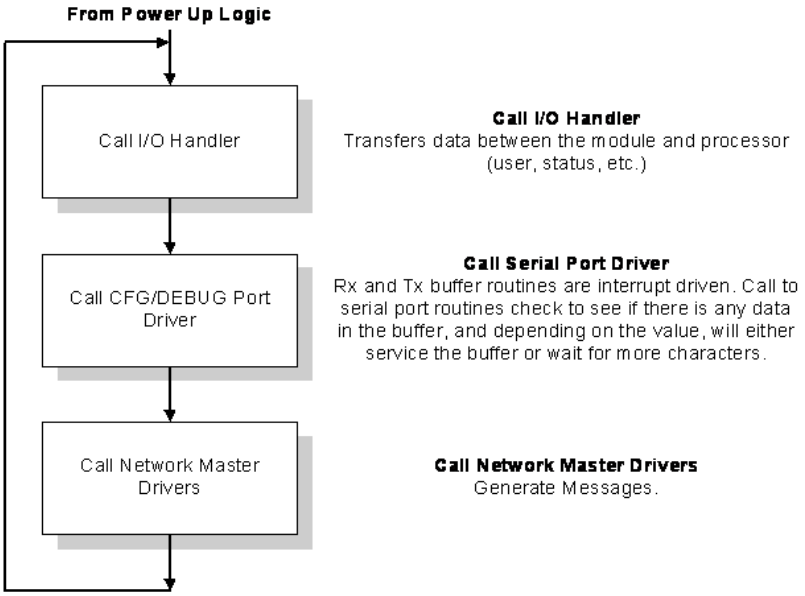
On power up, the module begins performing the following logical functions:

1. Initialize hardware components
   - Initialize Quantum backplane driver
   - Test and clear all RAM
   - Initialize the serial communication ports
2. Read configuration for module from HART.CFG file on Compact Flash Disk
3. Initialize the databases and ports
4. Set up the communication interface for the debug/configuration port

After the module has received the configuration, the module begins receiving and transmitting messages with devices on the serial networks.

Main Logic Loop

Upon completing the power up configuration process, the module enters an infinite loop that performs the following functions:



8.2.2 Backplane Data Transfer

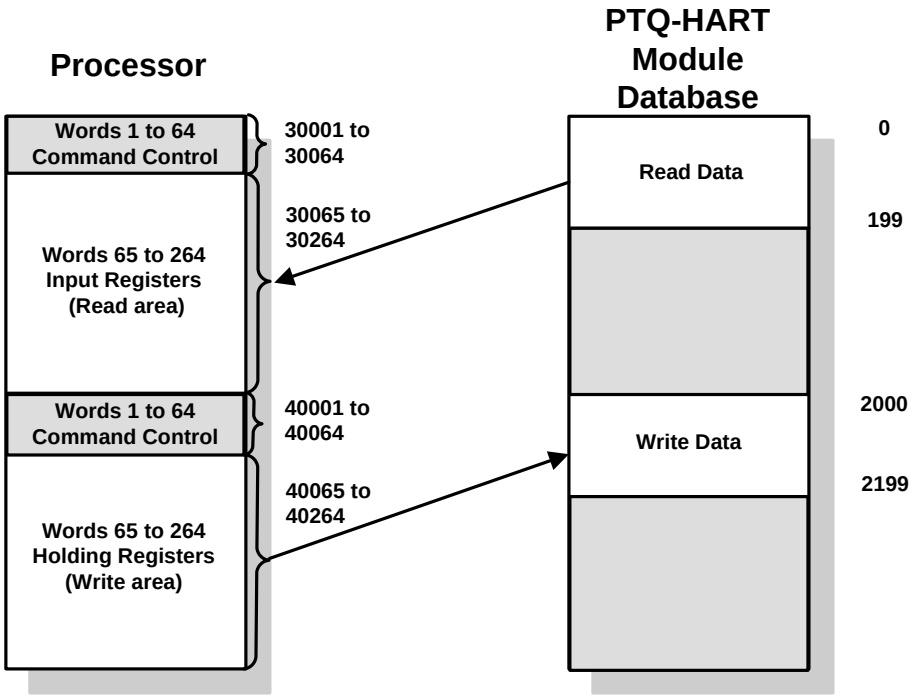
The current version of the PTQ-HART backplane driver (version 2.10 or newer), uses a Large I/O model, which differs from previous versions of the backplane driver in that it transfers all of the data in the Read and Write databases between the module and the processor on every scan.

The [Module] section of the configuration file defines the starting registers for read and write operations, as well as the number of registers to use for each data area.

```
#Used to define the data areas moved between the module and the processor
Read Register Start :     0  #Database start register to move to processor
Read Register Count :   200  #Number of words moved from module to processor
Write Register Start:  2000  #Database start register where data placed from
                              #processor
Write Register Count:   200  #Number of words moved from processor to module

# Used to define the area in the processor for the module to interface with.
# The first 64 words are reserved for special block (control block)
# functionality. At offset word 65 starts the actual database transfer area.
# For this example, 200 words of data will be copied starting from processor
# address 40065 (Quantum) or %MW65 (Unity)
# to PTQ database starting at address 2000. Also, 200 words of data will be
# copied starting from PTQ database address 0
# to processor at input register starting address 30065 (Quantum) or %IW65
# (Unity)
3x Register Start:        1  #3x start register where data moved from module
                              #to processor (1-n)
4x Register Start:        1  #4x start register where data moved from
processor
                              #to module (1-n)
```

The values in the example configuration file section above are illustrated in the following diagram.



The module transfers the entire read and write areas at the end of every processor scan. The module will hold the processor scan for a certain period of time, which allows the module to transfer the entire read and write areas. This means that the larger the read and write areas, the longer the processor scan time will be. Refer to Module Performance for more detailed information on determining scan times.

Note: The diagram above shows the memory addresses for a Quantum processor. If you are deploying the PTQ-HART with a Unity processor, substitute %MW for read only data, and %IW for read/write data.

Data Exchange

The module transfers all the configured read or write data at the end of each processor scan. You can configure up to 4000 words in each direction. The more data you configure, the longer the processor scan will be.

Words 0 through 63 in each read/write block are reserved for command control. Refer to Command Control (page 101) for more information on command control blocks. The following table shows the relationship between the processor memory and the module database areas.

Note: Refer to Backplane Data Transfer (page 98) for the example configuration values that are used in the following tables.

Module Database	Quantum Register	Unity Register	Description
Read Data	3x	%IW	Input Register
Write Data	4x	%MW	Holding Register

The data mapping in the following example shows the relationship between processor and PTQ-HART memory addresses, assuming a 4x register start value of 40001 and a PTQ-HART database start value of 0.

Processor Memory Address	Module Database Address
40065	0
40066	1
40067	2
40068	3
40069	4
…	…
40164	99

The data mapping in the following example shows the relationship between processor and PTQ-HART memory addresses, assuming a 3x register start value of 30001 and a PTQ-HART database start value of 2000.

Processor Memory Address	Module Database Address
30065	2000
30066	2001
30067	2002
30068	2003
30069	2004
…	…
30164	2099

8.2.3 Command Control

The first 64 words of each block are reserved for command control. Each command control block has a Block ID number (shown in parentheses below) that identifies the command control instruction. The PTQ-HART module supports the following command control blocks:

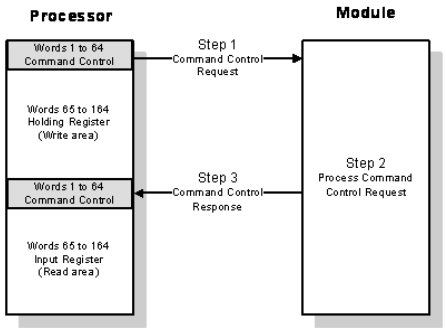
- Block ID 9902 - Command Enable Control (page 103)
- Block ID 9903 - Command Disable Control (page 103)
- Block ID 9998 or 9999 - Reboot Module (page 104)

The value in word 0 of this 64 word block is the block sequence number. This number identifies whether the contents of the block have changed. This is the actual trigger to send the control request to the module.

Processor logic must be built to handle the command control functionality. The logic would typically follow these steps:

1. Move the block request to output command control area.
2. Move a new value to the output block sequence number.
3. If the input block sequence number equals the output block sequence number + 1, copy the block response to appropriate variables in the module's memory.

Note: Command Control blocks are not copied to the module database. You must define variables in the module's main memory, and use processor logic to process the command control request.



The following table shows the contents of the command control area when a command control block such as 9970 (Read Module's Time to Processor) is issued.

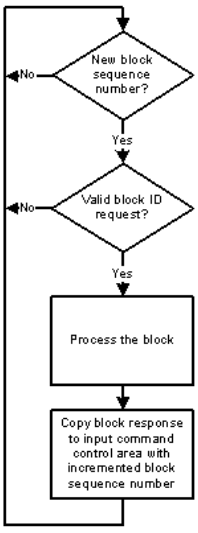
Note: The diagram above shows the memory addresses for a Quantum processor. If you are deploying the PTQ-HART with a Unity processor, substitute %MW for read only data, and %IW for read/write data.
Note: The processor memory locations in the example tables below use the 3x register start and 4x register start values defined in Backplane Data Transfer (page 98). You can configure any valid 3x and 4x start address that is not used by other processes.

Command Control Word	Description
40001	Output sequence number
40002	Block ID
40003	Block request word 1
40004	Block request word 2
40005	Block request word 3
…	…
40064	Block request word 62

The following table shows the results of the PTQ-HART response to the command control block.

Command Control Word	Description
30001	Block sequence number
30002	Block ID
30003	Block response word 1
30004	Block response word 2
30005	Block response word 3
…	…
30064	Block response word 62

The module recognizes that there is a new block request when it identifies that the block sequence number has changed. if the block ID is valid, the module will process the block and copy the response to the input command control area (3x for Quantum or %IW for Unity). The module will increment the block sequence number by one, as shown in the following illustration.



Special Functions

Special Function Block (9902)

This block is used by the processor to enable a set of commands that have their enable code set to 4 or 5 (one shot). The value referenced by the address associated with the command, Enable DB Address, will be set to -1 to enable the command for a one-shot (enable code 5) or continuous (enable code 4) execution. After the command is executed, the module resets the register to 0 to disable the command for the one-shot mode. The format of this command block is shown in the following table.

Word Offset in Block	Data Fields	Description
0	Sequence Number	This number triggers the request for the module. When this number changes, the module will process the command control request.
1	Block ID	This field contains the value of 9902 identifying the enable command to the module.
2	Port Number	This is the HART port number for the command. Each port has its own command list.
3	Command Count	This field contains the number of commands to enable in the command list. Valid values for this field are 1 to 60.
4 to 63	Command Numbers to Enable	These 60 words of data contain the command numbers in the command list to enable. The value referenced by the Enable DB Address parameter associated with each command will be set to -1.

There is no response to this block by the module.

Command Disabe Control Block (9903)

This block is used by the processor to disable a set of commands that have an enable code set to 4 or 5 (one shot). The value referenced by the address associated with the command, Enable DB Address, will be set to 0 to disable the command. The format for this command block is shown in the following table.

Word Offset in Block	Data Fields	Description
0	Sequence Number	This number triggers the request for the module. When this number changes, the module will process the command control request.
1	Block ID	This field contains the value of 9903 identifying the enable command to the module.
2	Port Number	This is the HART port number for the command. Each port has its own command list.
3	Command Count	This field contains the number of commands to enable in the command list. Valid values for this field are 1 to 60.
4 to 63	Command Numbers to Enable	These 60 words of data contain the command numbers in the command list to enable. The value referenced by the Enable DB Address parameter associated with each command will be set to 0.

There is no response to this block by the module.

Reboot Module Block (9998/9999)

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

Word Offset in Block	Data Fields	Description
0	Sequence Number	This number triggers the request for the module. When this number changes, the module will process the command control request.
1	Block ID	This field contains the value of 9998 or 9999 indicating a warm boot or cold boot to the module.
2 to 63	Spare	Not used.

8.2.4 HART Channels

The PTQ-HART module supports the HART protocol as a Master on up to two channels. Each channel is individually configurable.

The HART protocol uses the Bell 202 standard frequency shift-keying (FSK) signal to communicate at 1200 baud, superimposed at a low level on the 4 to 20 mA analog measurement signal. Having an average value of zero, and FSK signal causes no interference with the analog value. The HART devices are powered from this 4 to 20 mA analog loop.

Both HART protocol channels in the module generate the Bell 202 FSK signal to communicate in multi-drop mode with up to 15 HART devices and provide up to 250 mA supply for analog loop.

Auto-Poll Modes

Each HART channel can be set to operate in three different modes:

- Point-to-Point
- Multi-drop
- User Mode

Using the configuration file, choose the auto-poll mode through the Auto-Poll Code parameter (P, M, or N). In the first two modes, the module will automatically collect data from each HART instrument on the channel (auto-poll) and store the data in the module's database.

In User Mode, the module will only execute the commands in the user command list and will not automatically acquire data. Refer to the Auto-Polling section for more information.

Mode	Use Auto-Poll	Use Command List
N (None)	N	Y
M (Multi-drop)	Y	Y
P (Point-to-Point)	Y	Y

When configured for Point-to-Point communication, the channel will automatically poll data from the connected slave address 0.

When configured for Multi-drop, the channel will automatically poll data from the connected slaves in the network (address 1 to 15). Refer to the Auto-Polling section for more information about auto-poll mode.

User-configured commands can be issued on each channel to the HART devices. Up to 99 commands can be defined for each port. Data read from the devices is placed in the virtual database of the module, which is passed between the module and the processor.

Any write requests or device-specific commands for the HART slave devices are sourced with data from the virtual database, or from a configured constant data block. Within the commands, it can be specified whether or not to use the HART device's short or long address. If the long address is selected, the device is polled first with short address to ask for the long one. Then, the device is polled with the long address. The module does this processing automatically.

In a HART network, it is possible to have two masters. The module fully supports the existence of a second master, but it can reduce the throughput on the HART network. This facility is enabled or disabled in the module's configuration. If the ability to have a second master on the network is disabled, then the maximum communication throughput is achieved.

8.2.5 HART Command List

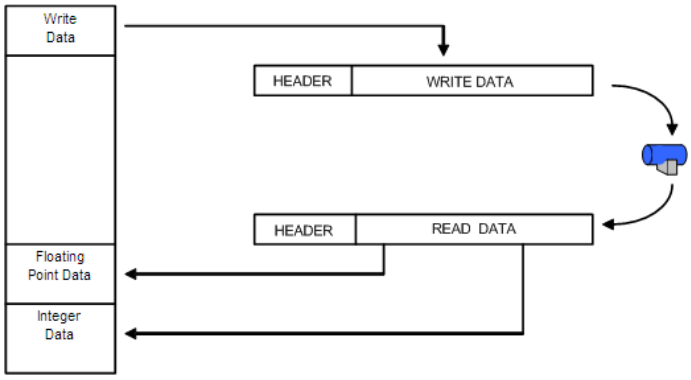
The HART Command List specifies the commands to be executed to the HART devices connected to a channel. A HART command can be seen as an outgoing message to the HART devices that provides Write Data for a specific command or a response message that carries process data (Read Data) back to the module. The PTQ module supports three kinds of data blocks in the Universal and Common Practice commands. These data blocks are:

- Integers
- IEEE 754 Floating Point Numbers (32 bits)
- Packed ASCII character strings

The Packed ASCII character strings are unpacked and placed with the integers data block.

For all commands, it is possible to select where the Write Data comes from; it can be in the module's internal database or it can be configured as a fixed data block in the command.

For response messages from HART devices, it is possible to configure where the Floating Point Data and Integer Data will be placed in the module's internal database, but this is only possible for the Universal and Common Practice commands. In the case of Device Specific commands, all the Read Data is placed in the Integer data section. See HART Command Support for a listing of supported HART commands.



Burst Mode

If a slave on a network will be placed in burst mode, its data can be placed in the module's database. For the command to be burst by slave device, enter a command in the user command list with the appropriate HART command number. Set the type field to 0 to disable the command. The parameters in the command will be used to store the data received from the bursting slave. There can only be one slave bursting on the network at any one time. This mode can be used for faster update of data from a slave.

8.2.6 Auto-Polling

This feature is enabled by setting the AUTO-POLL CODE (in the configuration file) to a value of P (point-to-point) or M (multi-drop). If the value N is entered for the parameter, the auto-poll feature is disabled. When the feature is disabled, the channel will only execute the commands enabled in the user command list. When the auto-polling mode is enabled, the module will automatically acquire data from the HART instruments attached to a channel without the use of user commands. If user commands are present and enabled when the feature is enabled, they will also be executed independent of auto-polling.

With the auto-poll feature enabled, the module automatically generates the following HART commands and stores the data in the module's database at the user-specified location:

CMD	Description
0	Read Unique Identifier
3	Read Current and Four Dynamic Variables
13	Read Tag, Descriptor and Date
14	Read PV Sensor Information
15	Read Output Information

If the unit is set for point-to-point mode, the module will automatically gather the information for the device with the polling address (short address) of zero and place the data into the database. Each device requires a 50-word database area with the format shown in the following table.

DB Byte Offset	Type	Description	Byte Cnt	Data Source	Use of Data
0	byte	Auto-polling command status bits	1	App	Status
1	byte	Last first status byte received from device	1	Resp	Status
2	byte	Last second status byte received from device	1	Resp	Status
3	byte	Manufacture ID Code	1	CMD 0	LongAddress
4	byte	Device Type Code	1	CMD 0	LongAddress
5	byte	Minimum number of preambles	1	CMD 0	Msgconstruction
6	byte	Universal Command Major Rev #	1	CMD 0	Msgchoice
7	byte	Device Revision Level	1	CMD 0	Info
8	byte	Software Revision Level	1	CMD 0	Info
9	byte	Hardware Revision Level/Physical Signaling Code	1	CMD 0	Info
10	byte	Device Flags	1	CMD 0	Info
11 to 13	byte	Device ID	3	CMD 0	Long Address
14	byte	Minimum number of preambles to be sent with the response message from the slave to the master.	1	CMD 0	

DB Byte Offset	Type	Description	Byte Cnt	Data Source	Use of Data
15	byte	Maximum number of device variables	1	CMD 0	Info
16 to 17	word	Configuration Change Counter	2	CMD 0	Info
18	byte	Extended Field Device Status	1	CMD 0	Info
19	byte	Primary variable units code	1	CMD 3	Cfg
20	byte	Secondary variable units code	1	CMD 3	Cfg
21	byte	Tertiary variable units code	1	CMD 3	Cfg
22	byte	Quaternary variable units code	1	CMD 3	Cfg
23 to 30	byte	Tag name	8	CMD 13	Info
31 to 46	byte	Descriptor	16	CMD 13	Info
47 to 49	byte	Tag/Descriptor data	3	CMD 13	Info
50 to 52	byte	Transducer serial number	3	CMD 14	Info
53	byte	Transducer limits and min span units code	1	CMD 14	Info
54	byte	PV alarm selection code	1	CMD 15	Info
55	byte	PV transfer function code	1	CMD 15	Info
56	byte	PV upper and lower range value units code	1	CMD 15	Info
57	byte	Write protection code	1	CMD 15	Status
58	byte	Private label distributor code	1	CMD 15	Info
59	byte	PV analogchannel flag	1	CMD 15	Info
		TOTAL BYTE COUNT	60		
		TOTAL WORD COUNT	30		

DB Byte Offset	Type	Description	Byte Cnt	Data Source	Use of Data
60 to 63	float	Primary variable value	4	CMD 3	Status
64 to 67	float	Secondary variable value	4	CMD 3	Status
68 to 71	float	Tertiary variable value	4	CMD 3	Status
72 to 75	float	Quaternary variable value	4	CMD 3	Status
76 to 79	float	Upper transducer limit	4	CMD 14	Cfg
80 to 83	float	Lower transducer limit	4	CMD 14	Cfg
84 to 87	float	Minimum span	4	CMD 14	Cfg
88 to 91	float	PV upper range value	4	CMD 15	Cfg
92 to 95	float	PV lower range value	4	CMD 15	Cfg
96 to 99	float	PV damping value (in seconds)	4	CMD 15	Cfg
		TOTAL FLOAT BYTE COUNT	40		
		TOTAL FLOAT WORD COUNT	20		

DB Regs/Device	50
Max DB Regs/channel for 15 Devices	750

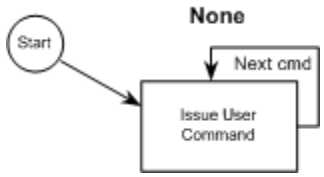
DB Regs/Device	50
Max DB Regs for HART Card	3000

The following table defines the auto-polling command status bits:

Bit #	Description
0	Long Address Set (command 0 successful)
1	Command 13 successful (configuration)
2	Command 14 successful (configuration)
3	Command 15 successful (configuration)
4	Command 3 successful (data polling)
5	Reserved
6	Reserved
7	Reserved

Auto-Poll Disabled Mode

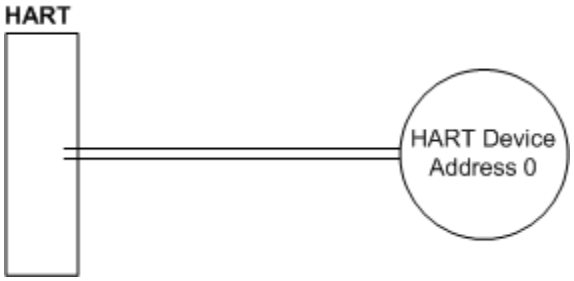
If the auto-polling feature is disabled (Auto-Poll Code = N), the module functions as shown in the following diagram:



Only the user commands are executed and all data is placed in and sourced from the module's internal database. The user is responsible for constructing all commands to control and monitor the instruments attached to the channel.

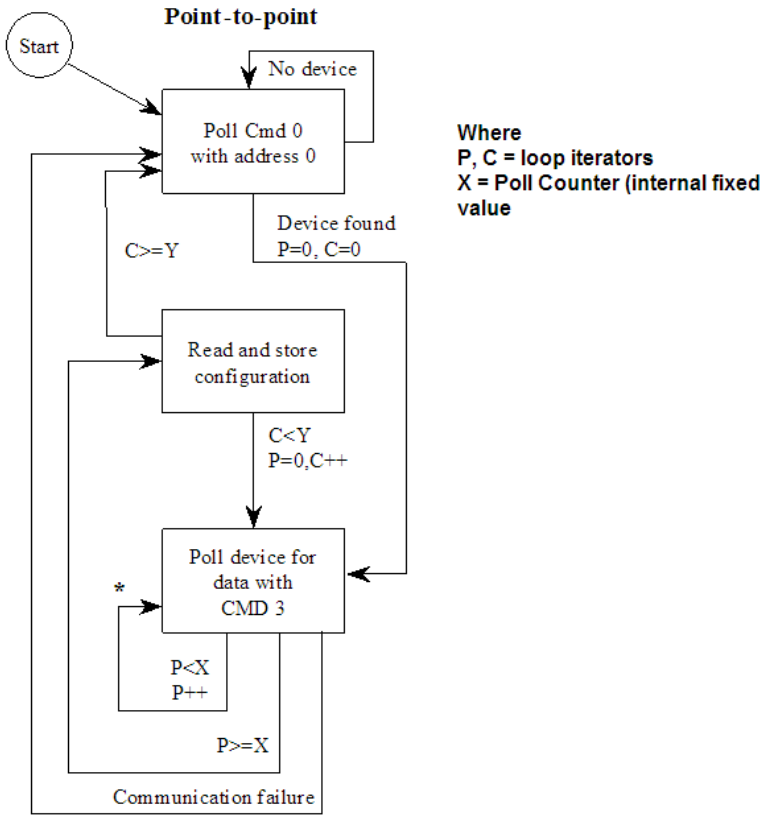
Point-to-Point Mode

Important: If the HART device address is 0 you must configure the channel for Point-to-Point mode.



In point-to-point mode, the module only polls for a single instrument with a polling address of zero. When the instrument is found by the channel, it continuously polls for the data using command 3. Occasionally, it will poll for the configuration information for the device. This is accomplished with HART commands 13, 14 and 15. Less frequently, the channel will perform a HART command 0 request to see if any of the data for the instrument has changed. If communications is lost with the device, the module will try to establish communications with the device using command 0. If user commands are present and enabled, they will be executed after each data poll.

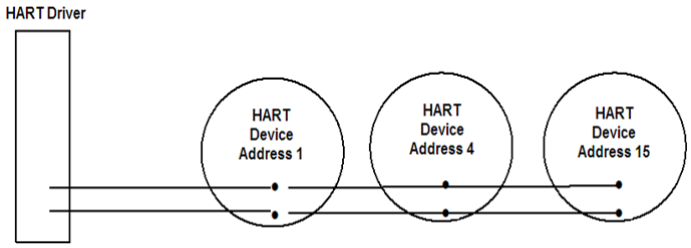
When the point-to-point mode of auto-polling is enabled (Auto-Poll Code = P), the following diagram applies to the channel operation:



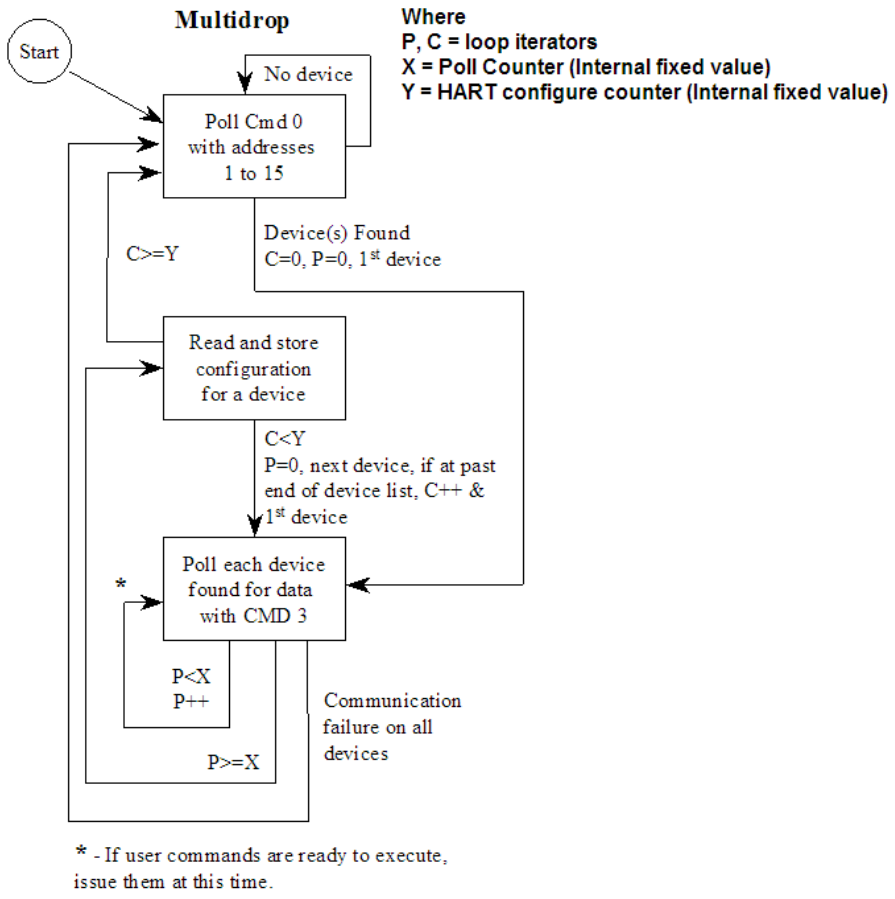
Multi-drop Mode

Important: If the HART device address is between 1 and 15, you must configure the channel for multi-drop mode.

If the unit is set for multi-drop mode, the module will poll each unit attached to the channel starting with polling address 1. The parameter MAX DEVICE COUNT in the configuration determines the maximum slave address number to be polled in multi-drop mode. For example, if the MAX DEVICE COUNT parameter is set to 3, the channel will poll for polling addresses 1, 2, and 3. It is important when assigning the device polling addresses for instruments on a channel to start with 1 and successfully increase the value by one until the last instrument is assigned an address. If you set the MAX DEVICE COUNT parameter to 2, and assign the two instruments addresses 1 and 15, the polling address 15 will never be used (only addresses 1 and 2). Therefore, the second instrument will never be polled and will be in communication failure. Up to 15 instruments can be assigned to a single channel (polling address 1 to 15).



If the module is configured for multi-drop auto-polling (Auto-Poll Code = M), the following diagram applies:



In multi-drop mode, the channel will poll for instruments 1 to the value set in the MAX DEVICE COUNT parameter. For each device found, it will continuously poll for data using command 3. After a certain number of iterations, the HART Driver will ready the configuration data for the next device (if there is one). Each time configuration information is acquired, it will be for a different slave device if more than 1 slave is recognized by the channel. Less frequently, the channel will poll for devices not found in the original search of the network using HART command 0. If communications is lost with all devices, the module will try to establish communications with the devices using command 0. If user commands are present and enabled, they will be executed after the data polling of the devices.

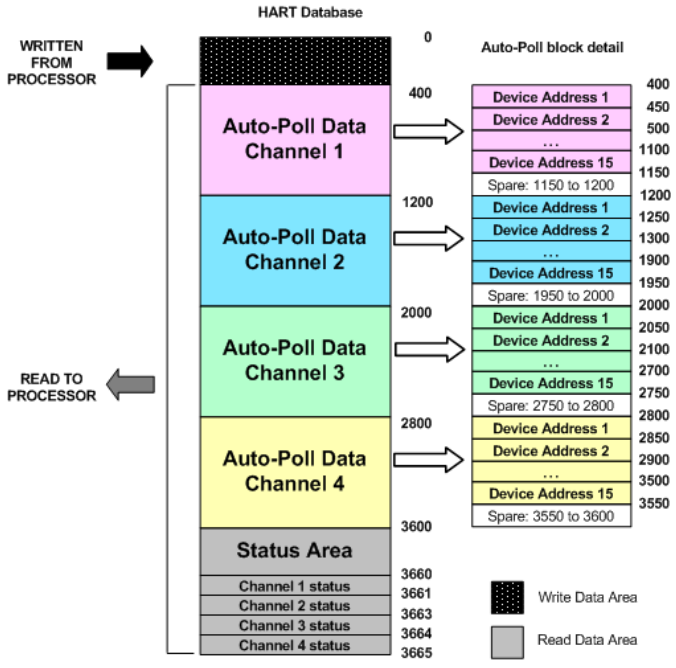
Multi-drop Mode Example:

The intent is to show when using Multi-drop mode how the Auto-Poll DB Address and Max Device Count parameters should be used.

If the configuration file sets the following parameters:

Parameter	Value	Config File Section
DB Address Status	3660	[HART PORT 0]
Auto-Poll DB Address	400	
Max Device Count	15	
DB Address Status	3661	[HART PORT 1]
Auto-Poll DB Address	1200	
Max Device Count	15	
DB Address Status	3662	[HART PORT 2]
Auto-Poll DB Address	2000	
Max Device Count	15	
DB Address Status	3663	[HART PORT 3]
Auto-Poll DB Address	2800	
Max Device Count	15	

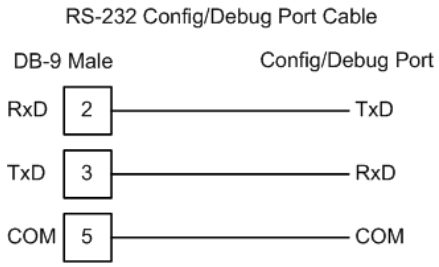
The HART database has the following structure.



The configuration file for this example shows that when the MAX Device Address parameter is configured as 15, the module will reserve 750 words for each channel (15 devices x 50 words each) even though not all devices may be present on the network.

8.3 RS-232 Configuration/Debug Port

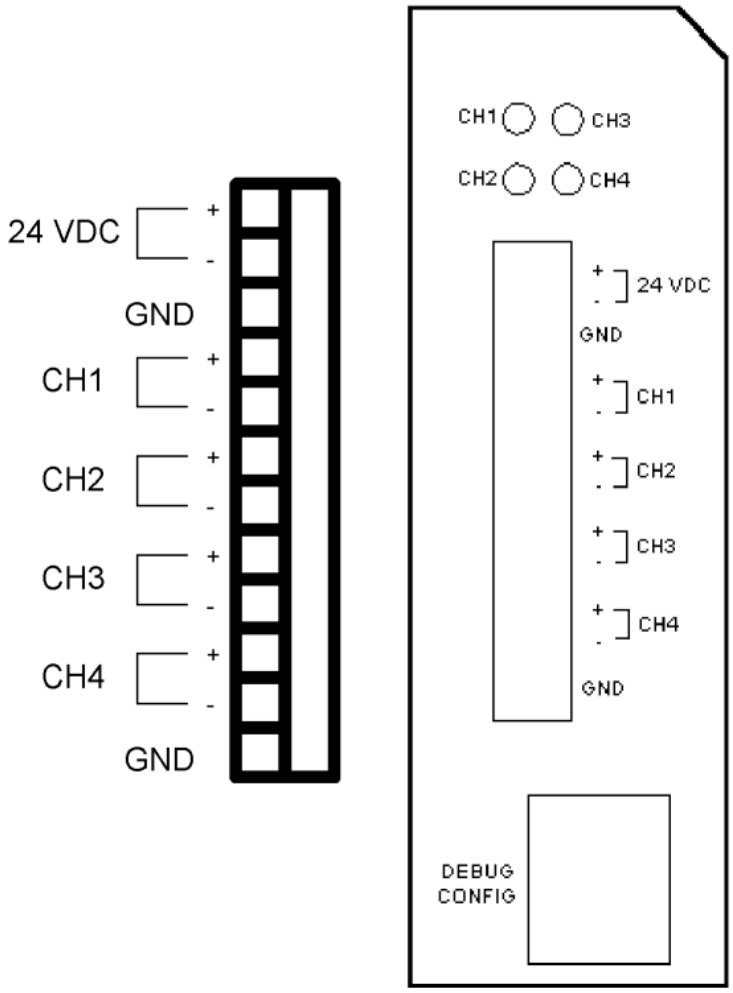
This port is physically a DB-9 connection. This port permits a PC based terminal emulation program to view configuration and status data in the module and to control the module. The cable for communications on this port is shown in the following diagram:



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

8.4 Application Port Connection

The PTQ-HART module has a single terminal connector to attach the module to the HART networks. The following diagrams display the configuration of the terminal connector:



8.5 Example Field Terminations

		Up to 15 instruments per channel	One instrument per channel	
		Multi-drop Mode (address 1 to 15)	**Point to Point Mode (address 0 to 15)**	**Point to Point Mode with Analog (address 0 only)**
Power Mode A 24VDC connected to HART power terminals 1 and 2 on terminal block	Master Powered instruments on all 4 channels	OK Install 250Ω 1KΩ	OK Install 1K resistor	OK
Power Mode B HART power terminals 1 and 2 on terminal block left open	Loop Powered instruments on 1 or more channels	OK	OK	OK - differential or single ended analog signal
	Self Powered instruments on 1 or more channels	Not possible	OK	OK -differential or single ended analog signal

8.5.1 Notes

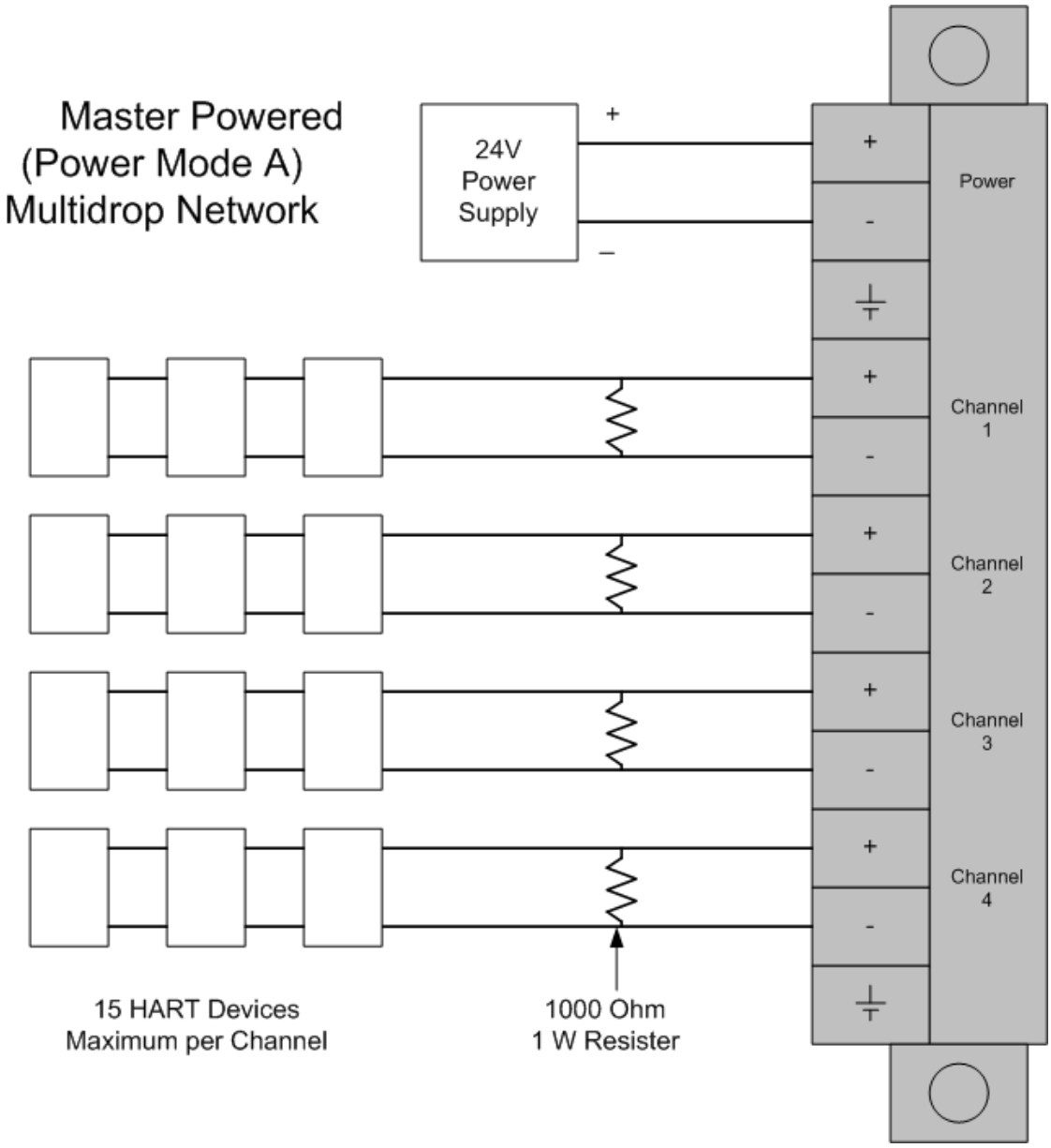
1 All four channels on the unit must be either Master Powered (Power Mode A) or must be Loop or Self powered (Power Mode B). You cannot mix Power Mode A and Power Mode B on the same unit. You can mix Loop powered and Self powered instruments on the same unit. However, you cannot mix Loop powered and Self powered instruments on the same channel.

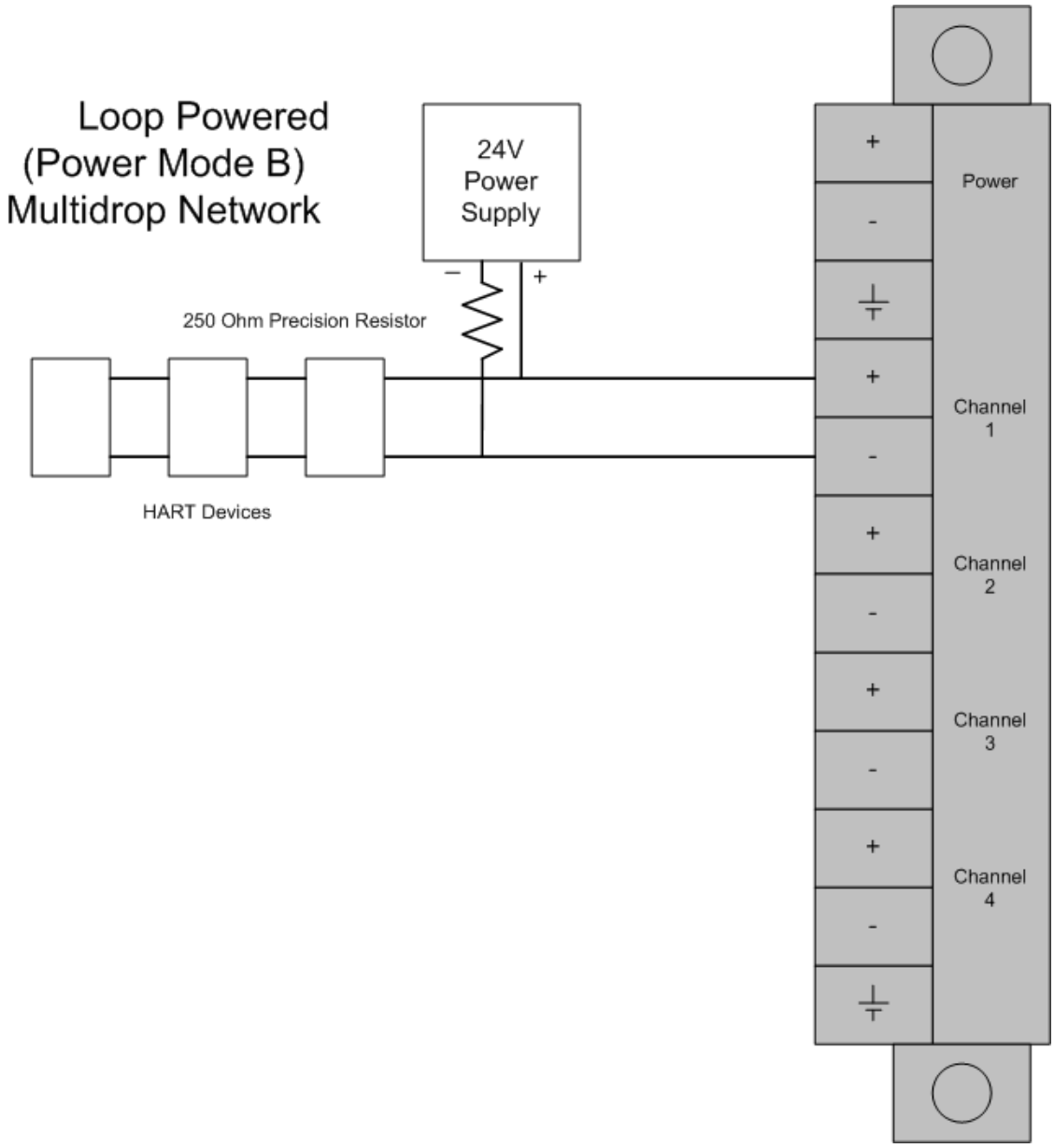
2 For Master Powered, Multi-drop Mode and Point to Point Mode, a 1K, 1W resistor must be connected across the + and - input terminals of each HART channel. Resistors are supplied with each product.

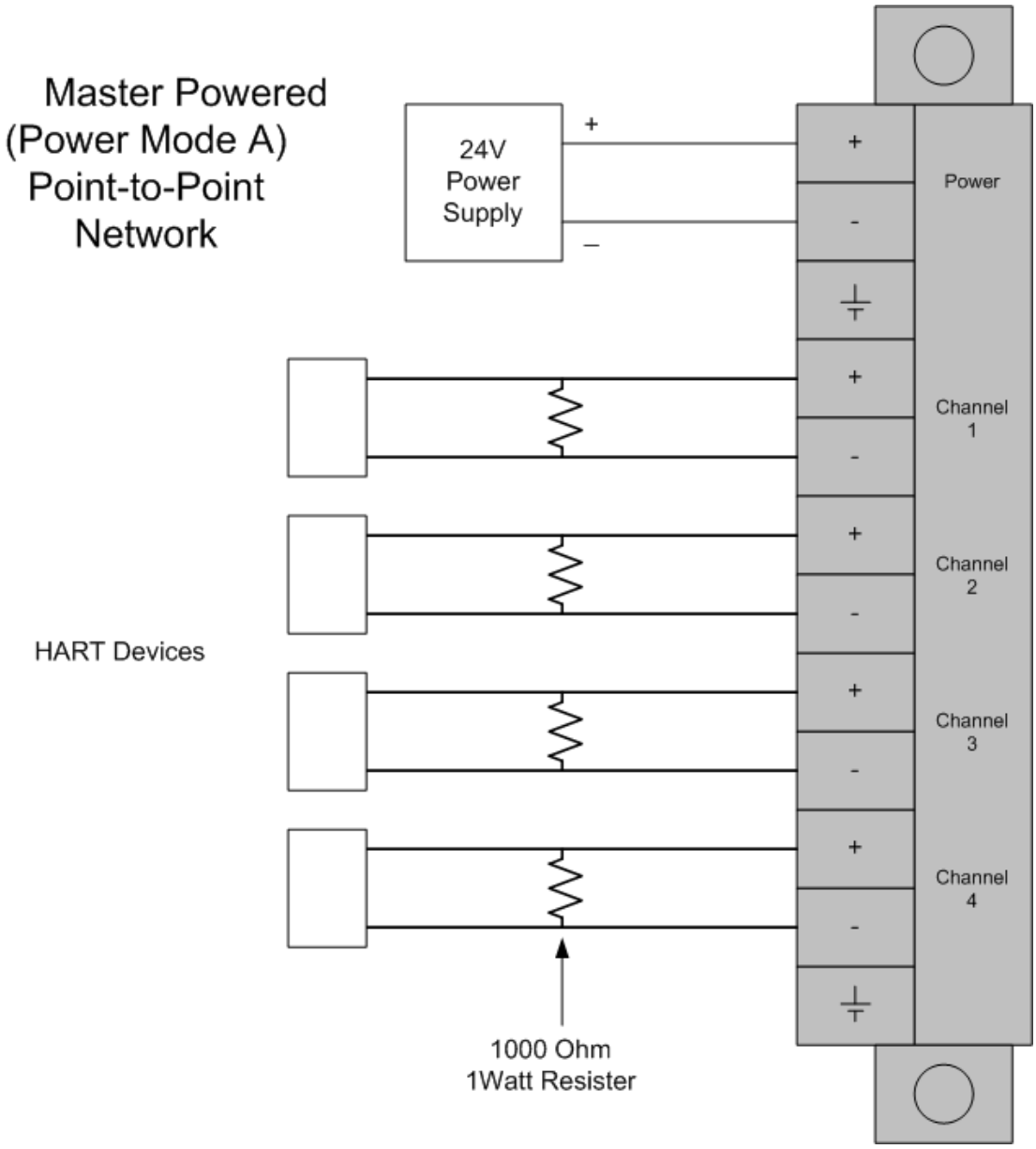
3 When configuring loop powered instruments in a Power Mode B type multi-drop network, use only one power supply per HART channel if you wish to maintain channel to channel isolation. If channel to channel isolation is not required, the same power supply may be used for all four channels. However, a separate series resistor is required for each channel. An appropriately sized resistor must be connected in series with the power supply for each channel. The value and power rating of the resistor is dependant upon the number of instruments in the loop and the length and gauge of the pair of wires connecting the instruments to the unit. 250 Ohm, 3 Watt is a good, first choice value.

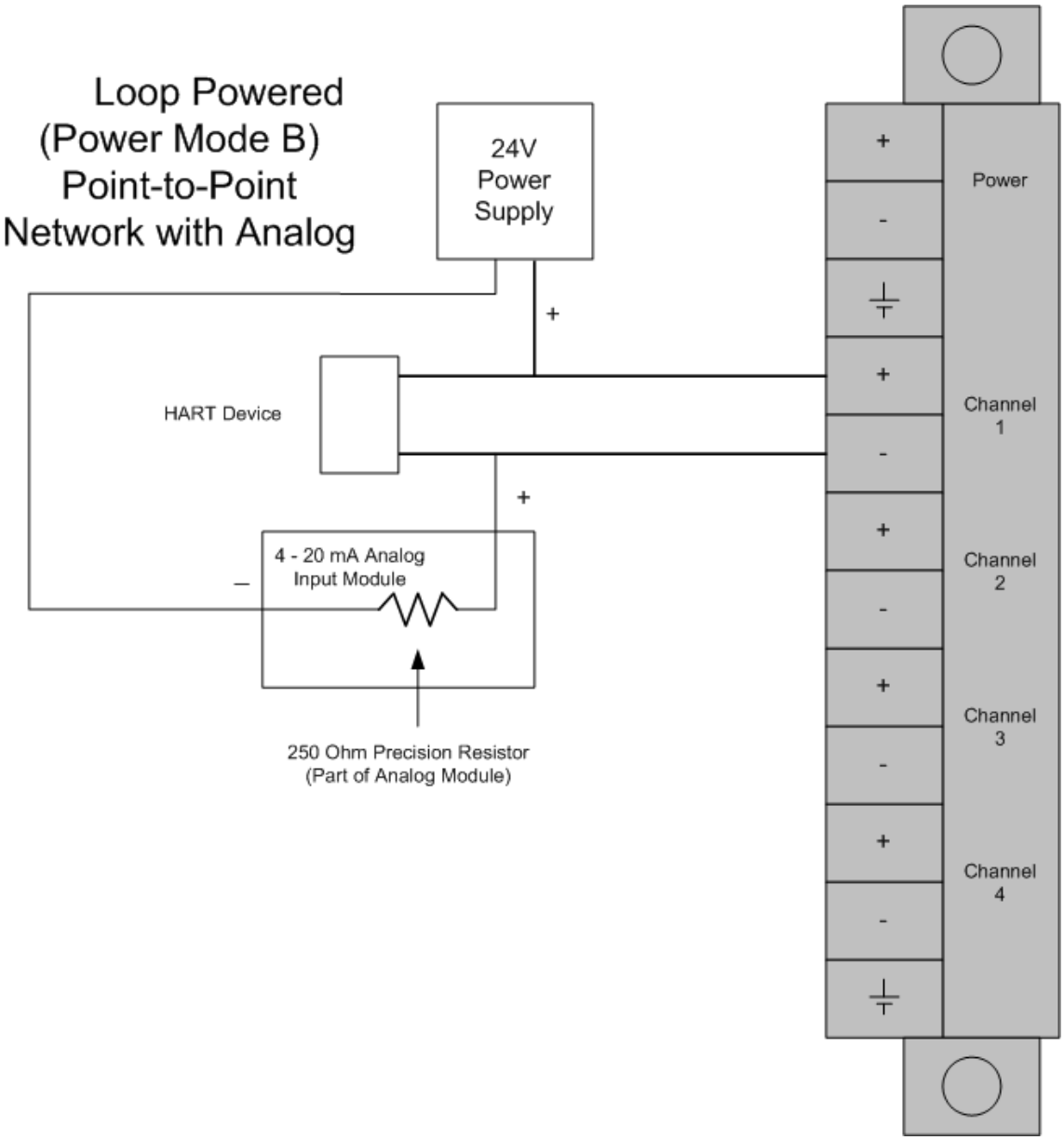
4 If field instrument wiring is shielded, terminate shields at the ground terminals of the HART connector. Do not terminate the shields at the instrument.

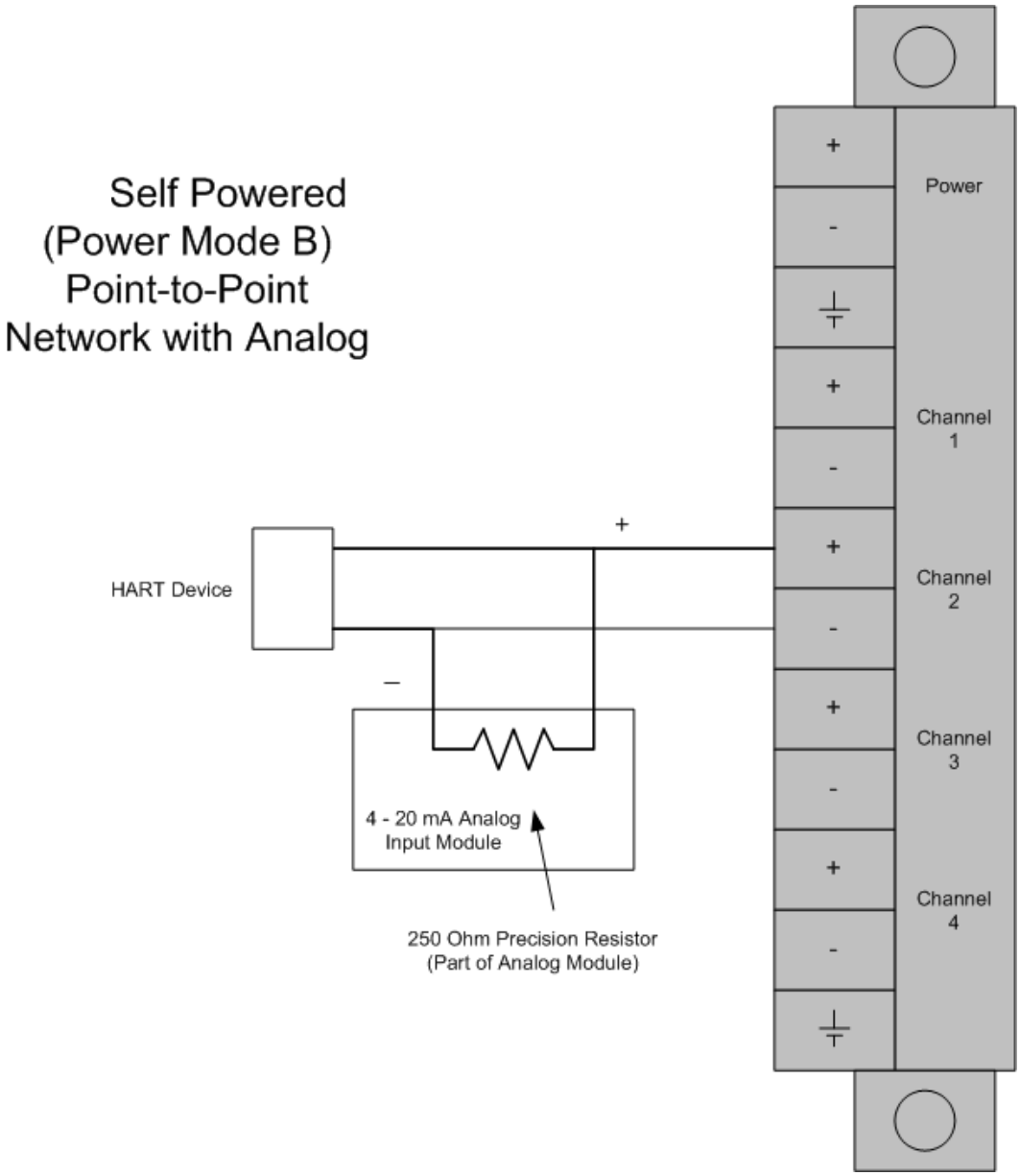
5 For multi-drop applications, the 1KΩ resistor should be normally used when one transmitter is connected to the port. If you increase the number of devices connected to a single port, you may need to reduce the size of the resistor.

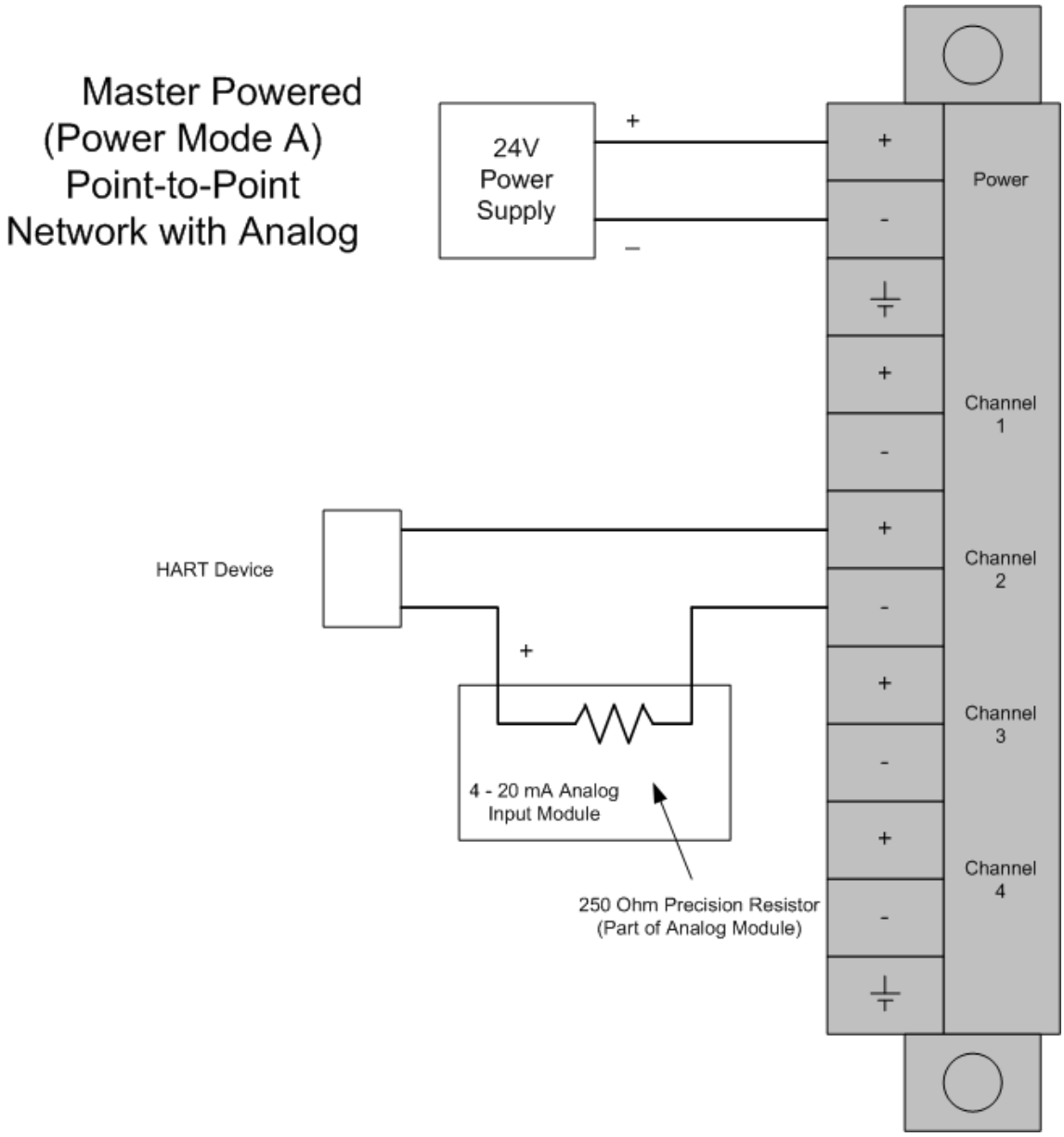



Loop Powered
(Power Mode B)
Multidrop Network
24V
Power
Supply
−
+
250 Ohm Precision Resistor
HART Devices
+
-
Power
+
-
Channel
1
+
-
Channel
2
+
-
Channel
3
+
-
Channel
4


Master Powered
(Power Mode A)
Point-to-Point
Network
24V
Power
Supply
+
–
+
-
Power
+
-
Channel
1
+
-
Channel
2
+
-
Channel
3
+
-
Channel
4
HART Devices
1000 Ohm
1Watt Resister


Loop Powered
(Power Mode B)
Point-to-Point
Network with Analog
24V
Power
Supply
+
HART Device
+
4 - 20 mA Analog
Input Module
–
250 Ohm Precision Resistor
(Part of Analog Module)
+
-
Power
+
-
Channel
1
+
-
Channel
2
+
-
Channel
3
+
-
Channel
4


Self Powered
(Power Mode B)
Point-to-Point
Network with Analog
+
-
Power
+
-
Channel
1
+
-
Channel
2
+
-
Channel
3
+
-
Channel
4
HART Device
+
–
4 - 20 mA Analog
Input Module
250 Ohm Precision Resistor
(Part of Analog Module)


Master Powered
(Power Mode A)
Point-to-Point
Network with Analog
24V
Power
Supply
+
–
+
-
Power
Channel
1
Channel
2
Channel
3
Channel
4
HART Device
+
4 - 20 mA Analog
Input Module
250 Ohm Precision Resistor
(Part of Analog Module)

8.6 Thermal Connector Port Details

The ProTalk module contains one thermal connector for 24VDC supply and four HART channels.

8.7 PTQ-HART Error Status Table

This section contains a listing of the PTQ-HART module's status data area.

8.7.1 Error/Status Data Block

Offset	Content	Description
4000	Program Scan Count	This value is incremented each time a complete program cycle occurs in the module.
4001 to 4003	Product Code	These two registers contain the product code of "HRT5 "
4004 to 4005	Product Version	These two registers contain the product version for the currently running software.
4006 to 4007	Operating System	These two registers contain the month and year values for the program operating system.
4008	Run Number	These two registers contain the run number value for the currently running software.
4009	Read Block Count	This field contains the total number of read blocks transferred from the module to the processor.
4010	Write Block Count	This field contains the total number of write blocks transferred from the processor to the module.
4011	Parse Block Count	This field contains the total number of blocks successfully parsed that were received from the processor.
4012	Reserved	Not used
4013	Command Block Count	This field contains the total number of command blocks received from the processor for enable and disable requests.
4014	Error Block Count	This field contains the total number of block errors recognized by the module.
4015	Port 1 State	Port 1 state machine value (used for debugging)
4016	Comm State	Port 1 communication state machine value (used for debugging)
4017	Device Status (bit mapped)	Each bit in this word corresponds to a slave address on the network starting at bit 1 for slave address 1. Bit 0 is not used. If the bit is set, the slave is in error. If the bit is clear (0), the slave is not in error.
4018	Device Poll List (bit mapped)	Each bit in this word corresponds to a slave address on the network starting at bit 1 for slave address 1. Bit 0 is not used. If the bit is set, the slave is in the poll list. If the bit is clear (0), the slave is not in the poll list.
4019	Device With Long Address (bit mapped)	Each bit in this word corresponds to a slave address on the network starting at bit 1 for slave address 1. Bit 0 is not used. If the bit is set, the slave uses the long address. If the bit is clear (0), the slave does not use the long address.
4020	Current Command	This field contains the index of the current command to execute.
4021	Command Request Count	This field contains the total number of request messages issued on the port.
4022	Command Response Count	This field contains the total number of response messages received from devices on the network.
4023	Configuration Error Word	Configuration error word (see table)

Offset	Content	Description
4024	Current Error Code	Current error code for port
4025	Last Error Code	Last error code reported for port
4026	Port 2 State	Port 2 state machine value (used for debugging)
4027	Comm State	Port 2 communication state machine value (used for debugging)
4028	Device Status (bit mapped)	Each bit in this word corresponds to a slave address on the network starting at bit 1 for slave address 1. Bit 0 is not used. If the bit is set, the slave is in error. If the bit is clear (0), the slave is not in error.
4029	Device Poll List (bit mapped)	Each bit in this word corresponds to a slave address on the network starting at bit 1 for slave address 1. Bit 0 is not used. If the bit is set, the slave is in the poll list. If the bit is clear (0), the slave is not in the poll list.
4030	Device With Long Address (bit mapped)	Each bit in this word corresponds to a slave address on the network starting at bit 1 for slave address 1. Bit 0 is not used. If the bit is set, the slave uses the long address. If the bit is clear (0), the slave does not use the long address.
4031	Current Command	This field contains the index of the current command to execute.
4032	Command Request Count	This field contains the total number of request messages issued on the port.
4033	Command Response Count	This field contains the total number of response messages received from devices on the network.
4034	Configuration Error Word	Configuration error word (see table).
4035	Current Error Code	Current error code for port.
4036	Last Error Code	Last error code reported for port.
4037	Port 3 State	Port 3 state machine value (used for debugging).
4038	Comm State	Port 3 communication state machine value (used for debugging).
4039	Device Status (bit mapped)	Each bit in this word corresponds to a slave address on the network starting at bit 1 for slave address 1. Bit 0 is not used. If the bit is set, the slave is in error. If the bit is clear (0), the slave is not in error.
4040	Device Poll List (bit mapped)	Each bit in this word corresponds to a slave address on the network starting at bit 1 for slave address 1. Bit 0 is not used. If the bit is set, the slave is in the poll list. If the bit is clear (0), the slave is not in the poll list.
4041	Device With Long Address (bit mapped)	Each bit in this word corresponds to a slave address on the network starting at bit 1 for slave address 1. Bit 0 is not used. If the bit is set, the slave uses the long address. If the bit is clear (0), the slave does not use the long address.
4042	Current Command	This field contains the index of the current command to execute.
4043	Command Request Count	This field contains the total number of request messages issued on the port.
4044	Command Response Count	This field contains the total number of response messages received from devices on the network.

Offset	Content	Description
4045	Configuration Error Word	Configuration error word (see table).
4046	Current Error Code	Current error code for port.
4047	Last Error Code	Last error code reported for port.
4048	Port 4 State	Port 4 state machine value (used for debugging).
4049	Comm State	Port 4 communication state machine value (used for debugging).
4050	Device Status (bit mapped)	Each bit in this word corresponds to a slave address on the network starting at bit 1 for slave address 1. Bit 0 is not used. If the bit is set, the slave is in error. If the bit is clear (0), the slave is not in error.
4051	Device Poll List (bit mapped)	Each bit in this word corresponds to a slave address on the network starting at bit 1 for slave address 1. Bit 0 is not used. If the bit is set, the slave is in the poll list. If the bit is clear (0), the slave is not in the poll list.
4052	Device With Long Address (bit mapped)	Each bit in this word corresponds to a slave address on the network starting at bit 1 for slave address 1. Bit 0 is not used. If the bit is set, the slave uses the long address. If the bit is clear (0), the slave does not use the long address.
4053	Current Command	This field contains the index of the current command to execute.
4054	Command Request Count	This field contains the total number of request messages issued on the port.
4055	Command Response Count	This field contains the total number of response messages received from devices on the network.
4056	Configuration Error Word	Configuration error word (see table).
4057	Current Error Code	Current error code for port.
4058	Last Error Code	Last error code reported for port.
4059 to 4246	Reserved	Not used.
4247	Read Block ID	Block identification code of 0 or -1 to indicate a status data block.

The following table defines the contents of the configuration error word. Each bit in the word corresponds to an error condition recognized when the module is configured. There is a separate word for each application port. This data is reported in the status data area previously defined.

Bit	Code	Description
0	0x0001	Enabled not set to Y or N
1	0x0002	Enable Handheld not set to Y or N
2	0x0004	Primary Master not set to Y or N
3	0x0008	Invalid Preambles (1 to 50)
4	0x0010	Invalid Short Address Retries (0 to 50)
5	0x0020	Invalid Long Address Retries (0 to 50)
6	0x0040	Invalid Retries After Error (0 to 50)
7	0x0080	Invalid Poll Time After Error (0 to 10000)
8	0x0100	Invalid DB Address Status
9	0x0200	Invalid Command Count
10	0x0400	Memory Error in allocating commands
11	0x0800	Memory Error in allocating command fixed data
12	0x1000	Memory Error in allocating TX/RX buffers
13	0x2000	HART Board not found
14	0x4000	Cannot initialize HART channel
15	0x8000	

Each command in the command list for each HART channel has a word value for a status error code. This error data list can be read using the Configuration/Debug Port and can be placed in the module's internal database using the Error/Status Offset parameter for each port. Accessing the Debug capabilities of the module is accomplished by connecting a PC to the Debug port, then running a terminal emulation program. This status error code is the first word of the Integer Data Block returned from every HART command executed. This word has information about the execution of the command by the HART device.

Refer to the following section for a listing of the Error Codes to interpret the status error codes present in the integer data area. The following illustration shows the location of each error word in the data block:

Word Offset	Description
0	Command 0 Error Value
…	Command 99 Error Value
…	
99	

The module error codes are listed in this section. Error codes returned from the HART device are placed at the first word of the integer data block in the Virtual Database. The error codes are formatted in the word as follows:

If the most-significant bit of the word contains "1", then the most significant byte has a communication error code. The least-significant byte of the word will contain "0".

If the most-significant bit of the word contains "0", the most significant byte contains a command error code. In this case, the least-significant byte contains device malfunction information.

If this word value is "-1" (or 0xFFFF), it means that the command timed out.

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

8.7.2 Protocol Error Codes

These are error codes that are part of the HART protocol. The standard HART error codes are shown in the following tables:

					Error Code Word										
First Byte								Second Byte							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0

BIT 7 OF FIRST BYTE = 1: COMMUNICATION ERROR	
FIRST BYTE	
BIT 6	Parity Error
BIT 5	Overrun Error
BIT 4	Framing Error
BIT 3	Checksum Error
BIT 2	Reserved
BIT 1	RX Buffer Overflow
BIT 0	Undefined
SECOND BYTE	
BIT 0 to BIT 7	All 0

BIT 7 OF FIRST BYTE = 0: COMMAND ERROR	
FIRST BYTE	
Bits 6 to 0 (not bit-mapped):	
0	No Error
1	Undefined
2	Invalid Selection
3	Passed Parameter Too Large

BIT 7 OF FIRST BYTE = 0: COMMAND ERROR	
4	Passed Parameter Too Small
5	Too Few Data Bytes Received
6	Transmitter-Specific Command Error
7	In Write-Protect Mode
8 to 15	Command Specific Errors (see Below)
16	Access Restricted
32	Device is Busy
64	Command not Implemented
SECOND BYTE	
BIT 7	Device Malfunction
BIT 6	Configuration Changed
BIT 5	Cold Start
BIT 4	Unused
BIT 3	Output Current Fixed
BIT 2	Analog Output Saturated
BIT 1	Variable (Not Primary) Out of Limits
BIT 0	Primary Variable Out of Limits

Command Specific Errors	
8	Update Failed Update In Progress Set to Nearest Possible Value
9	Applied Process Too High Lower Range Value Too High Not in Fixed Current Mode
10	Applied Process Too Low Lower Range Value Too Low Multi-drop Not Supported
11	In Multi-drop Mode Invalid Transmitter Variable Code Upper Range Value Too High
12	Invalid Unit Code Upper Range Value Too Low
13	Both Range Values Out of Limits
14	Pushed Upper Range Value Over Limit Span Too Small

The command error codes are constantly updated between the Quantum processor and the module. The command error codes are copied to the following address in the database:

Port	Start Address	End Address
0	4100	4199
1	4200	4299
2	4300	4399
3	4400	4499

8.8 HART Universal Commands

COMMAND 00 - Read Unique Identifier

Description

This command gets the long address of the HART device plus other manufacturer information like Manufacturer ID, Device Type Code, Software Revision, Hardware Revision, and so on.

Write Parameters

NONE

Floating Point Data Returned

NONE

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	
1	Constant "254"	Manufacturer Identification Code
2	Manufacturer Device Type Code	Number of Preambles
3	Universal Command Revision	Transmitter Specific Command Revision
4	Software Revision	Hardware Revision
5	Device Function Flags	Device ID Number 1
6	Device ID Number 2	Device ID Number 3

COMMAND 01 - Read Primary Variable

Description

This command gets the device Primary Variable and the Primary Variable Units

Write Parameters

NONE

Floating Point Data Returned

Word	High Byte	Low Byte
0	Primary Variable Value	
1		

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	
1	Primary Variable Units Code	0

COMMAND 02 - Read Current And Percent Of Range

Description

This command gets the current of the loop that is forced by the HART device and the Percent of Range of the Current.

Write Parameters

NONE

Floating Point Data Returned

Word	High Byte	Low Byte
0	Current (mA)	
1		
2	Percent of Range	
3		

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	

COMMAND 03 - Read Dynamic Variables

Description

This command gets the current and four (predefined) dynamic Variables.

Write Parameters

NONE

Floating Point Data Returned

Word	High Byte	Low Byte
0	Current (mA)	
1		
2	Primary Variable	
3		
4	Second Variable	
5		
6	Third Variable	
7		
8	Fourth Variable	
9		

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	
1	Primary Variable Units Code	Second Variable Units Code
2	Third Variable Units Code	Fourth Variable Units Code

COMMAND 06 - Write Polling Address

Description

This command sets the polling address of a HART device. Extreme care should be taken when you use this command because you can loose the communication with the device.

Write Parameters

Word	High Byte	Low Byte
0	Polling Address	Polling Address

Floating Point Data Returned

NONE

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	
1	Polling Address	0

COMMAND 12 - Read Message

Description

This command reads an ASCII message contained in the HART Device and written by the Write Message command 17.

Write Parameters

NONE

Floating Point Data Returned

NONE

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	
1	Message ASCII Character 0	Message ASCII Character 1
2	Message ASCII Character 2	Message ASCII Character 3
.	.	
.	.	
.	.	
14	Message ASCII Character 28	Message ASCII Character 29
15	Message ASCII Character 30	Message ASCII Character 31

COMMAND 13 - Read Tag, Descriptor and Date

Description

This command reads an ASCII Tag which identifies the device, an ASCII descriptor of the device and the last Date it has been configured.

Write Parameters

NONE

Floating Point Data Returned

NONE

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	
1	TAG ASCII Character 0	TAG ASCII Character 1
2	TAG ASCII Character 2	TAG ASCII Character 3
3	TAG ASCII Character 4	TAG ASCII Character 5
4	TAG ASCII Character 6	TAG ASCII Character 7
5	Descriptor ASCII Character 0	Message ASCII Character 1
.	.	
.	.	
.	.	
12	Descriptor ASCII Character 14	Message ASCII Character 15
13	Date	Date
14	Date	0

COMMAND 14 - Read PV Sensor Info

Description

This command gets information about the Primary Variable sensor, like limits and span.

Write Parameters

NONE

Floating Point Data Returned

Word	High Byte	Low Byte
0	Upper Sensor Limit	
1		
2	Lower Sensor Limit	
3		
4	Minimum Span	
5		

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	
1	Sensor Serial Number 0	Sensor Serial Number 1
2	Sensor Serial Number 2	Unit Codes for Sensor Limits and Span

COMMAND 15 - Read Output Information

Description

This command gets information about the Primary Variable Output Information.

Write Parameters

NONE

Floating Point Data Returned

Word	High Byte	Low Byte
0	Upper Range Value	
1		
2	Lower Range Value	
3		
4	Damping Value (Sec)	
5		

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	
1	Alarm Select Code	Transfer Function Code
2	Primary Variable Range Units Code	Write-Protect Code
2	Private-Label Distributor Code	0

COMMAND 16 - Read Final Assembly Number

Description

This command reads the final assembly number of the HART device.

Write Parameters

NONE

Floating Point Data Returned

NONE

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	
1	Final Assembly Number 0	Final Assembly Number 1
2	Final Assembly Number 2	0

COMMAND 17 - Write Message

Description

This command writes an ASCII message contained in the HART Device and that can be read with command 12.

Write Parameters

Word	High Byte	Low Byte
0	Packed ASCII Message Byte 0	Packed ASCII Message Byte 1
1	Packed ASCII Message Byte 2	Packed ASCII Message Byte 3
.	.	
.	.	
.	.	
11	Packed ASCII Message Byte 22	Packed ASCII Message Byte 23

Floating Point Data Returned

NONE

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	

COMMAND 18 - Write Tag, Descriptor and Date

Description

This command writes an ASCII Tag which identifies the device, an ASCII descriptor of the device and the last Date it has been configured.

Write Parameters

Word	High Byte	Low Byte
0	Packed ASCII TAG Byte 0	Packed ASCII TAG Byte 1
1	Packed ASCII TAG Byte 2	Packed ASCII TAG Byte 3
2	Packed ASCII TAG Byte 4	Packed ASCII TAG Byte 5
3	Packed ASCII Descriptor Byte 0	Packed ASCII Descriptor Byte 1
.	.	
.	.	
.	.	
8	Packed ASCII Descriptor Byte 10	Packed ASCII Descriptor Byte 11
9	Date	Date
10	Date	0

Floating Point Data Returned

NONE

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	
1	TAG ASCII Character 0	TAG ASCII Character 1
2	TAG ASCII Character 2	TAG ASCII Character 3
3	TAG ASCII Character 4	TAG ASCII Character 5
4	TAG ASCII Character 6	TAG ASCII Character 7
5	Descriptor ASCII Character 0	Message ASCII Character 1
.	.	
.	.	
.	.	
12	Descriptor ASCII Character 14	Message ASCII Character 15
13	Date	Date
14	Date	0

COMMAND 19 - Write Final Assembly Number

Description

This command writes the final assembly number of the HART device.

Write Parameters

Word	High Byte	Low Byte
0	Final Assembly Number 0	Final Assembly Number 1
1	Final Assembly Number 2	Final Assembly Number 2

Floating Point Data Returned:

NONE

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	
1	Final Assembly Number 0	Final Assembly Number 1
2	Final Assembly Number 2	0

8.9 HART Common Practice Commands

COMMAND 33 - Read Transmitter Variables

Description

This command gets four user selected dynamic Variables.

Write Parameters

Word	High Byte	Low Byte
0	Transmitter Variable Code For Slot 0	Transmitter Variable Code For Slot 1
1	Transmitter Variable Code For Slot 2	Transmitter Variable Code For Slot 3

Floating Point Data Returned

Word	High Byte	Low Byte
0	Variable for Slot 0	
1		
2	Variable for Slot 0	
3		
4	Variable for Slot 2	
5		
6	Variable for Slot 3	
7		

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	
1	Transmitter Variable Code For Slot 0	Units Code for Slot 0 Variable
2	Transmitter Variable Code For Slot 1	Units Code for Slot 1 Variable
3	Transmitter Variable Code For Slot 2	Units Code for Slot 2 Variable
4	Transmitter Variable Code For Slot 3	Units Code for Slot 3 Variable

COMMAND 34 - Write Damping Value

Description

This command writes the damping value of a HART device.

Write Parameters

Word	High Byte	Low Byte
0	Floating Point Damping Value (Sec)	
1		

Floating Point Data Returned

Word	High Byte	Low Byte
0	Floating Point Damping Value (Sec)	
1		

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	

COMMAND 35 - Write Range Values

Description

This command writes the Upper and Lower range of the Primary Variable

Write Parameters

Word	High Byte	Low Byte
0	Range Units Code	Floating Point Upper Range Value (Byte 0)
1	Floating Point Upper Range Value (Byte 1)	Floating Point Upper Range Value (Byte 2)
2	Floating Point Upper Range Value (Byte 3)	Floating Point Lower Range Value (Byte 0)
3	Floating Point Lower Range Value (Byte 1)	Floating Point Lower Range Value (Byte 2)
4	Floating Point Lower Range Value (Byte 3)	Floating Point Lower Range Value (Byte 3)

Floating Point Data Returned

Word	High Byte	Low Byte
0	Upper Range Value	
1		
2	Lower Range Value	
3		

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	
1	Range Units Code	0

COMMAND 36 - Set Upper Range Value

Description

This command is similar in effect to push the SPAN button of the HART device but doing it remotely through the HART network.

Write Parameters

NONE

Floating Point Data Returned

NONE

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	

COMMAND 37 - Set Lower Range Value

Description

This command is similar in effect to push the ZERO button of the HART device but doing it remotely through the HART network.

Write Parameters

NONE

Floating Point Data Returned

NONE

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	

COMMAND 38 - Reset Configuration Changed Flag

Description

This command resets the status bit that indicates that configuration has been changed.

Write Parameters

NONE

Floating Point Data Returned

NONE

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	

COMMAND 39 - EEPROM Control

Description

This command operates over the EEPROM changing its settings.

Write Parameters

Word	High Byte	Low Byte
0	EEPROM Control Code	EEPROM Control Code

Floating Point Data Returned

NONE

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	
1	EEPROM Control Code	0

COMMAND 40 - Enter Exit Fixed Current Mode

Description

This command writes the damping value of a HART device.

Write Parameters

Word	High Byte	Low Byte
0	Floating Point Current (mA)	
1		

Floating Point Data Returned

Word	High Byte	Low Byte
0	Floating Point Current (mA)	
1		

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	

COMMAND 41 - Perform Transmitter Self Test

Description

This command starts the HART device Self Test to find if there is any problem with the device hardware.

Write Parameters

NONE

Floating Point Data Returned

NONE

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	

COMMAND 42 - Perform Master Reset

Description

This command performs a master reset in the HART device.

Write Parameters

NONE

Floating Point Data Returned

NONE

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	

COMMAND 43 - Set PV Zero

Description

This command forces the Primary Value to Zero.

Write Parameters

NONE

Floating Point Data Returned

NONE

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	

COMMAND 44 - Write PV Units

Description

This command changes the setting of the Primary Variable units.

Write Parameters

Word	High Byte	Low Byte
0	Primary Variable Units Code	Primary Variable Units Code

Floating Point Data Returned

NONE

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	
1	Primary Variable Units Code	0

COMMAND 45 - Trim DAC Zero

Description

This command calibrates the device Digital to analog converter zero. This is done by writing the same current value that is measured in the loop.

Write Parameters

Word	High Byte	Low Byte
0	Floating Point measured current (mA)	
1		

Floating Point Data Returned

Word	High Byte	Low Byte
0	Floating Point Current (mA)	
1		

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	

COMMAND 46 - Trim DAC Gain

Description

This command calibrates the device Digital to analog converter gain. This is done by writing the same current value that is measured in the loop.

Write Parameters

Word	High Byte	Low Byte
0	Floating Point Measured Current (mA)	
1		

Floating Point Data Returned

Word	High Byte	Low Byte
0	Floating Point Current (mA)	
1		

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	

COMMAND 47 - Write Transfer Function

Description

This command changes the setting of the Transfer Function of the HART device.

Write Parameters

Word	High Byte	Low Byte
0	Transfer Function Code	Transfer Function Code

Floating Point Data Returned

NONE

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	
1	Transfer Function Code	0

COMMAND 48 - Read Additional Transmitter Status

Description

This command gets extended information about the status of the Transmitter. This information is specific for each transmitter.

Write Parameters

NONE

Floating Point Data Returned

NONE

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	
1	Additional Status (Byte 0)	Additional Status (Byte 1)
2	Additional Status (Byte 2)	Additional Status (Byte 3)
.	.	
.	.	
.	.	
13	Additional Status (Byte 24)	0

COMMAND 49 - Write PV Sensor Serial Number

Description

This command changes the sensor serial number of the Primary Variable.

Write Parameters

Word	High Byte	Low Byte
0	Sensor Serial Number (Byte 0)	Sensor Serial Number (Byte 1)
1	Sensor Serial Number (Byte 2)	Sensor Serial Number (Byte 2)

Floating Point Data Returned

NONE

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	
1	Sensor Serial Number (Byte 0)	Sensor Serial Number (Byte 1)
2	Sensor Serial Number (Byte 2)	0

COMMAND 50 - Read Dynamic Variable Assignments

Description

This command gets the actual assignment of the Dynamic Variables returned with command 3.

Write Parameters

NONE

Floating Point Data Returned

NONE

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	
1	Transmitter Variable Code For Primary Variable	Transmitter Variable Code For Second Variable
2	Transmitter Variable Code For Third Variable	Transmitter Variable Code For Fourth Variable

COMMAND 51 - Write Dynamic Variable Assignments

Description

This command sets the assignment of the Dynamic Variables returned with command 3.

Write Parameters

Word	High Byte	Low Byte
0	Transmitter Variable Code For Primary Variable	Transmitter Variable Code For Second Variable
1	Transmitter Variable Code For Third Variable	Transmitter Variable Code For Fourth Variable

Floating Point Data Returned

NONE

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	
1	Transmitter Variable Code For Primary Variable	Transmitter Variable Code For Second Variable
2	Transmitter Variable Code For Third Variable	Transmitter Variable Code For Fourth Variable

COMMAND 52 - Set Transmitter Variable Zero

Description

This command forces a selected transmitter variable to zero.

Write Parameters

Word	High Byte	Low Byte
0	Transmitter Variable Code	Transmitter Variable Code

Floating Point Data Returned

NONE

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	
1	Primary Variable Units Code	0

COMMAND 53 - Write Transmitter Variable Units

Description

This command changes a selected transmitter variable unit.

Write Parameters:

Word	High Byte	Low Byte
0	Transmitter Variable Code	Transmitter Variable Units Code

Floating Point Data Returned

NONE

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	
1	Primary Variable Units Code	Transmitter Variable Units Code

COMMAND 54 - Read Transmitter Variable Information

Description

This command gets information about any selected transmitter variable sensor.

Write Parameters

Word	High Byte	Low Byte
0	Transmitter Variable Code	Transmitter Variable Code

Floating Point Data Returned

Word	High Byte	Low Byte
0	Transmitter Variable Upper Limit	
1		
2	Transmitter Variable Lower Limit	
3		
4	Transmitter Variable Damping Value (Sec)	
5		

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	
1	Transmitter Variable Code	Sensor Serial Number 0
1	Sensor Serial Number 1	Sensor Serial Number 2
2	Unit Code for Limits	0

COMMAND 55 - Write Transmitter Variable Damping Value

Description

This command writes the Damping Value of a user selected transmitter variable.

Write Parameters

Word	High Byte	Low Byte
0	Transmitter Variable Code	Transmitter Variable Damping Value (Byte 0)
1	Transmitter Variable Damping Value (Byte 1)	Transmitter Variable Damping Value (Byte 2)
2	Transmitter Variable Damping Value (Byte 3)	Transmitter Variable Damping Value (Byte 3)

Floating Point Data Returned

Word	High Byte	Low Byte
0	Transmitter Variable Damping Value	
1		

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	
1	Transmitter Variable Code	0

COMMAND 56 - Write Transmitter Variable Sensor Serial Number

Description

This command writes the Serial Number of a user selected transmitter variable.

Write Parameters

Word	High Byte	Low Byte
0	Transmitter Variable Code	Transmitter Variable Sensor Serial Number (Byte 0)
1	Transmitter Variable Sensor Serial Number (Byte 1)	Transmitter Variable Sensor Serial Number (Byte 2)

Floating Point Data Returned

NONE

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	
1	Transmitter Variable Code	Transmitter Variable Sensor Serial Number (Byte 0)
2	Transmitter Variable Sensor Serial Number (Byte 1)	Transmitter Variable Sensor Serial Number (Byte 2)

COMMAND 57 - Read Unit Tag Descriptor Date

Description

This command reads an ASCII Tag which identifies the device, an ASCII descriptor of the device and the last Date it has been configured.

Write Parameters

NONE

Floating Point Data Returned

NONE

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	
1	TAG ASCII Character 0	TAG ASCII Character 1
2	TAG ASCII Character 2	TAG ASCII Character 3
3	TAG ASCII Character 4	TAG ASCII Character 5
4	TAG ASCII Character 6	TAG ASCII Character 7
5	Descriptor ASCII Character 0	Message ASCII Character 1
.	.	
.	.	
.	.	
12	Descriptor ASCII Character 14	Message ASCII Character 15
13	Date	Date
14	Date	0

COMMAND 58 - Write Unit Tag Descriptor Date

Description

This command writes an ASCII Tag which identifies the device, an ASCII descriptor of the device and the last Date it has been configured.

Write Parameters:

Word	High Byte	Low Byte
0	Packed ASCII TAG Byte 0	Packed ASCII TAG Byte 1
1	Packed ASCII TAG Byte 2	Packed ASCII TAG Byte 3
2	Packed ASCII TAG Byte 4	Packed ASCII TAG Byte 5
3	Packed ASCII Descriptor Byte 0	Packed ASCII Descriptor Byte 1
.	.	
.	.	
.	.	
8	Packed ASCII Descriptor Byte 10	Packed ASCII Descriptor Byte 11
9	Date	Date
10	Date	0

Floating Point Data Returned

NONE

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	
1	TAG ASCII Character 0	TAG ASCII Character 1
2	TAG ASCII Character 2	TAG ASCII Character 3
3	TAG ASCII Character 4	TAG ASCII Character 5
4	TAG ASCII Character 6	TAG ASCII Character 7
5	Descriptor ASCII Character 0	Message ASCII Character 1
.	.	
.	.	
.	.	
12	Descriptor ASCII Character 14	Message ASCII Character 15
13	Date	Date
14	Date	0

COMMAND 59 - Write Number Of Response Preambles

Description

This command sets the number of preambles that the HART slave will use in every command response.

Write Parameters

Word	High Byte	Low Byte
0	Number of Response Preambles	Number of Response Preambles

Floating Point Data Returned

NONE

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	
1	Number of Response Preambles	0

COMMAND 108 - Write Burst Mode Command Number

Description

This command sets the command number that the HART device will use in Burst Mode.

Write Parameters

Word	High Byte	Low Byte
0	Burst Mode Command Number	Burst Mode Command Number

Floating Point Data Returned

NONE

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	
1	Burst Mode Command Number	0

COMMAND 109 - Burst Mode Control

Description

This command sets the HART device Burst Mode.

Write Parameters

Word	High Byte	Low Byte
0	Burst Mode Control Code	Burst Mode Control Code

Floating Point Data Returned

NONE

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	
1	Burst Mode Control Code	0

1 = ON

0 = Off

COMMAND 110 - Read All Dynamic Variables

Description

This command gets all dynamic Variables.

Write Parameters

NONE

Floating Point Data Returned

Word	High Byte	Low Byte
0	Primary Variable	
1		
2	Second Variable	
3		
4	Third Variable	
5		
6	Fourth Variable	
7		

Integer Data Returned

Word	High Byte	Low Byte
0	STATUS WORD	
1	Primary Variable Units Code	Second Variable Units Code
2	Third Variable Units Code	Fourth Variable Units Code

8.10 Frequently Asked Questions

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

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

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

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

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

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

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

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

How do I configure the module?

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

What software application is required for my Ladder Logic?

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

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 number of words being transferred per command, the amount of other network traffic at the time data is being transferred, and overall processor scan times.

Is a .MDC available for configuration of the Module?

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

Does the module work in a remote rack?

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

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

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

Should I configure the port to use Multi-drop or Point to Point mode?

If you are considering using more than one device connected to the port you should configure the port to use Multi-drop mode. In this case the port will poll all slave addresses between 1 and *"Max Device Count"* (this is a parameter that can be configured in the configuration file).

If the port is configured for point to point operation only slave address 0 will be polled.

9 Support, Service & Warranty

In This Chapter

- Contacting Technical Support 177
- Return Material Authorization (RMA) Policies and Conditions..... 179
- LIMITED WARRANTY..... 181

Contacting Technical Support

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

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

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

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

Note: *For technical support calls within the United States, an after-hours answering system allows 24-hour/7-days-a-week pager access to one of our qualified Technical and/or Application Support Engineers.*

Internet	Web Site: www.prosoft-technology.com/support E-mail address: support@prosoft-technology.com
Asia Pacific (location in Malaysia)	Tel: +603.7724.2080, E-mail: asiapc@prosoft-technology.com Languages spoken include: Chinese, English
Asia Pacific (location in China)	Tel: +86.21.5187.7337 x888, E-mail: asiapc@prosoft-technology.com Languages spoken include: Chinese, English
Europe (location in Toulouse, France)	Tel: +33 (0) 5.34.36.87.20, E-mail: support.EMEA@prosoft-technology.com Languages spoken include: French, English
Europe (location in Dubai, UAE)	Tel: +971-4-214-6911, E-mail: mea@prosoft-technology.com Languages spoken include: English, Hindi

North America (location in California)	Tel: +1.661.716.5100, E-mail: support@prosoft-technology.com Languages spoken include: English, Spanish
Latin America (Oficina Regional)	Tel: +1-281-2989109, E-Mail: latinam@prosoft-technology.com Languages spoken include: Spanish, English
Latin America (location in Puebla, Mexico)	Tel: +52-222-3-99-6565, E-mail: soporte@prosoft-technology.com Languages spoken include: Spanish
Brasil (location in Sao Paulo)	Tel: +55-11-5083-3776, E-mail: brasil@prosoft-technology.com Languages spoken include: Portuguese, English

9.1 Return Material Authorization (RMA) Policies and Conditions

The following Return Material Authorization (RMA) Policies and Conditions (collectively, "RMA Policies") apply to any returned product. These RMA Policies are subject to change by ProSoft Technology, Inc., without notice. For warranty information, see Limited Warranty (page 181). In the event of any inconsistency between the RMA Policies and the Warranty, the Warranty shall govern.

9.1.1 Returning Any Product

a) In order to return a Product for repair, exchange, or otherwise, the Customer must obtain a Return Material Authorization (RMA) number from ProSoft Technology and comply with ProSoft Technology shipping instructions.
b) In the event that the Customer experiences a problem with the Product for any reason, Customer should contact ProSoft Technical Support at one of the telephone numbers listed above (page 177). A Technical Support Engineer will request that you perform several tests in an attempt to isolate the problem. If after completing these tests, the Product is found to be the source of the problem, we will issue an RMA.
c) All returned Products must be shipped freight prepaid, in the original shipping container or equivalent, to the location specified by ProSoft Technology, and be accompanied by proof of purchase and receipt date. The RMA number is to be prominently marked on the outside of the shipping box. Customer agrees to insure the Product or assume the risk of loss or damage in transit. Products shipped to ProSoft Technology using a shipment method other than that specified by ProSoft Technology, or shipped without an RMA number will be returned to the Customer, freight collect. Contact ProSoft Technical Support for further information.
d) A 10% restocking fee applies to all warranty credit returns, whereby a Customer has an application change, ordered too many, does not need, etc. Returns for credit require that all accessory parts included in the original box (i.e.; antennas, cables) be returned. Failure to return these items will result in a deduction from the total credit due for each missing item.

9.1.2 Returning Units Under Warranty

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

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

9.1.3 Returning Units Out of Warranty

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

The following is a list of non-repairable units:

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

9.2 LIMITED WARRANTY

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

9.2.1 What Is Covered By This Warranty

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

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

9.2.2 What Is Not Covered By This Warranty

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

9.2.3 Disclaimer Regarding High Risk Activities

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

9.2.4 Intellectual Property Indemnity

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

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

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

9.2.5 Disclaimer of all Other Warranties

The Warranty set forth in What Is Covered By This Warranty (page 181) are in lieu of all other warranties, express or implied, including but not limited to the implied warranties of merchantability and fitness for a particular purpose.

9.2.6 Limitation of Remedies **

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

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

9.2.7 Time Limit for Bringing Suit

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

9.2.8 No Other Warranties

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

9.2.9 Allocation of Risks

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

9.2.10 Controlling Law and Severability

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

Index

[

[HART PORT x COMMANDS] • 61
[HART PORT x] • 58
[Module] • 55

3

3x Register Start • 57

4

4x Register Start • 57

A

Adding the PTQ Module to the Project • 18, 30
Allocation of Risks • 184
Analyzing Data for Port 1 • 85
Application Port Connection • 116
Auto-Poll Code • 59
Auto-Poll DB Address • 60
Auto-Poll Disabled Mode • 110
Auto-Poll Modes • 106
Auto-Poll Swap Float • 60
Auto-Polling • 108

B

Backplane Data Transfer • 98, 100, 101
Block 9902
  Command Control • 65, 101, 103
Block 9903
  Command Disable Control • 65, 101, 104
Block 9998 or 9999
  Reboot Module • 101, 105
Building the Project • 19
Burst Mode • 107
Byte Count • 68

C

COMMAND 00 - Read Unique Identifier • 132
COMMAND 01 - Read Primary Variable • 133
COMMAND 02 - Read Current And Percent Of Range • 134
COMMAND 03 - Read Dynamic Variables • 135
COMMAND 06 - Write Polling Address • 136
COMMAND 108 - Write Burst Mode Command Number • 172
COMMAND 109 - Burst Mode Control • 173
COMMAND 110 - Read All Dynamic Variables • 174
COMMAND 12 - Read Message • 137
COMMAND 13 - Read Tag, Descriptor and Date • 138
COMMAND 14 - Read PV Sensor Info • 139
COMMAND 15 - Read Output Information • 140
COMMAND 16 - Read Final Assembly Number • 141
COMMAND 17 - Write Message • 142
COMMAND 18 - Write Tag, Descriptor and Date • 143
COMMAND 19 - Write Final Assembly Number • 144
COMMAND 33 - Read Transmitter Variables • 145
COMMAND 34 - Write Damping Value • 146
COMMAND 35 - Write Range Values • 147
COMMAND 36 - Set Upper Range Value • 148
COMMAND 37 - Set Lower Range Value • 149
COMMAND 38 - Reset Configuration Changed Flag • 150
COMMAND 39 - EEPROM Control • 151
COMMAND 40 - Enter Exit Fixed Current Mode • 152
COMMAND 41 - Perform Transmitter Self Test • 153
COMMAND 42 - Perform Master Reset • 154
COMMAND 43 - Set PV Zero • 155
COMMAND 44 - Write PV Units • 156
COMMAND 45 - Trim DAC Zero • 157
COMMAND 46 - Trim DAC Gain • 158
COMMAND 47 - Write Transfer Function • 159
COMMAND 48 - Read Additional Transmitter Status • 160
COMMAND 49 - Write PV Sensor Serial Number • 161
COMMAND 50 - Read Dynamic Variable Assignments • 162
COMMAND 51 - Write Dynamic Variable Assignments • 163
COMMAND 52 - Set Transmitter Variable Zero • 164
COMMAND 53 - Write Transmitter Variable Units • 165
COMMAND 54 - Read Transmitter Variable Information • 166
COMMAND 55 - Write Transmitter Variable Damping Value • 167
COMMAND 56 - Write Transmitter Variable Sensor Serial Number • 168
COMMAND 57 - Read Unit Tag Descriptor Date • 169
COMMAND 58 - Write Unit Tag Descriptor Date • 170
COMMAND 59 - Write Number Of Response Preambles • 171
Command Control • 100, 101
Command Count • 59
Command List Overview • 61
Commands Supported by the Module • 62, 64
Configuring Module Parameters • 54
Configuring the Processor with Concept • 25
Configuring the Processor with ProWORX • 41
Configuring the Processor with Unity Pro • 15
Connect the PC to the ProTalk Configuration/Debug Port • 47
Connect Your PC to the Processor • 20
Connecting to the Processor with TCPIP • 22
Contacting Technical Support • 177, 179
Controlling Law and Severability • 185
Creating a New Project • 16, 28

D

Data Analyzer • 83, 85
Data Exchange • 100
Database View Menu • 81, 89

DB Address Status • 59
Diagnostics and Troubleshooting • 9, 75
Disclaimer of all Other Warranties • 183
Disclaimer Regarding High Risk Activities • 182
Displaying the Current Page of Registers Again • 89
Displaying Timing Marks in the Data Analyzer • 86
Done DB Address • 68
Downloading the Project to the Module Using a Serial COM port • 74
Downloading the Project to the Processor • 23, 35

E

Enable • 65
Enable DB Address • 67
Enabled • 58
Error/Status Data Block • 125
Error/Status Offset • 55, 60
Example Field Terminations • 117
Example of HART Command Function 3 • 70
Example of HART Command Function 34 • 72
Exiting the Program • 82

F

Failure Flag Count • 55
Fixed Data • 69
Float DB Address • 65
FP Word Count • 66
Frequently Asked Questions • 175
Function Code • 66
Functional Overview • 9, 97
Functional Specifications • 96

G

General Concepts • 97
General Specifications • 94
Guide to the PTQ-HART User Manual • 9

H

Hardware and Software Requirements • 12
Hardware Specifications • 95
HART Channels • 105
HART Command Entry Formats • 64
Hart Command Examples • 70
HART Command List • 107
HART Common Practice Commands • 61, 145
HART Common Practice Commands Set • 62
HART Master Menu • 83
HART Universal Commands • 61, 132
HART Universal Commands Set • 62
How to Contact Us • 2

I

Information for Concept Version 2.6 Users • 26
Information for ProTalk® Product Users • 3
Initializing Output Data • 55
Installing MDC Configuration Files • 26
Installing ProSoft Configuration Builder Software • 14
Installing the ProTalk Module in the Quantum Rack • 46
Int Word Count • 67
Int. DB Address • 67
Intellectual Property Indemnity • 183

K

Keystrokes • 79

L

LED Status Indicators • 76
Limitation of Remedies ** • 184
LIMITED WARRANTY • 179, 181

M

Main Logic Loop • 97
Main Menu • 80
Master Command Error List Menu • 87
Master Command List Menu • 88
Max Device Count • 60
Module Configuration • 51
Module Name • 55
Module Power Up • 97
Moving Back Through 5 Pages of Commands • 87
Moving Back Through 5 Pages of Registers • 90
Moving Forward (Skipping) Through 5 Pages of Commands • 88
Moving Forward (Skipping) Through 5 Pages of Registers • 90
Multi-drop Mode • 112

N

Navigation • 79
No Other Warranties • 184
Notes • 117

O

Opening the Command Error List Menu • 83
Opening the Command List Menu • 83
Opening the Data Analyzer Menu • 83
Opening the Database View Menu • 81
Opening the HART Master Menu • 81

P

Package Contents • 12
PC and PC Software • 13
Pinouts • 3, 117
Point-to-Point Mode • 110
Poll Interval • 65
Preambles • 58
Primary Master • 58
Printing a Configuration File • 54
Product Specifications • 9, 94
ProSoft Technology® Product Documentation • 2
Protocol Error Codes • 129
PTQ-HART Error Status Table • 125

Q

Quantum Hardware • 13

R

Read Register Count • 56
Read Register Start • 56
Reading Status Data from the Module • 91
Redisplaying the Current Page • 87, 88
Redisplaying the Menu • 83
Reference • 9, 93
Removing Timing Marks in the Data Analyzer • 86
Renaming PCB Objects • 54
Retry Count • 58
Return Material Authorization (RMA) Policies and Conditions • 179
Returning Any Product • 179
Returning to the Main Menu • 84, 87, 88, 89, 90
Returning Units Out of Warranty • 180
Returning Units Under Warranty • 180
RS-232 Configuration/Debug Port • 115

S

Set Up the Project • 52
Setting up Data Memory in Project • 32, 49
Setting Up the ProTalk Module • 45
Short Address • 66
Slave List Status • 59
Special Functions • 103
Start Here • 9, 11
Starting the Data Analyzer • 86
Stopping the Data Analyzer • 87
Support, Service & Warranty • 9, 177
Swap Code • 66, 67, 68

T

Thermal Connector Port Details • 124
Time Limit for Bringing Suit • 184
Transferring the Configuration File from The Module to the PC • 81
Transferring the Configuration File from the PC to the Module • 81

U

Use Long • 67
Using ProSoft Configuration Builder • 52
Using ProSoft Configuration Builder (PCB) for Diagnostics • 77
Using the Diagnostic Window in ProSoft Configuration Builder • 77

V

Verifying Communication Between the Processor and the Module • 49
Verifying Successful Download • 37
Viewing Block Transfer Statistics • 80
Viewing Data in ASCII (Text) Format • 86, 90
Viewing Data in Decimal Format • 9, 90
Viewing Data in Floating-Point Format • 90
Viewing Data in Hexadecimal Format • 86, 90
Viewing Module Configuration • 80
Viewing Port Status and Configuration • 84
Viewing Register Pages • 89
Viewing the Master Command List Help • 84
Viewing the Next 50 Commands • 88
Viewing the Next Page of Commands • 88
Viewing the Next Page of Registers • 90
Viewing the Previous 50 Commands • 88
Viewing the Previous Page of Commands • 88
Viewing the Previous Page of Registers • 90
Viewing the Slave Status List • 83
Viewing Version Information • 82

W

Warm Booting the Module • 82
Warnings • 3
What Is Covered By This Warranty • 181, 183
What Is Not Covered By This Warranty • 182
Write DB Address • 68
Write Register Count • 56
Write Register Start • 56

Y

Your Feedback Please • 2