



  
**ProLinX**<sup>®</sup>  
**IEC-101 Slave**

**ProLinX Communication Gateway**

IEC 60870-5-101 v3 Slave  
(Firmware version 3.xx)

August 6, 2010

## Your Feedback Please

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

### **ProSoft Technology**

5201 Truxtun Ave., 3rd Floor

Bakersfield, CA 93309

+1 (661) 716-5100

+1 (661) 716-5101 (Fax)

[www.prosoft-technology.com](http://www.prosoft-technology.com)

[support@prosoft-technology.com](mailto:support@prosoft-technology.com)

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

101S v3 Protocol Manual

8/5/2010

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

## **ProSoft Technology<sup>®</sup> 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](http://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

## Important Installation Instructions

Power, Input, and Output (I/O) wiring must be in accordance with Class I, Division 2 wiring methods, Article 501-4 (b) of the National Electrical Code, NFPA 70 for installation in the U.S., or as specified in Section 18-1J2 of the Canadian Electrical Code for installations in Canada, and in accordance with the authority having jurisdiction. The following warnings must be heeded:

- A** WARNING - EXPLOSION HAZARD - SUBSTITUTION OF COMPONENTS MAY IMPAIR SUITABILITY FOR CLASS I, DIV. 2;
- B** WARNING - EXPLOSION HAZARD - WHEN IN HAZARDOUS LOCATIONS, TURN OFF POWER BEFORE REPLACING OR WIRING MODULES
- C** WARNING - EXPLOSION HAZARD - DO NOT DISCONNECT EQUIPMENT UNLESS POWER HAS BEEN SWITCHED OFF OR THE AREA IS KNOWN TO BE NON-HAZARDOUS.
- D** THIS DEVICE SHALL BE POWERED BY CLASS 2 OUTPUTS ONLY.

## All ProLinx® Products

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'EQUIPMENT, COUPER LE COURANT OU S'ASSURER QUE L'EMPLACEMENT EST DÉSIGNÉ NON DANGEREUX.

## Markings

|        |   |
|--------|---|
| UL/cUL | ISA 12.12.01 Class I, Div 2 Groups A, B, C, D |
| cUL    | C22.2 No. 213-M1987                           |



183151

CL I Div 2 GPs A, B, C, D

II 3 G

Ex nA nL IIC X

0°C <= Ta <= 60°C

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

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

G – Equipment protected against explosive gasses.

## ProLinx Gateways with Ethernet Ports

Series C ProLinx™ Gateways with Ethernet ports do **NOT** include the HTML Web Server. The HTML Web Server must be ordered as an option. This option requires a factory-installed hardware addition. The HTML Web Server now supports:

- 8 MB file storage for HTML files and associated graphics files (previously limited to 384K)
- 32K maximum HTML page size (previously limited to 16K)

### ***To upgrade a previously purchased Series C model:***

Contact your ProSoft Technology distributor to order the upgrade and obtain a Returned Merchandise Authorization (RMA) to return the unit to ProSoft Technology.

### ***To order a ProLinx Plus gateway with the -WEB option***

Add **-WEB** to the standard ProLinx part number. For example, **5201-MNET-MCM-WEB**.



# Contents

|   |   |
|---|---|
| Your Feedback Please.....                                 | 2 |
| ProSoft Technology® Product Documentation.....            | 2 |
| Important Installation Instructions.....                  | 3 |
| All ProLinx® Products.....                                | 3 |
| ProLinx Gateways with Ethernet Ports.....                 | 3 |
| To upgrade a previously purchased Series C model:.....    | 3 |
| To order a ProLinx Plus gateway with the -WEB option..... | 3 |

## Guide to the 101S v3 Protocol Manual **9**

### **1 Start Here **11****

|       |  |    |
|-------|--|----|
| 1.1   | What's New .....                                       | 11 |
| 1.2   | ProLinx Reference Guide.....                           | 11 |
| 1.3   | Installing ProSoft Configuration Builder Software..... | 12 |
| 1.3.1 | Using the Online Help .....                            | 12 |
| 1.4   | Using ProSoft Configuration Builder .....              | 13 |
| 1.4.1 | Setting Up the Project .....                           | 13 |
| 1.4.2 | Renaming PCB Objects .....                             | 15 |
| 1.4.3 | Configuring Module Parameters .....                    | 15 |
| 1.4.4 | Creating Optional Comment Entries .....                | 16 |
| 1.4.5 | Printing a Configuration File.....                     | 16 |
| 1.4.6 | Using the CommonNet Data Map .....                     | 17 |
| 1.4.7 | Downloading the Project to the Module .....            | 21 |

### **2 101S Protocol Configuration **23****

|        |                                     |    |
|--------|-------------------------------------|----|
| 2.1    | [IEC-870-5-101 Port 0] .....        | 24 |
| 2.1.1  | Enabled .....                       | 24 |
| 2.1.2  | Time DB Offset.....                 | 24 |
| 2.1.3  | Disable Time Sync Events .....      | 25 |
| 2.1.4  | Data Link Address Value.....        | 25 |
| 2.1.5  | Data link address length.....       | 25 |
| 2.1.6  | Common Address of ASDU Val .....    | 25 |
| 2.1.7  | Common Address of ASDU Len .....    | 25 |
| 2.1.8  | Inform. Object Address Len.....     | 26 |
| 2.1.9  | Cyclic Data Transmission.....       | 26 |
| 2.1.10 | Select/Operate Timeout .....        | 26 |
| 2.1.11 | Use ACTTERM with Set Point .....    | 26 |
| 2.1.12 | Use ACTTERM with Step.....          | 26 |
| 2.1.13 | Single char ACK F0, 1 or 3.....     | 26 |
| 2.1.14 | Single char ACK C1 or C2.....       | 27 |
| 2.1.15 | Maximum ASDU Resp Len .....         | 27 |
| 2.1.16 | Cause of Trans(mission) Octets..... | 27 |
| 2.1.17 | Freeze Start Type.....              | 27 |
| 2.1.18 | Interval for Freeze .....           | 29 |
| 2.1.19 | Set Priority Queues .....           | 30 |
| 2.1.20 | Event Priority .....                | 31 |
| 2.1.21 | Cyclic Set IV Time - PLX 101S ..... | 31 |

|        |                               |    |
|--------|-------------------------------|----|
| 2.1.22 | IV Check Delay Time .....     | 33 |
| 2.1.23 | IV Fail Count .....           | 33 |
| 2.1.24 | Event Scan Delay .....        | 33 |
| 2.1.25 | Scan Events.....              | 34 |
| 2.1.26 | Time Type .....               | 35 |
| 2.1.27 | Use Balanced Mode .....       | 35 |
| 2.1.28 | Retry Count.....              | 35 |
| 2.1.29 | Response Timeout.....         | 36 |
| 2.1.30 | Baud Rate .....               | 36 |
| 2.1.31 | Parity.....                   | 36 |
| 2.1.32 | RTS On.....                   | 36 |
| 2.1.33 | RTS Off.....                  | 36 |
| 2.1.34 | Minimum Delay .....           | 36 |
| 2.1.35 | Receive Timeout.....          | 37 |
| 2.1.36 | Hardware Handshaking .....    | 37 |
| 2.2    | [IEC-870-5-101 Database]..... | 38 |
| 2.2.1  | Short Pulse Time .....        | 38 |
| 2.2.2  | Long Pulse Time .....         | 38 |
| 2.2.3  | Point Count .....             | 39 |
| 2.2.4  | Sequence Flag.....            | 40 |
| 2.2.5  | Parameter Offset .....        | 41 |
| 2.3    | [M_SP_NA_1] .....             | 42 |
| 2.4    | [M_DP_NA_1] .....             | 42 |
| 2.5    | [M_ST_NA_1] .....             | 43 |
| 2.6    | [M_BO_NA_1 104].....          | 43 |
| 2.7    | [M_ME_NA_1].....              | 44 |
| 2.8    | [M_ME_NB_1].....              | 44 |
| 2.9    | [M_ME_NC_1] .....             | 45 |
| 2.10   | [M_IT_NA_1].....              | 45 |
| 2.11   | [C_SC_NA_1] .....             | 46 |
| 2.12   | [C_DC_NA_1] .....             | 46 |
| 2.13   | [C_RC_NA_1].....              | 47 |
| 2.14   | [C_BO_NA_1 104] .....         | 47 |
| 2.15   | [C_SE_NA_1].....              | 48 |
| 2.16   | [C_SE_NB_1].....              | 48 |
| 2.17   | [C_SE_NC_1] .....             | 49 |
| 2.18   | Group Codes.....              | 50 |

**3 Module Communication Ports 51**

|       |  |    |
|-------|--|----|
| 3.1   | Serial Port Cable Connections: Config/Debug and Port 0.....    | 51 |
| 3.2   | RS-232 Configuration/Debug Port.....                           | 52 |
| 3.3   | Application Serial Port(s) .....                               | 53 |
| 3.3.1 | RS-232: Modem Connection (Hardware Handshaking Required) ..... | 53 |
| 3.3.2 | RS-232: Null Modem Connection (Hardware Handshaking).....      | 53 |
| 3.3.3 | RS-232: Null Modem Connection (No Hardware Handshaking) .....  | 54 |
| 3.3.4 | RS-422.....  | 55 |
| 3.3.5 | RS-485 Application Port(s) .....                               | 55 |
| 3.3.6 | RS-485 and RS-422 Tip .....                                    | 55 |

|          |   |            |
|----------|---|------------|
| <b>4</b> | <b>Diagnostics and Troubleshooting</b>                            | <b>57</b>  |
| 4.1      | LED Indicators .....  | 57         |
| 4.1.1    | Base Module LEDs.....   | 57         |
| 4.1.2    | Mini-DIN8 Serial Port.....  | 58         |
| 4.2      | Using ProSoft Configuration Builder (PCB) for Diagnostics.....    | 59         |
| 4.2.1    | Using the Diagnostic Window in ProSoft Configuration Builder..... | 59         |
| 4.2.2    | Navigation .....  | 61         |
| 4.2.3    | Keystrokes.....   | 61         |
| <b>5</b> | <b>Reference</b>  | <b>63</b>  |
| 5.1      | Product Specifications .....                                      | 63         |
| 5.1.1    | General Specifications .....                                      | 63         |
| 5.1.2    | Functional Specifications - IEC 60870-5-101 Slave.....            | 64         |
| 5.1.3    | Internal Database .....   | 64         |
| 5.1.4    | Hardware Specifications.....                                      | 66         |
| 5.2      | IEC-60870-5-101 Slave Protocol Implementation.....                | 67         |
| 5.2.1    | Data Flow Between the Module and External Host .....              | 68         |
| 5.2.2    | General Parameter Configuration .....                             | 72         |
| 5.2.3    | Monitor Direction and Control Direction .....                     | 80         |
| 5.2.4    | Using Monitor Points .....  | 81         |
| 5.2.5    | Using Control (Command) Points .....                              | 88         |
| 5.2.6    | Events.....   | 99         |
| 5.2.7    | Command Block Functionality.....                                  | 113        |
| 5.2.8    | Slave Error and Status .....                                      | 128        |
| <b>6</b> | <b>IEC 60870-5-101 Slave Interoperability Statement</b>           | <b>129</b> |
| 6.1      | System or Device .....  | 130        |
| 6.2      | Network Configuration.....  | 130        |
| 6.3      | Physical Layer .....  | 130        |
| 6.3.1    | Transmission Speed (Control Direction) .....                      | 130        |
| 6.3.2    | Transmission Speed (Monitor Direction).....                       | 130        |
| 6.4      | Link Layer.....   | 131        |
| 6.5      | Application Layer.....  | 131        |
| 6.5.1    | Transmission Mode for Application Data .....                      | 131        |
| 6.5.2    | Common Address of ASDU - 101S.....                                | 131        |
| 6.5.3    | Information Object Address.....                                   | 131        |
| 6.5.4    | Cause of Transmission.....  | 131        |
| 6.5.5    | Length of APDU - 101S.....  | 131        |
| 6.6      | Selection of Standard ASDUs .....                                 | 132        |
| 6.6.1    | Process Information in Monitor Direction .....                    | 132        |
| 6.6.2    | Process Information in Control Direction.....                     | 133        |
| 6.6.3    | System Information in Monitor Direction .....                     | 134        |
| 6.6.4    | System Information in Control Direction.....                      | 134        |
| 6.6.5    | Parameter in Control Direction .....                              | 134        |
| 6.6.6    | File Transfer .....   | 134        |
| 6.7      | Type Identifier and Cause of Transmission Assignments.....        | 135        |
| 6.8      | Basic Application Functions .....                                 | 137        |
| 6.8.1    | Station Initialization .....                                      | 137        |
| 6.8.2    | Cyclic Data Transmission.....                                     | 137        |
| 6.8.3    | Read Procedure .....  | 137        |

|              |   |            |
|--------------|---|------------|
| 6.8.4        | Spontaneous Transmission .....                                    | 137        |
| 6.8.5        | Double Transmission of Information Objects.....                   | 138        |
| 6.8.6        | Station Interrogation .....                                       | 138        |
| 6.8.7        | Clock Synchronization .....                                       | 138        |
| 6.8.8        | Command Transmission.....   | 139        |
| 6.8.9        | Transmission of Integrated Totals .....                           | 139        |
| 6.8.10       | Parameter Loading .....   | 140        |
| 6.8.11       | Parameter Activation .....  | 140        |
| 6.8.12       | Test Procedure .....  | 140        |
| 6.8.13       | File Transfer.....  | 140        |
| 6.8.14       | Background Scan .....   | 141        |
| 6.8.15       | Aquisition of transmission delay .....                            | 141        |
| <b>7</b>     | <b>101S v3 Database Design Forms</b>                              | <b>143</b> |
| 7.1          | M_SP_NA, M_DP_NA, M_ST_NA, M_BO_NA, and M_IT_NA Form .....        | 144        |
| 7.2          | M_ME_NA and M_ME_NB Form.....                                     | 145        |
| 7.3          | Form for All C_ (Command) Data Types, Except C_RC_NA .....        | 146        |
| 7.4          | C_RC_NA Form.....   | 147        |
| <b>8</b>     | <b>Support, Service &amp; Warranty</b>                            | <b>149</b> |
| 8.1          | How to Contact Us: Technical Support.....                         | 149        |
| 8.2          | Return Material Authorization (RMA) Policies and Conditions ..... | 150        |
| 8.2.1        | All Product Returns:.....   | 150        |
| 8.2.2        | Procedures for Return of Units Under Warranty:.....               | 151        |
| 8.2.3        | Procedures for Return of Units Out of Warranty:.....              | 151        |
| 8.3          | LIMITED WARRANTY .....  | 152        |
| 8.3.1        | What Is Covered By This Warranty .....                            | 152        |
| 8.3.2        | What Is Not Covered By This Warranty.....                         | 153        |
| 8.3.3        | Disclaimer Regarding High Risk Activities.....                    | 153        |
| 8.3.4        | Intellectual Property Indemnity .....                             | 154        |
| 8.3.5        | Disclaimer of all Other Warranties .....                          | 154        |
| 8.3.6        | Limitation of Remedies ** .....                                   | 155        |
| 8.3.7        | Time Limit for Bringing Suit.....                                 | 155        |
| 8.3.8        | No Other Warranties .....   | 155        |
| 8.3.9        | Allocation of Risks .....   | 155        |
| 8.3.10       | Controlling Law and Severability .....                            | 156        |
| <b>Index</b> |   | <b>157</b> |



## Guide to the 101S v3 Protocol Manual

| Function   |   | Section to Read  | Details   |
|--|---|--|---|
| Introduction<br>(Must Do)                                  | → | Start Here (page 11)   | This section introduces the customer to the gateway. Included are: package contents, system requirements, hardware installation, and basic configuration. |
| Diagnostic and Troubleshooting                             | → | Diagnostics and Troubleshooting (page 57)  | This section describes Diagnostic and Troubleshooting procedures.   |
| Reference<br>Product Specifications<br>Functional Overview | → | Reference (page 63)<br>Product Specifications (page 63)<br>Functional Overview (page 67) | These sections contain general references associated with this product, Specifications, and the Functional Overview.                                      |
| Support, Service, and Warranty<br>Index                    | → | Support, Service and Warranty (page 149)<br>Index  | This section contains Support, Service and Warranty information.<br>Index of chapters.  |



# 1 Start Here

## *In This Chapter*

|   |    |
|---|----|
| ❖ What's New .....  | 11 |
| ❖ ProLinx Reference Guide .....                           | 11 |
| ❖ Installing ProSoft Configuration Builder Software ..... | 12 |
| ❖ Using ProSoft Configuration Builder .....               | 13 |

For most applications, the installation and configuration steps described in this section will work without additional programming. ProSoft Technology strongly recommends that you complete the steps in this chapter before developing a custom application.

## 1.1 What's New

This new and enhanced version 3 of ProSoft Technology's IEC 60870-5-101 Slave implementation includes several new features not found on previous versions. Version 3 now supports 32-bit bitstring data types. Version 3 also supports more Causes of Transmission than previous versions. For additional details, please see the Interoperability Statement (page 129).

## 1.2 ProLinx Reference Guide

The *ProLinx Reference Guide* on the ProSoft Solutions CD-ROM provides detailed information on the entire range of ProLinx gateways. If you have any questions that are not answered in the 101S Protocol Manual, please refer to the *ProLinx Reference Guide*.

### 1.3 Installing ProSoft Configuration Builder Software

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

#### **To install 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 gateway.

#### **To install 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 gateway.
- 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.

#### **1.3.1 Using the Online Help**

Most of the information needed to help you use ProSoft Configuration Builder is provided in a Help System that is always available whenever you are running ProSoft Configuration Builder. The Help System does not require an Internet connection.

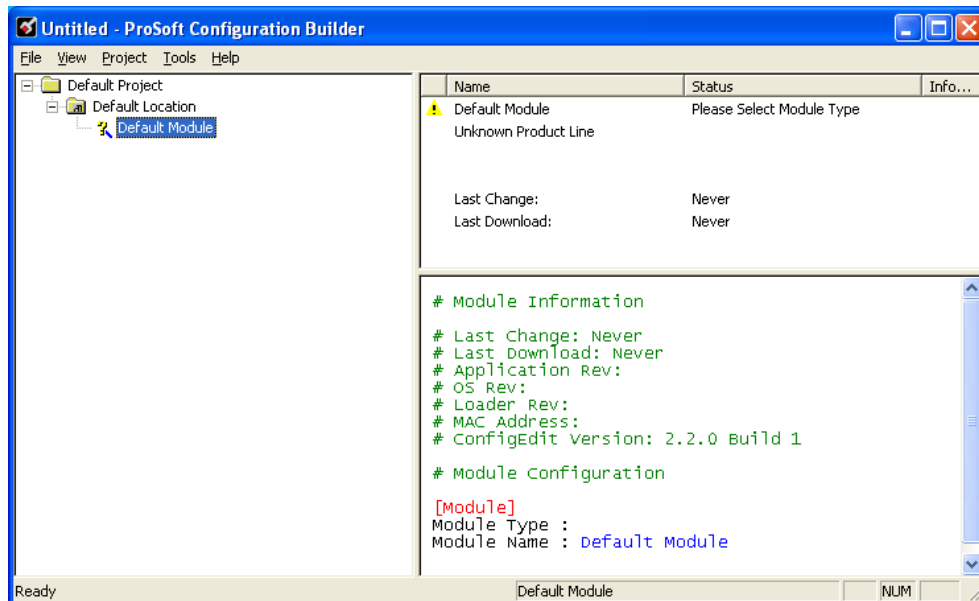
To view the help pages, start ProSoft Configuration Builder, open the **HELP** menu, and then choose **CONTENTS**.

## 1.4 Using ProSoft Configuration Builder

*ProSoft Configuration Builder (PCB)* provides a quick and easy way to manage gateway 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.

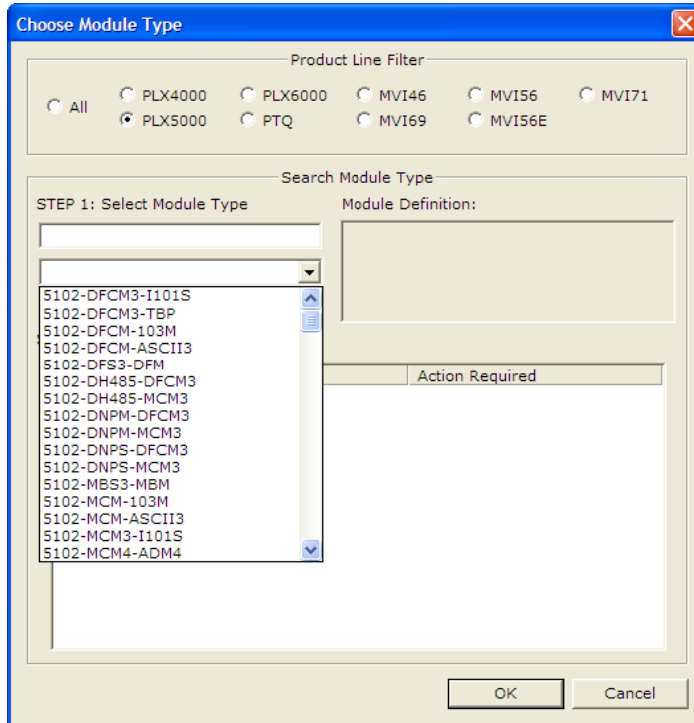
### 1.4.1 Setting Up the Project

To begin, start ProSoft Configuration Builder (PCB). If you have used other Windows configuration tools before, you will find the screen layout familiar. ProSoft Configuration Builder's (PCB'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 PCB, 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 PCB window with a new project.



***To add the gateway to the project***

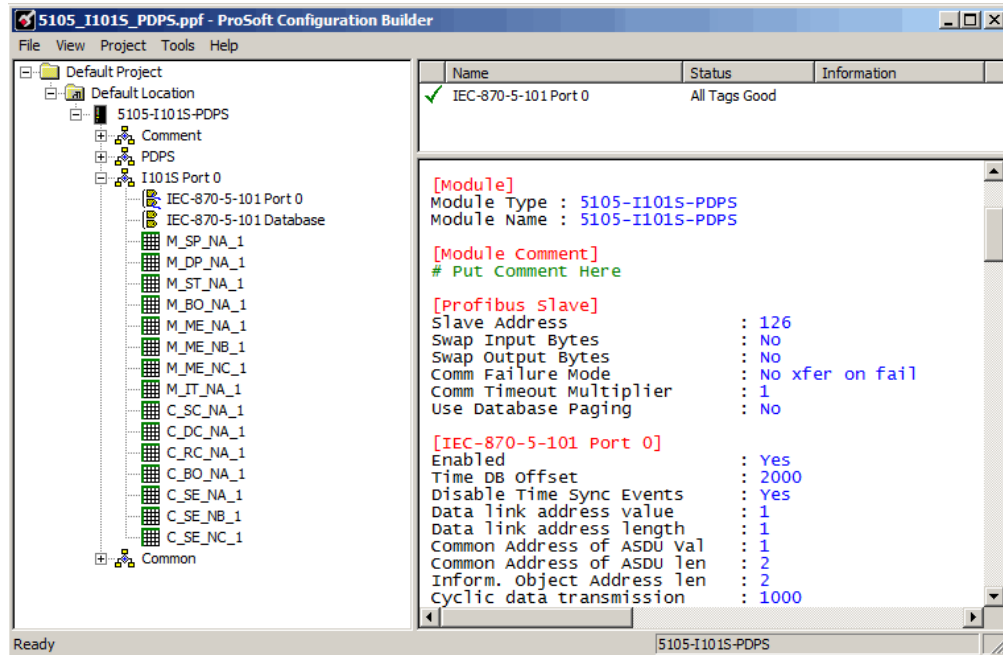
- 1 Use the mouse to select **DEFAULT MODULE** in the tree view, and then click the right mouse button to open a shortcut menu.
- 2 On the shortcut menu, choose **CHOOSE MODULE TYPE**. This action opens the *Choose Module Type* dialog box.



- 3 In the *Product Line Filter* area of the dialog box, select the appropriate product type radio button.
- 4 In the *STEP 1: Select Module Type* dropdown list, select the model number that matches your gateway, and then click **OK** to save your settings and return to the *PCB Main* window.

## 1.4.2 Renaming PCB Objects

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





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


### To rename an object

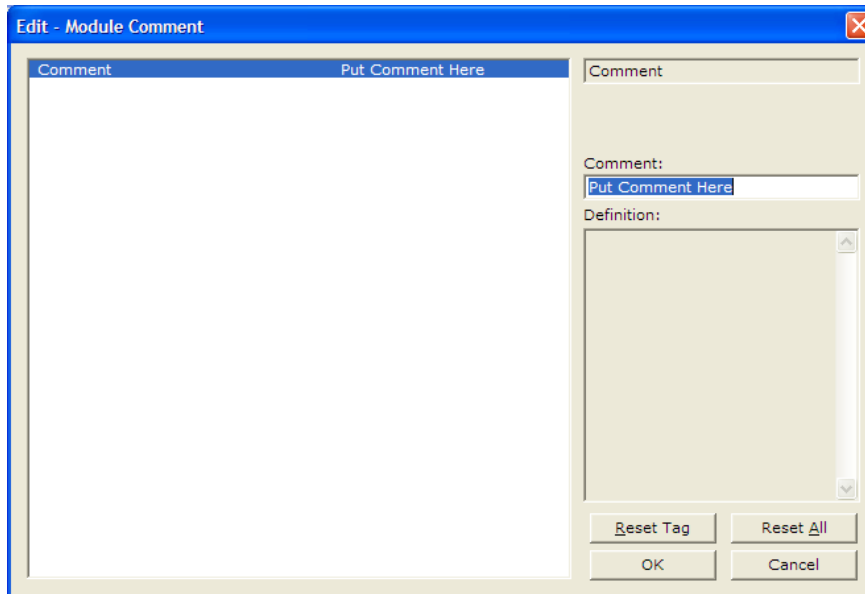
- 1 Select the object, and then click the right mouse button to open a shortcut menu. From the shortcut menu, choose **RENAME**.
- 2 Type the name to assign to the object.
- 3 Click away from the object to save the new name.

## 1.4.3 Configuring Module Parameters

- 1 Click on the **[+]** sign next to the gateway icon to expand gateway information.
- 2 Click on the **[+]** sign next to any  icon to view gateway 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.

### 1.4.4 Creating Optional Comment Entries

- 1 Click the **[+]** to the left of the  **Comment** icon to expand the module comments.
- 2 Double-click the  **Module Comment** icon. The *Edit - Module Comment* dialog box appears.



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

### 1.4.5 Printing a Configuration File

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



### 1.4.6 Using the CommonNet Data Map

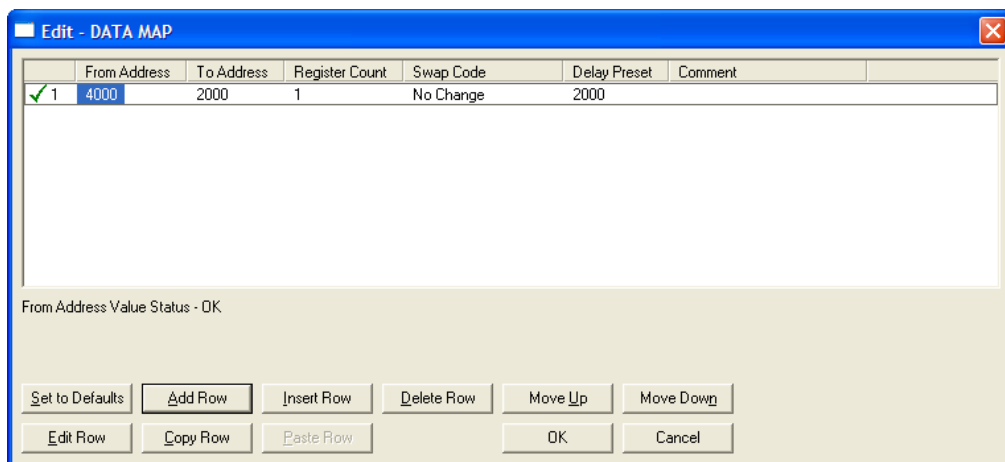
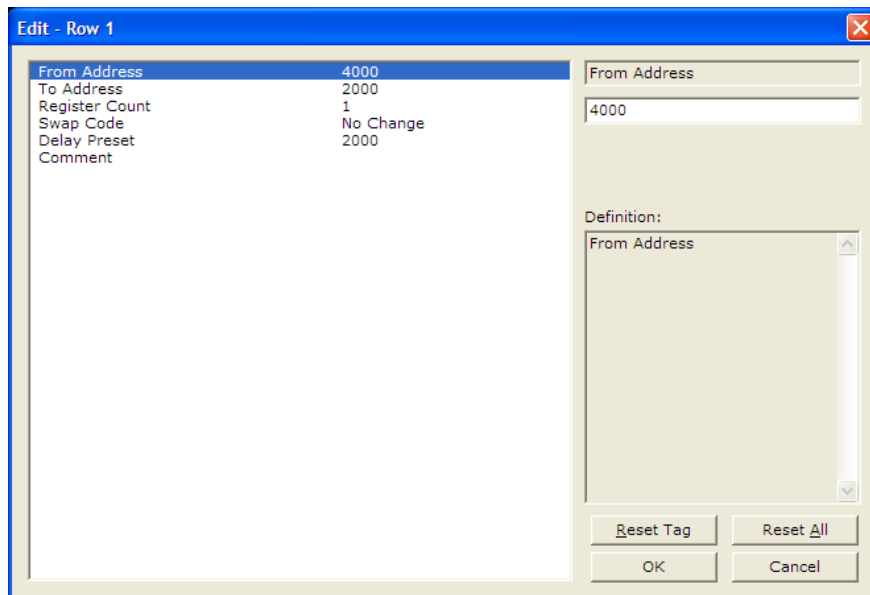
The *Data Map* section allows you to copy data between areas in the gateway's internal database.

You can copy a maximum of 100 registers per *Data Map* command, and you can configure a maximum of 200 separate copy commands.

You can copy data from the error or status tables in upper memory to internal database registers in the *User Data memory* area.

You can rearrange the byte and/or word order during the copy process. For example, by rearranging byte or word order, you can convert floating-point values to the correct format for a different protocol.

You can also use the *Data Map* to condense widely dispersed data into one contiguous data block, making it easier to access.



From Address

**0** to highest *Status Data* address

This field specifies the beginning internal database register address for the copy operation. This address can be any valid address in the *User Data Area* or the *Status Data Area* of the gateway.

To Address

**0** to **3999**

This parameter specifies the beginning destination register address for the copy operation. This address must always be within the *User Data registers* area. Take care to specify a destination address that will not overwrite data that has been stored in memory by one of the communication protocols running on the gateway.

Register Count

**1** to **100**

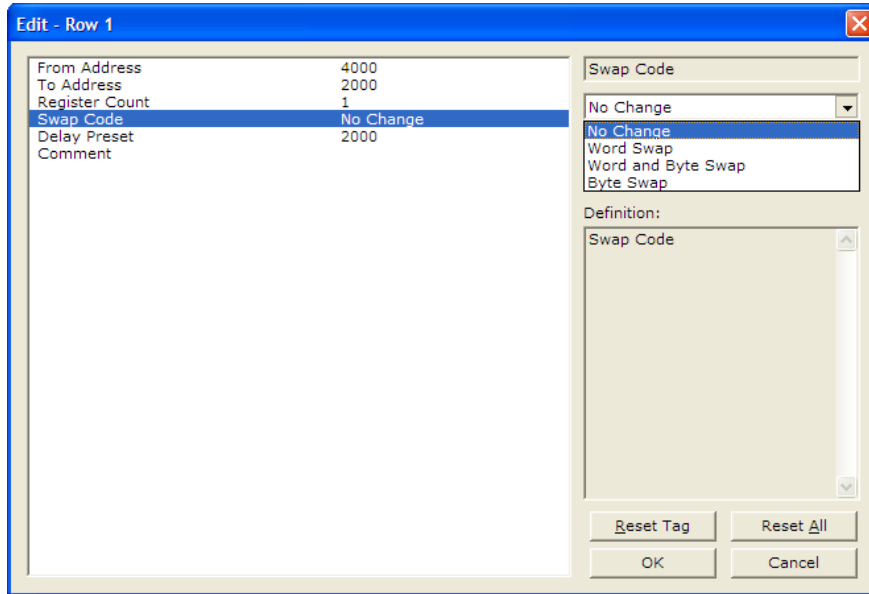
This parameter specifies the number of registers to copy.

Swap Code

**NO CHANGE, WORD SWAP, WORD AND BYTE SWAP, BYTE SWAP**

You may need to swap the order of the bytes in the registers during the copy process in order to change the alignment of bytes between dissimilar protocols. This parameter is helpful when dealing with floating-point or other multi-register values, as there is no standard method of storage of these data types in slave devices.

The following table defines the values and their associated operations:



| Swap Code          | Description  |
|--------------------|--|
| No Swap            | No change is made in the byte ordering (1234 = 1234)                       |
| Word Swap          | The words are swapped (1234=3412)  |
| Word and Byte Swap | The words are swapped, then the bytes in each word are swapped (1234=4321) |
| Bytes              | The bytes in each word are swapped (1234=2143)                             |

### Delay Preset

This parameter sets an interval for each *Data Map* copy operation. The value you put for the *Delay Preset* is not a fixed amount of time. It is the number of firmware scans that must transpire between copy operations.

The firmware scan cycle can take a variable amount of time, depending on the level of activity of the protocol drivers running on the ProLinx gateway and the level of activity on the gateway's communication ports. Each firmware scan can take from 1 to several milliseconds to complete. Therefore, *Data Map* copy operations cannot be expected to happen at regular intervals.

If multiple copy operations (several rows in the *Data map* section) happen too frequently or all happen in the same update interval, they could delay the process scan of the gateway protocols, which could result in slow data updates or missed data on communication ports. To avoid these potential problems, you should set the *Delay Preset* to different values for each row in the *Data Map* section and set them to higher, rather than lower, numbers.

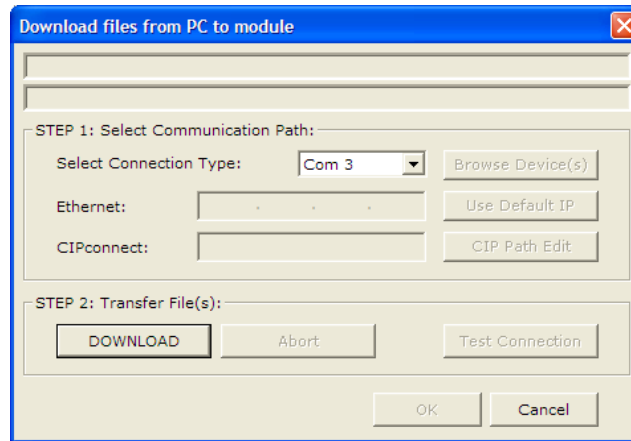
For example, *Delay Preset* values below 1000 could begin to cause a noticeable delay in data updates through the communication ports. And you should not set all *Delay Presets* to the same value. Instead, use different values for each row in the Data Map such as 1000, 1001, and 1002 or any other different *Delay Preset* values you like. This will prevent the copies from happening concurrently and prevent possible process scan delays.

### 1.4.7 Downloading the Project to the Module

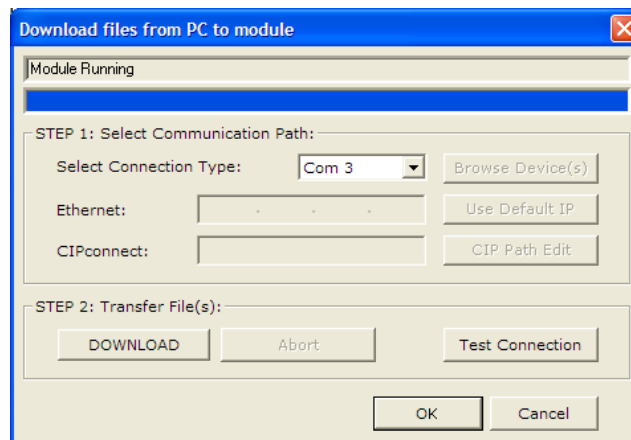
For the gateway to use the settings you configured, you must download (copy) the updated *Project* file from your PC to the gateway.

#### To Download the Project File

- 1 In the tree view in *ProSoft Configuration Builder*, click once to select the gateway.
- 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 gateway 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*.





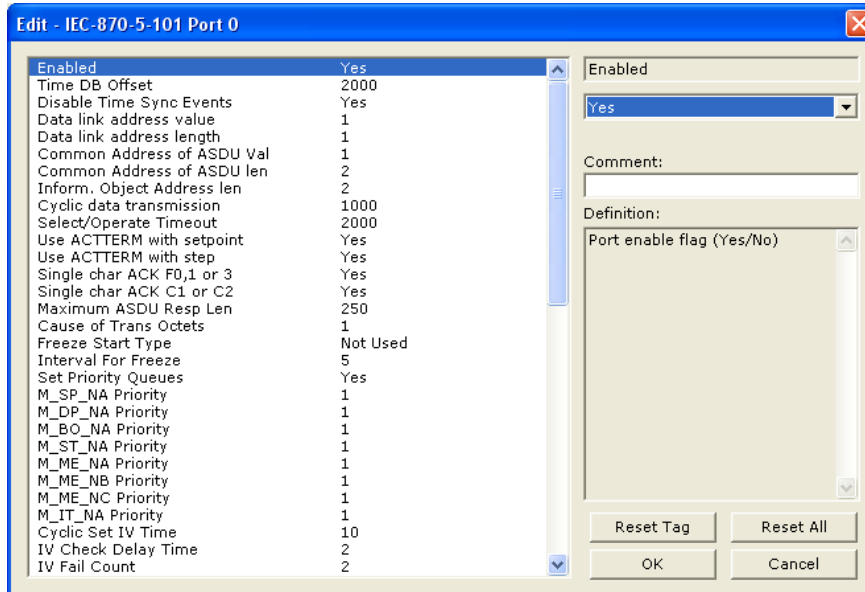
## 2 101S Protocol Configuration

### In This Chapter

|                                 |    |
|---------------------------------|----|
| ❖ [IEC-870-5-101 Port 0] .....  | 24 |
| ❖ [IEC-870-5-101 Database]..... | 38 |
| ❖ [M_SP_NA_1].....              | 42 |
| ❖ [M_DP_NA_1] .....             | 42 |
| ❖ [M_ST_NA_1].....              | 43 |
| ❖ [M_BO_NA_1 104] .....         | 43 |
| ❖ [M_ME_NA_1].....              | 44 |
| ❖ [M_ME_NB_1].....              | 44 |
| ❖ [M_ME_NC_1].....              | 45 |
| ❖ [M_IT_NA_1].....              | 45 |
| ❖ [C_SC_NA_1].....              | 46 |
| ❖ [C_DC_NA_1].....              | 46 |
| ❖ [C_RC_NA_1].....              | 47 |
| ❖ [C_BO_NA_1 104].....          | 47 |
| ❖ [C_SE_NA_1].....              | 48 |
| ❖ [C_SE_NB_1].....              | 48 |
| ❖ [C_SE_NC_1].....              | 49 |
| ❖ Group Codes.....              | 50 |

## 2.1 [IEC-870-5-101 Port 0]

This section provides information required to configure a slave application with the gateway.



### 2.1.1 Enabled

#### YES or NO

Enables or disables the primary port. If the port is not enabled (No), then the gateway will not use the port. If the port is enabled (Yes), the gateway will emulate an IEC 60870-101 slave device on the port.

### 2.1.2 Time DB Offset

#### -1, or 0 to 3994

This parameter defines the location in the database where the time maintained for the IEC protocol is copied. This time is updated whenever a time synchronization command is received from the host and continually as the program runs. If the parameter is set to -1, the time will not be placed in the database.



### **2.1.3 Disable Time Sync Events**

**YES or NO**

This feature can be used when the Master receives the event timestamp with only minutes and milliseconds information for each event message (CP24 time type). If the parameter is set to **N**, the spontaneous time sync events will be generated to indicate the change of hour. If this parameter is set to **Y**, then the spontaneous time sync event messages are not generated to indicate the change of hour.

### **2.1.4 Data Link Address Value**

**0 to 65535**

This parameter defines the Data Link Address for the emulated device on the gateway. This address identifies the gateway on the network along with the common address of ASDU.

### **2.1.5 Data link address length**

**0, 1, or 2**

This parameter specifies the number of octets used for the data link address. This parameter must be set the same for all devices on the network. A value of 0 is only valid when the balanced mode is used. If unbalanced mode is used, a value of 1 or 2 must be used.

### **2.1.6 Common Address of ASDU Val**

**0 to 65535**

This parameter specifies the common address of the ASDU (section address) for access to data in the gateway. There is only one value entered for access to all data in the gateway.

Refer to ASDU Configuration for more information on Application Service Data Unit configuration.

### **2.1.7 Common Address of ASDU Len**

**1 or 2**

This parameter specifies the number of octets used for the common address of ASDU. This parameter must be set the same for all devices on the network.

### **2.1.8 Inform. Object Address Len**

**1, 2, or 3**

This parameter specifies the number of octets used to define the address of an information object (point address).

### **2.1.9 Cyclic Data Transmission**

**0 to 4,294,967,296** (2 raised to the power of 32) milliseconds

This parameter defines the number of milliseconds between cyclic updates. The range of values allow for update times between 1 millisecond and 49.7 days. Set this parameter to 0 to disable cyclic data reporting.

### **2.1.10 Select/Operate Timeout**

**0 to 4,294,967,296** (2 raised to the power of 32) milliseconds

This parameter sets the number of milliseconds to wait for a valid Execute command after receiving a Select command. The range of values allows for update times between 1 millisecond and 49.7 days. Set this parameter to **0** to disable this feature.

### **2.1.11 Use ACTTERM with Set Point**

**Y - YES or N - NO**

This parameter determines if an ACTTERM will be sent. If the parameter is set to **YES**, then set point commands will issue an ACTTERM when the command is complete. If the parameter is set to **No**, ACTCON is the last response to a set point command.

### **2.1.12 Use ACTTERM with Step**

**Y - YES or N - NO**

This parameter determines if an ACTTERM will be sent. If the parameter is set to **YES**, then step commands will issue an ACTTERM when the command is complete. If the parameter is set to **No**, ACTCON is the last response to a step command.

### **2.1.13 Single char ACK F0, 1 or 3**

**Y - YES or N - NO**

If set to Yes, a single character ACK (0xE5) will be sent instead of a fixed length ACK (secondary function code 0) in response to a primary link function code 0, 1 or 3 if there is no access demand for class 1 data (ACD=1). If set to No, the fixed length ACK will be sent.

### **2.1.14 Single char ACK C1 or C2**

**Y -YES or N - NO**

If set to Yes, a single character ACK (0xE5) will be sent instead of a fixed length NACK (secondary function code 9) when no response user data is available. If set to No, the fixed length NACK will be sent.

### **2.1.15 Maximum ASDU Resp Len**

**25 to 252**

Sets the maximum ASDU response message length (usually 252).

### **2.1.16 Cause of Trans(mission) Octets**

**1 or 2**

Specifies the number of COT octets (1 or 2)

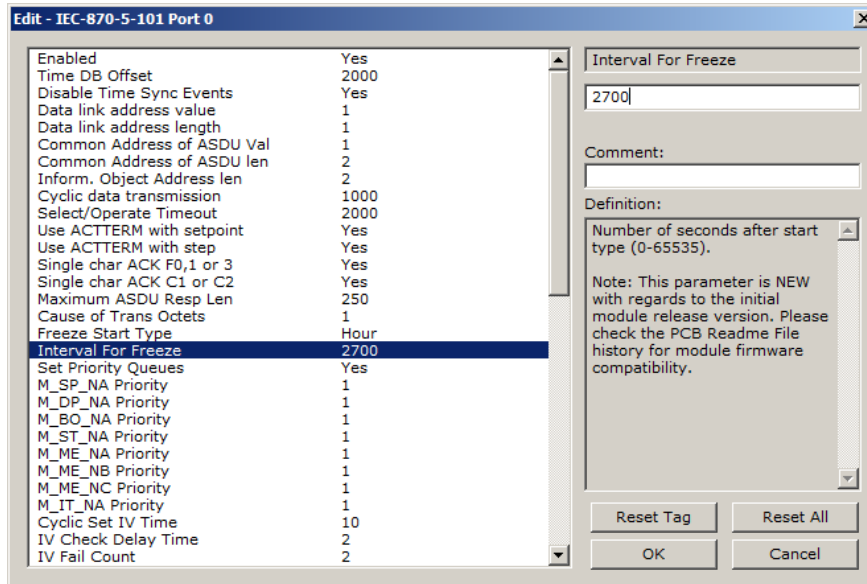
### **2.1.17 Freeze Start Type**

**D = DAY, H = HOUR, M = MINUTE, N = NOT USED**

The *Freeze Start Type* parameter defines when the gateway starts sending the M\_IT messages.

**Example I - Freeze Start Type**

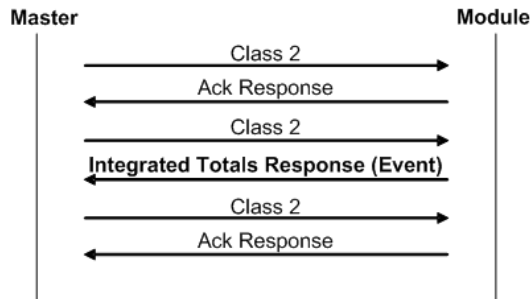
If the gateway should send the counter points on the hourly turn around time and also 45 minutes later, the Mode A parameters should be configured as follows:



So the gateway would send events as follows (Hours:Minutes:Seconds):

- 17:00:00
- 17:45:00
- 18:00:00
- 18:45:00
- 19:00:00
- 19:45:00
- ...

The following illustration shows a typical communication example when the Mode A is selected during unbalanced mode operation:

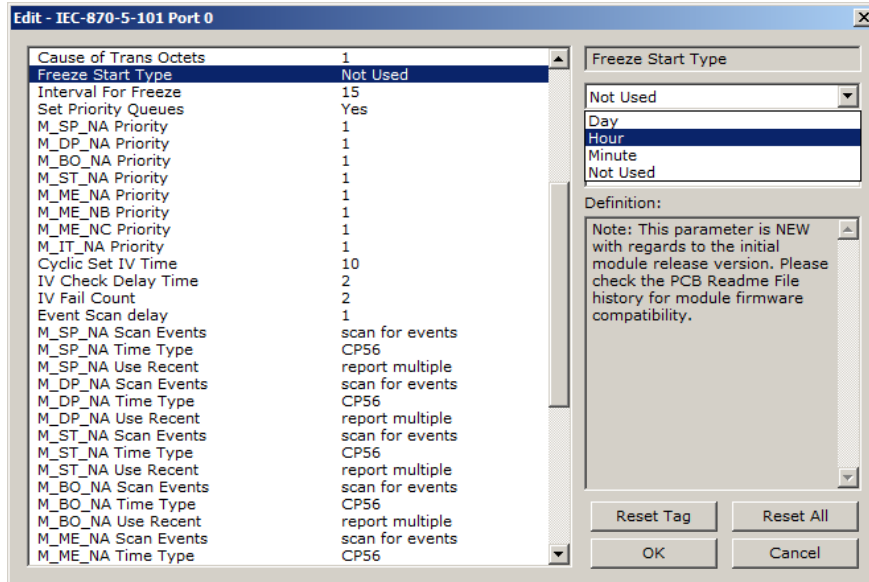


### Example II - Freeze Start Type

If the gateway powers up with the following date and time settings on its internal clock:

03/25/2004 18:07:42

And if you configure the *Interval For Freeze* parameter to 15 seconds, as follows:



The Freeze Start Type parameter determines when the gateway will begin sending these messages, as follows:

| If the Freeze Start Type is set to: | Then Messages will start being sent at:             |
|-------------------------------------|---|
| D                                   | 03/26/2004 00:00:00 and every 15 seconds thereafter |
| H                                   | 03/25/2004 19:00:00 and every 15 seconds thereafter |
| M                                   | 03/25/2004 18:08:00 and every 15 seconds thereafter |

Once message transmission begins, the gateway will freeze and transmit counter values at whatever interval has been configured using the *Interval For Freeze* parameter. In this example, that would be every 15-seconds.

### 2.1.18 Interval for Freeze

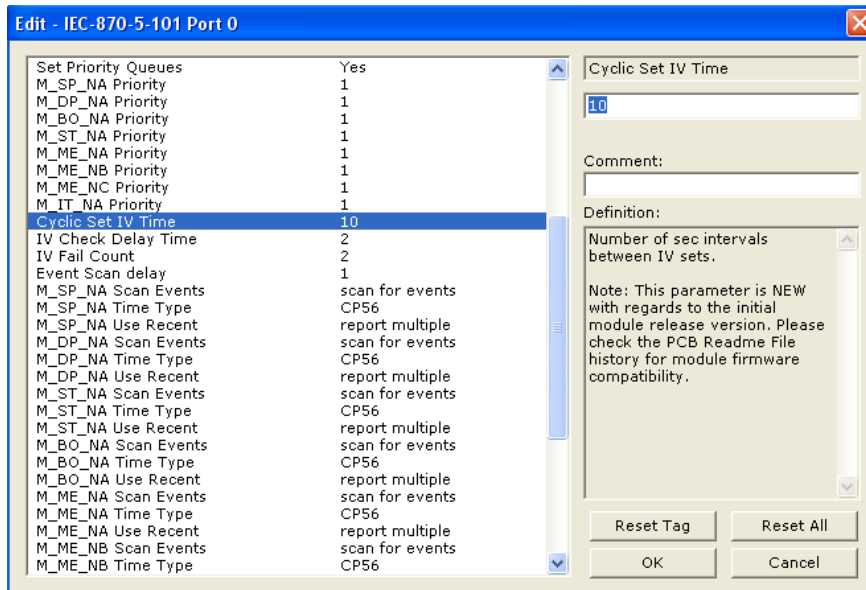
0 to 65535 seconds

*Freeze Start Type* and *Interval for Freeze* are used if Mode A operation is to be used for the counter freeze operation. If they are not used, the gateway will operate in Mode D.

### 2.1.19 Set Priority Queues

#### YES or NO

This section defines priority queues for the gateway. You can assign priorities to data types that return events so that events of some data types will be returned before events of other data types. If this feature is utilized, each data type must be assigned a unique index from 0 to 7. The lower the index, the higher the priority (0=highest priority, 7=lowest priority).



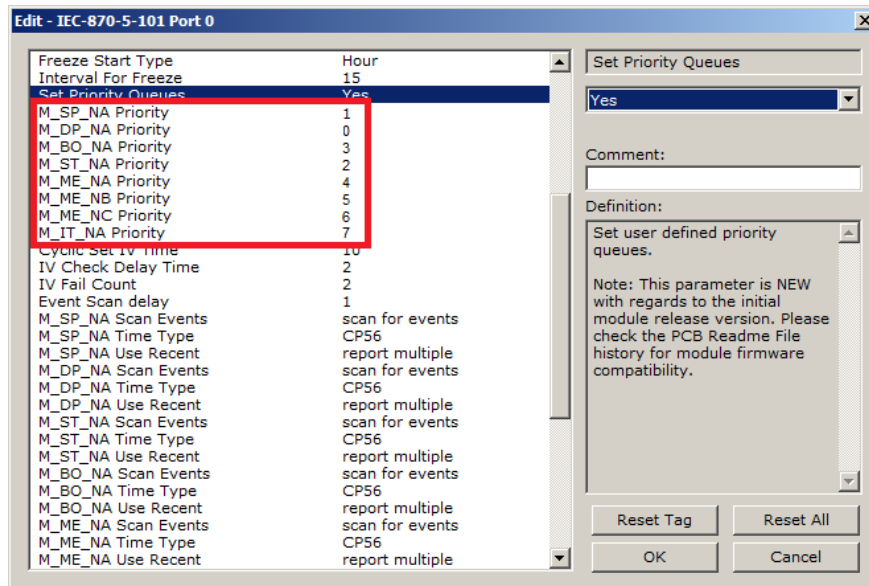
Events of ASDUs with lower numbers will be reported before events from other ASDUs with higher numbers.

**Note:** In some configurations, depending on the total number of events generated for all data types, setting priorities may cause some events to be lost, as the event buffers for low priority queues may overflow before their events can be reported.

For more information, refer to Event Priority (page 31).

### 2.1.20 Event Priority

Event Priority permits the user to set reporting priorities for data change events generated for each data type. The configuration file contains the following parameters to support this feature:



The *Set Priority Queues* parameter must be enabled for this feature to be used. Each of the Monitored Point ASDUs must be assigned a unique priority index from 0 to 7. When events of the ASDU with a lower priority number are present, they will always be reported before events from any other ASDUs with higher priority numbers.

### 2.1.21 Cyclic Set IV Time - PLX 101S

1 to **65535** seconds, or **0** to disable Invalid Bit Monitoring

The parameter should be set to a value significantly greater than the value of the *IV Check Delay Time* parameter, multiplied by the value of the *IV Fail Count* parameter. Example:

If, *IV Check Delay* = **3**;

And, *IV Fail Count* = **5**;

Then set *Cyclic Set IV Time* significantly greater than 15; preferably some multiple of this value, like **30, 45, 60**, or more.

The *Cyclic Set IV Time* parameter is one of three parameters needed to establish a fail-safe data validation system for this protocol implementation. This parameter is used with the *IV Check Delay Time* and the *IV Fail Count* parameters to create a way to alert the remote IEC-101 Master to situations where data being sent by the gateway might be invalid.

The gateway will automatically set ON (set to one) the IV Bits of all IEC database Monitor Points (M\_xx\_xx point) that have been configured to use invalid bit monitoring. This means that all IEC database Monitor Points configured with IV Bit addresses greater than zero (0) will have their IV Bit addresses in the gateway database set to a value of one (1) by the gateway firmware in a timed, periodic cycle, based on the number of seconds entered in *Cyclic Set IV Time*.

The IV Bits are checked at the interval specified by the *IV Check Delay Time* parameter. At the end of each check delay interval, if an IV Bit for any IEC database point is ON, an IV Bit Check Fail accumulator for that point will be incremented. If the value in any IV Bit Check Fail accumulator becomes equal to the value of the *IV Fail Count* parameter, the gateway will consider the data value of that point invalid and will report the invalid status of that point to the Master along with the data value.

If the IV Bit is cleared (reset, set OFF, set to zero) before the IV Bit Check Fail accumulator value becomes equal to the value of the *IV Fail Count* parameter, the IV Bit Check Fail accumulator will be reset to 0. The gateway will consider the data value of that point valid and will report the valid status of that point to the Master along with data value.

To properly use data validity checking with a ProLinx Gateway, the application that communicates to the gateway through the other gateway protocol (Modbus, EtherNet/IP, etc.) must be programmed to provide data validity information, as well as the actual data to be transferred. This other application must write bit values of zero (0) or one (1) to the proper gateway database addresses along with the data for each IEC database point.

This means the other protocol application will be responsible for clearing the IV Bits for data points which contain valid data at a time interval less than the amount of time required to accumulate the required number of IV Bit Check failures. If the other protocol application does not clear the IV Bits, then those IEC database monitor point values will be reported to the Master as invalid.

**NOTE:** If one or more of the *Cyclic Set IV Time*, the *IV Check Delay Time*, or the *IV Fail Count* parameters are set to zero, the data validity checking feature will be disabled for all points in the IEC database and the data reported to the remote Master will always be reported as valid data. If you wish to have actual data validity information relayed to the Master, be sure the *Cyclic Set IV Time*, the *IV Check Delay Time*, and the *IV Fail Count* parameters are set to non-zero values and be sure the application has been programmed to properly manipulate the IV Bit addresses in the gateway database. Once data validity checking has been enabled, you can still turn off validity checking on a point-by-point basis by setting the IV Bit Address to zero (0) for any point you wish to exclude from data validity checking and reporting.

For more information on the data validity features of this protocol implementation, refer to Invalid Bit Monitoring.



### **2.1.22 IV Check Delay Time**

**1** to **65535** seconds, or **0** to disable Invalid Bit Monitoring

This parameter sets the number of seconds between IV Bit value checks. Every IEC database Monitor information object which has an IV Bit address set greater than 0 will have that bit address checked at the interval specified by this parameter.

Setting this parameter to zero (**0**) will disable data validity checking.

### **2.1.23 IV Fail Count**

**1** to **65535** seconds, or **0** to disable Invalid Bit Monitoring

This parameter sets the number of successive IV Bit check failures which must occur before the data from an IEC database Monitor information object will be reported to the remote Client as invalid data.

An IV Bit Check Failure occurs when the IV Bit value in the gateway database is set ON {when it contains a value of one (1)} at the time an IV Bit Check is performed. IV Bit Check failures are counted and held in separate IV Bit Check Failure accumulators for each IEC Monitor information object configured for validity checking. If the value in any point's failure accumulator becomes equal to the value set in this *IV Fail Count* parameter, the gateway will consider data from this point as invalid and report to the Client this invalid status, along with the point's data value.

Setting this parameter to zero (**0**) will disable data validity checking.

### **2.1.24 Event Scan Delay**

**1** to **65535** milliseconds, or **0** to disable

If set to **0**, the feature will be disabled and the gateway will not generate any events. If set from **1** to **65535**, the parameter represents the number of milliseconds between event scanning. This parameter defines how often the program will scan for new events in the databases.

**NOTE:** The lower you set this value, the more negative effect it will have on overall gateway data update and transfer performance. Scanning for new events causes an interrupt of other data manipulation functions. The more often these scan interrupts occur, the less processor time the gateway will have available for other data operations. When choosing a value for this parameter, you should consider such factors as how often events might be generated by your application and how often you need to recognize and respond to such events. This will allow you to set this value as high as it can be set and still maintain your overall event capturing and reporting strategy. Doing so can reduce the amount of interrupts, prevent excessive event scanning, and maximize data update and transmission rates.

### 2.1.25 Scan Events

[0] = No SCANNING or [1] = SCAN FOR EVENTS

Separate parameters exist for each monitored point type and define whether or not events of that particular point type will be generated by the gateway. If the *Scan Events* parameter is set to zero [0] or **No SCANNING**, then events will not be generated. If this parameter is set to [1] or **SCAN FOR EVENTS**, events will be scanned and generated on data change.

```
→ M_SP_NA Scan Events           : scan for events
   M_SP_NA Time Type           : CP56
   M_SP_NA Use Recent          : report multiple
→ M_DP_NA Scan Events           : scan for events
   M_DP_NA Time Type           : CP56
   M_DP_NA Use Recent          : report multiple
→ M_ST_NA Scan Events           : scan for events
   M_ST_NA Time Type           : CP56
   M_ST_NA Use Recent          : report multiple
→ M_BO_NA Scan Events           : scan for events
   M_BO_NA Time Type           : CP56
   M_BO_NA Use Recent          : report multiple
→ M_ME_NA Scan Events           : scan for events
   M_ME_NA Time Type           : CP56
   M_ME_NA Use Recent          : report multiple
→ M_ME_NB Scan Events           : scan for events
   M_ME_NB Time Type           : CP56
   M_ME_NB Use Recent          : report multiple
→ M_ME_NC Scan Events           : scan for events
   M_ME_NC Time Type           : CP56
   M_ME_NC Use Recent          : report multiple
   M_IT_NA Time Type           : CP56
```

### 2.1.26 Time Type

[0] = None, [1] = CP24, or [2] = CP56

Separate parameters exist for multiple point types. This parameter defines the time format used with data events.

```
→ M_SP_NA Scan Events : scan for events
→ M_SP_NA Time Type : CP56
M_SP_NA Use Recent : report multiple
M_DP_NA Scan Events : scan for events
→ M_DP_NA Time Type : CP56
M_DP_NA Use Recent : report multiple
M_ST_NA Scan Events : scan for events
→ M_ST_NA Time Type : CP56
M_ST_NA Use Recent : report multiple
M_BO_NA Scan Events : scan for events
→ M_BO_NA Time Type : CP56
M_BO_NA Use Recent : report multiple
M_ME_NA Scan Events : scan for events
→ M_ME_NA Time Type : CP56
M_ME_NA Use Recent : report multiple
M_ME_NB Scan Events : scan for events
→ M_ME_NB Time Type : CP56
M_ME_NB Use Recent : report multiple
M_ME_NC Scan Events : scan for events
→ M_ME_NC Time Type : CP56
M_ME_NC Use Recent : report multiple
→ M_IT_NA Time Type : CP56
```

### 2.1.27 Use Balanced Mode

Y - YES or N - NO

This parameter specifies if the port will use balanced mode. If balanced mode is used, only one controlled station will be permitted on the port. If unbalanced mode is used, multiple controlled stations can be used on a port.

Refer to gateway Initialization for more information on these modes.

### 2.1.28 Retry Count

0 to 255

In balanced mode, this parameter specifies the number of retries (0 to 255) if a response is not received. In unbalanced mode, this parameter is ignored.

### **2.1.29 Response Timeout**

**0 to 65535** milliseconds

This parameter specifies the minimum number of milliseconds to wait for a response to a primary message. Do not set this parameter too small or timeout conditions may prevent successful data transmission. If the timeout is recognized, the message will be retransmitted up to the number of times specified in the Retry Count parameter. This parameter is only used in balance mode.

### **2.1.30 Baud Rate**

**300, 600, 1200, 2400, 4800, 9600, 19200, or 38400**

This parameter specifies the baud rate for the primary port on the gateway. Baud rates from 300 to 38400 are supported on the gateway.

### **2.1.31 Parity**

*None, Odd, Even*

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

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

### **2.1.32 RTS On**

**0 to 65535** milliseconds

This parameter sets the number of milliseconds to delay after *Ready To Send* (RTS) is asserted before data will be transmitted.

### **2.1.33 RTS Off**

**0 to 65535** milliseconds

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

### **2.1.34 Minimum Delay**

**0 to 65535** milliseconds

This parameter defines the minimum number of milliseconds to wait before a response is sent from the unit.

### **2.1.35 Receive Timeout**

**0** to **65535** milliseconds

This parameter specifies the minimum number of milliseconds to wait after the first byte of a frame is received before a timeout condition is set. Be careful not to set this parameter too small. If the timeout condition is set, all bytes in the frame received will be discarded.

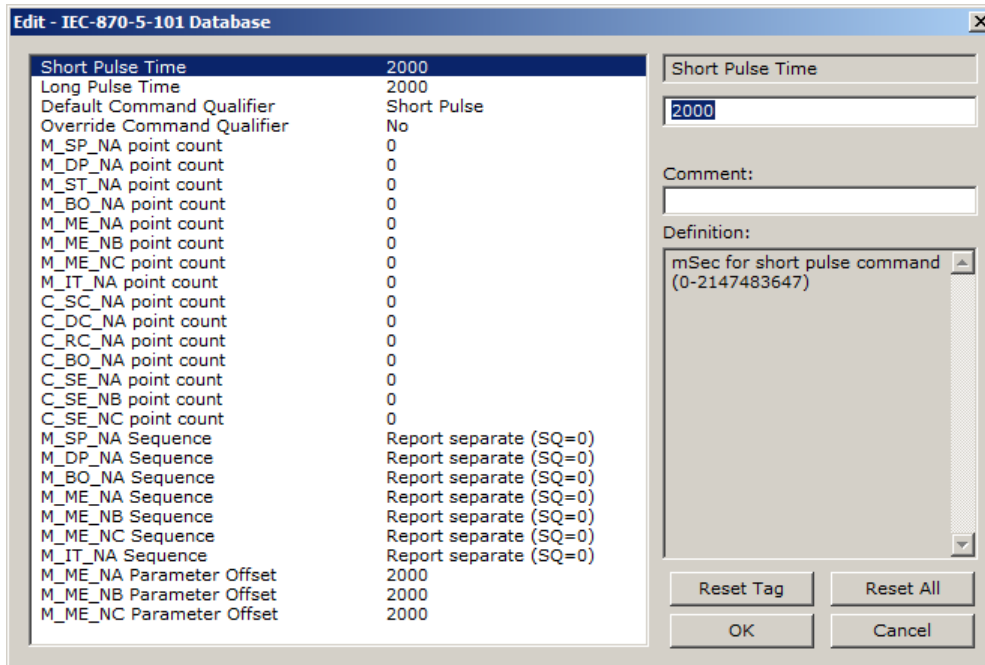
### **2.1.36 Hardware Handshaking**

**NONE, RTS/CTS, DTR/DSR, MODEM**

Selects the hardware handshaking type. **RTS/CTS**, **DTR/DSR** and **MODEM** options are most commonly used for older telephone line modems. **RTS/CTS** and **DTR/DSR** are also commonly used with leased-line data modems. The **MODEM** option is also commonly used with dial-up modems. When the messages must go out if the modem is not connected, the gateway will assert the DTR line which will cause the modem to dial the number in the modem's configuration. The gateway does not have the functionality to dial the number to be called for the modem. The gateway reconnects after 30 seconds of inactivity.

## 2.2 [IEC-870-5-101 Database]

This section configures the IEC-870-5-101 Database.



### 2.2.1 Short Pulse Time

0 to 2,147,483,647 (2 raised to the power of 31, minus 1) milliseconds

This parameter defines the number of milliseconds to be associated with a short pulse command.

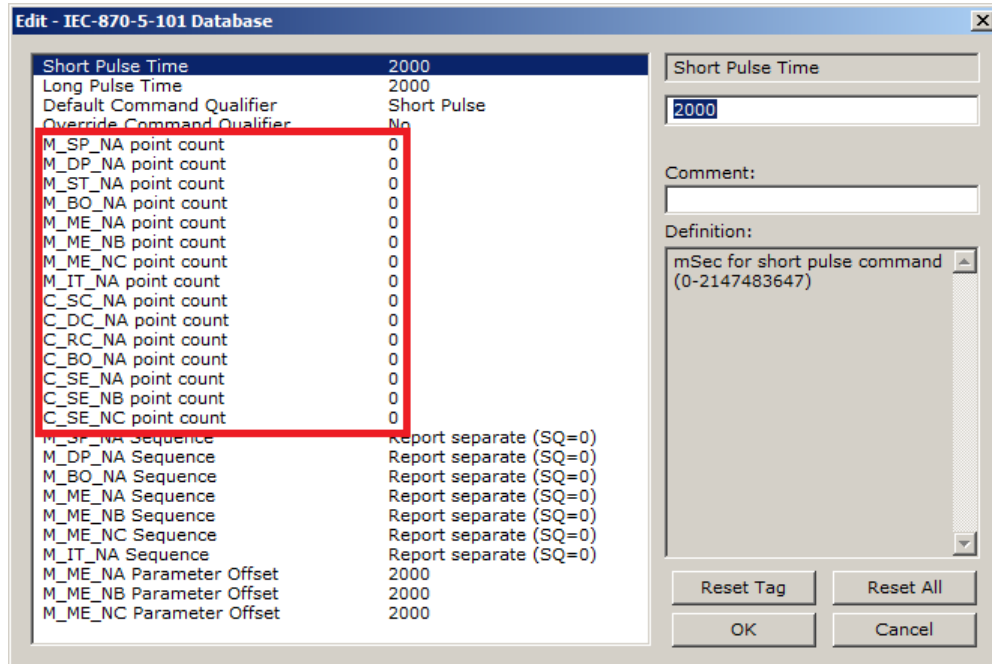
### 2.2.2 Long Pulse Time

0 to 2,147,483,647 (2 raised to the power of 31, minus 1) milliseconds

This parameter defines the number of milliseconds to be associated with a long pulse command.

### 2.2.3 Point Count

Point Count configuration ranges in the following configuration items are based on the assumption that you will be using only one of the available data types for your application. The number of point counts you configure will have an effect on gateway performance, in particular, the accuracy of the gateway's internal clock.



| Parameter           | Description   | Range     |
|---------------------|---|-----------|
| M_SP_NA point count | The number of point values assigned in monitored single-point database.             | 0 to 1000 |
| M_DP_NA point count | The number of point values assigned in monitored double-point database.             | 0 to 1000 |
| M_ST_NA point count | The number of point values assigned in monitored step-point database.               | 0 to 1000 |
| M_BO_NA point count | The number of 32-bit values assigned in monitored bitstring database                | 0 to 1000 |
| M_ME_NA point count | The number of point values assigned in monitored normalized-point database.         | 0 to 1000 |
| M_ME_NB point count | The number of point values assigned in monitored scaled-point database.             | 0 to 1000 |
| M_ME_NC point count | The number of point values assigned in monitored scaled short-float point database. | 0 to 50   |
| M_IT_NA point count | The number of point values assigned in monitored counter-point database.            | 0 to 99   |
| C_SC_NA point count | The number of point values assigned in command single-point database.               | 0 to 1000 |
| C_DC_NA point count | The number of point values assigned in command dual-point database.                 | 0 to 1000 |

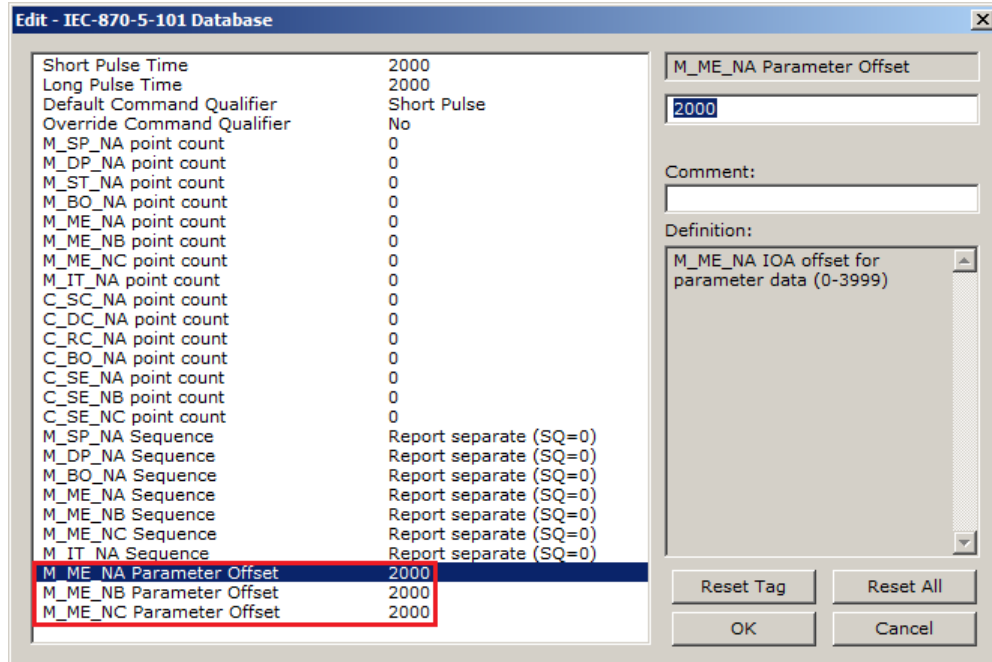
| Parameter           | Description  | Range     |
|---------------------|--|-----------|
| C_RC_NA point count | The number of point values assigned in command step-point database.        | 0 to 1000 |
| C_BO_NA point count | The number of 32-bit values assigned in command bitstring database         | 0 to 1000 |
| C_SE_NA point count | The number of point values assigned in command normalized-point database.  | 0 to 1000 |
| C_SE_NB point count | The number of point values assigned in command scaled-point database.      | 0 to 1000 |
| C_SE_NC point count | The number of point values assigned in command short-float point database. | 0 to 50   |

### 2.2.4 Sequence Flag

In order to save bandwidth, you can configure the gateway to use the Sequence Flag feature. If this feature is not selected, the gateway will send the Information Object Address (IOA) and the data value for every Monitor information object in a poll response to the Master. If this parameter is enabled, the gateway will turn on the Sequence Flag bit in a Monitor response; send the Information Object Address of only the first object in the response, along with its data value; and send only the data values for all the other information objects in the response. The Master assumes that all other points use IOAs in a contiguous order (using the first point as the starting reference address). Since the IOAs are omitted for all but the first point, this feature allows each message packet to contain more process data, which improves network throughput and optimizes bandwidth utilization.



## 2.2.5 Parameter Offset



This parameter specifies the IOA (Information Object Address) offset to the parameter data for the normalized parameter data. The value entered is added to the Information Object address for the associated point to compute the parameter IOA address.

The Master may send a "Parameter of Measured Normalized" or "Parameter of Measured Scaled" command using the parameter IOA in order to change the deadband values for specific points.

**Note:** The Low Limit and High Limit values are always calculated based on the deadband value as described in the following table.

| Point     | Value   |
|-----------|---|
| Threshold | Determined by the deadband set in the configuration file or altered by the write command. |
| Low       | Last reported event value - threshold.  |
| High      | Last reported event value + threshold.  |

For example, if the configuration sets two M\_ME\_NA points with IOA (Point #) of 600 and 601 and an M\_ME\_NA Parameter Offset value of 3000, it would result in the parameter points shown in the table.

### 2.3 [M\_SP\_NA\_1]

This section defines the monitored single-point database. This information is sourced from the database and is transferred to the remote Master unit.

This section takes the following parameters:

- **POINT #**
- **DB ADDRESS**
- **GROUPS** (page 50)
- **IV DB BIT**

Each point consumes one bit (1 = On, 0 = Off state). The **DB ADDRESS VALUE** corresponds to the bit offset address in the database.

The **DB ADDRESS** should be located in a database area that is being constantly moved from the gateway to the Master.

For additional information on how to set these parameters, see the *Reference* chapter of this manual.

### 2.4 [M\_DP\_NA\_1]

This section defines the monitored dual-point database. This information is sourced from the database and is transferred to the remote Master unit.

This section takes the following parameters:

- **POINT #**
- **DB ADDRESS**
- **GROUPS** (page 50)
- **IV DB BIT**

Each point consumes two bits (00 = intermediate, 01 = off, 10 = on and 11 = intermediate). The **DB ADDRESS** value corresponds to the bit offset address in the database.

The **DB ADDRESS** should be located in a database area that is being constantly moved from the gateway to the Master.

For additional information on how to set these parameters, see the *Reference* chapter of this manual.

## 2.5 [M\_ST\_NA\_1]

This section defines the monitored step database. This information is sourced from the database and is transferred to the remote Master unit.

This section takes the following parameters:

- **POINT #**
- **DB ADDRESS**
- **GROUPS** (page 50)
- **IV DB BIT**

Each point consumes one, 8-bit byte. The **DB ADDRESS** value corresponds to the byte offset address in the database.

The **DB ADDRESS** should be located in a database area that is being constantly moved from the gateway to the Master.

For additional information on how to set these parameters, see the *Reference* chapter of this manual.

## 2.6 [M\_BO\_NA\_1 104]

This section defines the *Monitored 32-bit Bitstring Data Point* database. This information is sourced from the database and is transferred to the remote Master unit.

This section takes the following parameters:

- Point #
- DB Address
- Groups (page 50)
- IV DB Bit

Each point is four 8-bit bytes (two 16-bit words) and the *DB Address* value corresponds to the double word offset in the gateway memory database.

The *DB address* should be located in a database area that is being constantly moved from the gateway to the Master.

For additional information on how to set these parameters, see the *Reference* chapter of this manual.

## 2.7 [M\_ME\_NA\_1]

This section defines the monitored measured value, normalized database. This information is sourced from the database and is transferred to the remote Master unit.

To determine the IOA (Information Object Address) for each object, add the **POINT #** in the following section to the value of the M\_ME\_NA parameter offset parameter set in the previous section.

This section takes the following parameters:

- **POINT #**
- **DB ADDRESS**
- **GROUPS** (page 50)
- **DEFAULT DEADBAND**
- **IV DB BIT**

Each point consumes one, 16-bit word. The **DB ADDRESS** value corresponds to the word offset address in the database.

The **DB ADDRESS** should be located in a database area that is being constantly moved from the gateway to the Master.

For additional information on how to set these parameters, see the *Reference* chapter of this manual.

## 2.8 [M\_ME\_NB\_1]

This section defines the monitored measured value, scaled database. This information is sourced from the database and is transferred to the remote Master unit.

To determine the IOA (Information Object Address) for each object, add the **POINT #** in the following section to the value of the M\_ME\_NB parameter offset parameter set in the previous section.

This section takes the following parameters:

- **POINT #**
- **DB ADDRESS**
- **GROUPS** (page 50)
- **DEFAULT DEADBAND**
- **IV DB BIT**

Each point consumes one, 16-bit word. The **DB ADDRESS** value corresponds to the word offset address in the database.

The **DB ADDRESS** should be located in a database area that is being constantly moved from the gateway to the Master.

For additional information on how to set these parameters, see the *Reference* chapter of this manual.

## 2.9 [M\_ME\_NC\_1]

This section defines the monitored short-float point database. This information is sourced from the database and is transferred to the remote Master unit.

To determine the IOA (Information Object Address) for each object, add the **POINT #** in the following section to the value of the M\_ME\_NC Parameter Offset parameter set in the previous section.

This section takes the following parameters:

- **POINT #**
- **DB ADDRESS**
- **GROUPS** (page 50)
- **DEFAULT DEADBAND**
- **IV DB BIT**

Each point consumes four 8-bit bytes (two 16-bit words). The **DB ADDRESS** value corresponds to the double-word offset address in the database.

The **DB ADDRESS** should be located in a database area that is being constantly moved from the gateway to the Master.

For additional information on how to set these parameters, see the *Reference* chapter of this manual.

## 2.10 [M\_IT\_NA\_1]

This section defines the monitored integrated totals (counter) database. This information is sourced from the database and is transferred to the remote Master unit. This section takes the following parameters:

- **POINT #**
- **DB ADDRESS**
- **GROUPS** (page 50)
- **IV DB BIT**

Each point consumes four 8-bit bytes (two 16-bit words). The **DB ADDRESS** value corresponds to the double-word offset address in the database.

The **DB ADDRESS** should be located in a database area that is being constantly moved from the gateway to the Master.

For additional information on how to set these parameters, see the *Reference* chapter of this manual.

## 2.11 [C\_SC\_NA\_1]

This section defines the single point command database. This information is sourced from the remote Master and is transferred to the database.

You can associate a command with a monitored single-point database value to coordinate the command/monitor operation. You must enter the correct **MONITOR POINT #** and **MONITOR DB ADDRESS** values in the table. If the **REQUIRE SELECT** parameter is not set to zero, a select command must be received before an execute command will be processed.

This section takes the following parameters:

- **POINT #**
- **DB ADDRESS**
- **MONITOR POINT #**
- **MONITOR DB ADDRESS**
- **REQUIRE SELECT**

Each point consumes one bit. The **DB ADDRESS** value corresponds to the bit offset in the database.

The **DB ADDRESS** should be located in a database area that is being constantly moved from the Master to the gateway.

For additional information on how to set these parameters, see the *Reference* chapter of this manual.

## 2.12 [C\_DC\_NA\_1]

This section defines the double point command database. This information is sourced from the remote Master and is transferred to the database.

You can associate a command with a monitored double point database value to coordinate the command/monitor operation. You must enter the correct **MONITOR POINT #** and **MONITOR DB ADDRESS** values in the table. If the **REQUIRE SELECT** parameter is not set to zero, a select command must be received before an execute command will be processed.

This section takes the following parameters:

- **POINT #**
- **DB ADDRESS**
- **MONITOR POINT #**
- **MONITOR DB ADDRESS**
- **REQUIRE SELECT**

Each point consumes two bits. The **DB ADDRESS** value corresponds to the bit offset in the database.

The **DB ADDRESS** should be located in a database area that is being constantly moved from the Master to the gateway.

For additional information on how to set these parameters, see the *Reference* chapter of this manual.

### 2.13 [C\_RC\_NA\_1]

This section defines the step command database. This information is sourced from the remote Master and is transferred to the database.

You can associate a command with a monitored single-point database value to coordinate the command/monitor operation. You must enter the correct **MONITOR POINT #** and **MONITOR DB ADDRESS** values in the table. If the **REQUIRE SELECT** parameter is not set to zero, a select command must be received before an execute command will be processed.

This section takes the following parameters:

- **POINT #**
- **DB ADDRESS**
- **MONITOR POINT #**
- **MONITOR DB ADDRESS**

Each point consumes one byte. **THE DB ADDRESS** value corresponds to the byte offset in the database.

The **DB ADDRESS** should be located in a database area that is being constantly moved from the Master to the gateway.

For additional information on how to set these parameters, see the *Reference* chapter of this manual.

### 2.14 [C\_BO\_NA\_1 104]

This section defines the *Command 32-bit Bitstring Point* database. This information is sourced from the remote Master and is transferred to the database.

You can associate a command with a *Monitored Double-Point* database value to coordinate the command/monitor operation. You must enter the correct *Monitor Point #* and *Monitor DB Address* values in the table. If the *Require Select* parameter is not set to zero, a select command must be received before an execute command will be processed.

This section takes the following parameters:

- Point #
- DB Address
- Monitor Point #
- Monitor DB Address

Each point is four 8-bit bytes and the *DB Address* value corresponds to the double word offset in the gateway memory database.

The *DB address* should be located in a database area that is being constantly moved from the Master to the gateway.

For additional information on how to set these parameters, see the *Reference* chapter of this manual.

## 2.15 [C\_SE\_NA\_1]

This section defines the normalized setpoint database. This information is sourced from the remote Master and is transferred to the database.

You can associate a command with a monitored single-point database value to coordinate the command/monitor operation. You must enter the correct **MONITOR POINT #** and **MONITOR DB ADDRESS** values in the table. If the **REQUIRE SELECT** parameter is not set to zero, a select command must be received before an execute command will be processed.

This section takes the following parameters:

- **POINT #**
- **DB ADDRESS**
- **MONITOR POINT #**
- **MONITOR DB ADDRESS**
- **REQUIRE SELECT**

Each point consumes one word. The **DB ADDRESS** value corresponds to the word offset in the database.

The **DB ADDRESS** should be located in a database area that is being constantly moved from the Master to the gateway.

For additional information on how to set these parameters, see the *Reference* chapter of this manual.

## 2.16 [C\_SE\_NB\_1]

This section defines the scaled setpoint database. This information is sourced from the remote Master and is transferred to the database.

You can associate a command with a monitored single-point database value to coordinate the command/monitor operation. You must enter the correct **MONITOR POINT #** and **MONITOR DB ADDRESS** values in the table. If the **REQUIRE SELECT** parameter is not set to zero, a select command must be received before an execute command will be processed.

This section takes the following parameters:

- **POINT #**
- **DB ADDRESS**
- **MONITOR POINT #**
- **MONITOR DB ADDRESS**
- **REQUIRE SELECT**

Each point consumes one word. The **DB ADDRESS** value corresponds to the word offset in the database.

The **DB ADDRESS** should be located in a database area that is being constantly moved from the Master to the gateway.

For additional information on how to set these parameters, see the *Reference* chapter of this manual.



## 2.17 [C\_SE\_NC\_1]

This section defines the short-float setpoint database. This information is sourced from the remote Master and is transferred to the database.

You can associate a command with a monitored single-point database value to coordinate the command/monitor operation. You must enter the correct **MONITOR POINT #** and **MONITOR DB ADDRESS** values in the table. If the **REQUIRE SELECT** parameter is not set to zero, a select command must be received before an execute command will be processed.

This section takes the following parameters:

- **POINT #**
- **DB ADDRESS**
- **MONITOR POINT #**
- **MONITOR DB ADDRESS**
- **REQUIRE SELECT**

Each point consumes two words. The **DB ADDRESS** value corresponds to the double-word offset in the database.

The **DB ADDRESS** should be located in a database area that is being constantly moved from the Master to the gateway.

For additional information on how to set these parameters, see the *Reference* chapter of this manual.

## 2.18 Group Codes

Note that the highest bit assigns the point to the cyclic data group. This data will be cyclically reported to the Master at the frequency set in the configuration value "Cyclic Data Transmission" Parameter.

| Group Code | Description   |
|------------|---|
| 0x00000001 | Interrogated by general interrogation (station or global) |
| 0x00000002 | Interrogated by group 1 interrogation                     |
| 0x00000004 | Interrogated by group 2 interrogation                     |
| 0x00000008 | Interrogated by group 3 interrogation                     |
| 0x00000010 | Interrogated by group 4 interrogation                     |
| 0x00000020 | Interrogated by group 5 interrogation                     |
| 0x00000040 | Interrogated by group 6 interrogation                     |
| 0x00000080 | Interrogated by group 7 interrogation                     |
| 0x00000100 | Interrogated by group 8 interrogation                     |
| 0x00000200 | Interrogated by group 9 interrogation                     |
| 0x00000400 | Interrogated by group 10 interrogation                    |
| 0x00000800 | Interrogated by group 11 interrogation                    |
| 0x00001000 | Interrogated by group 12 interrogation                    |
| 0x00002000 | Interrogated by group 13 interrogation                    |
| 0x00004000 | Interrogated by group 14 interrogation                    |
| 0x00008000 | Interrogated by group 15 interrogation                    |
| 0x00010000 | Interrogated by group 16 interrogation                    |
| 0x00020000 | Interrogated by general counter request                   |
| 0x00040000 | Interrogated by group 1 counter request                   |
| 0x00080000 | Interrogated by group 2 counter request                   |
| 0x00100000 | Interrogated by group 3 counter request                   |
| 0x00200000 | Interrogated by group 4 counter request                   |
| 0x40000000 | Disable event scanning of this point                      |
| 0x80000000 | Periodic/cyclic data returned from unit                   |

If the highest bit (bit 31) is set, data will be produced by the driver for the specified point at the rate set for periodic data generation. Bit 30 (0x40000000) enables scanning of this point for event generation. If the bit is clear and the data type is set for scanning, events will be generated for the point. If the bit is set, events will not be generated for the point. This feature can be used to select which points will generate events for the controlling station and can get rid of event data that is not important to the application

## 3 Gateway Communication Ports

### *In This Chapter*

- ❖ Serial Port Cable Connections: Config/Debug and Port 0 ..... 51
- ❖ RS-232 Configuration/Debug Port..... 52
- ❖ Application Serial Port(s)..... 53

### 3.1 Serial Port Cable Connections: Config/Debug and Port 0

This section contains information on the cable and pin assignments for the ProLinx Communication Gateway serial ports (RS-232/422/485).

The gateway will come with one to five serial ports, depending on the protocol combination and hardware configuration purchased. In all cases, the protocol serial ports will have the same pin-outs.

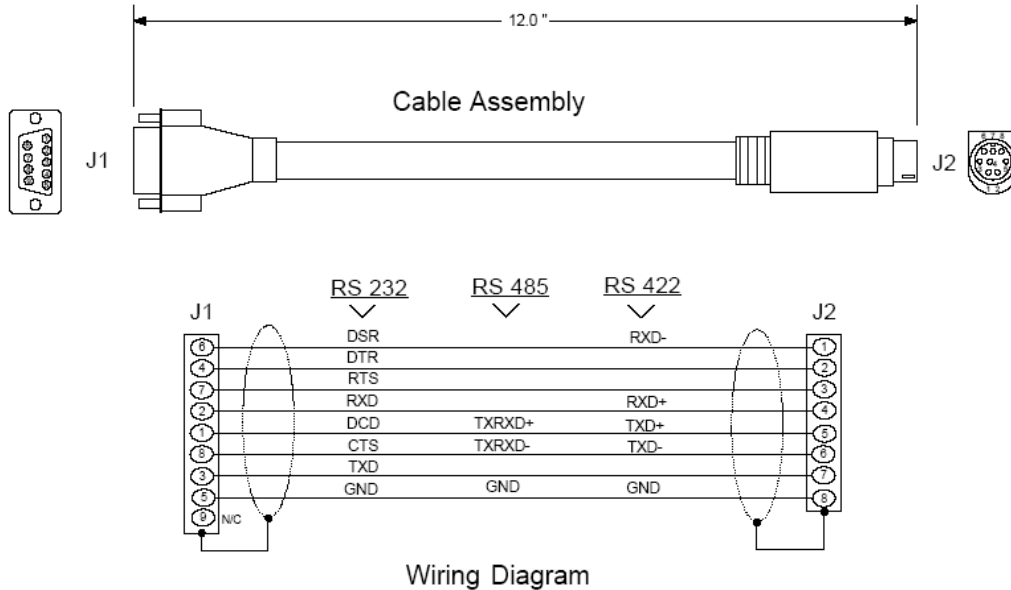
Each serial port uses an eight-pin, Mini-DIN8F (female) connector. A 6-inch Mini-DIN8M (male) to DB-9M (male) adapter cable is provided for each active serial port. The DB-9M provides connections for RS-232, RS-422 and RS-485 serial interfaces. The diagrams in the following topics detail the pin assignments for several possible physical connections.

The following table describes the relationship between the port labeling on the front of the ProLinx gateway and the application.

| Port Label | Function   |
|------------|--|
| Debug      | Debug/Configuration Port (Operates in RS-232 Mode Only)        |
| Port 0     | Application Port 0 (RS-232, RS-422, or RS-485 Modes Available) |

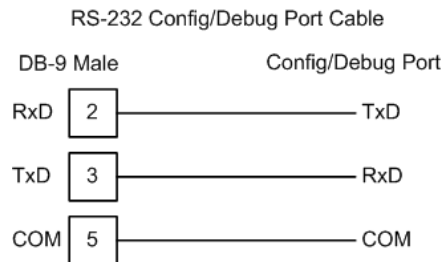
Please check and install the gateway's Port 0 jumper to match the RS interface type you wish to use.

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



### 3.2 RS-232 Configuration/Debug Port

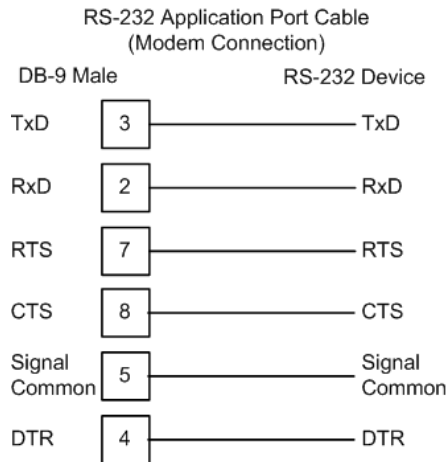
This port is physically an eight-pin, Mini-DIN8F connection. A Mini-DIN8M to DB9M adapter cable is included with the gateway. This port permits a PC-based terminal emulation program to view configuration and status data in the gateway and to control the gateway. Here are the cable pinouts for RS-232 communication on this port.



### 3.3 Application Serial Port(s)

#### 3.3.1 RS-232: Modem Connection (Hardware Handshaking Required)

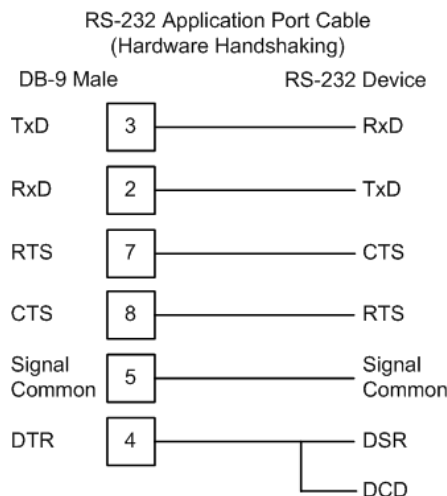
This type of connection is required between the gateway and a modem or other communication device.



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

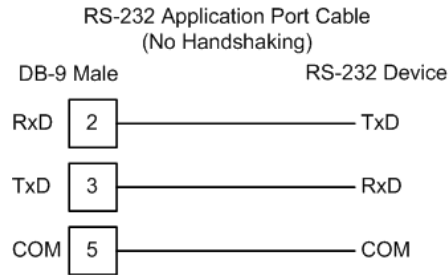
#### 3.3.2 RS-232: Null Modem Connection (Hardware Handshaking)

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

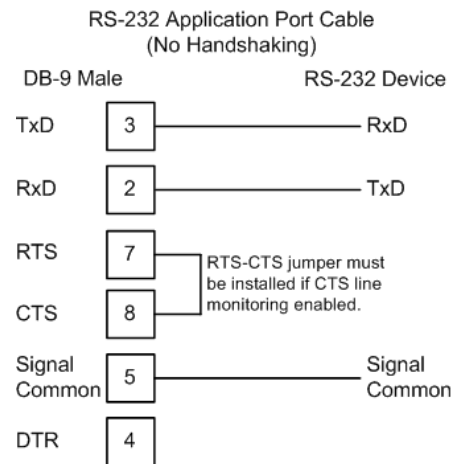


### 3.3.3 RS-232: Null Modem Connection (No Hardware Handshaking)

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

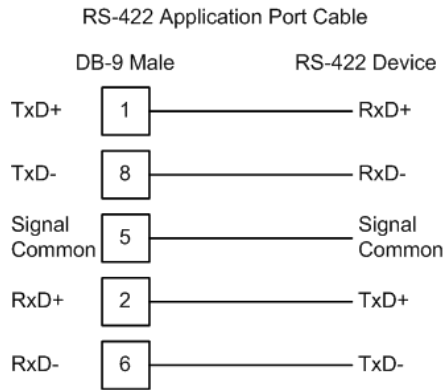


**Note:** For most null modem connections where hardware handshaking is not required, the *Use CTS Line* parameter should be set to **N** and no jumper will be required between Pins 7 (RTS) and 8 (CTS) on the connector. If the port is configured with the *Use CTS Line* set to **Y**, then a jumper is required between the RTS and the CTS lines on the port connection.



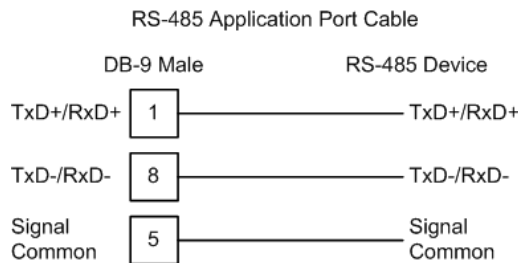
### 3.3.4 RS-422

The RS-422 interface requires a single four or five wire cable. The Common connection is optional, depending on the RS-422 network devices used. The cable required for this interface is shown below:



### 3.3.5 RS-485 Application Port(s)

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



**Note:** Terminating resistors are generally not required on the RS-485 network, unless you are experiencing communication problems that can be attributed to signal echoes or reflections. In these cases, installing a 120-ohm terminating resistor between pins 1 and 8 on the gateway connector end of the RS-485 line may improve communication quality.

### 3.3.6 RS-485 and RS-422 Tip

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





## 4 Diagnostics and Troubleshooting

### *In This Chapter*

- ❖ LED Indicators..... 57
- ❖ Using ProSoft Configuration Builder (PCB) for Diagnostics..... 59

There are two ways to troubleshoot ProLinx Gateways:

- Using the LEDs located on the front of the gateway
- Using the Debug port that provides a view into the gateway's internal database.

### 4.1 LED Indicators

LEDs provide visual indications of potential problems. The following LEDs are found on all ProLinx gateways.

#### 4.1.1 Base Gateway LEDs

| LED   | State       | Description   |
|-------|-------------|---|
| Power | Off         | Power is not connected to the power terminals or source is insufficient to properly power the gateway (minimum required is 800mA at 24 Vdc)   |
|       | Green Solid | Power is connected to the power terminals.  |
| Fault | Off         | Normal operation.   |
|       | Red Solid   | A critical error has occurred. Program executable has failed or has been user-terminated and is no longer running. Press Reset p/b or cycle power to clear error. If not, use the Debug procedures described later in this manual.  |
| Cfg   | Off         | Normal operation.   |
|       | Amber Solid | The unit is in configuration mode. The configuration file is currently being downloaded or, after power-up, is being read, the unit is implementing the configuration values, and initializing the hardware. This will occur during power cycle, or after pressing the reset button. It also occurs after a cold/warm boot command is received. |
| Err   | Off         | Normal operation.   |
|       | Flashing    | An error condition has been detected and is occurring on one of the application ports. Check configuration and troubleshoot for communication errors.   |
|       | Solid Red   | This error flag is cleared at the start of each command attempt (Master/Client) or on each receipt of data (slave/adaptor/server); so, if this condition exists, it indicates a large number of errors are occurring in the application (due to bad configuration) or on one or more ports (network communication failures).                    |

### 4.1.2 Mini-DIN8 Serial Port

| LED          | State              | Description  |
|--------------|--------------------|--|
| ERR          | Off                | Normal state. When off and Port Active LED indicates activity, there are no communication errors.  |
|              | Red On or Flashing | Activity on this LED indicates that some communication error was detected, either during transmit or receive. To determine the exact error, use the Debug port procedures described in this section. |
| ACT (Active) | Off                | No activity on this port.  |
|              | Flashing Green     | The port is either actively transmitting or receiving data.  |

## 4.2 Using ProSoft Configuration Builder (PCB) for Diagnostics

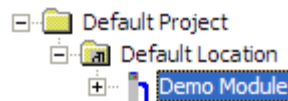
The *Configuration and Debug* menu for this gateway is arranged as a tree structure, with the *Main* menu at the top of the tree, and one or more submenus for each menu command. The first menu you see when you connect to the gateway is the *Main* menu.

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 gateway does not respond to mouse movements or clicks. The command executes as soon as you press the [command letter] — you do not need to press **[ENTER]**. When you type a [command letter], a new screen will be displayed in your terminal application.

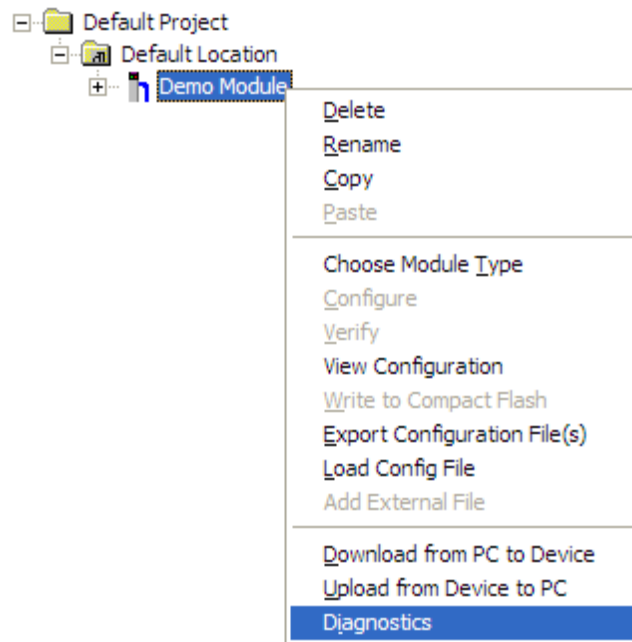
### 4.2.1 Using the Diagnostic Window in ProSoft Configuration Builder

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

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

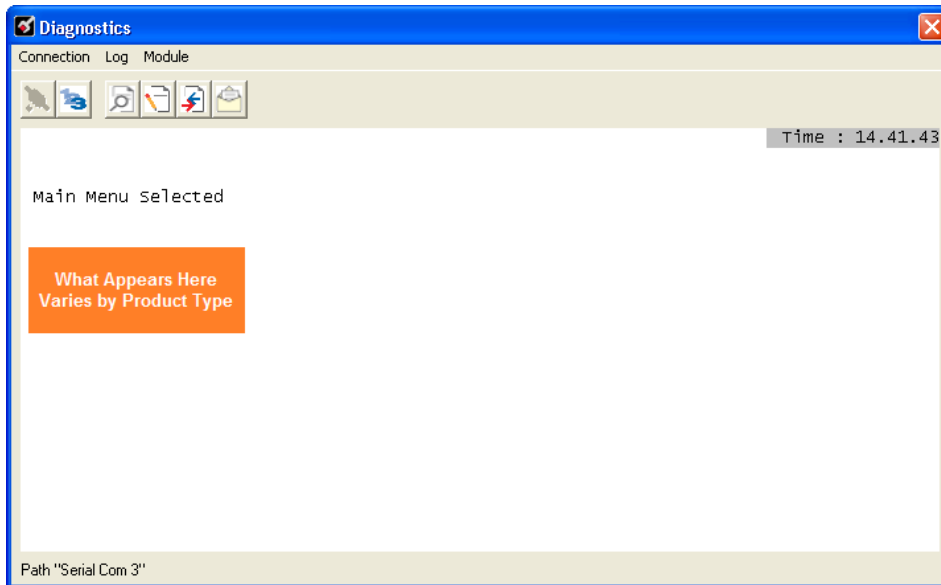


- 2 On the shortcut menu, choose **DIAGNOSTICS**.



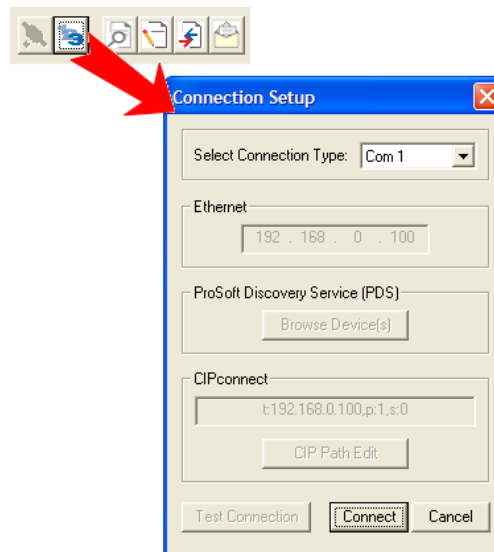
This action opens the *Diagnostics* dialog box.

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



If there is no response from the gateway, 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 gateway.



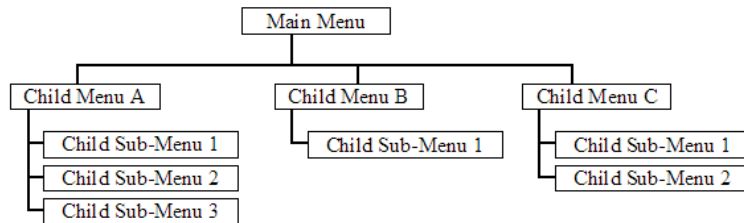
- 2 Verify that the null modem cable is connected properly between your computer's serial port and the gateway. A regular serial cable will not work.
- 3 On computers with more than one serial port, verify that your communication program is connected to the same port that is connected to the gateway.

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

## 4.2.2 Navigation

All of the submenus for this gateway 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 gateway, and briefly discusses the commands available to you.

## 4.2.3 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 gateway and may not be used interchangeably.



## 5 Reference

### *In This Chapter*

- ❖ Product Specifications ..... 63
- ❖ IEC-60870-5-101 Slave Protocol Implementation ..... 67

The standards used in developing the product are listed in the following table.

| Publication                    | Title   |
|--------------------------------|---|
| IEC 60870-5-101                | Companion Standard for Basic Telecontrol Tasks            |
| IEC 60870-5-101<br>Amendment 1 | Companion Standard for Basic Telecontrol Tasks            |
| IEC 60870-5-1                  | Transmission Frame Formats                                |
| IEC 60870-5-2                  | Link Transmission Procedures                              |
| IEC 60870-5-3                  | General Structure of Application Data                     |
| IEC 60870-5-4                  | Definition and Coding of Application Information Elements |
| IEC 60870-5-5                  | Basic Application Functions                               |

Refer to these standards for any questions on the protocol supported.

### 5.1 Product Specifications

ProLinx Communication gateways are the ideal solution for bridging communication between networks that use differing protocols. ProLinx gateways provide several protocol combinations, allowing for a wide range of flexibility in establishing a network, or fitting into and expanding your current infrastructure. Coupled with ProSoft's radio technology, you can easily integrate devices into your system wirelessly, and without the need to purchase additional radios to do so.

#### **5.1.1 General Specifications**

ProLinx<sup>®</sup> Communication gateways provide connectivity for two or more dissimilar network types. The gateway, enclosed in sturdy extruded aluminum, are stand-alone, DIN-rail-mounted solutions that provide data transfer between many of today's most widely used industrial automation protocols.

### 5.1.2 Functional Specifications - IEC 60870-5-101 Slave

The 101S gateway acts as an input/output device between the IEC 60870-5-101 Master and other protocols and networks, as well as several proprietary interfaces.

Protocol specifications include:

- User-definable gateway memory usage
- Storage of IEC time used in gateway is available in the database
- Protocol implementation conforms to the IEC 60870-5-101 specification with fully configurable parameters
- Priority Queues
- Invalid Bit Monitoring
- Supports Balanced and Unbalanced Mode
- Supports CP24 and CP56 time formats for events
- Event generation configurable per point or data type

---

#### General Parameters

---

|                          |  |
|--------------------------|--|
| Communication parameters | Baud Rate: 110 to 38,400 baud<br>Stop Bits: 1 or 2. Data Bits: 5 or 8<br>Parity: None, Even, Odd<br>RTS Timing Delays: 0 to 65535 milliseconds |
|--------------------------|--|

---

#### IEC 60870-5-101 Slave

---

|                         |  |
|-------------------------|--|
| Configurable Parameters | Data link address length<br>Common Address of ASDU length<br>Inform. Object Address length<br>Select/Operate Timeout<br>Event Scan delay<br>Use Balanced Mode<br>Short Pulse Time<br>Long Pulse Time<br>Time DB Offset |
|-------------------------|--|

---

### 5.1.3 Internal Database

The ProLinx gateway contains an internal database that consists of areas for application data, status information, and configuration information.

The internal database is shared between all ports on the gateway and is used as a conduit to pass information from a device on one network to one or more devices on another network.



### Application Data Area

The protocol drivers exchange data by storing and retrieving data from a shared application memory data area. The database is used as a source for data to be sent to remote devices and holds data received from the remote devices. For protocol drivers that act as Masters or Clients, commands defined in the configuration file (stored in the configuration data area) control how the data is to be handled in the database. For protocol drivers that act as slaves or servers, the remote Master or Client must be properly programmed to send data to or request data from the correct memory addresses in the gateway application data area.

### Status Data Area

This area stores error codes, counters, and other status information.

### Configuration File

This file contains gateway configuration information such as port configuration, network information, and command configuration. This configuration file is transferred to or from the gateway.

### Optional Web Server for Ethernet Solutions

An HTML server is available for Ethernet ProLinx gateways. With this option, HTML pages can be created to:

- Display gateway internal register and status values
- Accept user data input values via POST commands for setpoint, on/off control, and so on.
- Provide limited graphic file support

| Type                         | Specifications  |
|------------------------------|---|
| HTML Server (See note below) | Key features of the HTML server include:<br>Max HTML page size: 1MB<br>Max File Storage: 32MB<br>Supported context types: jpeg, bmp, css<br>Supported data types: bit, ASCII, integer, float<br>Sockets: Up to five connections. Note that this limits the number of simultaneous graphic file and frame references per HTML page                                   |
| FTP Server                   | Permits remote HTML file transfer between the gateway and remote host. Capabilities of the FTP Server include:<br>Single socket connection<br>Non-passive transfers only<br>WS_FTP or Command Line FTP recommended<br>CuteFTP, Internet Explorer, Netscape, or NCFTP all support multiple socket connections and therefore are not supported by the ProLinx gateway |

**Important Note:** The Web Server is not designed to act like, or replace, the powerful web servers available in the marketplace today. Please check application specifics with Technical Support if you are unsure if your application will work with the ProLinx Web Server.

### 5.1.4 Hardware Specifications

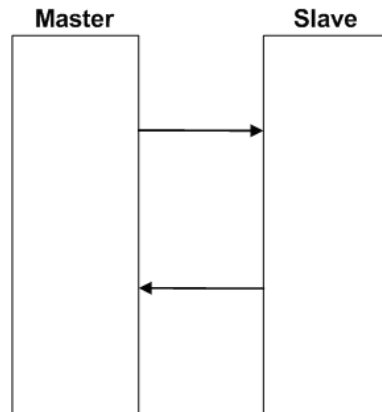
| Specification   | Description   |
|---|---|
| Power Supply  | 24 Vdc nominal<br>18 to 32 Vdc allowed<br>Positive, Negative, GND Terminals<br>2.5 mm screwdriver blade   |
| Current Load  | 500 mA maximum @ 32 Vdc maximum   |
| Operating Temperature   | -4° F to 122° F (-20° C to 50° C)   |
| Storage Temperature   | -40° F to 185° F (-40° C to 85° C)  |
| Relative Humidity   | 5 to 95% (without condensation)   |
| Dimensions<br>(Height x Width x<br>Depth)                           | Standard: 5.20 in x 2.07 in x 4.52 in<br>(13.2 cm x 5.25 cm x 11.48 cm)<br>Extended: 5.20 in x 2.73 in x 4.52 in<br>(13.2 cm x 6.934 cm x 11.48 cm)   |
| LED Indicators<br>(On all gateways)                                 | Power and Hardware Fault<br>Configuration and Application Communication Status<br>Serial Configuration Port Activity and Error  |
| Configuration<br>Serial Port  | DB-9M RS-232 only<br>No hardware handshaking  |
| Ethernet Port<br>(Ethernet protocol<br>gateways only)               | 10Base-T half-duplex RJ45 Connector<br>Link LED and Activity LED indicators<br>Electrical Isolation 1500 V rms at 50 Hz to 60 Hz for 60 s,<br>applied as specified in section 5.3.2 of IEC 60950: 1991<br>Ethernet Broadcast Storm Resiliency = less than or equal<br>to 5000 [ARP] frames-per-second and less than or equal to<br>5 minutes duration |
| Application Serial<br>Port(s)<br>(Serial protocol<br>gateways only) | RS-232/422/485<br>RS-232 handshaking configurable<br>RS-422/485 DB-9 to Screw Terminal Adapter<br><b>Note:</b> The number of serial application ports depends on<br>the gateway type, and the combination of protocols.   |
| Serial Port Isolation   | 2500 V RMS port signal isolation per UL 1577<br>3000 VDC min. isolation port to ground and port to logic  |
| Shipped with Each Unit  | Mini-DIN to DB-9M serial cables<br>4-foot RS-232 configuration cable<br>2.5mm screwdriver<br>CD (docs and Configuration utility)<br>RS-422/485 DB-9 to Screw Terminal Adapter for each<br>serial application port (serial protocols only)   |

## 5.2 IEC-60870-5-101 Slave Protocol Implementation

This section presents an overview of how the 101S works, while skipping the complex details of the protocol specification. If you require more information about the implementation of the protocol, refer to the protocol specification (IEC 60870-5-101 2003). For more information on configuring and modifying the implementation of the protocol with the 101S, refer to Configure the Module. (page 23)

The IEC-60870-5-101 protocol applies to telecontrol equipment and systems with coded bit serial data transmission for monitoring and controlling geographically widespread processes.

Any application using the IEC-60870-5-101 protocol will have one Master (controlling station) and one or more slaves (controlled stations). The Master will constantly monitor and control the data from each slave in the network.



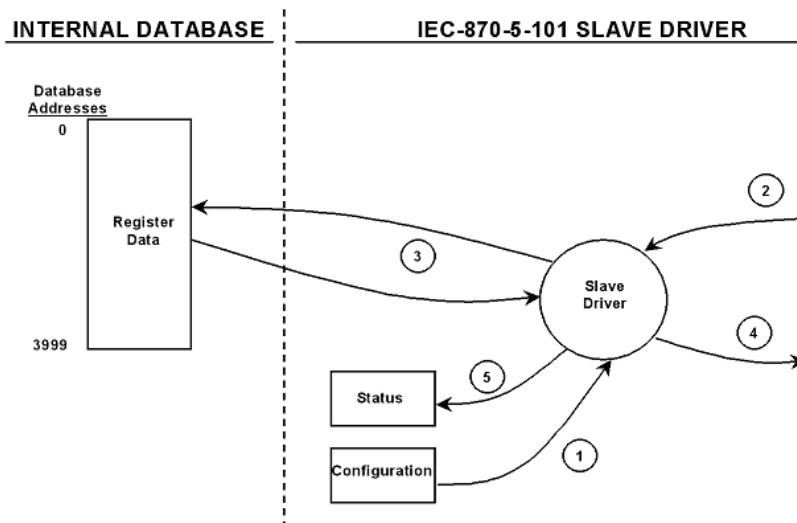
The gateway works as an IEC-60870-5-101 slave. It can send monitor data, receive commands or generate events to the Master unit, as explained in the following topics.

### 5.2.1 Data Flow Between the gateway and External Host

The following section describes the flow of data between the 101S gateway and the IEC 60870-5-101 Master unit.

#### Slave Driver

The slave driver allows the 101S gateway to respond to data read and write commands issued by a Master unit on the serial network. The following flow chart and associated table describe the flow of data into and out of the gateway.



- 1 The slave driver receives the configuration information from the configuration file in the gateway. This information configures the driver and defines the node characteristics.
- 2 A host device issues read or write commands to the gateway's node address. The driver qualifies the message before accepting it into the gateway.
- 3 Once the gateway accepts the message, the data is immediately transferred to or from the internal database in the gateway. If the command is a read command, the data is read out of the database and a response message is built. If the command is a write command, the data is written directly into the database and a response message is built. The gateway will also generate messages without being queried by the host. These messages include spontaneous and cyclic COT messages.
- 4 Once the data processing is complete, the response is issued to the originating Master node.
- 5 Counters are available in the Status Block that permits the level of activity of the driver to be determined.

The configuration section of this manual describes the complete list of parameters that must be defined for a slave. The IEC-60870-5-101 Interoperability Document for the 101S slave contains a listing of the protocol support supplied by the gateway.

### Databases

The read and write areas can be placed anywhere in the gateway's database area. Because each point is defined individually to the gateway, the data for a specific type need not be contiguous in the gateway's database. This means that the gateway error/status area can be passed to the controlling station using the M\_ME\_NB\_1 database. In the database definition for this type, establish a M\_ME\_NB\_1 information object for each status value to be monitored by the controlling station and use the gateway's status database as the DB Address.

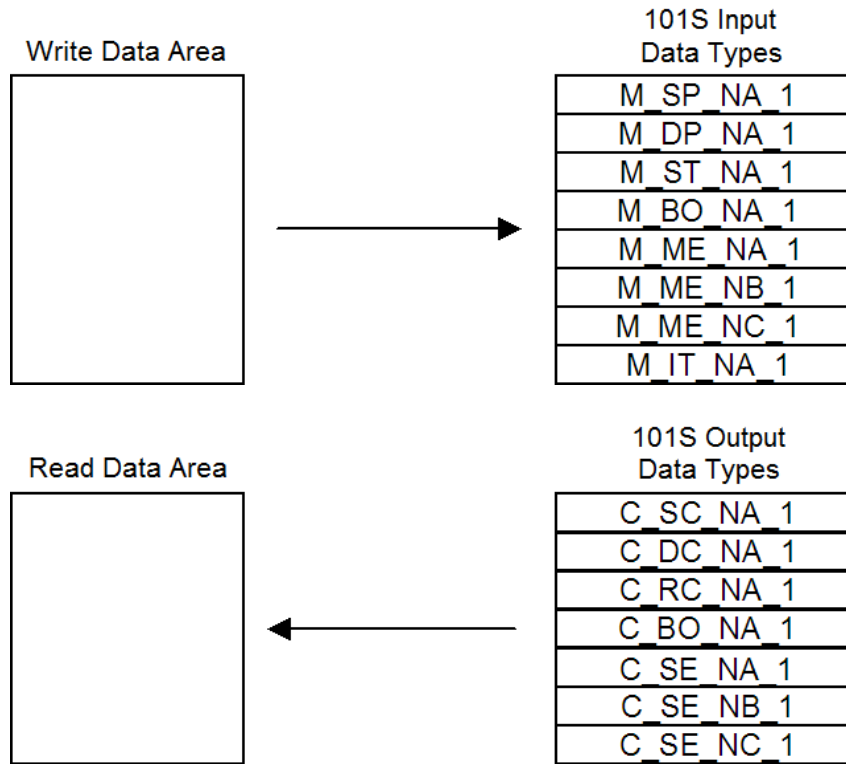
This section discusses the databases utilized by the gateway to support the IEC 60870-5-101 protocol. The user is responsible for defining the databases for their specific application. The size of each ASDU database and definition of each point within each data type is established in the gateway's configuration file or by using ProSoft Configuration Builder. Refer to the *Module Configuration* section for a complete discussion of configuration options. The following table summarizes the ASDU data types used in each of the 15 individual databases.

| Type ID | Type      | Description   | Data Representation  |
|---------|-----------|---|--|
| 1       | M_SP_NA_1 | Monitored Single-point Information: This data type stores a single binary input point. Associated time-tagged event information for this type are M_SP_TA_1 (2) and M_SP_TB_1 (30).   | Single bit value with 0=Off and 1=On.  |
| 3       | M_DP_NA_1 | Monitored Double-point Information: This data type stores a dual-point binary input value (that is, valve status). Associated time-tagged event information for this type are M_DP_TA_1 (4) and M_DP_TB_1 (31).   | Double-bit status with 00b (0 decimal) = indeterminate or intermediate, 01b (1 decimal) = Off, 10b (2 decimal) = On and 11b (3 decimal) = indeterminate.   |
| 5       | M_ST_NA_1 | Monitored Step-point Information: This data type is used for step position of transformers or other step position information. The value for the position ranges from -64 to 63. Associated time-tagged event information for this type are M_ST_TA_1 (6) and M_ST_TB_1 (32). | Step data is stored in a single character value with bits 0 to 6 (-64 to +63) representing the step position and bit 7 representing the following states: 0 = Equipment is not in transient state 1 = Equipment in transient state |
| 7       | M_BO_NA_1 | Monitored 32-Bit Bitstring Binary State Information: This data type is used to hold and transmit binary state data as bitstrings of 32 bits.  | Bitstring data is stored as groups of 32 bits, with each bit representing the state of an object: 0=Off and 1=On   |
| 9       | M_ME_TA_1 | Monitored Normalized Measured Value: This data type is used for analog input data. Associated time-tagged event information for this type are M_ME_TA_1 (10) and M_ME_TD_1 (34).  | Normalized values are stored in a (16-bit) word data area with a range of $-1..+1 \cdot 2^{-15}$   |
| 11      | M_ME_NB_1 | Monitored Scaled Measured Value: This data type is used for analog input data. Associated time-tagged event information for this type are M_ME_TB_1 (12) and M_ME_TE_1 (35).  | Scaled values are stored in a (16-bit) word data area with a range of $-2^{15}..+2^{15}-1$   |

| Type ID | Type      | Description  | Data Representation  |
|---------|-----------|--|--|
| 13      | M_ME_NC_1 | Monitored Measured Value, Short Floating-Point Number: This data type is used for analog input data stored in floating point format according to the IEEE STD 754, QDS format. Associated time-tagged event information for this type are M_ME_TC_1 (14) and M_ME_TE_1 (36). | Short floating-point number stored in IEEE STD 754 format (Fraction, Exponent, Sign)                             |
| 15      | M_IT_NA_1 | Monitored Integrated Total-point Information: This data type stores meter or other count data. Associated time-tagged event information for this type are M_IT_TA_1 (15) and M_IT_TB_1 (37).   | Binary counter data is stored in a double-word (32-bit) value with a range of $-2^{31}..+2^{31}-1$ .             |
| 45      | C_SC_NA_1 | Single-point Command: This command controls a single binary point such as a relay.   | Single bit value with 0 = Off and 1 = On   |
| 46      | C_DC_NA_1 | Double-point Command: This command controls a dual-point binary control device such as a trip/close relay.   | Double Command with 0 = Not permitted 1 = Off 2 = On 3 = Not permitted   |
| 47      | C_RC_NA_1 | Regulating Step Command: This command controls a stepping device such as a transformer.  | Regulating Step Command with 0 = Not permitted 1 = Next step lower 2 = Next step higher 3 = Not permitted        |
| 48      | C_SE_NA_1 | Setpoint Command, Normalized Value: This command controls an analog device.  | Normalized values are stored in a (16-bit) word data area with a range of $-1..+1-2^{-15}$                       |
| 49      | C_SE_NB_1 | Setpoint Command, Scaled Value: This command controls an analog device.  | Scaled values are stored in a (16-bit) word data area with a range of $-2^{15}..+2^{15}-1$                       |
| 50      | C_SE_NC_1 | Setpoint Command, Short Floating-Point Value: This data type is used for analog input data stored in floating point format according to the IEEE STD 754, QDS format.  | Short floating-point number stored in IEEE STD 754 format (Fraction, Exponent, Sign)                             |
| 51      | C_BO_NA_1 | 32-Bit Bitstring Binary State Command: This data type is used to hold and transmit binary state data as bitstrings of 32 bits.   | Bitstring data is stored as groups of 32 bits, with each bit representing the state of an object: 0=Off and 1=On |

A key concept in interfacing the protocol with the internal database is the relationship between the IEC-60870-5-101 databases and the data transfer operation between the driver and gateway's internal database. The control data types, C\_XX\_XX\_1, are transferred from the driver to the gateway's internal database. This data can then be used by the other gateway protocol or processor in which the gateway is installed to control other devices. The monitor data types (M\_XX\_XX\_1) are sourced from the gateway's internal database and is transferred to the remote Master on the IEC-60870-5-101 network.

The relationships between the data types and the read and write data areas of the gateway are displayed in the following diagram:



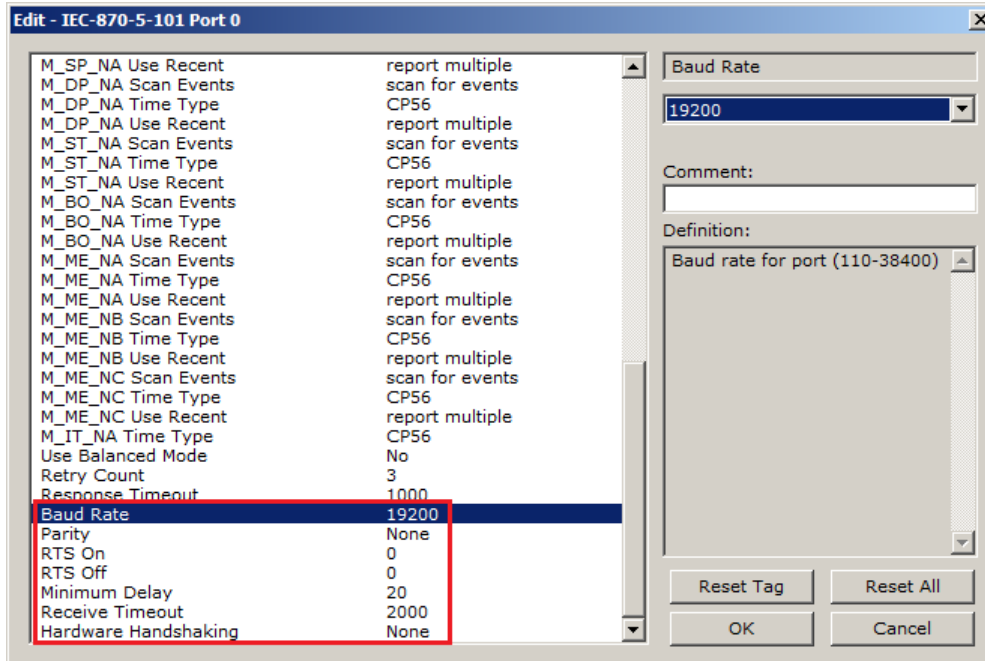
The Write Data Area is used to hold data for the 101S Monitored Point databases, which is data that will be sent by the 101S Slave driver from the gateway to the remote Master. The Read Data Area is used to hold data for the 101S Control Point databases, which is data that has been sent from the remote Master to the 101S driver to be stored in the gateway. The read and write areas can be placed anywhere in the gateway's database. Because the gateway address of each point is defined individually, the data area for a specific data type need not be contiguous in the gateway's memory. However, most users find it convenient to keep data of each type in contiguous address blocks.

Module error/status data can be passed to the Master using the M\_ME\_NB\_1 database. In the database definition for the type, establish a point for each status value to be monitored by the Master and set the gateway's database address for the point in the definition.

## 5.2.2 General Parameter Configuration

### Communication Parameters

The following parameters should be configured for serial communication:

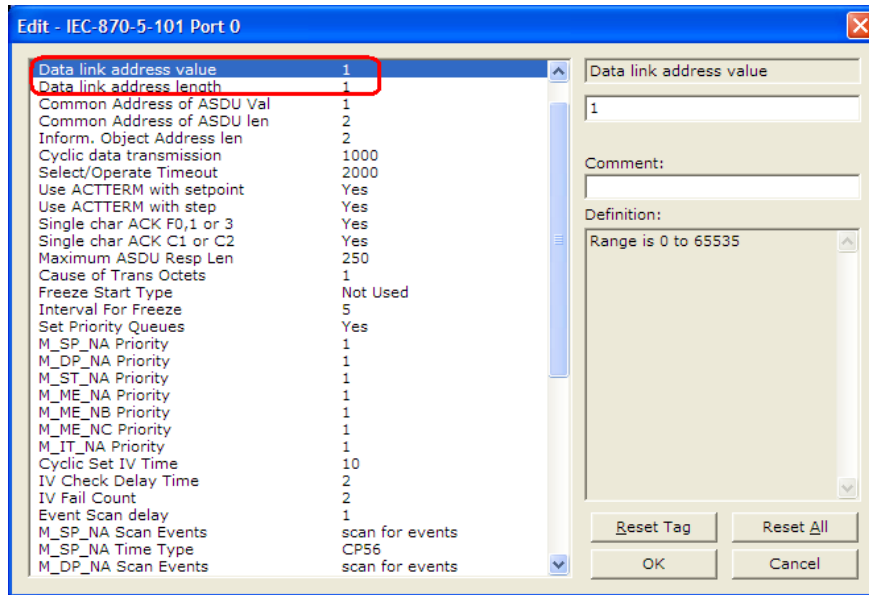


Adjust these parameters for your application.



### Data Link Configuration

The protocol specification document IEC 60870-5-2 specifies an unambiguous address (number) must be used for each link. Each address may be unique within a specific system, or may be unique within a group of links sharing a common channel. The protocol specification defines that the Data Link Address may have 0, 1 or 2 octets. The first option should be used only during Balanced Mode.



Configure the Data Link Address Length parameter to set the number of octets to be used for the Data Link Address value. It is essential that the Master unit also uses the same number of octets configured in the 101 Slave. Values shown here are for example only. Your application may require different settings.

- Data Link Address Length: 1

You must also configure the actual Data Link Address value using the following parameter:

- Data link address: 1

This value identifies the gateway's address in the network.

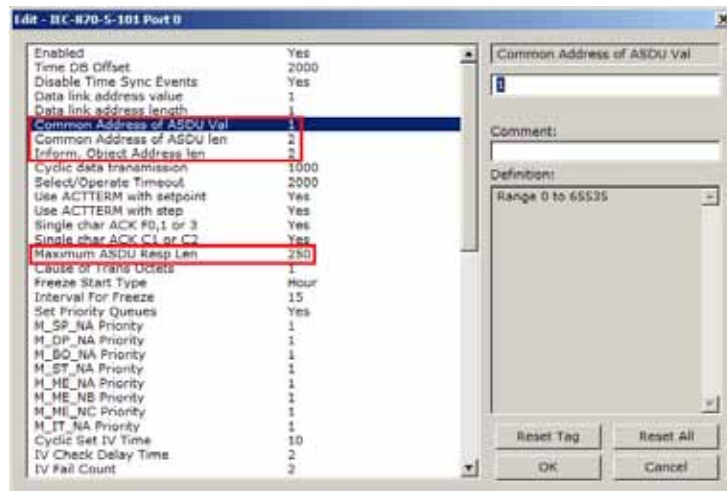
### ASDU Configuration

The protocol specification document IEC 60870-5-3 describes the Basic Application Data Units that are used in the protocol. It also defines the Application Service Data Unit (ASDU) used by the protocol for data communication.

You can configure the number of bytes to be used for the following ASDU components:

| Component                  | Abbreviation | Possible Number of Octets |
|----------------------------|--------------|---------------------------|
| Cause of Transmission      | COT          | 1 or 2                    |
| Common Address of ASDU     | CASDU        | 1 or 2                    |
| Information Object Address | IOA          | 1, 2 or 3                 |

These parameters must be configured to match the Master's configuration (values shown are for example only; your application may require different values):



The Common Address of ASDU is the station address. The gateway allows the addressing of only the whole station (some devices allows different Common Addresses to identify particular station sectors). You should configure the Common ASDU Address with the following parameter:

- Common Address of ASDU: 1

You should also configure the maximum number of bytes that the gateway will support for each ASDU response to the Master unit. The range is from 25 to 252 bytes. You should verify the maximum number of bytes supported by the Master because some IEC 60870-5-101 Master devices support messages with less than 252 bytes.

Configure the following parameter to set the maximum number of bytes to be transferred at every ASDU response.

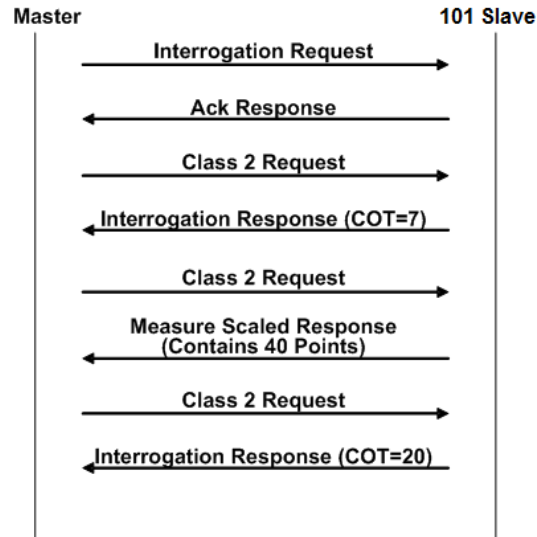
- Maximum ASDU Resp Len: 252

A value less than 252 can cause the gateway to break down the response to send all points using more response messages, with each message containing fewer bytes.

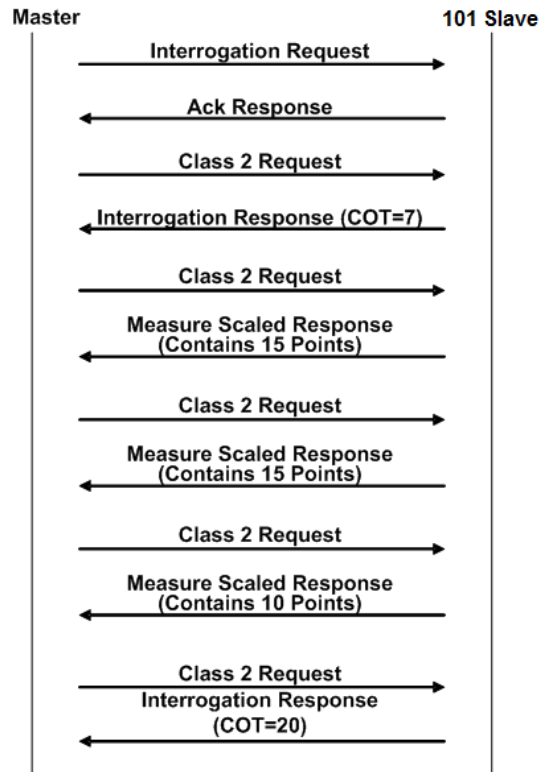
**Example - Changing the ASDU Length:**

Considering that the Master sends a General Interrogation request to poll forty M\_ME\_NB points (measured scaled points in Unbalanced Mode):

If Maximum ASDU Resp Len = 252:



This example shows that the gateway sends all 40 measured scaled points in one single message, if the message is not greater than 252 bytes. However, some Master devices may not support messages containing this number of bytes. If the Master for a given application supports only 100 bytes, the following communication procedure would occur:



As shown in the previous diagram, the gateway sends out three (3) messages (15 points, 15 points, and 10 points) instead of sending the whole 40 points in one single message.

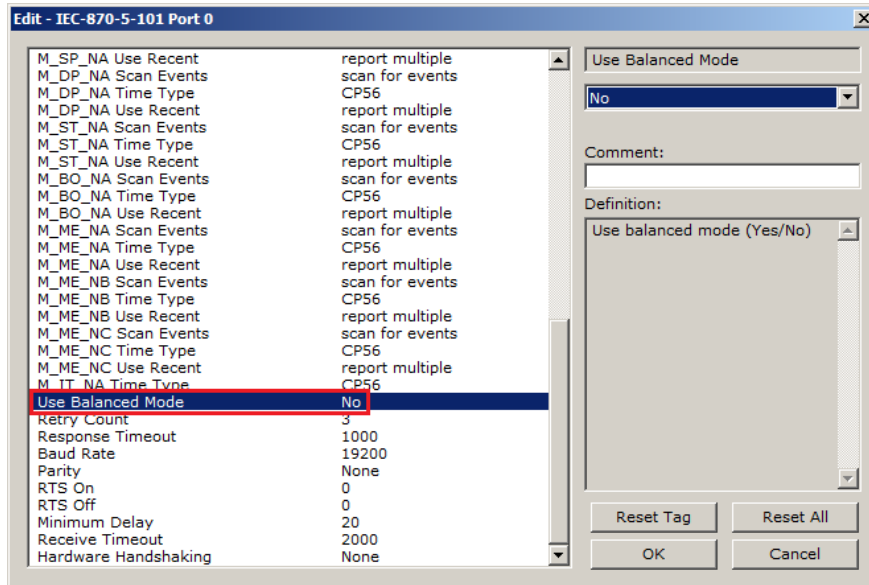
**Note:** This example shows the case where IOA Length = 3 bytes, COT Length = 2 bytes and CASDU Length = 2 bytes.

**Important:** If the database parser gets a point index that is not valid, the whole database is set as invalid and no points are reported. Because the index 0 is not valid (the protocol does not support this index value), the driver considers it invalid. For example, if you set the size of the ASDU to 1 and you set a value of 1000 for a point index, this is also invalid as the indexes can only go from 1 to 255.

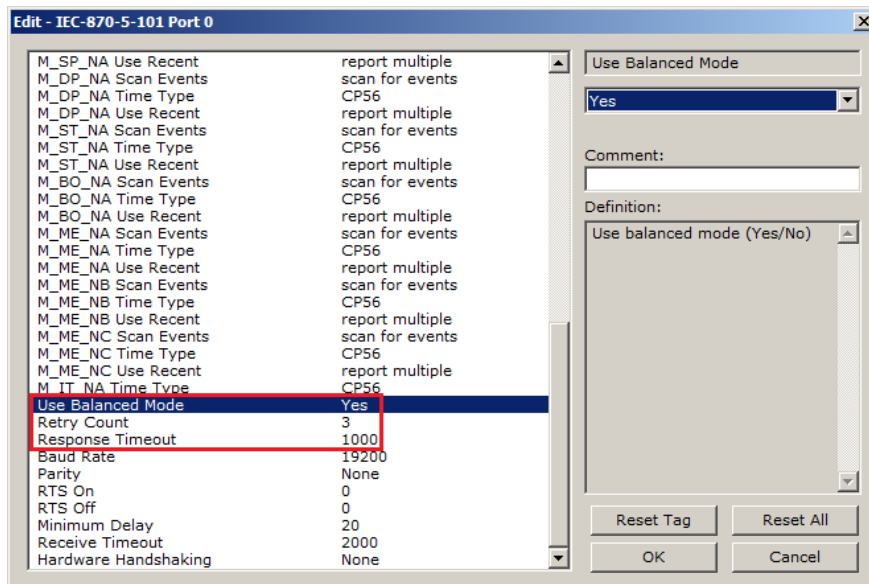
### Balanced and Unbalanced Modes

The gateway supports balanced and unbalanced modes. In balanced mode, each station may initiate message transfers. If the links from the Master unit to several slaves share a common physical channel, then these links must be operated in an unbalanced mode to avoid the possibility of more than one device attempting to transmit on the channel at the same time.

Select the communication mode with the following parameter:



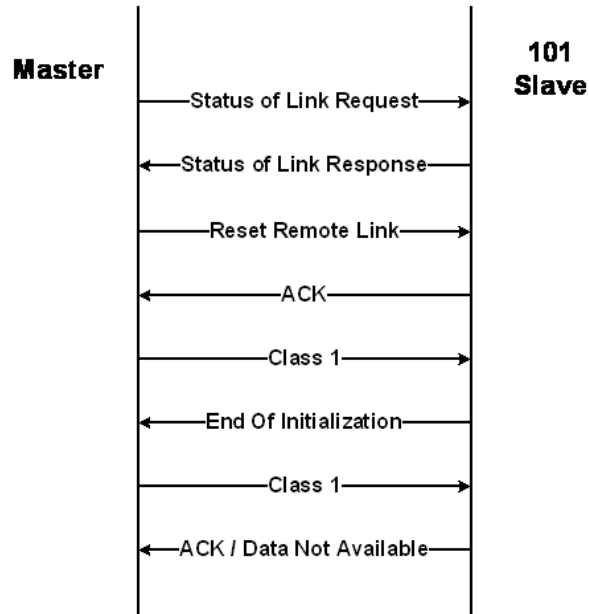
To use balanced mode, configure the following parameters:



**Unbalanced mode**

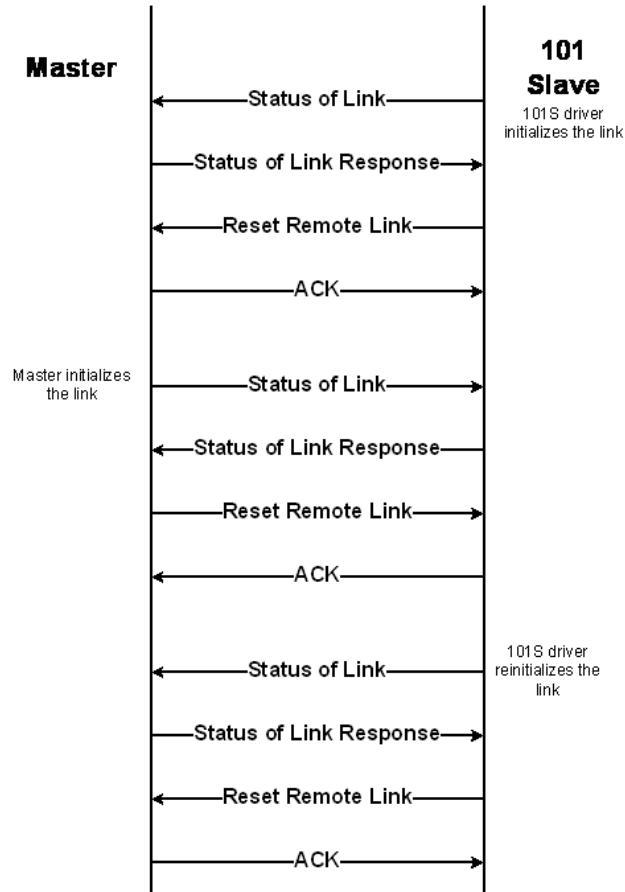
In order to start communications between the Master and the slave units, the Master tries to establish the link connection by transmitting repeated "Request Status of Link" at specific time out intervals. When the gateway's link is available, it will respond with a "Status of Link" response. Then, the Master transmits a "Reset of Remote Link" message and the slave responds with an Acknowledge (ACK) response. Then the Master sends two consecutive Class 1 requests. The slave responds the first Class 1 request with an "End of Initialization" response and the second Class 1 request with an ACK message.

The following illustration shows a typical initialization procedure for the unbalanced mode:



### Balanced Mode

During balanced mode, the link must be initialized in both directions. The 101S gateway also always reinitializes the link after it receives an initialization request from the Master. Therefore, the following initialization occurs during balanced mode, after 101S boots.

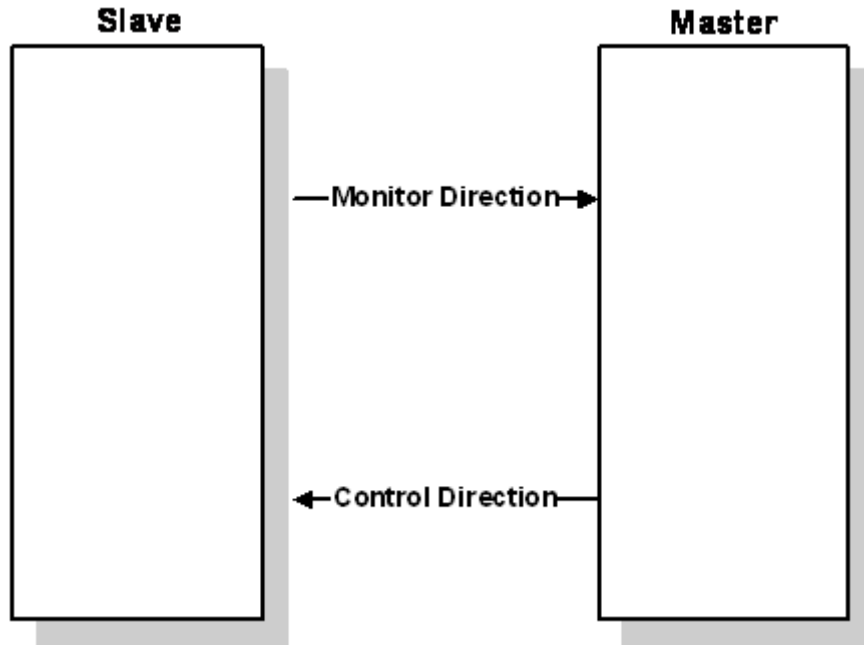


After the initialization procedure is completed, the Master and the 101S start communicating. During communication, if the Master fails to respond to a message from the gateway, the gateway will retry for a configured number of times. If the Master still fails to respond, the gateway will attempt to initialize the line again.

### 5.2.3 Monitor Direction and Control Direction

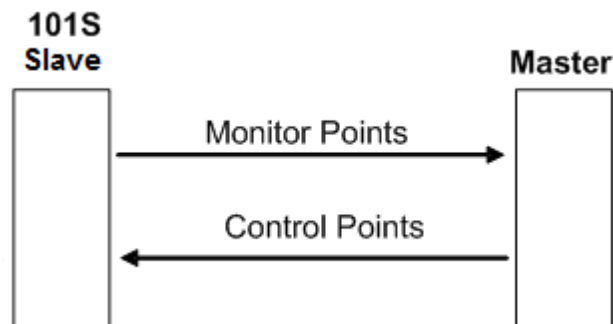
The protocol specification defines two directions of data: monitor direction and control direction.

- **Monitor Direction:** The direction of transmission from a slave to the Master
- **Control Direction:** The direction of transmission from the Master to the slave



The points that are typically transferred from the slave to the Master are also known as Monitor Points (or Monitor Information Objects). The points that are typically transferred from the Master to the slave are also known as Control Points (or Command Information Objects).

The following illustration describes the data transfer for control and monitor points between the 101S and the Master.



You must configure all points in the correct location in the 101S gateway database, so that they can be updated properly from the Master.



### 5.2.4 Using Monitor Points

The following monitor points are supported by the 101S gateway:

| Symbol  | Description                              | Data Size in Database | Addressing Type |
|---------|--|-----------------------|-----------------|
| M_SP_NA | Monitored Single-Points                  | 1 bit                 | Bit             |
| M_DP_NA | Monitored Dual-Points                    | 2 bits                | Bit             |
| M_ST_NA | Monitored Step-Points                    | 1 byte                | Byte            |
| M_ME_NA | Monitored Measured Normalized-Points     | 1 word                | Word            |
| M_BO_NA | Monitored 32-bit Bitstring Points        | 2 words               | Word            |
| M_ME_NB | Monitored Measured Scaled-Points         | 1 word                | Word            |
| M_ME_NC | Monitored Measured Short Floating Points | 2 words               | Double-Word     |
| M_IT_NA | Monitored Counter-Points                 | 2 words               | Double-Word     |

Each monitor point is identified by its Information Object Address (it should be unique for each Common ASDU Address in the network). For each monitor point, configure the following parameters:

**Point #** - The information object address of the point. It identifies the point in the network.

**DB Address** - The database location in the 101S gateway associated with the point. You must associate each point to a database address in the 101S gateway. The interpretation of this parameter depends on the point type configured. For example, for an M\_SP\_NA point, this value represents the bit address. For a M\_ME\_NA point, this value represents the Word address.

**Group(s)** - This is the group definition for the point. It sets how the point will be polled by the Master (cyclic or group interrogation). It can also be used to enable or disable the event generation for one specific point. The group parameter is discussed in the Data Communication section.

**Deadband** - Sets the deadband for each Measured point. If the value changes from more than the configured deadband, the gateway will generate an event for this point.

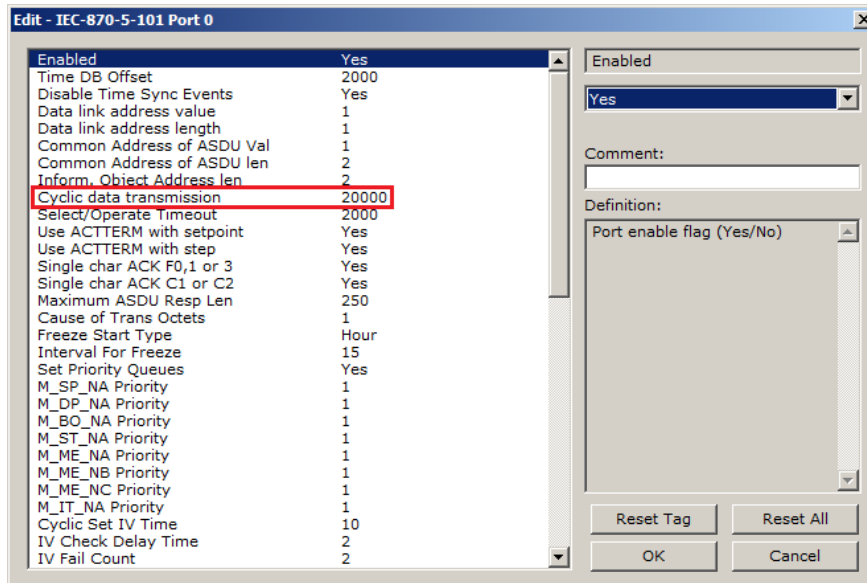
**IV DB Bit** - This feature allows the application to set the invalid (IV) quality bit of the protocol for all the monitored ASDU types supported. If you enable this feature, the Master can determine the individual IV quality bit status of each point you configured.

Monitor Data Transfer

In unbalanced mode, the polling procedure is initiated from the Master unit through Class 1 and Class 2 requests. In general, ASDUs containing periodic(cyclic) causes of transmission are typically assigned to be transmitted in response to Class 2 requests from the Master. All ASDUs containing time-tagged or spontaneous causes of transmission are typically assigned to be transmitted in response to Class 1 requests. Other ASDUs with other causes of transmission of low priority, such as background scans, may also be assigned to be transmitted in response to Class 2 requests.

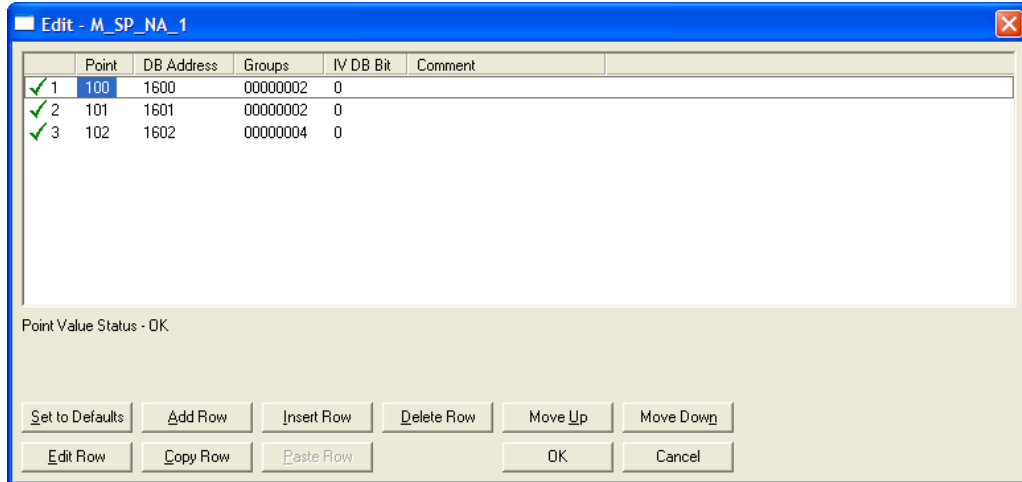
Typically, you should properly configure the group code (page 50) for each monitor point to define how the Master will poll for the point.

The gateway will periodically send all points configured for periodic/cyclic poll (0x80000000) at every x milliseconds, where x is configured with the *Cyclic Data Transmission* parameter:



### M\_SP\_NA

A *Monitor Single-Point* information object occupies one binary bit and uses bit addressing. For example, if you configured the following points as shown:



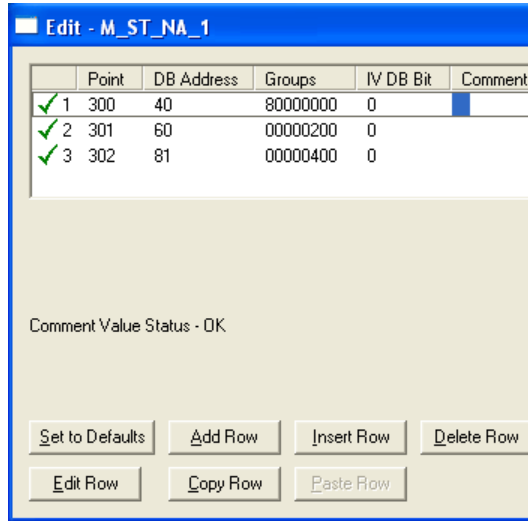
The following table describes how these points would be stored in the gateway:

| <b>Inf. Object Address</b> | <b>Module Database Address</b>       |
|----------------------------|--------------------------------------|
| 100                        | Bit 0 of word 100 (Bit address 1600) |
| 101                        | Bit 1 of word 100 (Bit address 1601) |
| 102                        | Bit 2 of word 100 (Bit address 1602) |

M\_ST\_NA

A *Monitor Step Position* information object occupies one byte and uses byte addressing.

For example, if you configured the following points:



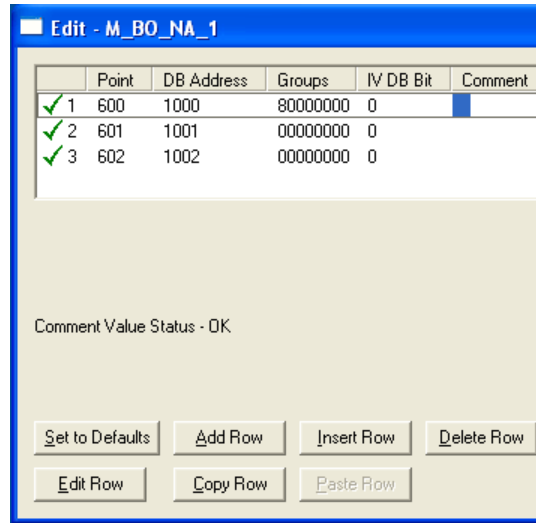
The following table describes how these points would be stored in the gateway.

| Inf. Object Address | Module Database Address                |
|---------------------|--|
| 300                 | Low byte of word 20 (Byte address 40)  |
| 301                 | Low byte of word 30 (Byte address 60)  |
| 302                 | High byte of word 40 (Byte address 81) |

M\_BO\_NA

A *Monitored 32-Bit Bitstring Point* occupies two words and uses double-word addressing.

For example, if you configured the following points:



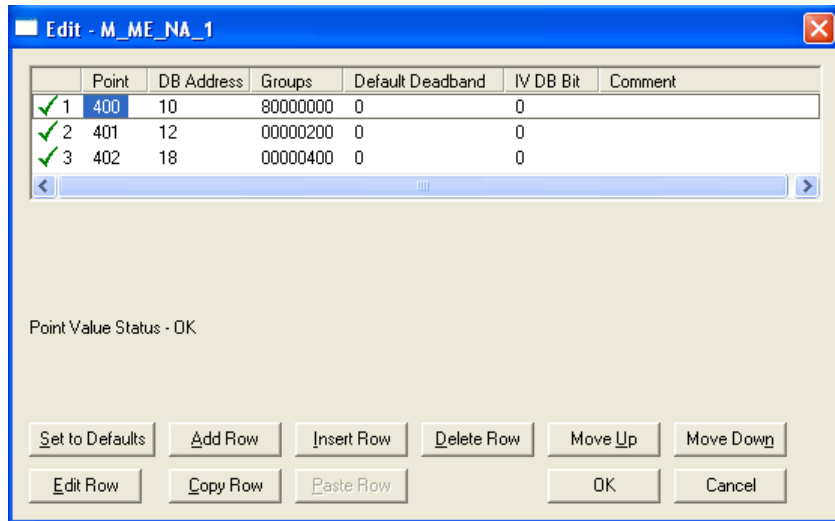
The following table describes how these points would be stored in the gateway memory database.

| Inf. Object Address | Module Database Address                        |
|---------------------|--|
| 600                 | Words 2000 and 2001 (Double-word address 1000) |
| 601                 | Words 2002 and 2003 (Double-word address 1001) |
| 602                 | Words 2004 and 2005 (Double-word address 1002) |

**M\_ME\_NA and M\_ME\_NB**

A *Monitor Normalized Measured* information object or *Monitor Scaled Measured* information object occupies one word and uses word addressing.

For example, if you configured the following points:



The following table describes how these points would be stored in the gateway.

| Inf. Object Address | Module Database Address   |
|---------------------|---------------------------|
| 400                 | Word 10 (Word address 10) |
| 401                 | Word 12 (Word address 12) |
| 402                 | Word 18 (Word address 18) |

*Monitor Normalized Measured* information objects use a data representation defined by the protocol specification to represent fractional decimal values. The following table describes the value for each bit as a reciprocal power of two (2), that is two (2) raised to the power of a negative exponent (-1 through -15). Bit 15 is the Sign Bit.

| Bit            | 15   | 14              | 13              | 12              | 11              | 10              | 9               | 8               | 7               | 6               | 5                | 4                | 3                | 2                | 1                | 0                |
|----------------|------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|------------------|------------------|------------------|------------------|------------------|------------------|
| Value          |      |                 |                 |                 | 800             | 400             | 200             | 100             | 80              | 40              | 20               | 10               |                  |                  |                  |                  |
| Hex(h)         | Sign | 4000h           | 2000h           | 1000h           | h               | h               | h               | h               | h               | h               | h                | h                | 8h               | 4h               | 2h               | 1h               |
| Decimal        |      | 2 <sup>-1</sup> | 2 <sup>-2</sup> | 2 <sup>-3</sup> | 2 <sup>-4</sup> | 2 <sup>-5</sup> | 2 <sup>-6</sup> | 2 <sup>-7</sup> | 2 <sup>-8</sup> | 2 <sup>-9</sup> | 2 <sup>-10</sup> | 2 <sup>-11</sup> | 2 <sup>-12</sup> | 2 <sup>-13</sup> | 2 <sup>-14</sup> | 2 <sup>-15</sup> |
| 4000h          | 0    | 1               | 0               | 0               | 0               | 0               | 0               | 0               | 0               | 0               | 0                | 0                | 0                | 0                | 0                | 0                |
| 0.5            |      |                 |                 |                 |                 |                 |                 |                 |                 |                 |                  |                  |                  |                  |                  |                  |
| 2000h          | 0    | 0               | 1               | 0               | 0               | 0               | 0               | 0               | 0               | 0               | 0                | 0                | 0                | 0                | 0                | 0                |
| 0.25           |      |                 |                 |                 |                 |                 |                 |                 |                 |                 |                  |                  |                  |                  |                  |                  |
| 1000h          | 0    | 0               | 0               | 1               | 0               | 0               | 0               | 0               | 0               | 0               | 0                | 0                | 0                | 0                | 0                | 0                |
| 0.125          |      |                 |                 |                 |                 |                 |                 |                 |                 |                 |                  |                  |                  |                  |                  |                  |
| 6000h          | 0    | 1               | 1               | 0               | 0               | 0               | 0               | 0               | 0               | 0               | 0                | 0                | 0                | 0                | 0                | 0                |
| 0.75           |      |                 |                 |                 |                 |                 |                 |                 |                 |                 |                  |                  |                  |                  |                  |                  |
| 3210h          | 0    | 0               | 1               | 1               | 0               | 0               | 1               | 0               | 0               | 0               | 0                | 1                | 0                | 0                | 0                | 0                |
| 0.395751953125 |      |                 |                 |                 |                 |                 |                 |                 |                 |                 |                  |                  |                  |                  |                  |                  |

**Examples:**

A value of 4000hex (only Bit 14 set, all others clear) is interpreted as 0.5 decimal

A value of 2000hex (only Bit 13 set, all others clear) is interpreted as 0.25 decimal

A value of 1000hex (only Bit 12 set, all others clear) is interpreted as 0.125 decimal

... and so on until...

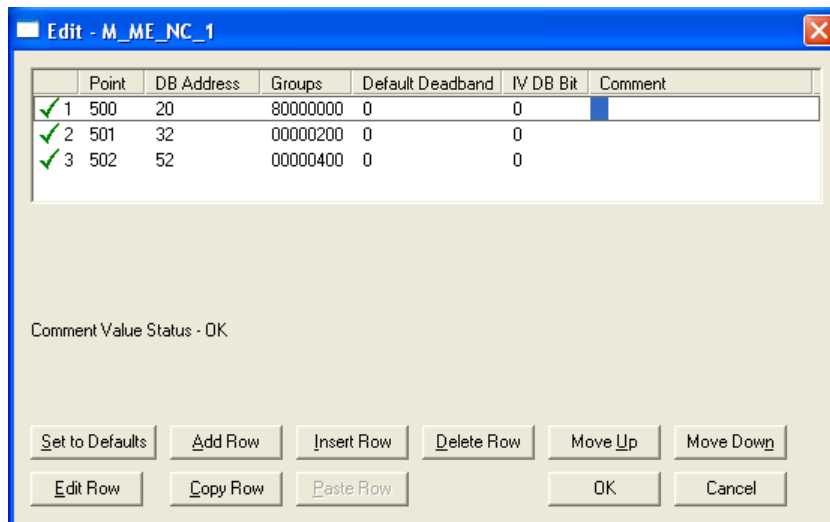
A value of 0001hex (Only Bit 0 set, all others clear) is interpreted as 0.000030517578125

Therefore, the actual data values transmitted may be any combination of the decimal values for any given bit pattern.

M ME NC and M IT NA

The *Monitor Short Floating-Point Measured Value* and *Monitored Integrated Totals* information objects occupy two words with double-word addressing.

For example, if you configured the following points:



The following table describes how these points would be stored in the gateway.

| Inf. Object Address | Module Database Address                   |
|---------------------|---|
| 500                 | Words 40 and 41 (Double-word address 20)  |
| 501                 | Words 64 and 65 (Double-word address 32)  |
| 502                 | Word 104 and 105 (Double-word address 52) |

### 5.2.5 Using Control (Command) Points

The 101S gateway supports the following Control points:

| ASDU Type | Information Object Description                       |
|-----------|--|
| C_SC_NA   | Control Single Command                               |
| C_DC_NA   | Control Double Command                               |
| C_RC_NA   | Control Regulating Step Command                      |
| C_BO_NA   | Control 32-Bit Bitstring Command                     |
| C_SE_NA   | Control Normalized Value Set Point Command           |
| C_SE_NB   | Control Scaled Value Set Point Command               |
| C_SE_NC   | Control Short Floating-Point Value Set Point Command |

Each Control point is identified by its Information Object Address. For each Control point, configure the following parameters:

*Point #* - This is the Information Object Address of the point. It identifies the point in the network. This address must be unique for each Common ASDU Address in the network.

*DB Address* - This is the database location in the 101S gateway associated with the point.

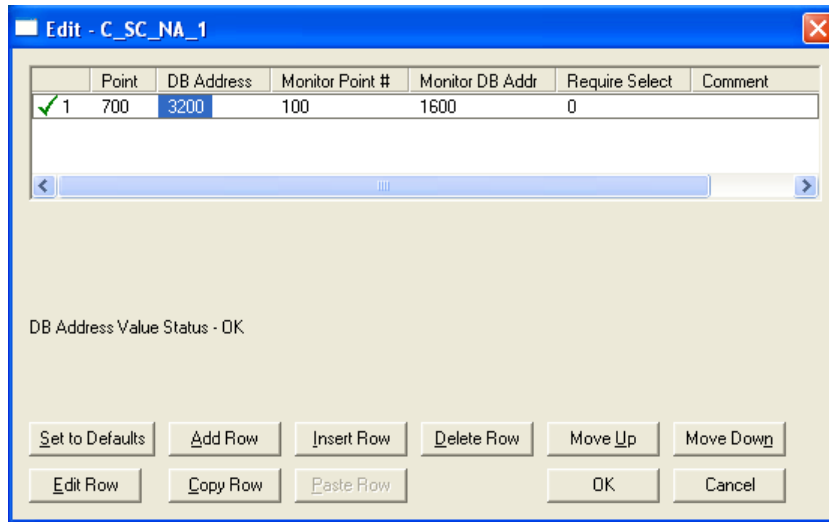
*Monitor Point #* - This is the Information Object Address of the associated Monitor point to be sent to the Client once the Control request is received.

*Monitor DB Address* - The user might (optionally) configure a Monitor point to be sent by the 101S gateway when it receives the command for that specific point.

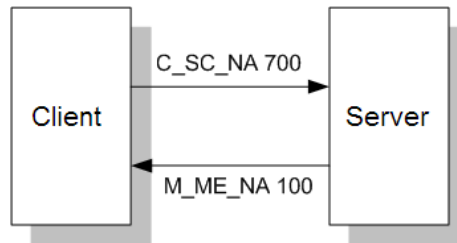
*Require Select* - This parameter configures the point to require a Select request before the Operate command.



### Example (C\_SC\_NA)



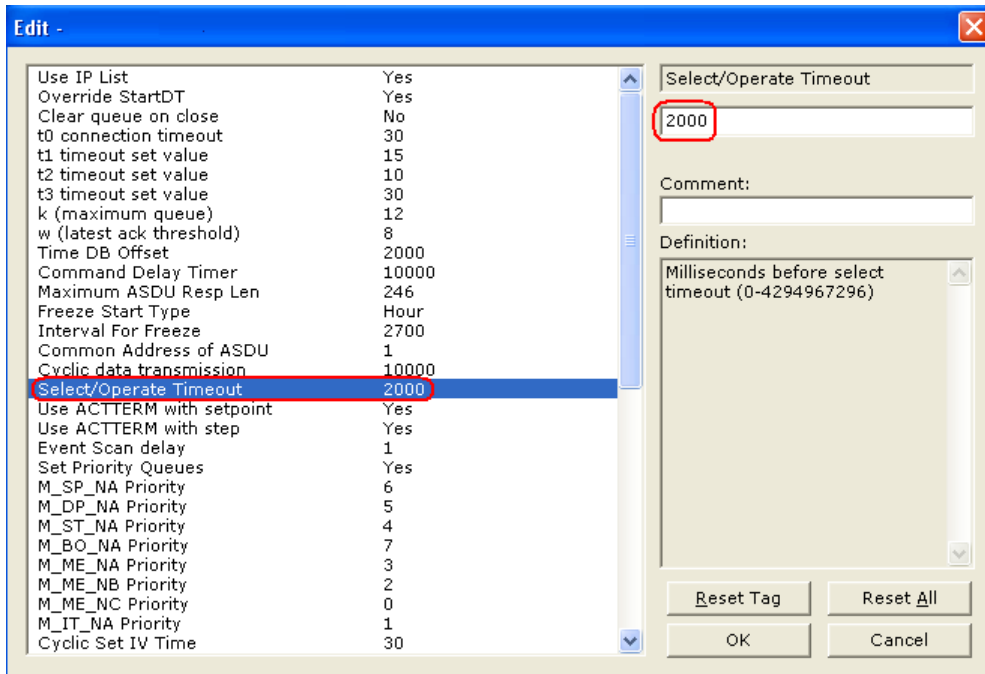
In the example above, each time the gateway receives a value of one (1) in *Control Single Command* 700, it will send a response by placing a value of one (1) in *Monitor Single-Point #* 100 at database bit address 1600.



Some of the Command points may be configured to require a Select command before they will act on an Execute command.

### Select/Operate Timeout Parameter

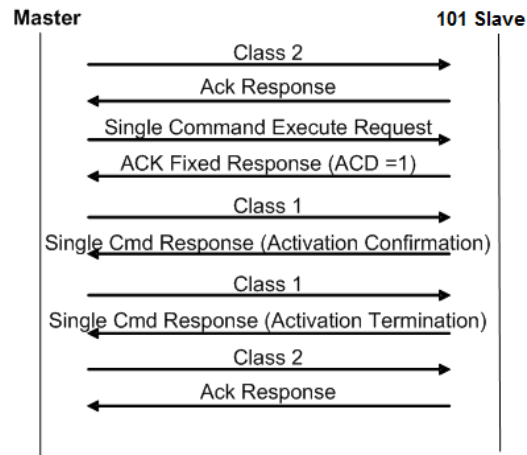
Use the *Select/Operate Timeout* parameter to configure the select/operate timeout period. After the gateway receives the Select command, it will wait for this period of time to receive the Execute command for the same point address. If the gateway does not receive an Execute command within this period of time, it will require another Select operation before the Execute operation will be accepted and acted upon.



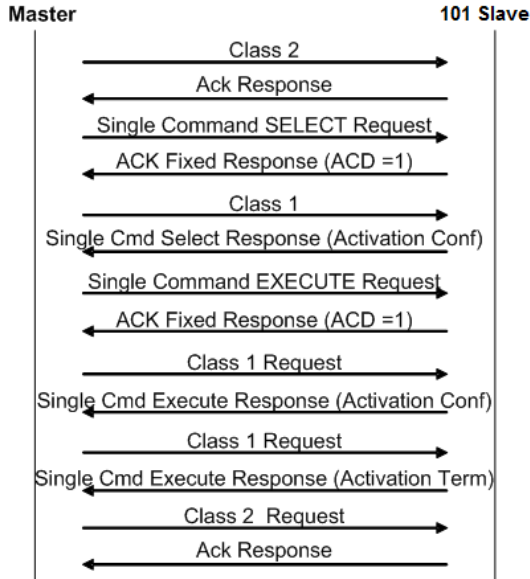
Control Data Transfer

The control communication typically occurs when the Master sends a command request to update the gateway's command points.

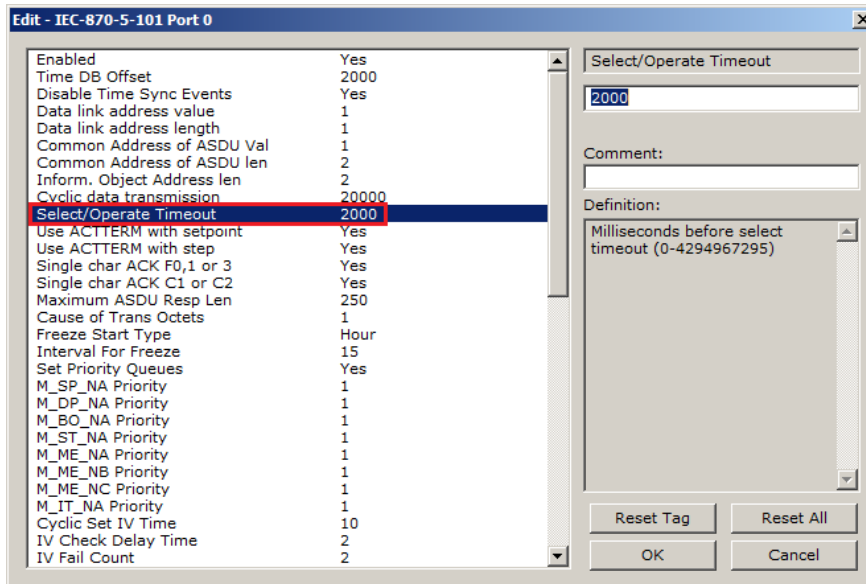
The following illustration shows a typical command communication between the Master and gateway during unbalanced operation.



Some of the command points may be configured to be selected before executed. The following illustration shows a typical command communication between the Master and gateway operation with a SELECT operation.



Refer to the following parameter to configure the select/operate timeout period. After the gateway receives the SELECT operation it will wait for this period of time for the EXECUTE operation. If the gateway does not receive an EXECUTE operation within this period of time it will require another SELECT operation before the EXECUTE operation.



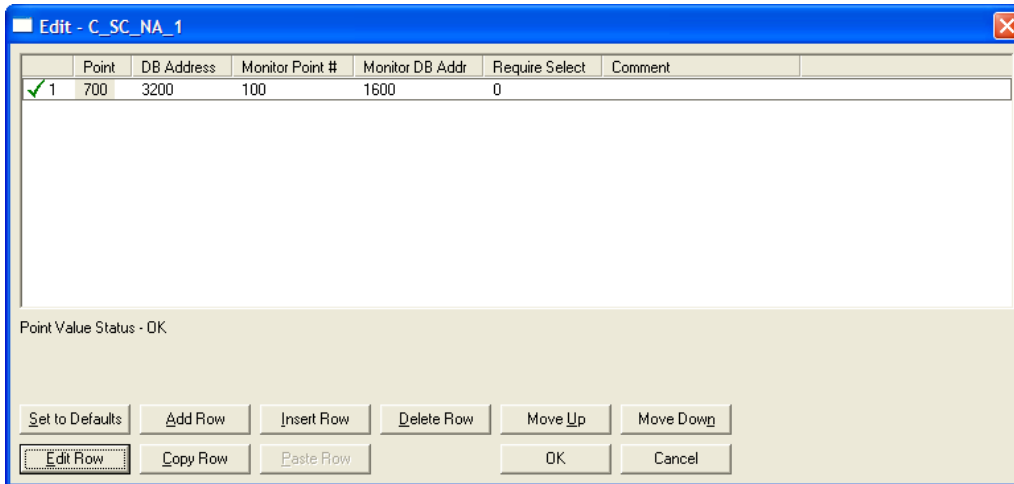
### Control Points Addressing

Control points must be configured in a database area that is updated in the gateway. You must associate each point to a database address in the 101S gateway. The interpretation of the *DB Address* parameter in the configuration tables depends on the point type configured and the type of addressing associated with that point type.

| <b>ASDU Type</b> | <b>Data Size</b> | <b>Addressing Type</b> |
|------------------|------------------|------------------------|
| C_SC_NA          | 1 bit            | Bit                    |
| C_DC_NA          | 2 bits           | Bit                    |
| C_RC_NA          | 1 byte           | Byte                   |
| C_BO_NA          | 2 words          | Double word            |
| C_SE_NA          | 1 word           | Word                   |
| C_SE_NB          | 1 word           | Word                   |
| C_SE_NC          | 2 words          | Double word            |

### C\_SC\_NA and C\_DC\_NA

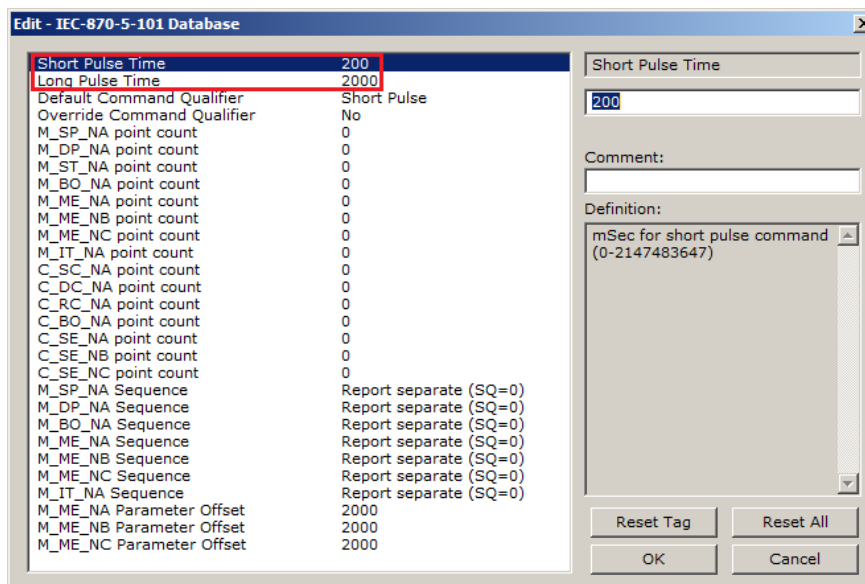
The single-point command and dual-point command points use one bit with bit-addressing. For example, if you configure the following points:



The following table describes how these points would be used.

| Inf. Object Address | Module Database Address |
|---------------------|-------------------------|
| 100                 | Bit 0 of word 100       |
| 101                 | Bit 1 of word 100       |
| 102                 | Bit 2 of word 100       |

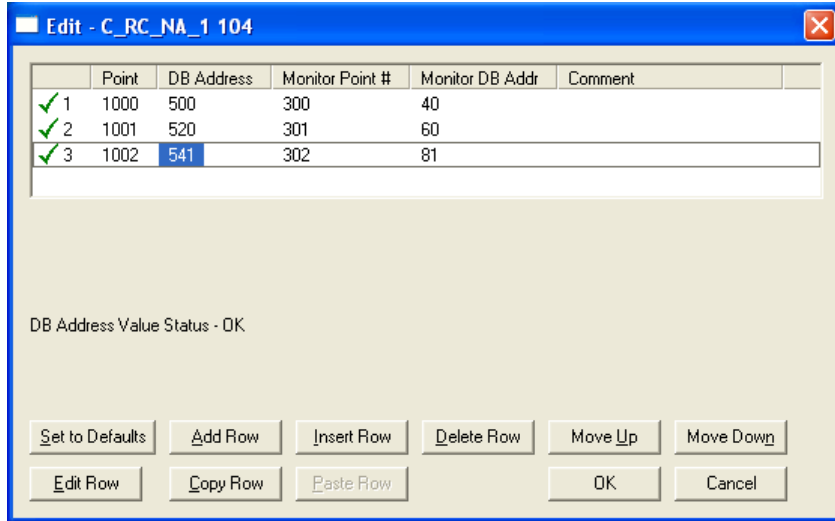
The protocol specification defines a qualifier value that is set by the Master to determine the duration of the pulse (short, long or persistent). Configure the *Short Pulse Time* and *Long Pulse Time* parameters, seen below, to set the duration of the short and long pulses:



### C\_RC\_NA

A *Control Regulating Step Command* information object occupies one byte and uses byte addressing.

For example, if you configured the following points:



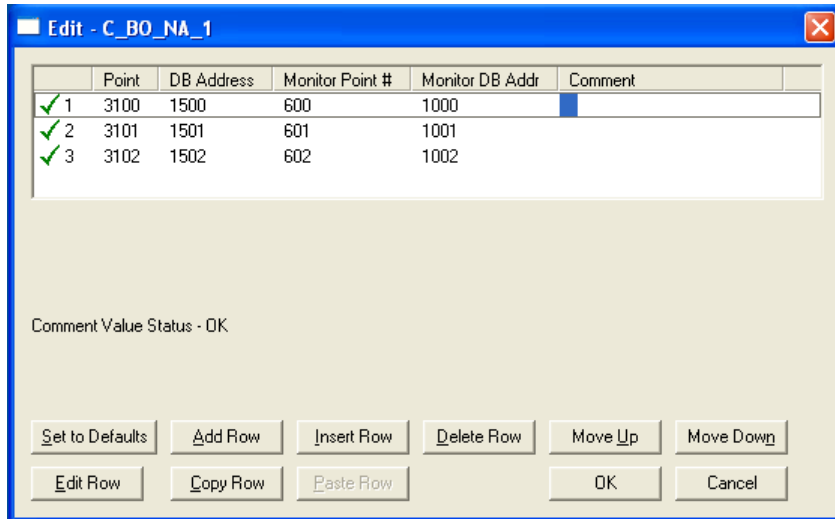
The following table describes how these points would be used.

| Inf. Object Address | Module Database Address   |
|---------------------|---|
| 1000                | Low Byte of word 250 (Byte address 500)   |
| 300                 | Low Byte of word 20 (Byte address 40. Must match configuration of point 300 in <i>M_ST_NA</i> .)  |
| 1001                | Low Byte of word 260 (Byte address 520)   |
| 301                 | Low Byte of word 30 (Byte address 60. Must match configuration of point 301 in <i>M_ST_NA</i> .)  |
| 1002                | High Byte of word 270 (Byte address 541)  |
| 302                 | High Byte of word 40 (Byte address 81. Must match configuration of point 302 in <i>M_ST_NA</i> .) |

### C\_BO\_NA

A *Control 32-Bit Bitstring Command* information object occupies two words and uses double-word addressing.

For example, if you configured the following points:



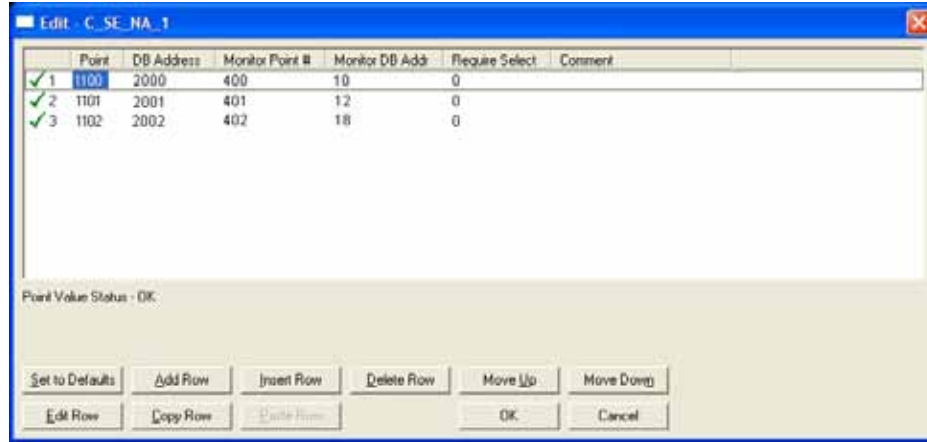
These points would be used as follows:

| Inf. Object Address | Module Database Address   |
|---------------------|---|
| 3100                | Words 3000 and 3001 (Double-word address 1500)  |
| 600                 | Words 2000 and 2001 (Double-word address 1000. Must match the configuration of point 600 in <i>M_BO_NA</i> .) |
| 3101                | Words 3002 and 3003 (Double-word address 1501)  |
| 601                 | Words 2002 and 2001 (Double-word address 1001. Must match the configuration of point 601 in <i>M_BO_NA</i> .) |
| 3102                | Words 3003 and 3004 (Double-word address 1502)  |
| 602                 | Words 2004 and 2005 (Double-word address 1002. Must match the configuration of point 602 in <i>M_BO_NA</i> .) |



### C\_SE\_NA and C\_SE\_NB

The *Control Normalized Value Set Point Command* information object and the *Control Scaled Value Set Point Command* information object use one word with word addressing. For example, if you configured the following points:



The following table describes how these points would be used.

| Inf. Object Address | Module Database Address  |
|---------------------|--|
| 1100                | Word 2000 (Word address 2000)  |
| 400                 | Word 10 (Word address 10. Must match the configuration of point 400 in <i>M_ME_NA</i> .) |
| 1101                | Word 2001 (Word address 2001)  |
| 401                 | Word 12 (Word address 12. Must match the configuration of point 401 in <i>M_ME_NA</i> .) |
| 1102                | Word 2002 (Word address 2002)  |
| 402                 | Word 18 (Word address 18. Must match the configuration of point 402 in <i>M_ME_NA</i> .) |

The *Control Normalized Measured* information objects use a data representation defined by the protocol specification to represent fractional decimal values. The following table describes the value for each bit as a reciprocal power of two (2), that is two (2) raised to the power of a negative exponent (-1 through -15). Bit 15 is the Sign Bit.

| Bit            | 15   | 14              | 13              | 12              | 11              | 10              | 9               | 8               | 7               | 6               | 5                | 4                | 3                | 2                | 1                | 0                |
|----------------|------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|------------------|------------------|------------------|------------------|------------------|------------------|
| Value          |      |                 |                 |                 | 800             | 400             | 200             | 100             | 80              | 40              | 20               | 10               | 8h               | 4h               | 2h               | 1h               |
| Hex(h)         | Sign | 4000h           | 2000h           | 1000h           | h               | h               | h               | h               | h               | h               | h                | h                | 2 <sup>-12</sup> | 2 <sup>-13</sup> | 2 <sup>-14</sup> | 2 <sup>-15</sup> |
| Decimal        |      | 2 <sup>-1</sup> | 2 <sup>-2</sup> | 2 <sup>-3</sup> | 2 <sup>-4</sup> | 2 <sup>-5</sup> | 2 <sup>-6</sup> | 2 <sup>-7</sup> | 2 <sup>-8</sup> | 2 <sup>-9</sup> | 2 <sup>-10</sup> | 2 <sup>-11</sup> |                  |                  |                  |                  |
| 4000h          | 0    | 1               | 0               | 0               | 0               | 0               | 0               | 0               | 0               | 0               | 0                | 0                | 0                | 0                | 0                | 0                |
| 0.5            |      |                 |                 |                 |                 |                 |                 |                 |                 |                 |                  |                  |                  |                  |                  |                  |
| 2000h          | 0    | 0               | 1               | 0               | 0               | 0               | 0               | 0               | 0               | 0               | 0                | 0                | 0                | 0                | 0                | 0                |
| 0.25           |      |                 |                 |                 |                 |                 |                 |                 |                 |                 |                  |                  |                  |                  |                  |                  |
| 1000h          | 0    | 0               | 0               | 1               | 0               | 0               | 0               | 0               | 0               | 0               | 0                | 0                | 0                | 0                | 0                | 0                |
| 0.125          |      |                 |                 |                 |                 |                 |                 |                 |                 |                 |                  |                  |                  |                  |                  |                  |
| 6000h          | 0    | 1               | 1               | 0               | 0               | 0               | 0               | 0               | 0               | 0               | 0                | 0                | 0                | 0                | 0                | 0                |
| 0.75           |      |                 |                 |                 |                 |                 |                 |                 |                 |                 |                  |                  |                  |                  |                  |                  |
| 3210h          | 0    | 0               | 1               | 1               | 0               | 0               | 1               | 0               | 0               | 0               | 0                | 1                | 0                | 0                | 0                | 0                |
| 0.395751953125 |      |                 |                 |                 |                 |                 |                 |                 |                 |                 |                  |                  |                  |                  |                  |                  |

**Examples:**

A value of 4000hex (only Bit 14 set, all others clear) is interpreted as 0.5 decimal

A value of 2000hex (only Bit 13 set, all others clear) is interpreted as 0.25 decimal

A value of 1000hex (only Bit 12 set, all others clear) is interpreted as 0.125 decimal

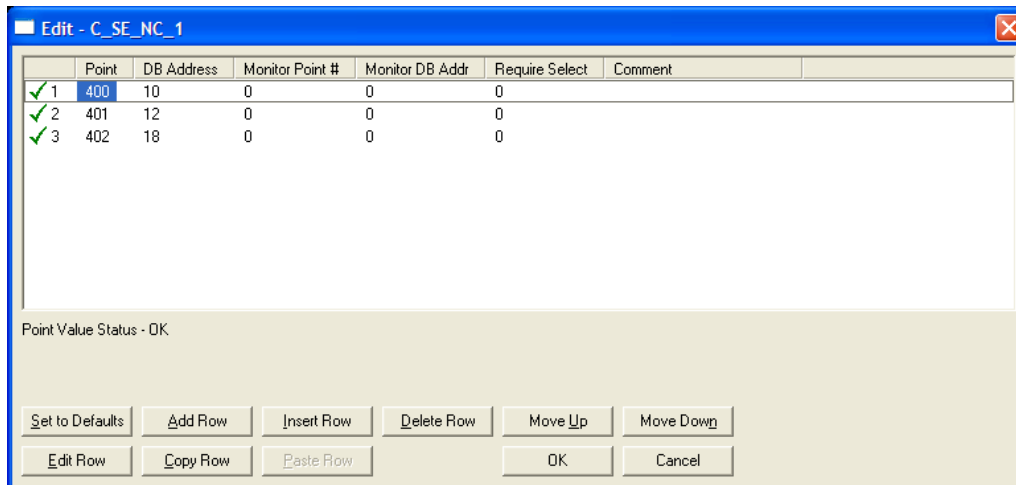
... and so on until...

A value of 0001hex (Only Bit 0 set, all others clear) is interpreted as 0.000030517578125

Therefore, the actual data values transmitted may be any combination of the decimal values for any given bit pattern.

**C\_SE\_NC**

The measured short floating point command uses two words with double-word addressing. For example, if you configured the following points:

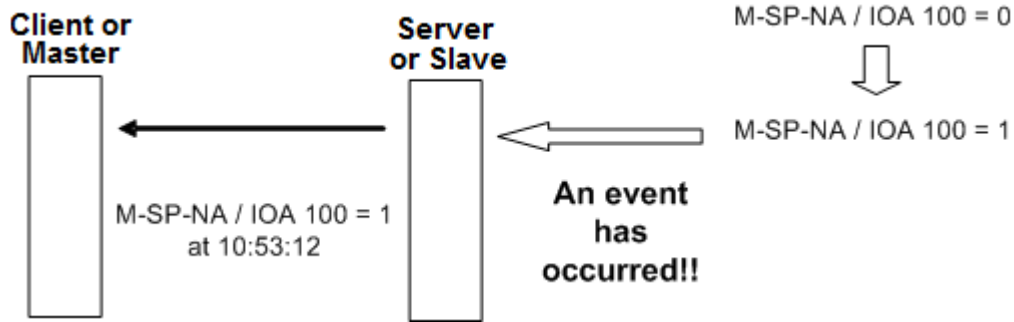


The following table describes how these points would be used.

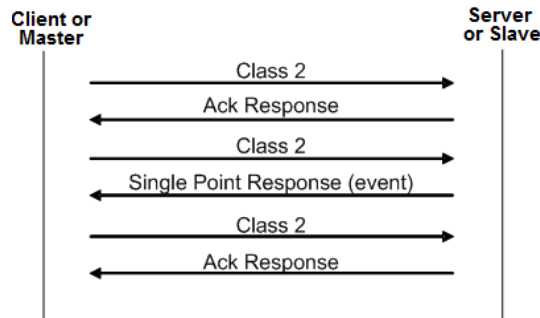
| Inf. Object Address | 101S v3 gateway Database Address |
|---------------------|----------------------------------|
| 400                 | Words 5 and 6                    |
| 401                 | Words 6 and 7                    |
| 402                 | Words 9 and 10                   |

### 5.2.6 Events

In order to improve the communication efficiency, most applications will require the Master to periodically poll for data changes with a higher priority than polling for monitor data. Every time data change occurs, the slave should send this information, typically with the date and time information on when the data change occurred.



The following illustration shows the event communication between the Master and the gateway during unbalanced mode:



The gateway can queue up to 99 events per data type. When the queue is full, the oldest event will be deleted each time a new event is added to the queue. You must configure the Master to poll the event queue frequently enough to avoid losing events.

The events can be returned for Class 1 or Class 2 requests according to the data type as follows:

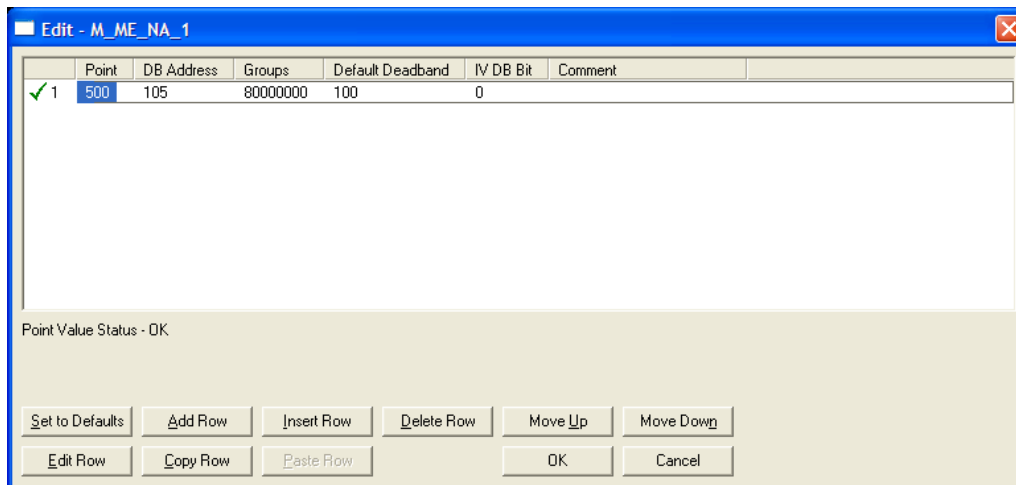
| Data Type | Class Request |
|-----------|---------------|
| M_SP_NA   | Class 1       |
| M_DP_NA   | Class 1       |
| M_ST_NA   | Class 1       |
| M_BO_NA   | Class 1       |
| M_ME_NA   | Class 2       |
| M_ME_NB   | Class 2       |
| M_ME_NC   | Class 2       |
| M_IT_NA   | Class 1       |

**Note:** In response to a Class 2 poll the gateway may respond with Class 1 data when there is no Class 2 data available. So eventually the events for M\_SP\_NA, M\_DP\_NA and M\_ST\_NA points may also be sent during a Class 2 response.

Deadbands

The Monitored Measured points (M\_ME\_NA and M\_ME\_NB) will generate events only if the data value changes an amount greater than or equal to the configured deadband value.

For example, if the following point is configured:



Then, if the current value for this point is 130, it would generate events only if:

- NEW VALUE is less than or equal to 30
- OR
- NEW VALUE is greater than or equal to 230.

Each monitored measured point can have its own deadband value assigned to it in the configuration file or by using the PCB configuration options.

A network Master may also dynamically change the deadband for each monitored point. The Master may send one of the following commands:

| Type | Command  |
|------|--|
| 110  | Parameter of Measured Normalized Data (M_ME_NA)      |
| 111  | Parameter of Measured Scaled Data (M_ME_NB)          |
| 112  | Parameter of Measured Short Floating Point (M_ME_NC) |

The protocol specification explains that the qualifier value for these commands should be configured as:

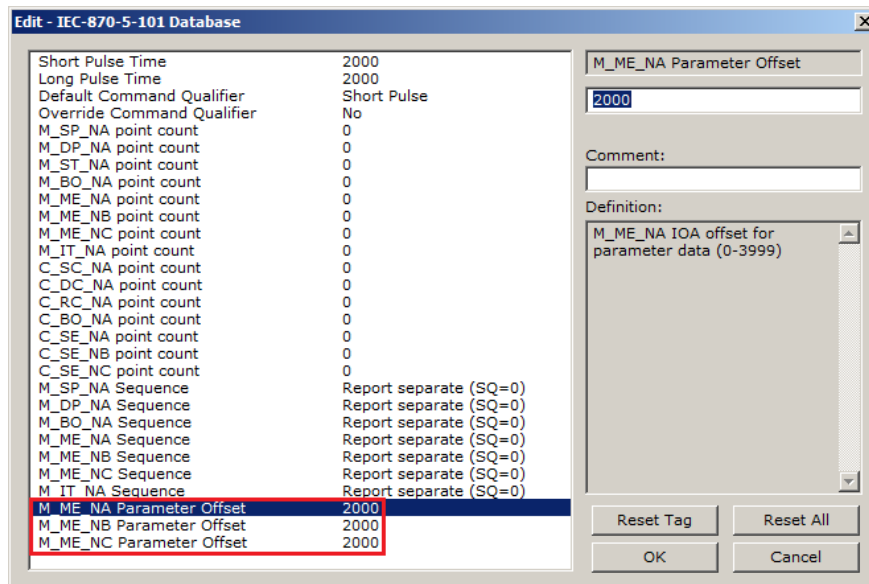
| Bits   | Value | Description   |
|--------|-------|---|
| 1 to 6 | 0     | Not Used  |
|        | 1     | Threshold Value (Deadband)                              |
|        | 2     | Smoothing Factor (filter time constant) - Not Supported |
|        | 3     | Low Limit Transmission of Measured Value                |
|        | 4     | High Limit Transmission of Measured Value               |
|        | 5..31 | Reserved  |
| 7      | 0     | No Change   |
|        | 1     | Change  |
| 8      | 0     | Operation   |
|        | 1     | Not in Operation  |

The gateway calculates the Low Limit and High Limit values using the formula below (so these values cannot be changed dynamically)

Low Limit = (LAST REPORTED VALUE) - Deadband

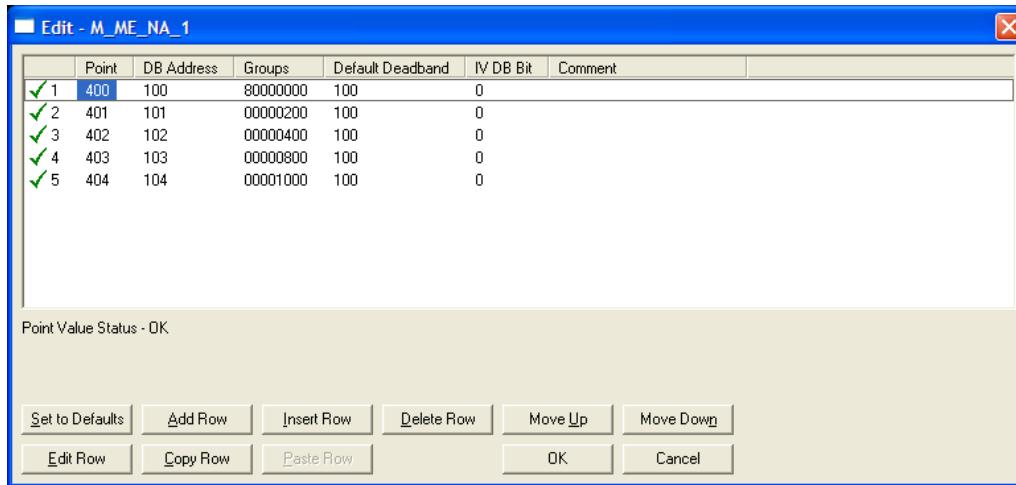
High Limit = (LAST REPORTED VALUE) + Deadband

These commands must be sent to a specific Information Object Address. The gateway associates each monitor measured point with a parameter point through the following configuration parameters:



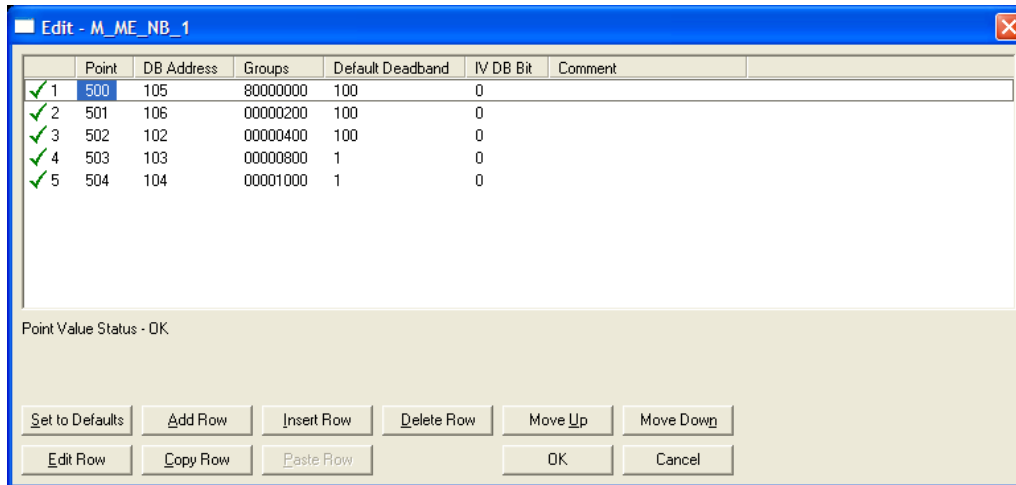
**Example - Parameter Data for Monitored Points**

If the following monitored measured points are configured:



The screenshot shows a dialog box titled "Edit - M\_ME\_NA\_1" with a table of five rows. Each row has a checkmark in the first column. The columns are: Point, DB Address, Groups, Default Deadband, IV DB Bit, and Comment. Below the table, it says "Point Value Status - OK" and there are buttons for "Set to Defaults", "Add Row", "Insert Row", "Delete Row", "Move Up", "Move Down", "Edit Row", "Copy Row", "Paste Row", "OK", and "Cancel".

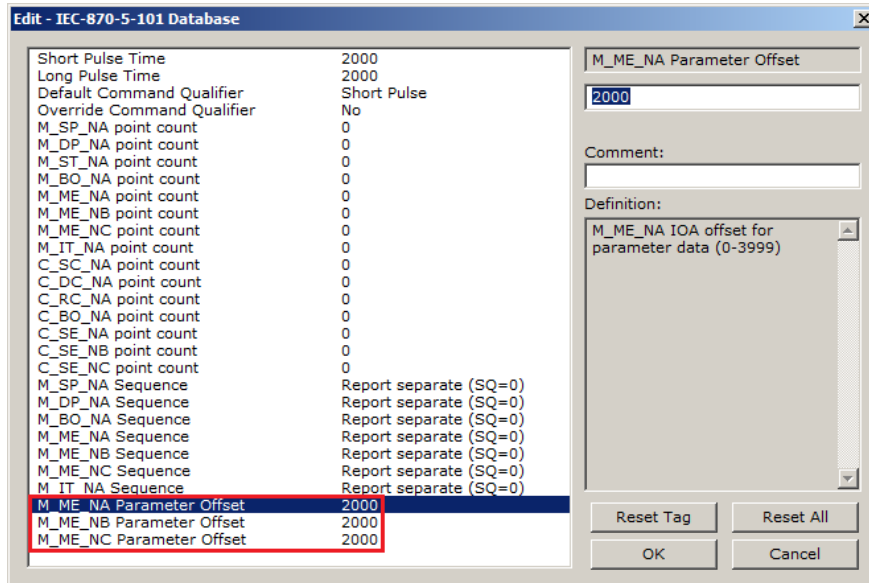
|     | Point | DB Address | Groups   | Default Deadband | IV DB Bit | Comment |
|-----|-------|------------|----------|------------------|-----------|---------|
| ✓ 1 | 400   | 100        | 80000000 | 100              | 0         |         |
| ✓ 2 | 401   | 101        | 00000200 | 100              | 0         |         |
| ✓ 3 | 402   | 102        | 00000400 | 100              | 0         |         |
| ✓ 4 | 403   | 103        | 00000800 | 100              | 0         |         |
| ✓ 5 | 404   | 104        | 00001000 | 100              | 0         |         |



The screenshot shows a dialog box titled "Edit - M\_ME\_NB\_1" with a table of five rows. Each row has a checkmark in the first column. The columns are: Point, DB Address, Groups, Default Deadband, IV DB Bit, and Comment. Below the table, it says "Point Value Status - OK" and there are buttons for "Set to Defaults", "Add Row", "Insert Row", "Delete Row", "Move Up", "Move Down", "Edit Row", "Copy Row", "Paste Row", "OK", and "Cancel".

|     | Point | DB Address | Groups   | Default Deadband | IV DB Bit | Comment |
|-----|-------|------------|----------|------------------|-----------|---------|
| ✓ 1 | 500   | 105        | 80000000 | 100              | 0         |         |
| ✓ 2 | 501   | 106        | 00000200 | 100              | 0         |         |
| ✓ 3 | 502   | 102        | 00000400 | 100              | 0         |         |
| ✓ 4 | 503   | 103        | 00000800 | 1                | 0         |         |
| ✓ 5 | 504   | 104        | 00001000 | 1                | 0         |         |

And if the parameter offsets are configured as follows:

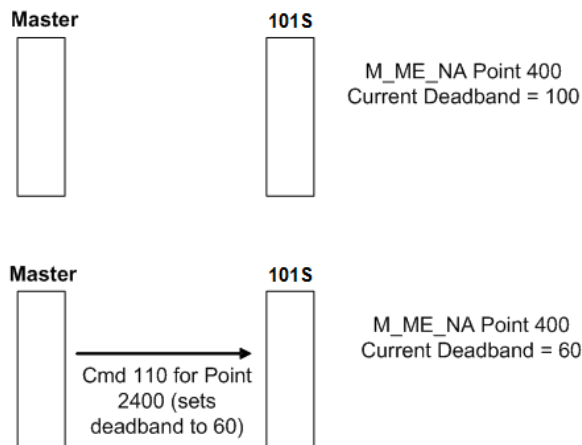


It would imply that the parameter points would be configured as follows:

### M\_ME\_NA

| Monitored | Measured | Normalized | Point | Associated Parameter Point |
|-----------|----------|------------|-------|----------------------------|
|           |          |            | 400   | 2400                       |
|           |          |            | 401   | 2401                       |
|           |          |            | 402   | 2402                       |
|           |          |            | 403   | 2403                       |
|           |          |            | 404   | 2404                       |

So, in order to send change the deadband for the M\_ME\_NA point 400, the Master would send a command type 110 to point 2400:



**M\_ME\_NB**

| Monitored Measured Normalized Point | Associated Parameter Point |
|-------------------------------------|----------------------------|
| 500                                 | 2500                       |
| 501                                 | 2501                       |
| 502                                 | 2502                       |
| 503                                 | 2503                       |
| 504                                 | 2504                       |

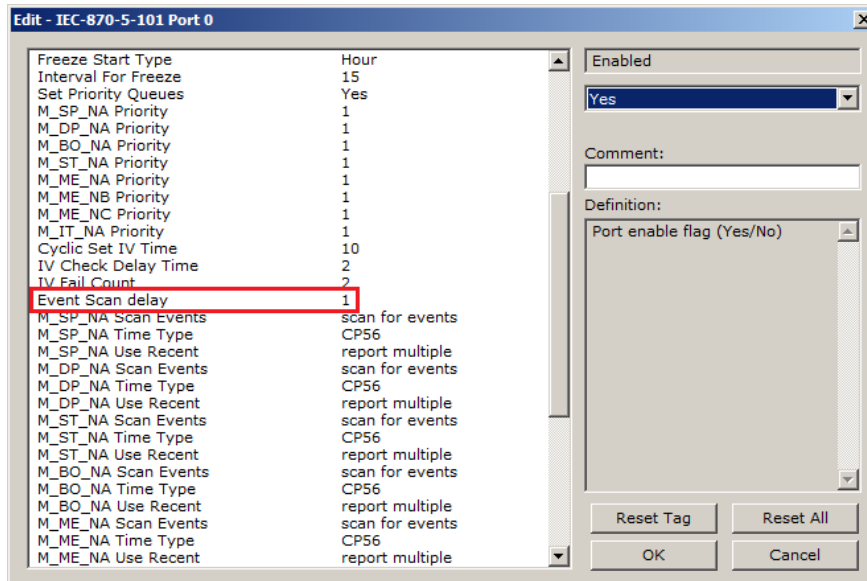
Controlling the Generation of Events

Some applications may require that only some points should generate events. Other applications may require that all configured points should generate events. The gateway offers considerable flexibility for event control. You can control how events will be generated at 3 different levels:

- 1 Application Level (all configured points)
- 2 Data Type Level (all configured points of specific ASDU type)
- 3 Point Level (each individual point)

**General Events (All Points)**

You can control how frequently the gateway will scan the database for events using the following configuration parameter:

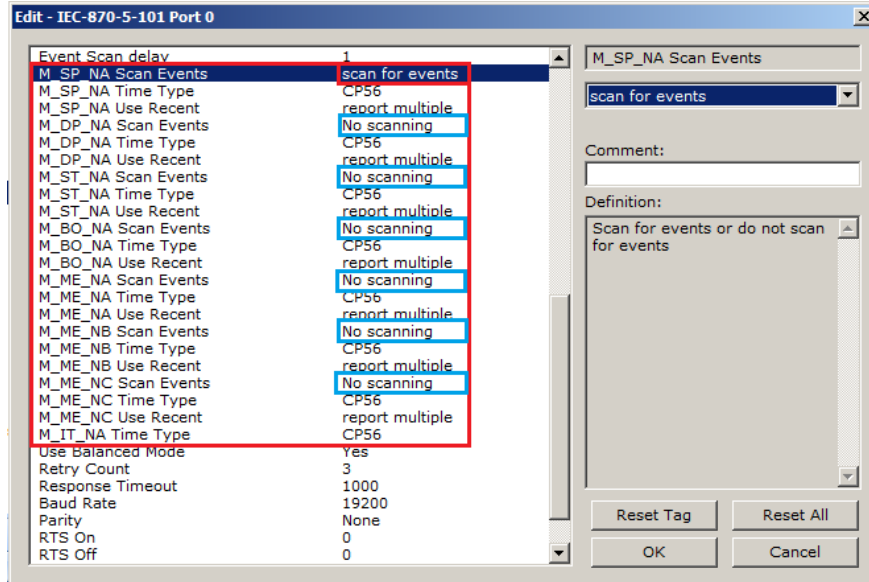


If this parameter is set to 0, the gateway will not generate events for any points. A non-zero value will configure how frequently the gateway will scan for events in the database.



### Data Type Level

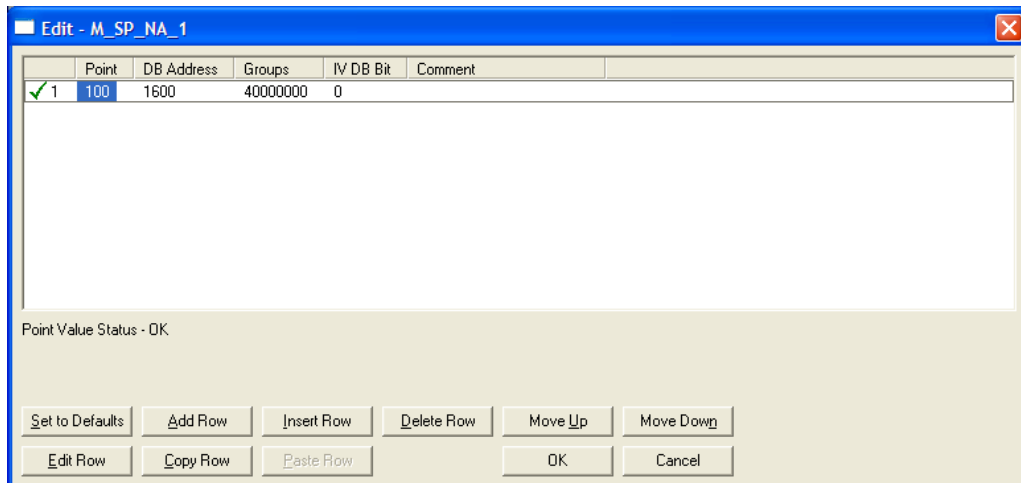
You can configure whether a data type should generate events. Each data type has a configuration parameter to control the generation of events:



In the example above, only the M\_SP\_NA points would generate events.

### Point Level

You can configure whether each point should generate events using the Group field for each point configuration. To disable event generation for a specific port, set the value to 40000000.



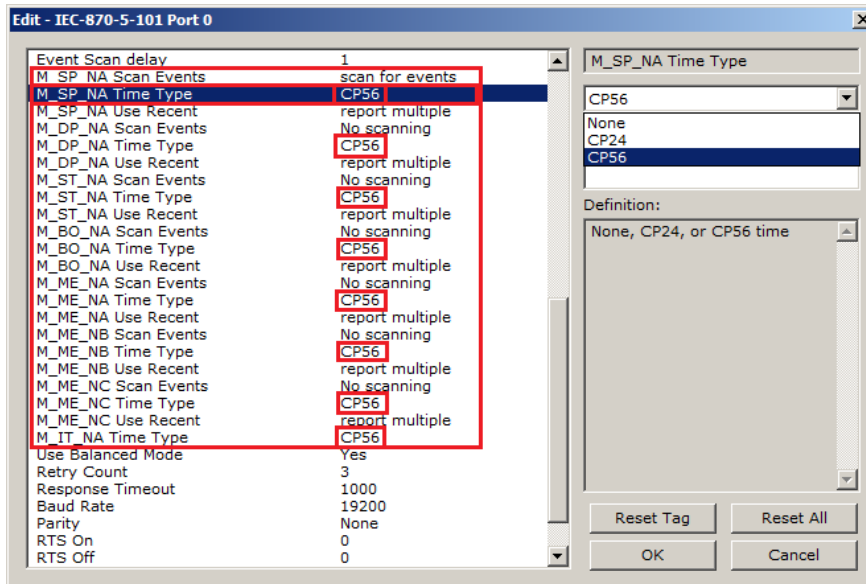
Time Information

Each event may also carry the date and time information when it occurred. The gateway supports the CP56 and CP24 time formats (as defined in the protocol specification).

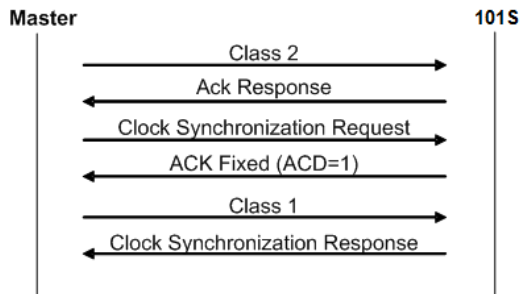
The CP56 time format contains the milliseconds, seconds, minute, hour, day, month and year when the event has occurred.

The CP24 time format contains the milliseconds and minutes when the event has occurred.

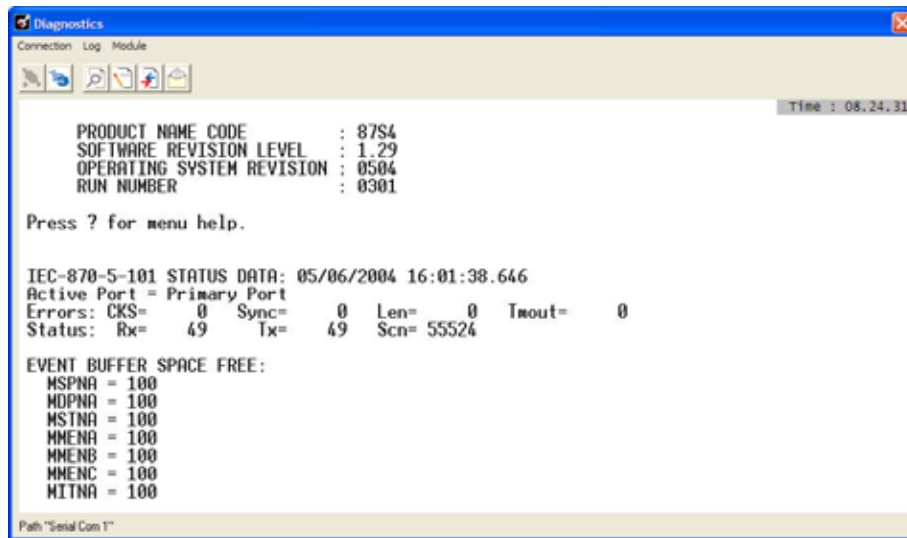
The gateway may also be configured to not send any time information with each event for certain data types. The following *Time Type* parameters may be used to control the time information for each data type:



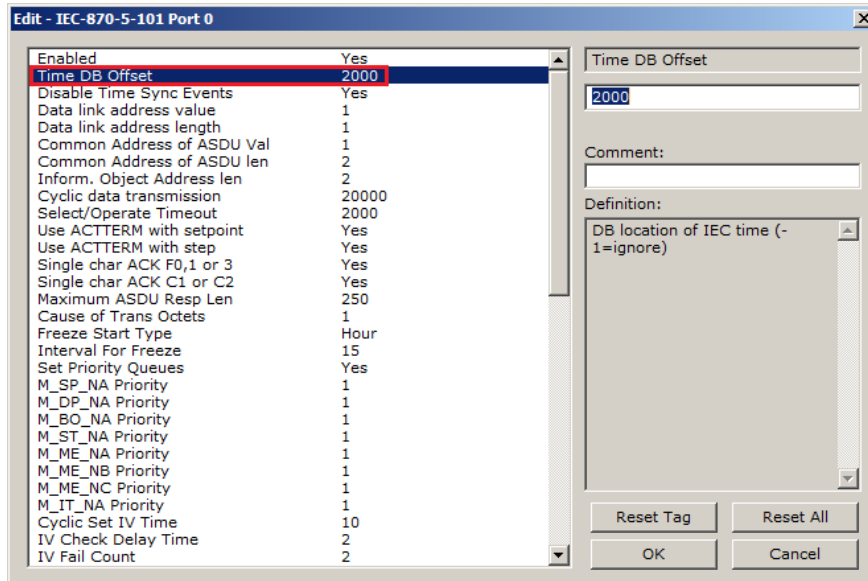
The Master should periodically send a Time Synchronization command to the gateway driver to synchronize its date and time information. This is a very important step in order to make sure that the Master and the slave are both using the same time information.



You can check the current date and time using the debug menu. Press **[E]** at the main menu to view the gateway Status Data screen:



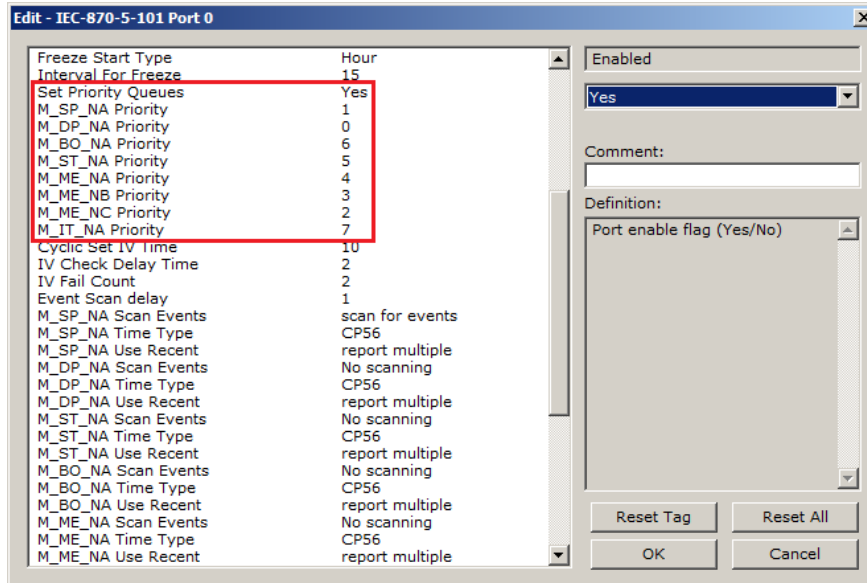
You can also configure the gateway to copy the date and time information to the gateway database with the *Time DB Offset* parameter:



As the gateway sends events to the Master (serial gateways) or Client (Ethernet gateways), it also sends a Clock Synchronization response (spontaneous COT) every hour for the transmission of the clock time to the Master (serial gateways) or Client (Ethernet gateways).

### Event Priority

Event Priority permits the user to set reporting priorities for data change events generated for each ASDU data type. The configuration file contains the following parameters to support this feature:



The *Set Priority Queues* parameter must be set to **YES** for this feature to be used. Each of the Monitored Point ASDUs must be assigned a unique priority index from 0 to 7. If Event Priority is enabled, whenever events of an ASDU with a lower priority number are present, they will always be reported before events from any other ASDUs with higher priority numbers. Multiple events will be assembled into each message packet for transmission. This methodology limits the likelihood of a buffer overflow and still maximizes the bandwidth usage on the communication channel.

**Example - Event Priority**

If the gateway is configured with the example values above, and the event queue contains the events generated in the following order:

| Event Order | ASDU    |
|-------------|---------|
| 1           | M_SP_NA |
| 2           | M_SP_NA |
| 3           | M_DP_NA |
| 4           | M_ST_NA |
| 5           | M_DP_NA |
| 6           | M_SP_NA |

Then, the gateway will respond to a *Class One* data request from the controlling station by returning the data in the event queues in the order shown in the following table.

| Packet Order | Content                   |
|--------------|---------------------------|
| 1            | M_DP_NA events 3 and 5    |
| 2            | M_SP_NA events 1, 2 and 6 |
| 3            | M_ST_NA event 4           |

**WARNING:** Events from the higher priority (lower-numbered) queues are always reported first, when present, before lower priority (higher-numbered) queues. If careful planning is not exercised when deciding how to configure Event Priority queues, or if large volumes of events occur events in the lower queues may be lost due to buffer overflow.

When using Event Priority, events from the highest priority queues, when present, will always be reported before events from lower priority queue. Therefore, if large volumes of events are generated, events in the lower priority queues may be lost due to event buffer overflows.

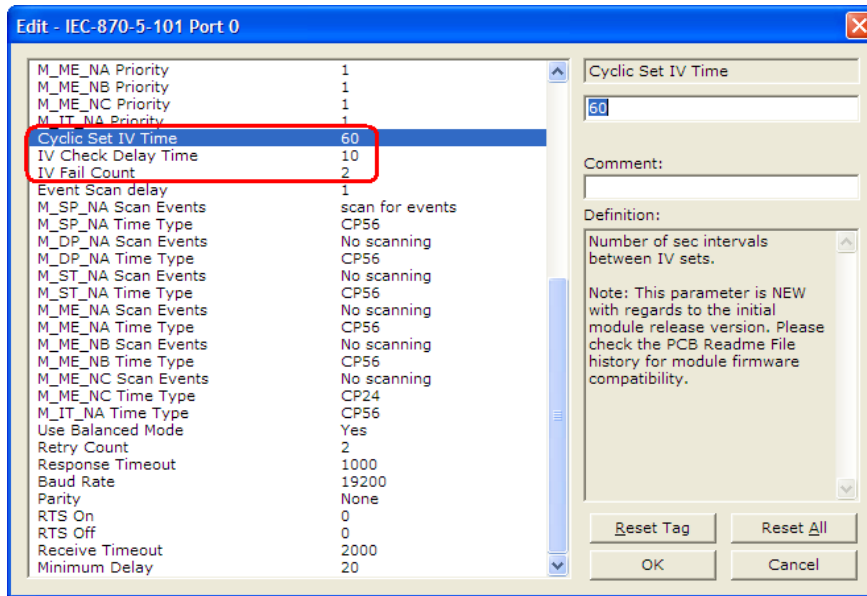
If the *Set Priority Queue* parameter is set to **NO**, this feature is not utilized. Events will be reported based on their order of occurrence, as determined by the event's timestamp. Multiple events will be assembled into message packets for transmission. This methodology limits the likelihood of a buffer overflow and still maximizes the bandwidth usage on the communication channel.

**Invalid Bit Monitoring**

This feature allows the application to set the invalid (IV) quality bit of data points for all supported Monitored ASDU types. If the feature is enabled, the status of each point configured by the user can have the individual IV quality bit determined by the processor.

The parameters required to support this feature are:

- Cyclic Set IV Time
- IV Check Delay Time
- IV Fail Count



To disable this feature, set the IV Fail Count parameter to 0. If the IV bit field is absent or set to 0, the invalid quality state for the point will always be reported as valid. If used, the Cyclic Set IV Time parameter must be at least 3 times larger than the IV Check Delay Time.

The **CYCLIC SET IV TIME** parameter determines the interval at which the driver will set all the IV bits for the points being monitored in the gateway. If the IV bit is ON for the number of times specified by the **IV FAIL COUNT** parameter, the point is in an invalid state. The driver will check the state of each bit at the frequency determined by the **IV CHECK DELAY TIME**.

The values for the parameters must permit the driver to properly execute the logic. For example, the value for **CYCLIC SET IV TIME** should be twice the **IV CHECK DELAY TIME**, multiplied by the **IV FAIL COUNT**. If the cyclic timer is set to a smaller value, the logic may not execute correctly.

In order to configure points for this feature, the **IV DB Bit** field for each data type must be set to the database configuration records in the configuration file. For example;

**Edit - M\_SP\_NA\_1**

|     | Point | DB Address | Groups   | IV DB Bit | Comment                           |
|-----|-------|------------|----------|-----------|-----------------------------------|
| ✓ 1 | 100   | 0          | 0FFFFFFF | 48        | P1-PSHH -- Discharge pressure SD  |
| ✓ 2 | 101   | 1          | 00000001 | 49        | P1-PSH -- High discharge pressure |
| ✓ 3 | 102   | 2          | 00000003 | 50        | P1-PSL -- Low suction pressure    |
| ✓ 4 | 103   | 3          | 00000001 | 51        | P1-FSL -- Low flow                |

Point Value Status - OK

**Edit - M\_DP\_NA\_1**

|     | Point | DB Address | Groups   | IV DB Bit | Comment                   |
|-----|-------|------------|----------|-----------|---------------------------|
| ✓ 1 | 200   | 16         | 0FFFFFFF | 52        | Pump 1 Status             |
| ✓ 2 | 201   | 18         | 00000001 | 52        | MOV101 -- position switch |
| ✓ 3 | 202   | 20         | 00000001 | 52        | MOV102 -- position switch |
| ✓ 4 | 203   | 22         | 00000001 | 52        | MOV103 -- position switch |

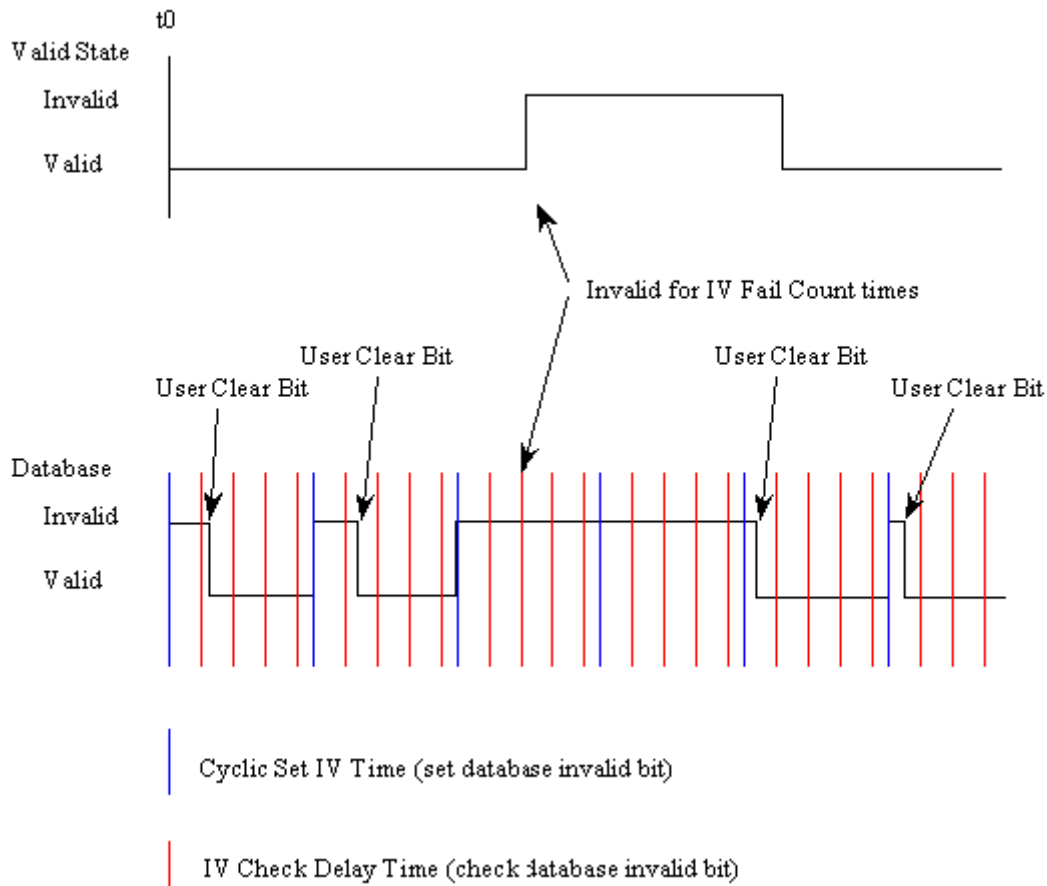
Point Value Status - OK

**Edit - M\_ME\_NA\_1**

|     | Point | DB Address | Groups   | Default Deadband | IV DB Bit | Comment               |
|-----|-------|------------|----------|------------------|-----------|-----------------------|
| ✓ 1 | 400   | 10         | 0FFFFFFF | 2000             | 53        | P1 suction pressure   |
| ✓ 2 | 401   | 11         | 00000001 | 1000             | 53        | P1 discharge pressure |
| ✓ 3 | 402   | 12         | 00000001 | 5000             | 53        | P2 suction pressure   |

Point Value Status - OK

The following illustration shows how these parameters are implemented:



If a database bit address (1 to 64000) is present, the application may consider the point with an invalid flag if the previous logic checks the IV bit as 1 during consecutive IV Check Delay scans. The IV bits would have to be reset to 0 to set the point to valid state.

The IV DB bit defined for each point can be unique, or many points may share the same bit. The last case could be used when the points on an I/O gateway are to be considered as one set. In this case only a single bit is required. For a point that is the result of a computation, the valid quality state could be set for each point individually.



---

## 5.2.7 Command Block Functionality

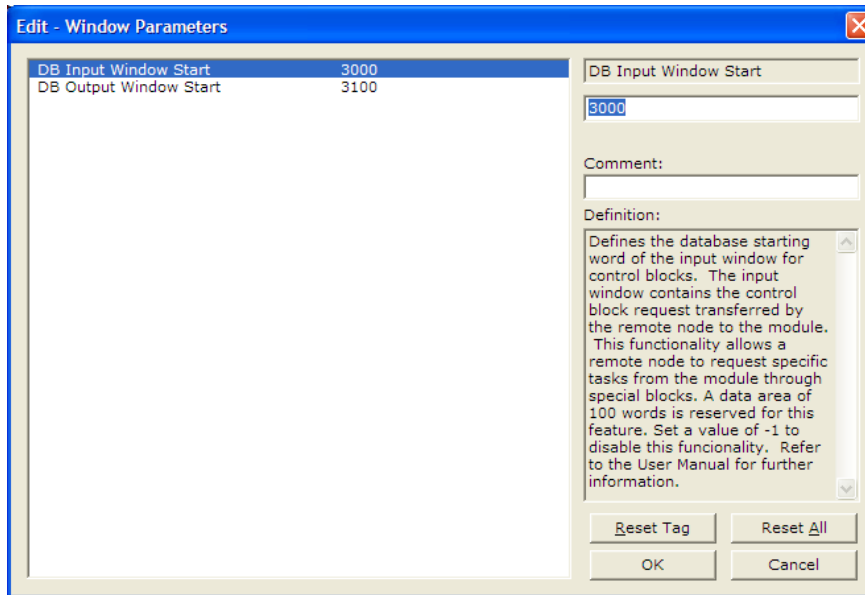
### Introduction

The command block functionality allows the ProLinx gateway to receive requests for special tasks through the other protocol channel (example: Modbus TCP/IP, DF1, and so on). To perform a command block request, a command block must be moved to the ProLinx gateway database starting at a specific address configured by the user. For example, this functionality allows a Modbus TCP/IP client to pass IEC-870-5-101 timestamp events to the IEC-870-5-101 client that is connected to the ProLinx gateway.

**Note:** The IEC 101S protocol Command Block functionality explained in the following topics is not available in all ProLinx Communication gateways. It is available in combination with only a limited number of other protocols. If you see a parameter edit box called *Window Parameters* in your ProSoft Configuration Builder I014S configuration options list, then this feature is supported on your gateway model and the next few topics will tell you how to use this feature. If you do not see the *Window Parameters* edit box option in the configuration tree, your gateway does not support this feature and you may safely ignore the next few topics.

### Configuration

In order to configure the ProLinx gateway to use the Command Block functionality (if present on your specific gateway model), there are two parameters to be configured, *DB Input Window Start* and *DB Output Window Start*, as seen below.



**Note:** If the PCB configuration for your specific gateway model does not include the window shown above, it means your model does not support Command Block functionality.

The *DB Input Window Start* parameter defines the ProLinx database starting register where the command block request data should be placed (transferred from the remote node to the ProLinx gateway). In the example above, a Modbus TCP/IP client would send a control block request to the ProLinx gateway starting at database start register 3000. A value of -1 in this parameter disables this functionality.

The *DB Output Window Start* parameter defines the ProLinx database starting register where the control block response should be placed (transferred from the ProLinx gateway to the remote node.) After receiving a command from a remote node, some control block requests require a response be returned from the gateway (for example, the control block to retrieve the current time from the gateway). For the example above, the remote node must read the control block response from the ProLinx gateway starting at database start register 3100. A value of -1 in this parameter disables this functionality.

### General Command Block Structure

A command request block will have the following general structure. The command request block should be copied starting at the database address given by the configuration parameter DB Input Window Start:

| Word Offset | Description                                  |
|-------------|--|
| 0           | Block ID1 - Command Request Without Response |
| 1           | Block ID2 - Command Request With Response    |
| 2 to 98     | Command Request Data                         |
| 99          | Block ID1 - Command Request Without Response |

Where:

- Command Request Without Response Block is transferred from the remote node to the ProLinx gateway to request a special task that does not require data back from the gateway. For example, a Set Time command block sets the gateway time but does not require data back from the gateway.
- Command Request With Response Block requires data back from the gateway. The response block is copied at a different database area. For example, a Get Time command block requests the gateway time information.

So a command block may:

- Perform command request without response only
- Perform command request with response only
- Perform both command request with response and command request without response

The ProLinx gateway recognizes a new block command request if:

- A new command request block ID is received at word offsets 0 and 99
- The same command request block ID is received at word offsets 0 and 99

If these conditions are fulfilled the gateway will process the request and response command blocks.

Once the command response block is processed, the response data will be copied to the ProLinx database with the following structure. The command response block will be copied starting at the database address given by the configuration parameter DB Output Window Start:

| Word Offset | Description                                  |
|-------------|--|
| 0           | Block ID2 - Command Request With Response    |
| 1           | Block ID1 - Command Request Without Response |
| 2 to 98     | Command Response Data                        |
| 99          | Block ID2 - Command Request With Response    |

The remote node must therefore read this block from the ProLinx gateway to retrieve the command block response.

Command Block List

The following command blocks are supported by the gateway:

| Block Range | Mode                             | Direction       | Description             |
|-------------|----------------------------------|-----------------|-------------------------|
| 100 and 101 | Command Request without Response | Node to ProLinx | Event Data              |
| 200 and 201 | Command Request without Response | Node to ProLinx | Set Time                |
| 200 and 201 | Command Request with Response    | ProLinx to Node | Get Time                |
| 202 and 203 | Command Request with Response    | ProLinx to Node | Get Event Buffer Counts |

Two block IDs per command type are required to allow the detection of a new block. So the remote node can switch between two block IDs when a new command block for the same type is requested. Even if the block only contains a request with response type, then offsets 0 and 99 must also trigger to a new value to allow the gateway to detect a new block.

The following general examples illustrate the block transfer mechanism.

For these examples, set the *DB Input Window Start* parameter to 3000 and set the *DB Output Window Start* parameter to 3100.

**Example 1 - Command Request Without Response only**

For this example, the remote node is passing a timestamp event to be transferred to the remote client connected to the ProLinx gateway.



The following block can be used for the first request.

Command Request Block

| Word Offset | Database Address | Value      | Description  |
|-------------|------------------|------------|--|
| 0           | 3000             | 100        | Block ID1 - Command Request Without Response   |
| 1           | 3001             | 0          | Block ID2 - Command Request With Response. Move a value of 0 or any other value out of the 200 to 203 range since no data is requested back from the gateway |
| 2 to 10     | 3002 to 3010     | Event Data | Timestamp event  |
| 99          | 3099             | 100        | Block ID1 - Command Request Without Response   |

Command Response Block

| Word Offset | Database Address | Value    | Description                                  |
|-------------|------------------|----------|--|
| 0           | 3100             | 0        | Block ID2 - Command Request With Response.   |
| 1           | 3101             | 100      | Block ID1 - Command Request Without Response |
| 2 to 98     | 3102 to 3198     | Reserved | Reserved - no update                         |
| 99          | 3199             | 0        | Block ID2 - Command Request With Response.   |

In order to send another command request to pass another event, use the following format:

Command Request Block

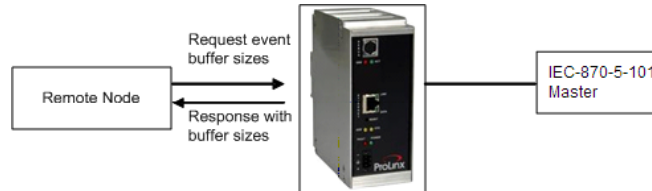
| Word Offset | Database Address | Value      | Description  |
|-------------|------------------|------------|--|
| 0           | 3000             | 101        | Block ID1 - Command Request Without Response   |
| 1           | 3001             | 0          | Block ID2 - Command Request With Response. Move a value of 0 or any other value out of the 200 to 203 range since no data is requested back from the gateway |
| 2 to 10     | 3002 to 3010     | Event Data | Timestamp event  |
| 99          | 3099             | 101        | Block ID1 - Command Request Without Response   |

Command Response Block

| <b>Word Offset</b> | <b>Database Address</b> | <b>Value</b> | <b>Description</b>                           |
|--------------------|-------------------------|--------------|--|
| 0                  | 3100                    | 0            | Block ID2 - Command Request With Response.   |
| 1                  | 3101                    | 101          | Block ID1 - Command Request Without Response |
| 2 to 98            | 3102 to 3198            | Reserved     | Reserved - no update                         |
| 99                 | 3199                    | 0            | Block ID2 - Command Request With Response.   |

**Example 2 - Command Request with Response Only**

For this example, the remote node is requesting the event buffer sizes from the ProLinx gateway. This is an example of a request that requires a response back from the gateway:



The following block can be used for the first request.

Command Request Block

| Word Offset | Database Address | Value    | Description   |
|-------------|------------------|----------|---|
| 0           | 3000             | 0        | Block ID1 - Command Request Without Response<br>Move a value of 0 or any other value out of the 100 to 101 and 200 to 201 range, because no request without response is performed |
| 1           | 3001             | 202      | Block ID2 - Command Request With Response.  |
| 2 to 98     | 3002 to 3098     | Not Used | Not Used  |
| 99          | 3099             | 0        | Block ID1 - Command Request Without Response<br>Move a value of 0 or any other value out of the 100 to 101 and 200 to 201 range, because no request without response is performed |

Command Response Block

| Word Offset | Database Address | Value         | Description                                  |
|-------------|------------------|---------------|--|
| 0           | 3100             | 202           | Block ID2 - Command Request With Response.   |
| 1           | 3101             | 0             | Block ID1 - Command Request Without Response |
| 2 to 8      | 3102 to 3108     | Event Buffers | Event Buffers                                |
| 99          | 3199             | 202           | Block ID2 - Command Request With Response.   |

In order to send another command request to pass another event, use the following format:

Command Request Block

| <b>Word Offset</b> | <b>Database Address</b> | <b>Value</b> | <b>Description</b>   |
|--------------------|-------------------------|--------------|--|
| 0                  | 3000                    | 1            | Block ID1 - Command Request Without Response<br>Command Request Without Response Move a value of 0 or any other value out of the 100 to 101 and 200 to 201 range, because no request without response is performed |
| 1                  | 3001                    | 203          | Block ID2 - Command Request With Response.   |
| 2 to 98            | 3002 to 3098            | Not Used     | Not Used   |
| 99                 | 3099                    | 1            | Block ID1 - Command Request Without Response<br>Command Request Without Response Move a value of 0 or any other value out of the 100 to 101 and 200 to 201 range, because no request without response is performed |

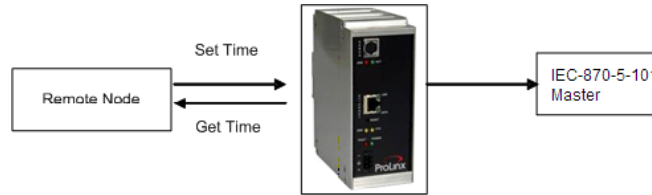
Command Response Block

| <b>Word Offset</b> | <b>Database Address</b> | <b>Value</b>  | <b>Description</b>                           |
|--------------------|-------------------------|---------------|--|
| 0                  | 3100                    | 203           | Block ID2 - Command Request With Response.   |
| 1                  | 3101                    | 1             | Block ID1 - Command Request Without Response |
| 2 to 8             | 3102 to 3108            | Event Buffers | Event Buffers                                |
| 99                 | 3199                    | 203           | Block ID2 - Command Request With Response.   |



**Example 3 - Command Request with Response and Command Request Without Response**

For this example, the remote node is setting the ProLinx gateway ProLinx gateway requesting the current time from the gateway.



The following block can be used for the first request.

Command Request Block

| Word Offset | Database Address | Value                    | Description   |
|-------------|------------------|--------------------------|---|
| 0           | 3000             | 200                      | Block ID1 - Command Request Without Response (Set Time) |
| 1           | 3001             | 200                      | Block ID2 - Command Request With Response (Get Time)    |
| 2 to 8      | 3002 to 3008     | New Time Sent to ProLinx | New Time Sent to ProLinx                                |
| 99          | 3099             | 200                      | Block ID1 - (Set Time)                                  |

Command Response Block

| Word Offset | Database Address | Value                        | Description   |
|-------------|------------------|------------------------------|---|
| 0           | 3100             | 200                          | Block ID2 - Command Request With Response (Get Time)    |
| 1           | 3101             | 200                          | Block ID1 - Command Request Without Response (Set Time) |
| 2 to 8      | 3002 to 3008     | Current Time sent by ProLinx | Current Time sent by ProLinx                            |
| 99          | 3199             | 200                          | Block ID2 - Command Request With Response (Get Time)    |

In order to send another command request to pass another event, use the following format:

Command Request Block

| <b>Word Offset</b> | <b>Database Address</b> | <b>Value</b>             | <b>Description</b>                                      |
|--------------------|-------------------------|--------------------------|---|
| 0                  | 3000                    | 201                      | Block ID1 - Command Request Without Response (Set Time) |
| 1                  | 3001                    | 201                      | Block ID2 - Command Request With Response (Get Time)    |
| 2 to 98            | 3002 to 3008            | New Time Sent to ProLinx | New Time Sent to ProLinx                                |
| 99                 | 3099                    | 201                      | Block ID1 - (Set Time)                                  |

Command Response Block

| <b>Word Offset</b> | <b>Database Address</b> | <b>Value</b>  | <b>Description</b>                                      |
|--------------------|-------------------------|---------------|---|
| 0                  | 3100                    | 201           | Block ID2 - Command Request With Response (Get Time)    |
| 1                  | 3101                    | 201           | Block ID1 - Command Request Without Response (Set Time) |
| 2 to 8             | 3102 to 3108            | Event Buffers | Current Time sent by ProLinx                            |
| 99                 | 3199                    | 201           | Block ID2 - Command Request With Response (Get Time)    |

**Event Data Command Block**

This functionality allows a remote node to pass timestamp events to be transferred to the IEC-870-5-101 server connected to the ProLinx gateway. The following topics describe the structure of the event data command block.

**Request Block**

Each Event Request block can send up to 10 events to the gateway. Refer to Events for more information about timestamped events. This block should only be used to pass events with a predefined timestamp (the gateway will also send timestamped events when database values change). While using the Event Request block, disable the events for those specific points to avoid multiple event generation (caused by point value update through the database).

| Start Word | End Word | Data Field(s)              | Description  |
|------------|----------|----------------------------|--|
| 0          | 0        | Block ID                   | 100 or 101 = Set Time (Block ID1)  |
| 1          | 1        | Data block being requested | Block ID2 (Command Request With Response)  |
| 2          | 2        | Event Count                | Number of events present in the block. This field can have a value from 1 to 12. |
| 3          | 10       | Event #1                   | Event data to add to event message queue.  |
| 11         | 18       | Event #2                   | Event data to add to event message queue.  |
| 19         | 26       | Event #3                   | Event data to add to event message queue.  |
| 27         | 34       | Event #4                   | Event data to add to event message queue.  |
| 35         | 42       | Event #5                   | Event data to add to event message queue.  |
| 43         | 50       | Event #6                   | Event data to add to event message queue.  |
| 51         | 58       | Event #7                   | Event data to add to event message queue.  |
| 59         | 66       | Event #8                   | Event data to add to event message queue.  |
| 67         | 74       | Event #9                   | Event data to add to event message queue.  |
| 75         | 82       | Event #10                  | Event data to add to event message queue.  |
| 83         | 90       | Event #11                  | Event data to add to event message queue.  |
| 91         | 98       | Event #12                  | Event data to add to event message queue.  |
| 99         | 99       | Block ID                   | 100 or 101 = Set Time (Block ID1)  |

Each event contains the following data structure:

| Start Byte | End Byte | Data Field(s)          | Description  |
|------------|----------|------------------------|--|
| 0          | 1        | DB Index               | This is the index for the point in the gateway's database. This corresponds to the order of point definition for the gateway data types. This is not the point address for the event.  |
| 2          | 2        | ASDU                   | This is the ASDU data type for the event message. <ul style="list-style-type: none"> <li>▪ 1=single point,</li> <li>▪ 3=double-point,</li> <li>▪ 5=step,</li> <li>▪ 9=normalized,</li> <li>▪ 11=scaled,</li> <li>▪ 13=short-float</li> <li>▪ 15=integrated total.</li> </ul> |
| 3          | 3        | Qualifier              | This is the qualifier code for the event message. Refer to the IEC protocol specification for a full listing of valid qualifier codes for each ASDU type.  |
| 4          | 5        | Year                   | This field contains the four-digit year for the event.   |
| 6          | 6        | Month                  | This field contains the month value for the event. Valid entry for this field is in the range of 1 to 12.  |
| 7          | 7        | Day                    | This field contains the day value for the event. Valid entry for this field is in the range of 1 to 31.  |
| 8          | 8        | Hour                   | This field contains the hour value for the event. Valid entry for this field is in the range of 0 to 23 (bits 0 to 4). Bit 7 of the byte can be used to set the summer time flag (0=standard, 1=summer time)   |
| 9          | 9        | Minute                 | This field contains the minute value for the event. Valid entry for this field is in the range of 0 to 59 (bits 0 to 5). Bit 7 of the byte can be used to set the invalid bit for the time (0=valid, 1=invalid).   |
| 10         | 11       | Seconds & Milliseconds | This field contains the seconds and milliseconds value for the event. Valid entry for this field is in the range of 0 to 59,999.   |
| 12         | 15       | Data                   | These bytes contain the data for the event. For single- and double-point, step events, the first byte is used. For measured value events, the first two bytes are used. For integrated total and short-float events, all four bytes are used.                                |

### Set Time Command Block

This functionality allows a remote node to set the clock of the ProLinX gateway.

#### **Request Block**

The following table describes the structure of the Set Time command block.

| <b>Start Word</b> | <b>End Word</b> | <b>Description</b>                        |
|-------------------|-----------------|---|
| 0                 | 0               | 200 or 201 = Set Time (Block ID1)         |
| 1                 | 1               | Block ID2 (Command Request With Response) |
| 2                 | 2               | Year                                      |
| 3                 | 3               | Month                                     |
| 4                 | 4               | Day                                       |
| 5                 | 5               | Hour                                      |
| 6                 | 6               | Minutes                                   |
| 7                 | 7               | Seconds & milliseconds                    |
| 8                 | 98              | Reserved for future use                   |
| 99                | 99              | 200 or 201 = Set Time (Block ID1)         |

**Get Time Command Block**

This functionality allows a remote node to retrieve the clock data from the ProLinX gateway.

**Request Block**

The following table describes the structure of the Get Time command block.

| Start Word | End Word | Description                                  |
|------------|----------|--|
| 0          | 0        | Block ID1 (Command Request Without Response) |
| 1          | 1        | 200 or 201 = Get Time (Block ID2)            |
| 2          | 2        | Year   |
| 3          | 3        | Month  |
| 4          | 4        | Day  |
| 5          | 5        | Hour   |
| 6          | 6        | Minutes                                      |
| 7          | 7        | Seconds & milliseconds                       |
| 8          | 98       | Reserved for future use                      |
| 99         | 99       | Block ID1 (Command Request Without Response) |

**Response Block**

| Start Word | End Word | Description                                  |
|------------|----------|--|
| 0          | 0        | 200 or 201 = Get Time (Block ID2)            |
| 1          | 1        | Block ID1 (Command Request Without Response) |
| 2          | 2        | Year   |
| 3          | 3        | Month  |
| 4          | 4        | Day  |
| 5          | 5        | Hour   |
| 6          | 6        | Minutes                                      |
| 7          | 7        | Seconds & milliseconds                       |
| 8          | 98       | Reserved for future use                      |
| 99         | 99       | 200 or 201 = Get Time (Block ID2)            |

**Get Event Buffer Counts Command Block**

This functionality allows a remote node to retrieve the event buffer space of each data type. The following tables describe the structure of the Get Event Buffer Sizes command block:

**Request Block**

| Start Word | End Word | Description                                     |
|------------|----------|---|
| 0          | 0        | Block ID1 (Command Request Without Response)    |
| 1          | 1        | 202 or 203 = Get Event Buffer Count (Block ID2) |
| 2          | 2        | Reserved  |
| 3          | 98       | Reserved  |
| 99         | 99       | Block ID1                                       |

**Response Block**

| Start Word | End Word | Description                                     |
|------------|----------|---|
| 0          | 0        | 202 or 203 = Get Event Buffer Count (Block ID2) |
| 1          | 1        | Block ID1 (Command Request Without Response)    |
| 2          | 2        | Single point event buffer space                 |
| 3          | 3        | Double point event buffer space                 |
| 4          | 4        | Step point event buffer space                   |
| 5          | 5        | Normalized event buffer space                   |
| 6          | 6        | Scaled event buffer space                       |
| 7          | 7        | Short-float event buffer space                  |
| 8          | 8        | Integrated totals event buffer space            |
| 9          | 98       | Reserved  |
| 99         | 99       | 202 or 203 = Get Event Buffer Count (Block ID2) |

### 5.2.8 Slave Error and Status

The 101S Driver Error and Status Data areas are discussed in this section. This data represents a collection of status, diagnostic and troubleshooting registers which may prove helpful in troubleshooting the 101S network and port operation. The data map functionality of the gateway must be utilized in order to map this data into the normal gateway database region (0 to 3999). All or any portion of the data can be moved using this facility.

The data area is filled with zeros whenever the gateway is initialized. This occurs during a cold-start (power-on), reset (reset push-button pressed) or a warm-boot operation (from a warm command or after loading of new configuration).

#### IEC870-5-101 Slave Driver Error/Status Data

| Offset         | Parameter                | Description   |
|----------------|--------------------------|---|
| 10100          | Bad Checksum Count       | This word contains the total number of frames received by the gateway that contain a bad check-sum values in the message.   |
| 10101          | Sync Error Count         | This word contains the total number of frames received by the gateway that have synchronization errors. Each frame in the protocol has a specific header that must be received in a fixed sequence. If this header is not received correctly, this word will be incremented, and the frame will be discarded. |
| 10102          | Frame Length Error Count | This word contains the total number of frames received by the gateway that do not have the correct length.  |
| 10103          | Timeout Error Count      | This word contains the total number of frames received by the gateway that were not received within the specified receive timeout parameter.  |
| 10104          | Receive Frame Count      | This word contains the total number of frames received by the gateway.  |
| 10105          | Transmit Frame Count     | This word contains the total number of frames transmitted by the gateway.   |
| 10106 to 10199 | No valid data            | Reserved.   |



## 6 IEC 60870-5-101 Slave Interoperability Statement

### *In This Chapter*

|   |     |
|---|-----|
| ❖ System or Device .....                                      | 130 |
| ❖ Network Configuration .....                                 | 130 |
| ❖ Physical Layer .....  | 130 |
| ❖ Link Layer .....  | 131 |
| ❖ Application Layer .....                                     | 131 |
| ❖ Selection of Standard ASDUs .....                           | 132 |
| ❖ Type Identifier and Cause of Transmission Assignments ..... | 135 |
| ❖ Basic Application Functions .....                           | 137 |

This companion standard presents sets of parameters and alternatives from which subsets have to be selected to implement particular telecontrol systems. Certain parameter values, such as the number of octets in the COMMON ADDRESS of ASDUs represent mutually exclusive alternatives. This means that only one value of the defined parameters is admitted per system. Other parameters, such as the listed set of different process information in command and in monitor direction allow the specification of the complete set or subsets, as appropriate for given applications. This clause summarizes the parameters of the previous clauses to facilitate a suitable selection for a specific application. If a system is composed of equipment originating from different manufacturers it is necessary that all partners agree on the selected parameters.

**Note:** In addition, the full specification of a system may require individual selection of certain parameters for certain parts of the system, such as the individual selection of scaling factors for individually addressable measured values.

The selected parameters should be marked in the white boxes as follows:

- Function or ASDU is not used
- Function or ASDU is used as standardized (default)

A black check box indicates that the option cannot be selected in this companion standard.

### 6.1 System or Device

- System definition
- Controlling station definition (Master)
- Controlled station definition (slave)

### 6.2 Network Configuration

(Network-specific parameter)

- Point-to-point  Multipoint-party line
- Multiple point-to-point  Multipoint-star

### 6.3 Physical Layer

(Network-specific parameter)

#### 6.3.1 Transmission Speed (Control Direction)

| Unbalanced interchange<br>Circuit V.24/V.28<br>Standard | Unbalanced interchange<br>Circuit V.24/V.28<br>Recommended if > 1200 bit/s | Balanced<br>interchange<br>Circuit X.24/X.27    |                                      |
|---|--|---|--------------------------------------|
| <input type="checkbox"/> 100 bit/s                      | <input checked="" type="checkbox"/> 2400 bit/s                             | <input checked="" type="checkbox"/> 2400 bit/s  | <input type="checkbox"/> 56000 bit/s |
| <input type="checkbox"/> 200 bit/s                      | <input checked="" type="checkbox"/> 4800 bit/s                             | <input checked="" type="checkbox"/> 4800 bit/s  | <input type="checkbox"/> 64000 bit/s |
| <input checked="" type="checkbox"/> 300 bit/s           | <input checked="" type="checkbox"/> 9600 bit/s                             | <input checked="" type="checkbox"/> 9600 bit/s  |                                      |
| <input checked="" type="checkbox"/> 600 bit/s           |  | <input checked="" type="checkbox"/> 19200 bit/s |                                      |
| <input checked="" type="checkbox"/> 1200 bit/s          |  | <input checked="" type="checkbox"/> 38400 bit/s |                                      |

#### 6.3.2 Transmission Speed (Monitor Direction)

| Unbalanced interchange<br>Circuit V.24/V.28<br>Standard | Unbalanced interchange<br>Circuit V.24/V.28<br>Recommended if > 1200 bit/s | Balanced<br>interchange<br>Circuit X.24/X.27    |                                      |
|---|--|---|--------------------------------------|
| <input type="checkbox"/> 100 bit/s                      | <input checked="" type="checkbox"/> 2400 bit/s                             | <input checked="" type="checkbox"/> 2400 bit/s  | <input type="checkbox"/> 56000 bit/s |
| <input type="checkbox"/> 200 bit/s                      | <input checked="" type="checkbox"/> 4800 bit/s                             | <input checked="" type="checkbox"/> 4800 bit/s  | <input type="checkbox"/> 64000 bit/s |
| <input checked="" type="checkbox"/> 300 bit/s           | <input checked="" type="checkbox"/> 9600 bit/s                             | <input checked="" type="checkbox"/> 9600 bit/s  |                                      |
| <input checked="" type="checkbox"/> 600 bit/s           |  | <input checked="" type="checkbox"/> 19200 bit/s |                                      |
| <input checked="" type="checkbox"/> 1200 bit/s          |  | <input checked="" type="checkbox"/> 38400 bit/s |                                      |

## 6.4 Link Layer

(Network-specific parameter)

Frame format FT 1.2, single character 1 and the fixed time out interval are used exclusively in this companion standard.

### Link transmission procedure

- Balanced transmission
- Unbalanced transmission

### Address field of link

- Not present (balanced transmission only)
- One octet
- Two octets
- Structured

### Frame length

- Unstructured
- 25 to 252 Maximum length L (number of octets)

## 6.5 Application Layer

### 6.5.1 Transmission Mode for Application Data

Mode 1 (Least significant octet first), as defined in clause 4.10 of IEC 60870-5-4, is used exclusively in this companion standard.

### 6.5.2 Common Address of ASDU - 101S

(System-specific parameter)

- One octet
- Two octets

### 6.5.3 Information Object Address

(System-specific parameter)

- One octet
- Structured
- Two octets
- Unstructured
- Three octets

### 6.5.4 Cause of Transmission

(System-specific parameter)

- One octet
- Two octets (with originator address)

### 6.5.5 Length of APDU - 101S

(System-specific parameter, specify the maximum length of the APDU per system)

The maximum length of the APDU in this protocol implementation is 252 (default). The maximum length may be reduced by the system.

The maximum length of APDU per system is 255.

## 6.6 Selection of Standard ASDUs

### 6.6.1 Process Information in Monitor Direction

(Station-specific parameter; mark each Type ID "X" if it is only used in the standard direction, "R" if only used in the reverse direction, and "B" if used in both directions)

|                                     |      |   |           |
|-------------------------------------|------|---|-----------|
| <input checked="" type="checkbox"/> | <1>  | := Single-point information   | M_SP_NA_1 |
| <input checked="" type="checkbox"/> | <2>  | := Single-point information with time tag                                     | M_SP_TA_1 |
| <input checked="" type="checkbox"/> | <3>  | := Double-point information   | M_DP_NA_1 |
| <input checked="" type="checkbox"/> | <4>  | := Double-point information with time tag                                     | M_DP_TA_1 |
| <input checked="" type="checkbox"/> | <5>  | := Step position information  | M_ST_NA_1 |
| <input checked="" type="checkbox"/> | <6>  | := Step position information with time tag                                    | M_ST_TA_1 |
| <input checked="" type="checkbox"/> | <7>  | := Bitstring of 32 bits   | M_BO_NA_1 |
| <input checked="" type="checkbox"/> | <8>  | := Bitstring of 32 bits with time tag   | M_BO_TA_1 |
| <input checked="" type="checkbox"/> | <9>  | := Measured value, normalized value   | M_ME_NA_1 |
| <input checked="" type="checkbox"/> | <10> | := Measured value, normalized value with time tag                             | M_ME_TA_1 |
| <input checked="" type="checkbox"/> | <11> | := Measured value, scaled value   | M_ME_NB_1 |
| <input checked="" type="checkbox"/> | <12> | := Measured value, scaled value with time tag                                 | M_ME_TB_1 |
| <input checked="" type="checkbox"/> | <13> | := Measured value, short floating-point value                                 | M_ME_NC_1 |
| <input checked="" type="checkbox"/> | <14> | := Measured value, short floating-point value<br>with time tag                | M_ME_TC_1 |
| <input checked="" type="checkbox"/> | <15> | := Integrated totals  | M_IT_NA_1 |
| <input checked="" type="checkbox"/> | <16> | := Integrated totals with time tag  | M_IT_TA_1 |
| <input type="checkbox"/>            | <17> | := Event of protection equipment with time tag                                | M_EP_TA_1 |
| <input type="checkbox"/>            | <18> | := Packed start events of protection equipment<br>with time tag               | M_EP_TB_1 |
| <input type="checkbox"/>            | <19> | := Packed output circuit information of protection<br>equipment with time tag | M_EP_TC_1 |
| <input type="checkbox"/>            | <20> | := Packed single-point information with status<br>change detection            | M_PS_NA_1 |
| <input type="checkbox"/>            | <21> | := Measured value, normalized value without<br>quality description            | M_ME_ND_1 |
| <input checked="" type="checkbox"/> | <30> | := Single-point information with time tag<br>CP56Time2a                       | M_SP_TB_1 |
| <input checked="" type="checkbox"/> | <31> | := Double-point information with time tag<br>CP56Time2a                       | M_DP_TB_1 |
| <input checked="" type="checkbox"/> | <32> | := Step position information with time tag<br>CP56Time2a                      | M_ST_TB_1 |
| <input checked="" type="checkbox"/> | <33> | := Bitstring of 32 bits with time tag CP56Time2a                              | M_BO_TB_1 |

|                                     |      |  |           |
|-------------------------------------|------|--|-----------|
| <input checked="" type="checkbox"/> | <34> | := Measured value, normalized value with time tag<br>CP56Time2a                          | M_ME_TD_1 |
| <input checked="" type="checkbox"/> | <35> | := Measured value, scaled value with time tag<br>CP56Time2a                              | M_ME_TE_1 |
| <input checked="" type="checkbox"/> | <36> | := Measured value, short floating-point value<br>with time tag CP56Time2a                | M_ME_TF_1 |
| <input checked="" type="checkbox"/> | <37> | := Integrated totals with time tag CP56Time2a  | M_IT_TB_1 |
| <input type="checkbox"/>            | <38> | := Event of protection equipment with time tag<br>CP56Time2a                             | M_EP_TD_1 |
| <input type="checkbox"/>            | <39> | := Packed start events of protection equipment<br>with time tag CP56Time2a               | M_EP_TE_1 |
| <input type="checkbox"/>            | <40> | := Packed output circuit information of protection<br>equipment with time tag CP56Time2a | M_EP_TF_1 |

### 6.6.2 Process Information in Control Direction

(Station-specific parameter; mark each Type ID "X" if it is only used in the standard direction, "R" if only used in the reverse direction, and "B" if used in both directions)

|                                     |      |  |           |
|-------------------------------------|------|--|-----------|
| <input checked="" type="checkbox"/> | <45> | := Single command  | C_SC_NA_1 |
| <input checked="" type="checkbox"/> | <46> | := Double command  | C_DC_NA_1 |
| <input checked="" type="checkbox"/> | <47> | := Regulating step command   | C_RC_NA_1 |
| <input checked="" type="checkbox"/> | <48> | := Set point command, normalized value                                       | C_SE_NA_1 |
| <input checked="" type="checkbox"/> | <49> | := Set point command, scaled value   | C_SE_NB_1 |
| <input checked="" type="checkbox"/> | <50> | := Set point command, short floating-point value                             | C_SE_NC_1 |
| <input checked="" type="checkbox"/> | <51> | := Bitstring of 32 bit   | C_BO_NA_1 |
| <input type="checkbox"/>            | <58> | := Single command with time tag CP56Time2a                                   | C_SC_TA_1 |
| <input type="checkbox"/>            | <59> | := Double command with time tag CP56Time2a                                   | C_DC_TA_1 |
| <input type="checkbox"/>            | <60> | := Regulating step command with time tag<br>CP56Time2a                       | C_RC_TA_1 |
| <input type="checkbox"/>            | <61> | := Set point command, normalized value<br>with time tag CP56Time2a           | C_SE_TA_1 |
| <input type="checkbox"/>            | <62> | := Set point command, scaled value<br>with time tag CP56Time2a               | C_SE_TB_1 |
| <input type="checkbox"/>            | <63> | := Set point command, short floating-point value<br>with time tag CP56Time2a | C_SE_TC_1 |
| <input type="checkbox"/>            | <64> | := Bitstring of 32 bits with time tag CP56Time2a                             | C_BO_TA_1 |

Either the ASDUs of the set <45>-<51> or of the set <58>-<64> are used.

### 6.6.3 System Information in Monitor Direction

(Station-specific parameter; mark "X" if used)

- |                                     |      |                          |           |
|-------------------------------------|------|--------------------------|-----------|
| <input checked="" type="checkbox"/> | <70> | := End of initialization | M_EI_NA_1 |
|-------------------------------------|------|--------------------------|-----------|

### 6.6.4 System Information in Control Direction

(Station-specific parameter; mark each Type ID "X" if it is only used in the standard direction, "R" if only used in the reverse direction, and "B" if used in both directions)

- |                                     |       |  |           |
|-------------------------------------|-------|--|-----------|
| <input checked="" type="checkbox"/> | <100> | := Interrogation command                       | C_IC_NA_1 |
| <input checked="" type="checkbox"/> | <101> | := Counter interrogation command               | C_CI_NA_1 |
| <input checked="" type="checkbox"/> | <102> | := Read command                                | C_RD_NA_1 |
| <input checked="" type="checkbox"/> | <103> | := Clock synchronization command               | C_CS_NA_1 |
| <input checked="" type="checkbox"/> | <104> | := Test command                                | C_TS_NB_1 |
| <input checked="" type="checkbox"/> | <105> | := Reset process command                       | C_RP_NC_1 |
| <input checked="" type="checkbox"/> | <106> | := Delay acquisition command <sup>Note 1</sup> | C_CD_NA_1 |

### 6.6.5 Parameter in Control Direction

(Station-specific parameter; mark each Type ID "X" if it is only used in the standard direction, "R" if only used in the reverse direction, and "B" if used in both directions)

- |                                     |       |  |           |
|-------------------------------------|-------|--|-----------|
| <input checked="" type="checkbox"/> | <110> | := Parameter of measured, normalized value           | P_ME_NA_1 |
| <input checked="" type="checkbox"/> | <111> | := Parameter of measured, scaled value               | P_ME_NB_1 |
| <input checked="" type="checkbox"/> | <112> | := Parameter of measured, short floating-point value | P_ME_NC_1 |
| <input type="checkbox"/>            | <113> | := Parameter activation                              | P_AC_NA_1 |

### 6.6.6 File Transfer

(Station-specific parameter; mark each Type ID "X" if it is only used in the standard direction, "R" if only used in the reverse direction, and "B" if used in both directions)

- |                          |       |   |           |
|--------------------------|-------|---|-----------|
| <input type="checkbox"/> | <120> | := File ready   | F_FR_NA_1 |
| <input type="checkbox"/> | <121> | := Section ready  | F_SR_NA_1 |
| <input type="checkbox"/> | <122> | := Call directory, select file, call file, call section | F_SC_NA_1 |
| <input type="checkbox"/> | <123> | := Last section, last segment                           | F_LS_NA_1 |
| <input type="checkbox"/> | <124> | := Ack file, ack section                                | F_AF_NA_1 |
| <input type="checkbox"/> | <125> | := Segment  | F_SG_NA_1 |
| <input type="checkbox"/> | <126> | := Directory  | F_DR_TA_1 |

## 6.7 Type Identifier and Cause of Transmission Assignments

(Station-specific parameters)

Shaded boxes: option not required

Black boxes: option not permitted in this companion standard

Blank boxes: functions or ASDU not used

Mark Type Identification/Cause of Transmission combinations: "X" if only used in standard direction, "R" if only used in reverse direction, and "B" if used in both directions.

| Type Identification | Cause of transmission |   |   |   |   |   |   |   |   |    |    |    |    |                |                |    |    |    |    |
|---------------------|-----------------------|---|---|---|---|---|---|---|---|----|----|----|----|----------------|----------------|----|----|----|----|
|                     | 1                     | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 20<br>to<br>36 | 37<br>to<br>41 | 44 | 45 | 46 | 47 |
| <1> M_SP_NA_1       |                       | X | X |   | X |   |   |   |   |    |    | X  |    |                | X              |    |    |    |    |
| <3> M_DP_NA_1       |                       | X | X |   | X |   |   |   |   |    |    | X  |    |                | X              |    |    |    |    |
| <5> M_ST_NA_1       |                       | X | X |   | X |   |   |   |   |    |    | X  |    |                | X              |    |    |    |    |
| <7> M_BO_NA_1       |                       | X | X |   | X |   |   |   |   |    |    | X  |    |                | X              |    |    |    |    |
| <9> M_ME_NA_1       | X                     |   | X |   | X |   |   |   |   |    |    | X  |    |                | X              |    |    |    |    |
| <11> M_ME_NB_1      | X                     |   | X |   | X |   |   |   |   |    |    | X  |    |                | X              |    |    |    |    |
| <13> M_ME_NC_1      | X                     |   | X |   | X |   |   |   |   |    |    | X  |    |                | X              |    |    |    |    |
| <15> M_IT_NA_1      |                       | X | X |   | X |   |   |   |   |    |    |    |    |                |                | X  |    |    |    |
| <20> M_PS_NA_1      |                       |   |   |   |   |   |   |   |   |    |    |    |    |                |                |    |    |    |    |
| <21> M_ME_ND_1      |                       |   |   |   |   |   |   |   |   |    |    |    |    |                |                |    |    |    |    |
| <30> M_SP_TB_1      |                       |   | X |   |   |   |   |   |   |    |    |    |    |                |                |    |    |    |    |
| <31> M_DP_TB_1      |                       |   | X |   |   |   |   |   |   |    |    |    |    |                |                |    |    |    |    |
| <32> M_ST_TB_1      |                       |   | X |   |   |   |   |   |   |    |    |    |    |                |                |    |    |    |    |
| <33> M_BO_TB_1      |                       |   | X |   |   |   |   |   |   |    |    |    |    |                |                |    |    |    |    |
| <34> M_ME_TD_1      |                       |   | X |   |   |   |   |   |   |    |    |    |    |                |                |    |    |    |    |
| <35> M_ME_TE_1      |                       |   | X |   |   |   |   |   |   |    |    |    |    |                |                |    |    |    |    |
| <36> M_ME_TF_1      |                       |   | X |   |   |   |   |   |   |    |    |    |    |                |                |    |    |    |    |
| <37> M_IT_TB_1      |                       |   | X |   |   |   |   |   |   |    |    |    |    |                |                | X  |    |    |    |
| <38> M_EP_TD_1      |                       |   |   |   |   |   |   |   |   |    |    |    |    |                |                |    |    |    |    |
| <39> M_EP_TE_1      |                       |   |   |   |   |   |   |   |   |    |    |    |    |                |                |    |    |    |    |
| <40> M_EP_TF_1      |                       |   |   |   |   |   |   |   |   |    |    |    |    |                |                |    |    |    |    |
| <45> C_SC_NA_1      |                       |   |   |   |   | X | X | X | X | X  |    |    |    |                |                | X  | X  | X  | X  |
| <46> C_DC_NA_1      |                       |   |   |   |   | X | X | X | X | X  |    |    |    |                |                | X  | X  | X  | X  |
| <47> C_RC_NA_1      |                       |   |   |   |   | X | X | X | X | X  |    |    |    |                |                | X  | X  | X  | X  |
| <48> C_SE_NA_1      |                       |   |   |   |   | X | X | X | X | X  |    |    |    |                |                | X  | X  | X  | X  |
| <49> C_SE_NB_1      |                       |   |   |   |   | X | X | X | X | X  |    |    |    |                |                | X  | X  | X  | X  |
| <50> C_SE_NC_1      |                       |   |   |   |   | X | X | X | X | X  |    |    |    |                |                | X  | X  | X  | X  |
| <51> C_BO_NA_1      |                       |   |   |   |   | X | X |   |   | X  |    |    |    |                |                | X  | X  | X  | X  |
| <58> C_SC_TA_1      |                       |   |   |   |   | X | X | X | X | X  |    |    |    |                |                | X  | X  | X  | X  |

|       | Type Identification | Cause of transmission |   |   |   |   |   |   |   |   |    |    |    |    |                |                |    |    |    |    |
|-------|---------------------|-----------------------|---|---|---|---|---|---|---|---|----|----|----|----|----------------|----------------|----|----|----|----|
|       |                     | 1                     | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 20<br>to<br>36 | 37<br>to<br>41 | 44 | 45 | 46 | 47 |
| <59>  | C_DC_TA_1           |                       |   |   |   |   | X | X | X | X | X  |    |    |    |                |                | X  | X  | X  | X  |
| <60>  | C_RC_TA_1           |                       |   |   |   |   | X | X | X | X | X  |    |    |    |                |                | X  | X  | X  | X  |
| <61>  | C_SE_TA_1           |                       |   |   |   |   | X | X | X | X | X  |    |    |    |                |                | X  | X  | X  | X  |
| <62>  | C_SE_TB_1           |                       |   |   |   |   | X | X | X | X | X  |    |    |    |                |                | X  | X  | X  | X  |
| <63>  | C_SE_TC_1           |                       |   |   |   |   | X | X | X | X | X  |    |    |    |                |                | X  | X  | X  | X  |
| <64>  | C_BO_TA_1           |                       |   |   |   |   | X | X |   |   | X  |    |    |    |                |                | X  | X  | X  | X  |
| <70>  | M_EI_NA_1           |                       |   |   | X |   |   |   |   |   |    |    |    |    |                |                |    |    |    |    |
| <100> | C_IC_NA_1           |                       |   |   |   |   | X | X | X | X | X  |    |    |    |                |                | X  | X  | X  | X  |
| <101> | C_CI_NA_1           |                       |   |   |   |   | X | X |   |   | X  |    |    |    |                |                | X  | X  | X  | X  |
| <102> | C_RD_NA_1           |                       |   |   | X |   |   |   |   |   |    |    |    |    |                |                | X  | X  | X  | X  |
| <103> | C_CS_NA_1           |                       |   |   |   |   | X | X |   |   |    |    |    |    |                |                | X  | X  | X  | X  |
| <105> | C_RP_NA_1           |                       |   |   |   |   | X | X |   |   |    |    |    |    |                |                | X  | X  | X  | X  |
| <107> | C_TS_TA_1           |                       |   |   |   |   | X | X |   |   |    |    |    |    |                |                | X  | X  | X  | X  |
| <110> | P_ME_NA_1           |                       |   |   |   |   | X | X |   |   |    |    |    |    |                |                | X  | X  | X  | X  |
| <111> | P_ME_NB_1           |                       |   |   |   |   | X | X |   |   |    |    |    |    |                |                | X  | X  | X  | X  |
| <112> | P_ME_NC_1           |                       |   |   |   |   | X | X |   |   |    |    |    |    |                |                | X  | X  | X  | X  |
| <113> | P_AC_NA_1           |                       |   |   |   |   |   |   |   |   |    |    |    |    |                |                |    |    |    |    |
| <120> | F_FR_NA_1           |                       |   |   |   |   |   |   |   |   |    |    |    |    |                |                |    |    |    |    |
| <121> | F_SR_NA_1           |                       |   |   |   |   |   |   |   |   |    |    |    |    |                |                |    |    |    |    |
| <122> | F_SC_NA_1           |                       |   |   |   |   |   |   |   |   |    |    |    |    |                |                |    |    |    |    |
| <123> | F_LS_NA_1           |                       |   |   |   |   |   |   |   |   |    |    |    |    |                |                |    |    |    |    |
| <124> | F_AF_NA_1           |                       |   |   |   |   |   |   |   |   |    |    |    |    |                |                |    |    |    |    |
| <125> | F_SG_NA_1           |                       |   |   |   |   |   |   |   |   |    |    |    |    |                |                |    |    |    |    |
| <126> | F_DR_TA_1           |                       |   |   |   |   |   |   |   |   |    |    |    |    |                |                |    |    |    |    |
| <127> | F_SC_NB_1           |                       |   |   |   |   |   |   |   |   |    |    |    |    |                |                |    |    |    |    |



## 6.8 Basic Application Functions

### 6.8.1 Station Initialization

(Station-specific parameter; mark "X" if function is used)

- Remote initialization

### 6.8.2 Cyclic Data Transmission

(Station-specific parameter; mark "X" if function is only used in the standard direction, "R" if only used in the reverse direction, and "B" if used in both directions)

- Cyclic data transmission

### 6.8.3 Read Procedure

(Station-specific parameter; mark "X" if function is only used in the standard direction, "R" if only used in the reverse direction, and "B" if used in both directions)

- Read procedure

### 6.8.4 Spontaneous Transmission

(Station-specific parameter; mark "X" if function is only used in the standard direction, "R" if only used in the reverse direction, and "B" if used in both directions)

- Spontaneous transmission

### 6.8.5 Double Transmission of Information Objects

(Station-specific parameter; mark each information type "X" where both a Type ID without time and corresponding Type ID with time are issued in response to a single spontaneous change of a monitored object)

The following type identifications may be transmitted in succession caused by a single status change of an information object. The particular Information Object Addresses for which double transmission is enabled are defined in a project-specific list.

- Single-point information: M\_SP\_NA\_1, M\_SP\_TA\_1, M\_SP\_TB\_1, and M\_PS\_NA\_1
- Double-point information: M\_DP\_NA\_1, MDP\_TA\_1, and M\_DP\_TB\_1
- Step position information: M\_ST\_NA\_1, M\_ST\_TA\_1, and M\_ST\_TB\_1
- Bitstring of 32 bits: M\_BO\_NA\_1, M\_BO\_TA\_1 and M\_BO\_TB\_1
- Measured value, normalized value: M\_ME\_NA\_1, M\_ME\_TA\_1, M\_ME\_ND\_1, and M\_ME\_TD\_1
- Measured value, scaled value: M\_ME\_NB\_1, M\_ME\_TB\_1, and M\_ME\_TE\_1
- Measured value, short floating-point value: M\_ME\_NC\_1, M\_ME\_TC\_1, and M\_ME\_TF\_1

### 6.8.6 Station Interrogation

(Station-specific parameter; mark "X" if function is only used in the standard direction, "R" if only used in the reverse direction, and "B" if used in both directions)

- global
- group 1
- group 2
- group 3
- group 4
- group 5
- group 6
- group 7
- group 8
- group 9
- group 10
- group 11
- group 12
- group 13
- group 14
- group 15
- group 16

### 6.8.7 Clock Synchronization

(Station-specific parameter; mark "X" if function is only used in the standard direction, "R" if only used in the reverse direction, and "B" if used in both directions)

- Clock synchronization
- Day of week used
- RES1, GEN (time tag substituted/not substituted) used
- SU-bit (summertime) used (Optional)

### 6.8.8 Command Transmission

(Object-specific parameter; mark "X" if function is only used in the standard direction, "R" if only used in the reverse direction, and "B" if used in both directions)

- Direct command transmission
- Direct set point command transmission
- Select and execute command
- Select and execute set point command
- C\_SE\_ACTTERM used <sup>note 2</sup>
- No additional definition
- Short pulse duration (duration determined by a system parameter in the outstation)
- Long pulse duration (duration determined by a system parameter in the outstation)
- Persistent output
- Supervision of maximum delay in command direction of commands and set point commands  
60 seconds Maximum allowable delay of commands and set point commands

### 6.8.9 Transmission of Integrated Totals

(Station- or object-specific parameter; mark "X" if function is only used in the standard direction, "R" if only used in the reverse direction, and "B" if used in both directions)

- Mode A: Local freeze with spontaneous transmission
- Mode B: Local freeze with counter interrogation
- Mode C: Freeze and transmit by counter-interrogation commands
- Mode D: Freeze by counter-interrogation command, frozen values reported spontaneously
  
- Counter read
- Counter freeze without reset
- Counter freeze with reset
- Counter reset
  
- General request counter
- Request counter group 1
- Request counter group 2
- Request counter group 3
- Request counter group 4

### **6.8.10 Parameter Loading**

(Object-specific parameter; mark "X" if function is only used in the standard direction, "R" if only used in the reverse direction, and "B" if used in both directions)

- Threshold value
- Smoothing factor
- Low limit for transmission of measured value
- High limit for transmission of measured value

### **6.8.11 Parameter Activation**

(Object-specific parameter; mark "X" if function is only used in the standard direction, "R" if only used in the reverse direction, and "B" if used in both directions)

- Act/deact of persistent cyclic or periodic transmission of the addressed object

### **6.8.12 Test Procedure**

(Station-specific parameter; mark "X" if function is only used in the standard direction, "R" if only used in the reverse direction, and "B" if used in both directions)

- Test procedure

### **6.8.13 File Transfer**

(Station-specific parameter; mark "X" if function is used)

File transfer in monitor direction

- Transparent file
- Transmission of disturbance data of protection equipment
- Transmission of sequence of events
- Transmission of sequence of recorded analogue values

File transfer in control direction

- Transparent file

#### **6.8.14 Background Scan**

(Station-specific parameter; mark "X" if function is only used in the standard direction, "R" if only used in the reverse direction, and "B" if used in both directions)

- Background scan

#### **6.8.15 Aquisition of transmission delay**

(station-specific parameter, mark with an "X" if function is used only in the standard direction, "R" if used only in the reverse direction, and "B" if used in both directions)

- Acquisition of transmission delay



## 7 101S v3 Database Design Forms

### In This Chapter

- ❖ M\_SP\_NA, M\_DP\_NA, M\_ST\_NA, M\_BO\_NA, and M\_IT\_NA Form.. 144
- ❖ M\_ME\_NA and M\_ME\_NB Form ..... 145
- ❖ Form for All C\_ (Command) Data Types, Except C\_RC\_NA..... 146
- ❖ C\_RC\_NA Form ..... 147

This section contains a set of forms that can be used to design the databases required by the gateway.













## 8 Support, Service & Warranty

### In This Chapter

- ❖ How to Contact Us: Technical Support..... 149
- ❖ Return Material Authorization (RMA) Policies and Conditions..... 150
- ❖ LIMITED WARRANTY..... 152

### 8.1 How to Contact Us: 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 gateway, 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.

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

---

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

---

## 8.2 Return Material Authorization (RMA) Policies and Conditions

The following RMA Policies and Conditions (collectively, "RMA Policies") apply to any returned Product. These RMA Policies are subject to change by ProSoft without notice. For warranty information, see Limited Warranty (page 152). In the event of any inconsistency between the RMA Policies and the Warranty, the Warranty shall govern.

### 8.2.1 All Product Returns:

- a) In order to return a Product for repair, exchange or otherwise, the Customer must obtain a Return Material Authorization (RMA) number from ProSoft and comply with ProSoft shipping instructions.
- b) In the event that the Customer experiences a problem with the Product for any reason, Customer should contact ProSoft Technical Support at one of the telephone numbers listed above (page 149). A Technical Support Engineer will request that you perform several tests in an attempt to isolate the problem. If after completing these tests, the Product is found to be the source of the problem, we will issue an RMA.
- c) All returned Products must be shipped freight prepaid, in the original shipping container or equivalent, to the location specified by ProSoft, and be accompanied by proof of purchase and receipt date. The RMA number is to be prominently marked on the outside of the shipping box. Customer agrees to insure the Product or assume the risk of loss or damage in transit. Products shipped to ProSoft using a shipment method other than that specified by ProSoft, or shipped without an RMA number will be returned to the Customer, freight collect. Contact ProSoft Technical Support for further information.
- d) A 10% restocking fee applies to all warranty credit returns, whereby a Customer has an application change, ordered too many, does not need, etc. Returns for credit require that all accessory parts included in the original box (i.e.; antennas, cables) be returned. Failure to return these items will result in a deduction from the total credit due for each missing item.

### **8.2.2 Procedures for Return of 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 at designated location referenced on the Return Material Authorization
  - i. If a defect is found and is determined to be customer generated, or if the defect is otherwise not covered by ProSoft's warranty, there will be no credit given. Customer will be contacted and can request module be returned at their expense;
  - ii. If defect is customer generated and is repairable, customer can authorize ProSoft repair the unit by providing a purchase order for 30% of the current list price plus freight charges, duties and taxes as applicable.

### **8.2.3 Procedures for Return of Units Out of Warranty:**

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

**The following is a list of non-repairable units:**

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

### 8.3 LIMITED WARRANTY

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

#### 8.3.1 What Is Covered By This Warranty

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



### **8.3.2 What Is Not Covered By This Warranty**

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

### **8.3.3 Disclaimer Regarding High Risk Activities**

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

### **8.3.4 Intellectual Property Indemnity**

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

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

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

### **8.3.5 Disclaimer of all Other Warranties**

The Warranty set forth in What Is Covered By This Warranty (page 152) 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.

### **8.3.6 Limitation of Remedies \*\***

In no event will ProSoft or its Dealer be liable for any special, incidental or consequential damages based on breach of warranty, breach of contract, negligence, strict tort or any other legal theory. Damages that ProSoft or its Dealer will not be responsible for 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.

### **8.3.7 Time Limit for Bringing Suit**

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

### **8.3.8 No Other Warranties**

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

### **8.3.9 Allocation of Risks**

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

### ***8.3.10 Controlling Law and Severability***

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

## Index

### [

[C\_BO\_NA\_1 104] • 47  
[C\_DC\_NA\_1] • 46  
[C\_RC\_NA\_1] • 47  
[C\_SC\_NA\_1] • 46  
[C\_SE\_NA\_1] • 48  
[C\_SE\_NB\_1] • 48  
[C\_SE\_NC\_1] • 49  
[IEC-870-5-101 Database] • 38  
[IEC-870-5-101 Port 0] • 24  
[M\_BO\_NA\_1 104] • 43  
[M\_DP\_NA\_1] • 42  
[M\_IT\_NA\_1] • 45  
[M\_ME\_NA\_1] • 44  
[M\_ME\_NB\_1] • 44  
[M\_ME\_NC\_1] • 45  
[M\_SP\_NA\_1] • 42  
[M\_ST\_NA\_1] • 43

### 1

101S Protocol Configuration • 23, 67  
101S v3 Database Design Forms • 143

### A

All Product Returns: • 150  
All ProLinx® Products • 3  
Allocation of Risks • 155  
Application Data Area • 65  
Application Layer • 131  
Application Serial Port(s) • 53  
Aquisition of transmission delay • 141  
ASDU Configuration • 74

### B

Background Scan • 141  
Balanced Mode • 79  
Base Module LEDs • 57  
Basic Application Functions • 137  
Baud Rate • 36

### C

C\_BO\_NA • 96  
C\_RC\_NA • 95  
C\_RC\_NA Form • 147  
C\_SC\_NA and C\_DC\_NA • 94  
C\_SE\_NA and C\_SE\_NB • 97  
C\_SE\_NC • 98  
Cause of Trans(mission) Octets • 27  
Cause of Transmission • 131  
Clock Synchronization • 138  
Command Block Functionality • 113

Command Block List • 116  
Command Request Block • 117, 119, 120, 121, 122  
Command Response Block • 117, 118, 119, 120, 121, 122  
Command Transmission • 139  
Common Address of ASDU - 101S • 131  
Common Address of ASDU Len • 25  
Common Address of ASDU Val • 25  
Communication Parameters • 72  
Configuration • 114  
Configuration File • 65  
Configuring Module Parameters • 15  
Control Data Transfer • 91  
Control Points Addressing • 93  
Controlling Law and Severability • 156  
Controlling the Generation of Events • 104  
Creating Optional Comment Entries • 16  
Cyclic Data Transmission • 26, 137  
Cyclic Set IV Time - PLX 101S • 31

### D

Data Flow Between the Module and External Host • 68  
Data link address length • 25  
Data Link Address Value • 25  
Data Link Configuration • 73  
Data Type Level • 105  
Databases • 69  
Deadbands • 100  
Delay Preset • 20  
Diagnostics and Troubleshooting • 9, 57  
Disable Time Sync Events • 25  
Disclaimer of all Other Warranties • 154  
Disclaimer Regarding High Risk Activities • 153  
Double Transmission of Information Objects • 138  
Downloading the Project to the Module • 21

### E

Enabled • 24  
Event Data Command Block • 123  
Event Priority • 30, 31, 108  
Event Scan Delay • 33  
Events • 99  
Example 1 - Command Request Without Response only • 117  
Example 2 - Command Request with Response Only • 119  
Example 3 - Command Request with Response and Command Request Without Response • 121

### F

File Transfer • 134, 140  
Form for All C\_ (Command) Data Types, Except C\_RC\_NA • 146  
Freeze Start Type • 27  
From Address • 18  
Functional Specifications - IEC 60870-5-101 Slave • 64

## G

General Command Block Structure • 115  
General Events (All Points) • 104  
General Parameter Configuration • 72  
General Specifications • 63  
Get Event Buffer Counts Command Block • 127  
Get Time Command Block • 126  
Group Codes • 42, 43, 44, 45, 50, 82  
Guide to the 101S v3 Protocol Manual • 9

## H

Hardware Handshaking • 37  
Hardware Specifications • 66  
How to Contact Us  
    Technical Support • 149, 150

## I

IEC 60870-5-101 Slave Interoperability Statement • 11, 129  
IEC-60870-5-101 Slave Protocol Implementation • 9, 67  
IEC870-5-101 Slave Driver Error/Status Data • 128  
Important Installation Instructions • 3  
Inform. Object Address Len • 26  
Information Object Address • 131  
Installing ProSoft Configuration Builder Software • 12  
Intellectual Property Indemnity • 154  
Internal Database • 64  
Interval for Freeze • 29  
Introduction • 113  
Invalid Bit Monitoring • 110  
IV Check Delay Time • 33  
IV Fail Count • 33

## K

Keystrokes • 61

## L

LED Indicators • 57  
Length of APDU - 101S • 131  
Limitation of Remedies \*\* • 155  
LIMITED WARRANTY • 150, 152  
Link Layer • 131  
Long Pulse Time • 38

## M

M\_BO\_NA • 85  
M\_ME\_NA • 103  
M\_ME\_NA and M\_ME\_NB • 86  
M\_ME\_NA and M\_ME\_NB Form • 145  
M\_ME\_NB • 104  
M\_ME\_NC and M\_IT\_NA • 87  
M\_SP\_NA • 83  
M\_SP\_NA, M\_DP\_NA, M\_ST\_NA, M\_BO\_NA, and M\_IT\_NA Form • 144  
M\_ST\_NA • 84  
Maximum ASDU Resp Len • 27

Mini-DIN8 Serial Port • 58  
Minimum Delay • 36  
Module Communication Ports • 51  
Monitor Data Transfer • 82  
Monitor Direction and Control Direction • 80

## N

Navigation • 61  
Network Configuration • 130  
No Other Warranties • 155

## O

Optional Web Server for Ethernet Solutions • 65

## P

Parameter Activation • 140  
Parameter in Control Direction • 134  
Parameter Loading • 140  
Parameter Offset • 41  
Parity • 36  
Physical Layer • 130  
Pinouts • 3, 51  
Point Count • 39  
Point Level • 105  
Printing a Configuration File • 16  
Procedures for Return of Units Out of Warranty: • 151  
Procedures for Return of Units Under Warranty: • 151  
Process Information in Control Direction • 133  
Process Information in Monitor Direction • 132  
Product Specifications • 9, 63  
ProLinX Gateways with Ethernet Ports • 3  
ProLinX Reference Guide • 11  
ProSoft Technology® Product Documentation • 2

## R

Read Procedure • 137  
Receive Timeout • 37  
Reference • 9, 63  
Register Count • 18  
Renaming PCB Objects • 15  
Request Block • 123, 125, 126  
Response Block • 126  
Response Timeout • 36  
Retry Count • 35  
Return Material Authorization (RMA) Policies and Conditions • 150  
RS-232  
    Modem Connection (Hardware Handshaking Required) • 53  
    Null Modem Connection (Hardware Handshaking) • 53  
    Null Modem Connection (No Hardware Handshaking) • 54  
RS-232 Configuration/Debug Port • 52  
RS-422 • 55  
RS-485 and RS-422 Tip • 55  
RS-485 Application Port(s) • 55  
RTS Off • 36

RTS On • 36

## S

Scan Events • 34  
Select/Operate Timeout • 26  
Selection of Standard ASDUs • 132  
Sequence Flag • 40  
Serial Port Cable Connections  
    Config/Debug and Port 0 • 51  
Set Priority Queues • 30  
Set Time Command Block • 125  
Setting Up the Project • 13  
Short Pulse Time • 38  
Single char ACK C1 or C2 • 27  
Single char ACK F0, 1 or 3 • 26  
Slave Driver • 68  
Slave Error and Status • 128  
Spontaneous Transmission • 137  
Start Here • 9, 11  
Station Initialization • 137  
Station Interrogation • 138  
Status Data Area • 65  
Support, Service & Warranty • 9, 149  
Swap Code • 18  
System Information in Control Direction • 134  
System Information in Monitor Direction • 134  
System or Device • 130

## T

Test Procedure • 140  
Time DB Offset • 24  
Time Information • 106  
Time Limit for Bringing Suit • 155  
Time Type • 35  
To Address • 18  
To order a ProLinx Plus gateway with the -WEB option  
    • 3  
To upgrade a previously purchased Series C model: •  
    3  
Transmission Mode for Application Data • 131  
Transmission of Integrated Totals • 139  
Transmission Speed (Control Direction) • 130  
Transmission Speed (Monitor Direction) • 130  
Type Identifier and Cause of Transmission  
    Assignments • 135

## U

Unbalanced mode • 78  
Use ACTTERM with Set Point • 26  
Use ACTTERM with Step • 26  
Use Balanced Mode • 35  
Using Control (Command) Points • 88  
Using Monitor Points • 81  
Using ProSoft Configuration Builder • 13  
Using ProSoft Configuration Builder (PCB) for  
    Diagnostics • 59  
Using the CommonNet Data Map • 17  
Using the Diagnostic Window in ProSoft Configuration  
    Builder • 59

Using the Online Help • 12

## W

What Is Covered By This Warranty • 152, 154  
What Is Not Covered By This Warranty • 153  
What's New • 11

## Y

Your Feedback Please • 2