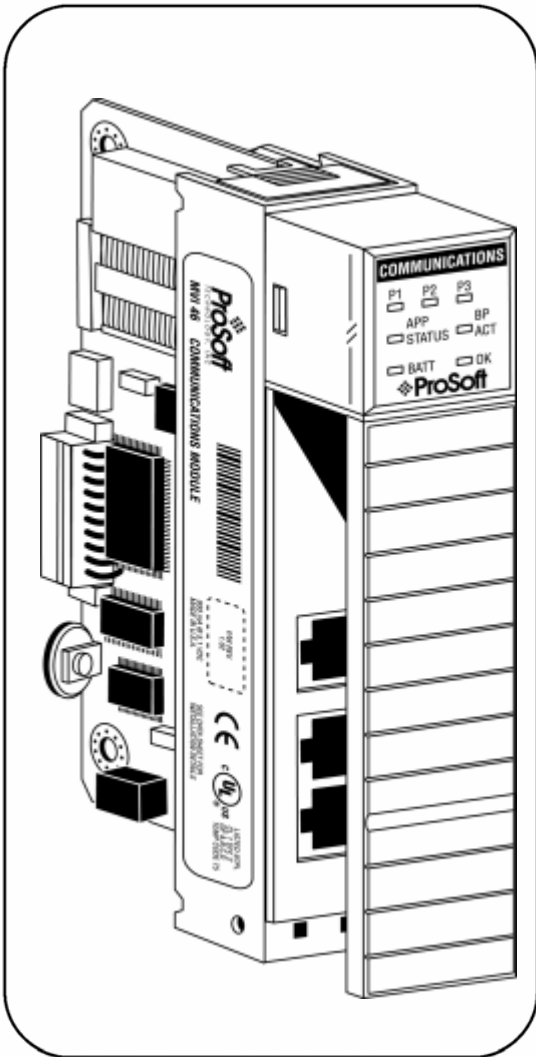


# inRAx



## MVI46-103M

SLC Platform

IEC 60870-5-103 Master Communication  
Module

User Manual

September 12, 2006

**ProSoft**  
TECHNOLOGY

# Please Read This Notice

Successful application of this module requires a reasonable working knowledge of the Rockwell Automation SLC hardware, the MVI46-103M Module and the application in which the combination is to be used. For this reason, it is important that those responsible for implementation satisfy themselves that the combination will meet the needs of the application without exposing personnel or equipment to unsafe or inappropriate working conditions.

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

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

Reproduction of the contents of this manual, in whole or in part, without written permission from ProSoft Technology, Inc. is prohibited.

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

## Your Feedback Please

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

### **ProSoft Technology, Inc.**

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

Copyright © ProSoft Technology, Inc. 2000 - 2006. All Rights Reserved.

MVI46-103M User Manual  
September 12, 2006

# Contents

|   |           |
|---|-----------|
| <b>PLEASE READ THIS NOTICE</b> .....                              | <b>2</b>  |
| <b>Your Feedback Please</b> .....                                 | <b>2</b>  |
| <b>1 PRODUCT SPECIFICATIONS</b> .....                             | <b>7</b>  |
| <b>1.1 General Specifications</b> .....                           | <b>7</b>  |
| 1.1.1 Physical.....   | 7         |
| 1.1.2 SLC Interface .....   | 8         |
| <b>1.2 Hardware Specifications</b> .....                          | <b>8</b>  |
| <b>2 QUICK START</b> .....  | <b>9</b>  |
| <b>3 FUNCTIONAL OVERVIEW</b> .....                                | <b>15</b> |
| <b>3.1 General Concepts</b> .....                                 | <b>15</b> |
| 3.1.1 Module Power Up .....                                       | 15        |
| 3.1.2 Main Logic Loop.....  | 16        |
| 3.1.3 SLC Processor Not in Run.....                               | 16        |
| 3.1.4 Backplane Data Transfer .....                               | 17        |
| 3.1.5 Data Types and Mapping.....                                 | 17        |
| 3.1.6 Command Control Blocks .....                                | 20        |
| <b>3.2 Master Driver</b> .....                                    | <b>27</b> |
| <b>4 MODULE CONFIGURATION</b> .....                               | <b>29</b> |
| <b>4.1 Installing and Configuring the Module</b> .....            | <b>29</b> |
| <b>4.2 Module Data</b> .....                                      | <b>31</b> |
| <b>4.3 Configuration File</b> .....                               | <b>31</b> |
| <b>4.4 Uploading and Downloading the Configuration File</b> ..... | <b>35</b> |
| 4.4.1 Transferring the Configuration File to Your PC.....         | 35        |
| 4.4.2 Transferring the Configuration File to the Module .....     | 37        |
| <b>5 LADDER LOGIC</b> .....                                       | <b>41</b> |

---

|          |   |           |
|----------|---|-----------|
| 5.1      | <b>Main Routine</b> .....   | 41        |
| 5.2      | <b>Data Transfer (U:3)</b> .....  | 42        |
| 5.3      | <b>Control Routine (U:4)</b> .....  | 43        |
| <b>6</b> | <b>DIAGNOSTICS AND TROUBLESHOOTING</b> .....                                | <b>47</b> |
| 6.1      | <b>The Configuration/Debug Menu</b> .....                                   | <b>47</b> |
| 6.1.1    | Navigation.....   | 47        |
| 6.2      | <b>Required Hardware</b> .....  | <b>48</b> |
| 6.3      | <b>Required Software</b> .....  | <b>49</b> |
| 6.4      | <b>Using the Configuration/Debug Port</b> .....                             | <b>49</b> |
| 6.4.1    | Main Menu.....  | 50        |
| 6.4.2    | Database View Menu .....  | 54        |
| 6.4.3    | IEC-103 Master Driver Menu.....   | 56        |
| 6.4.4    | IEC-870-Master Command List Menu.....                                       | 60        |
| 6.4.5    | Port Configuration Menu .....   | 61        |
| 6.4.6    | Port Status Menu.....   | 62        |
| 6.4.7    | Data Analyzer .....   | 63        |
| 6.4.8    | Session Configuration Menu .....  | 66        |
| 6.4.9    | Sector Configuration Menu.....  | 67        |
| 6.4.10   | Sector Database Menu.....   | 68        |
| 6.5      | <b>LED Status Indicators</b> .....  | <b>69</b> |
| 6.5.1    | Clearing a Fault Condition.....   | 70        |
| 6.5.2    | Troubleshooting.....  | 70        |
| <b>7</b> | <b>REFERENCE</b> .....  | <b>73</b> |
| 7.1      | <b>Cable Connections</b> .....  | <b>73</b> |
| 7.1.1    | RS-232 .....  | 74        |
| 7.1.2    | RS-232 Configuration/Debug Port.....  | 76        |
| 7.1.3    | RS-485 .....  | 76        |
| 7.1.4    | RS-422 .....  | 77        |
| 7.2      | <b>Setting Jumpers</b> .....  | <b>77</b> |
| 7.3      | <b>MVI46-103M Status Data Area</b> .....                                    | <b>78</b> |
| 7.3.1    | MVI46-IEC 60870-5-103 Master Communication Module Error/Status Data Format. | 78        |

---

|            |   |            |
|------------|---|------------|
| 7.3.2      | MVI46-IEC 60870-5-103 Master Communication Module Error Codes .....   | 80         |
| <b>7.4</b> | <b>MVI46-103M Configuration Data Definition.....</b>                  | <b>82</b>  |
| 7.4.1      | MVI46 IEC 60870-5-103 Master Communication Module Configuration ..... | 82         |
| <b>7.5</b> | <b>MVI46-103M Configuration File Example .....</b>                    | <b>86</b>  |
| <b>7.6</b> | <b>Database Form.....</b>   | <b>158</b> |
| <b>7.7</b> | <b>Command List Form.....</b>   | <b>160</b> |
| <b>7.8</b> | <b>Protocol Support .....</b>   | <b>162</b> |
| 7.8.1      | List of Type Identification Codes.....                                | 162        |
| 7.8.2      | List of Cause of Transmission Codes .....                             | 163        |
| 7.8.3      | List of Function Types.....   | 164        |
| 7.8.4      | Information Numbers Used In Monitor Direction.....                    | 164        |
| 7.8.5      | Information Numbers Used In Control Direction .....                   | 167        |
| 7.8.6      | Definition and Presentation of ASDU'S In Monitor Direction .....      | 168        |
| 7.8.7      | Definition and Presentation Of ASDU'S In Control Direction.....       | 171        |
| <b>7.9</b> | <b>Protocol Interoperability Documentation.....</b>                   | <b>172</b> |
| 7.9.1      | Physical Layer.....   | 173        |
| 7.9.2      | Electrical Interface .....  | 173        |
| 7.9.3      | Optical Interface.....  | 173        |
| 7.9.4      | Transmission speed.....   | 174        |
| 7.9.5      | Link Layer.....   | 174        |
| 7.9.6      | Application Layer .....   | 174        |
| 7.9.7      | Transmission mode for application data.....                           | 174        |
| 7.9.8      | Common address of ASDU.....   | 174        |
| 7.9.9      | Selection of standard information numbers in monitor direction.....   | 174        |
| 7.9.10     | System functions in monitor direction .....                           | 174        |
| 7.9.11     | Status indications in monitor direction .....                         | 174        |
| 7.9.12     | Supervision indications in monitor direction.....                     | 175        |
| 7.9.13     | Earth fault indications in monitor direction .....                    | 176        |
| 7.9.14     | Fault indications in monitor direction .....                          | 176        |
| 7.9.15     | Auto-reclosure indications in monitor direction.....                  | 177        |
| 7.9.16     | Measurands in monitor direction.....                                  | 177        |

---

|   |  |            |
|---|--|------------|
| 7.9.17  | Generic functions in monitor direction .....                         | 177        |
| 7.9.18  | Selection of standard information numbers in control direction ..... | 178        |
| 7.9.19  | System functions in control direction.....                           | 178        |
| 7.9.20  | General commands in control direction.....                           | 178        |
| 7.9.21  | Generic functions in control direction .....                         | 178        |
| 7.9.22  | Basic application functions .....                                    | 179        |
| 7.9.23  | Miscellaneous.....   | 179        |
| <b>SUPPORT, SERVICE &amp; WARRANTY .....</b>                |  | <b>181</b> |
| <b>Module Service and Repair .....</b>                      |  | <b>181</b> |
| <b>General Warranty Policy – Terms and Conditions .....</b> |  | <b>182</b> |
| <b>Limitation of Liability.....</b>                         |  | <b>183</b> |
| <b>RMA Procedures .....</b>                                 |  | <b>183</b> |
| <b>INDEX.....</b>   |  | <b>185</b> |

# 1 Product Specifications

## *In This Chapter*

- General Specifications ..... 7
- Hardware Specifications ..... 8

The MVI46-103M ("IEC 60870-5-103 Master Communication Module") allows Rockwell Automation SLC I/O compatible processors to interface easily with IEC 60870-5-103 slave (controlled unit) protection devices. Compatible devices include relays, breakers, sub-station communication modules and other serial communication devices.

## 1.1 General Specifications

The MVI46-103M module interfaces up to 32 serial communication devices with the Rockwell Automation SLC processor. Two communication ports on the module act as controlling devices (masters) to interface with controlled devices on their own networks. Each port is individually configurable. Data is exchanged between the serial network and the Rockwell Automation processor using the internal database contained in the module and direct control by the controller's ladder logic.

Some of the general specifications include:

- Built in accordance to the approved international specification
- Two independent master ports completely user configurable
- Up to 32 sessions (controlled devices)
- Up to five sectors (separate databases) for each session
- Individual database definition for each sector
- 1000 commands to control stations
- SLC processor can issue control commands directly to the module or a controlled device
- Pass-through of event messages from controlled device to processor for logging of time-tagged events

### 1.1.1 *Physical*

This module is designed by ProSoft Technology and incorporates licensed technology from Rockwell Automation (SLC backplane technology).

- SLC Form Factor - Single Slot

Connections:

- 1 - RJ45 RS-232 Configuration Tool Connector
- 2 - RJ45 RS-232/422/485 Application ports

### **1.1.2 SLC Interface**

- Operation via simple ladder logic
- Complete set up and monitoring of module through RSLogix 500 software and user constructed configuration file (IEC103M.CFG)
- SLC backplane interface via M-File access
- All data related to the module is contained in user data files to ease in the monitoring and interfacing with the module
- Control of module and controlled devices on serial network available from ladder logic

## **1.2 Hardware Specifications**

The MVI46-103M module is designed by ProSoft Technology and incorporates licensed technology from Rockwell Automation (SLC backplane technology).

- Current Loads: 800 ma @ 5V (from backplane)
- Operating Temperature: 0 to 60°C (32 to 140°F)
- Storage Temperature: -40 to 85°C (-40 to 185°F)
- Relative Humidity: 5 to 95% (non-condensing)
- Configuration Connector: RJ45 RS-232 Connector (RJ45 to DB-9 cable shipped with unit)
- Application Port Connector: RJ45-RS-232/422/485 Connector (RJ45 to DB-9 cables shipped with unit)



## 2 Quick Start

This section provides the steps required to configure the module. After you download the sample configuration file, please perform the following steps:

### Step 1 - Configure the Number of Slaves (Sessions)

The IEC 60870-5-103 protocol is a master-slave protocol where the slaves are typically protection equipments for substations. The MVI46-103M module supports up to 32 slaves (total) connected to its ports. Each slave has to be configured as a *session*. Refer to the configuration file to enter the number of slaves that will be connected to the MVI46-103M module:

```
[IEC-870-5-103 Master]
Session Count : 5 #1 to 32 - maximum number of slaves on all channels
```

In the example above, the module will only poll sessions 0 to 4. The module would not poll sessions 5 to 15.

*In the Step 3, the user will configure each session as an actual slave in the network.*

### Step 2 - Configure the Port Communication Parameters

The user should configure the port communication parameters in order to enable data transfer between the master and the slave(s). The port communication parameters include: baud rate, parity, RTS ON, RTS OFF and Minimum Delay. The IEC 60870-5-103 protocol uses two baud rates: 19200 or 9600 kb/s and even parity.

Refer to the [IEC-870-5-103 Master Port 0] section in the configuration file in order to configure the communication parameters for the 103M port:

```
[IEC-870-5-103 Master Port 0]
# Communication Parameters
Baud Rate      : 19200 #Baud rate for port 9600-19200
Parity         : E #N=None, O=Odd, E=Even, M=Mark, S=Space
RTS On         : 0 #0-65536 mSec before message
RTS Off        : 1 #0-65536 mSec after message
Minimum Delay  : 10 #Minimum # of mSec before response sent
Receive Timeout : 2000 #Maximum mSec from first char to last to wait
# These parameters are protocol specific
Single char ACK F0,1 or 3 : Y #Single E5 resp to ACK func 0, 1 & 3 req (Y/N)
```

The user should also configure the jumpers located at the back at the module in order to select the correct communication mode: RS-232, RS-422 or RS-485.

### Step 3 - Configure the Session (Slave) Poll Parameters

According to the IEC 60870-5-103 protocol, the master cyclically polls data from the slaves. The data is classified into two classes; Class 1 and Class 2. Events belong to Class 1 and analog data to Class 2. The module can request data through Class 1 or Class 2 requests. Responses to control command and general interrogation commands are also sent as Class 1 data.

Please refer to the [IEC-103 Master Session x] section in the configuration file in order to configure how each slave will be polled.

Initially, the user should enter the MVI46-103M port number that will be connected to the session (slave) using the "Communication Port" parameter. Valid values are 0 or 1.

These parameters include the *Data Link Address*, which is the slave address that identifies all protection equipment on the network. There should be a unique number for each slave on the network. There are also certain parameters that pertain to how the Class 1 and Class 2 polls will be used for data transfer.

You must enter the number of sectors for each session using the Sector Count parameter. The module accepts up to 3 sectors per session.

```
[IEC-103 Master Session 0]
Communication Port      :      0 #Index of COM port for session (0 or 1)
Sector Count           :      5 #5 is max for this version of app
Data Link Address      :      0 #Range is 0 to 65535 DL address of slave

Failure Delay          :      3 #Min Sec to delay before poll of offline slave
                        :      #(0 to 2000 seconds)
Confirm Timeout        :    20000 #0 to 2^32-1 mSec to wait for DL confirm
Retry Count            :      5 #0 to 255 retries for if no confirm
Response Timeout       :    5000 #Timeout for confirm of req (0 to 2^32-1)
C1/C2 Poll Count Pend :      6 #class 1 or 2 polls before next slave tried (0-
65535)
Class 1 Polls          :     10 #Max class 1 polls to this session
Class 1 Pend delay     :    1000 #Min mSec delay between call (0 to 2^32-1)
Class 2 Pend delay     :    1000 #Min mSec delay between call (0 to 2^32-1)
Class 1 Poll delay     :    1000 #Min mSec delay between call (0 to 2^32-1)
Class 2 Poll delay     :    1000 #Min mSec delay between call (0 to 2^32-1)
```

This step should be repeated for each session to be used. For example, if the user selected 8 sessions during **Step 1**, he or she should configure sessions 0 to 7:

```
[IEC-103 Master Session 0]
[IEC-103 Master Session 1]
[IEC-103 Master Session 2]
[IEC-103 Master Session 3]
[IEC-103 Master Session 4]
[IEC-103 Master Session 5]
[IEC-103 Master Session 6]
[IEC-103 Master Session 7]
```

## Step 4 - Sector (Data Set) Configuration

For each session (slave), you must configure one or more sectors (maximum of 5). A sector is a data set defined by the vendor. Each sector is identified by the Common ASDU Address parameter in the [IEC-103 Master Session x Sector 0] area in the configuration file. This area also contains some parameters that will affect the module initialization procedure.

```
[IEC-103 Master Session 0 Sector 0]
Common ASDU Address      :      0 #Range 0 to 255 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync         :      Y #Send time sync message when first online
Online General Int       :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync            :      Y #Send time sync message when first online
EOI General Int          :      Y #Send general interrogation

# ASDU      Database  Function   Point
# Type      Index     Code       Index
START
    1        0         128        16
    1        16        128        18
    1        32        128        19
    1        48        128        20
    1        64        128        21
    1        80        128        22
    1        96        128        23
    1       112        128        24
    1       128        128        25
    1       144        128        26
END
```

This step should be repeated for each sector used by the application. The module will only use the sectors configured in the previous step.

## Step 5 - Monitor Point Configuration (Monitor Direction)

After the slave receives a Class 1 or Class 2 request from the master, it responds with a message containing data. Each piece of equipment is normally configured to respond with specific points when it is being polled with a Class 2 request. During a Class 2 response, the slave may set a control bit (ACD) to inform the master that there are new events to be transmitted. Then, the master will send a Class 1 poll to read the events from the slave.

The IEC 60870-5-103 protocol states that the data is transferred between the master and slave using an ASDU (Application Service Data Unit) format. Each format is given by:

- Type Identification
- Variable Structure Qualifier

- Type Identification
- Cause Of Transmission
- Common Address of ASDU
- Function Type
- Information Number
- Data...
- Data...
- ...

The user should refer to the protection equipment specification for the following information about each point:

- Type: Type of the message
- Function Type: Type of protection function
- Information Number: Point Identification

This information will identify each point in the MVI46-103M configuration file. The user has to configure the points that will be updated in the MVI46-103M database when a Class 2 or Class 1 response containing data is sent from the slave. The user can refer to [IEC-103 Master Session x Sector y] section in the configuration file in order to configure each point:

```
# ASDU      Database  Function  Point
# Type      Index      Code      Index
START
    1         0        128       16
    1         16       128       17
    1         32       160       18
    1         48       176       19
    1         64       192       20
    1         80       128       21
END
```

Where the user should enter:

**ASDU Type:** ASDU type for the point

**Function Type:** Function type for the point

**Point Index:** Information number for the point

**Database Index:** The MVI46-103M database location where the value will be copied. Special attention should be considered since the type of addressing will depend on the ASDU type:

| ASDU Type | DB Addressing                                  |
|-----------|--|
| 1         | Bit address with each point occupying 2 bits   |
| 2         | Bit address with each point occupying 2 bits   |
| 3         | Word address with each point occupying 4 words |
| 4         | Double-word address for the single float value |

| ASDU Type | DB Addressing                                   |
|-----------|---|
| 5         | Byte address with each point occupying 12 bytes |
| 9         | Word address with each point occupying 9 words  |

For example, in order to configure the following points:

- Time-tagged message point with information number 17 (teleprotection active) and distance protection function (128). The value will be copied to bits 0 and 1 in word 1 (second word) in the module's database.
- Measurands I point with information number 144 (measurands I) and overcurrent protection function (160). The value will be copied to word 50 in the module's database.

The following configuration information should be entered:

```
# ASDU   Database  Function  Point
# Type   Index      Code      Index
START
    1      16       128      17
    1      50       160     144
END
```

Every time the module responds with a Class 1 or Class 2 poll with these points, the module will update its value to the database.

All the points configured in this section are sent from the slave to the master. The protocol specification refers to this data flow as the *Monitor Direction*.

This step should be repeated for each sector.

### Step 6 - Command Configuration (Control Direction)

The user might also configure the master to send commands to slaves. The IEC 60870-5-103 protocol specification refers to this data flow as *Control Direction*. The commands include general commands, interrogation requests, and time synchronization requests. In order to configure a command, the user should refer to the [IEC-103 Master Commands] section:

```
[IEC-103 Master Commands]
# Enable Database  Poll Session Sector Data  Func  Point Ovrdr Ovrdr
# Code  Index Interval  Index  Index Type  Code  Index Flag  Val
START
    1    0      0      0      0  6 255  0  0  0
    1   10      0      0      0  7 255  0  0  0
END
```

When sending a General Command, the user might associate the source data with a register in the MVI46-103M database to be sent to the remote slave. The following example will send 8 commands to the slave configured as Session 0/Sector 0. When using a General Command, the bit addressing should be used:

```
# Enable   DB      Poll  Session  Sector  Data  Func  Point  Ovrdr  Ovrdr
# Code  Index  Interval  Index  Index  Type  Code  Index  Flag  Val
START
```

|   |       |   |   |   |    |     |    |   |   |
|---|-------|---|---|---|----|-----|----|---|---|
| 1 | 16000 | 0 | 0 | 0 | 20 | 128 | 16 | 0 | 0 |
| 1 | 16016 | 0 | 0 | 0 | 20 | 128 | 17 | 0 | 0 |
| 1 | 16032 | 0 | 0 | 0 | 20 | 128 | 18 | 0 | 0 |
| 1 | 16048 | 0 | 0 | 0 | 20 | 128 | 19 | 0 | 0 |
| 1 | 16064 | 0 | 0 | 0 | 20 | 128 | 23 | 0 | 0 |
| 1 | 16080 | 0 | 0 | 0 | 20 | 128 | 24 | 0 | 0 |
| 1 | 16096 | 0 | 0 | 0 | 20 | 128 | 25 | 0 | 0 |
| 1 | 16112 | 0 | 0 | 0 | 20 | 128 | 26 | 0 | 0 |

END

The user should refer to the device specification for the Point Index (Information Number) listing available for control direction.

The module can also send a periodic General Interrogation command in order to initialize and refresh the event-updated points in its database. The slave keeps a list of all data subject to General Interrogation.

**Step 7 - Transfer the Configuration (on page 37) from the Computer to the module.**

### 3 Functional Overview

*In This Chapter*

- General Concepts ..... 15
- Master Driver..... 27

This chapter provides an overview of how the MVI46-103M module transfers data using the 103M protocol. You should understand the important concepts in this chapter before you begin installing and configuring the module.

The standards used to build the module are listed in the following table:

| PUBLICATION             | TITLE   |
|-------------------------|---|
| IEC 60870-5-103         | Companion Standard for the informative interface of protection equipment. |
| IEC 60870-5-103 Annex A | Generic functions --Examples of constructing a directory                  |
| 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   |

These documents should be obtained, reviewed, and understood in order to fully appreciate the protocol implementation. Most of the complexity of the protocol is hidden from the user and simplified in the application of the module. Detailed questions of about the protocol can be answered by reading these documents. In addition to calling our technical support group, there is also help available for the protocol using the following mail list Web Site:

***www.TriangleMicroWorks.com/iec870-5***  
(<http://www.trianglemicroworks.com/iec870-5>). Go to this site to join the mail list and to review questions and answers from mail list users.

#### 3.1 General Concepts

The following discussion explains several concepts that are important for understanding the operation of the MVI46-103M module.

##### 3.1.1 **Module Power Up**

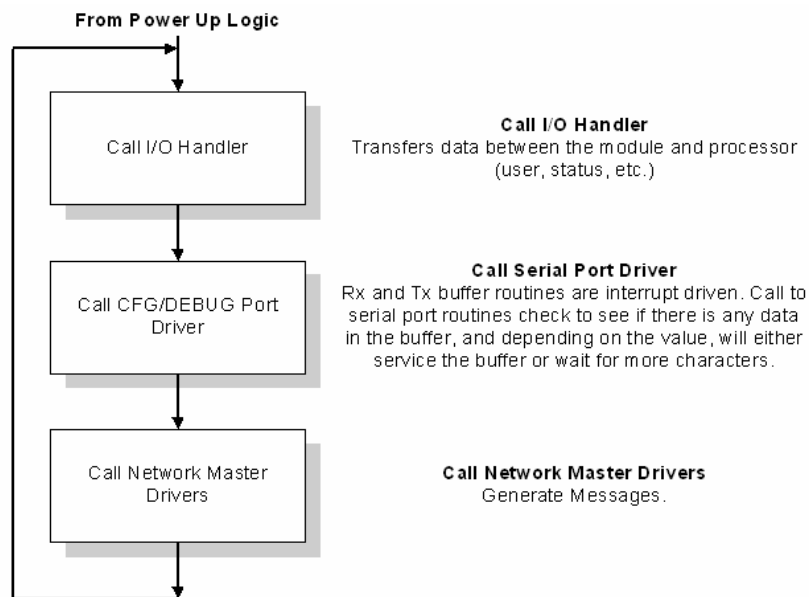
On power up the module begins performing the following logical functions:

- 1 Initialize hardware components
  - Initialize SLC backplane driver
  - Test and clear all RAM
  - Initialize the serial communication ports
- 2 Read configuration for module from IEC103M.CFG file on Compact Flash Disk
- 3 Initialize the databases and ports
- 4 Set up the serial communication interface for the debug/configuration port

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

### 3.1.2 Main Logic Loop

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



### 3.1.3 SLC Processor Not in Run

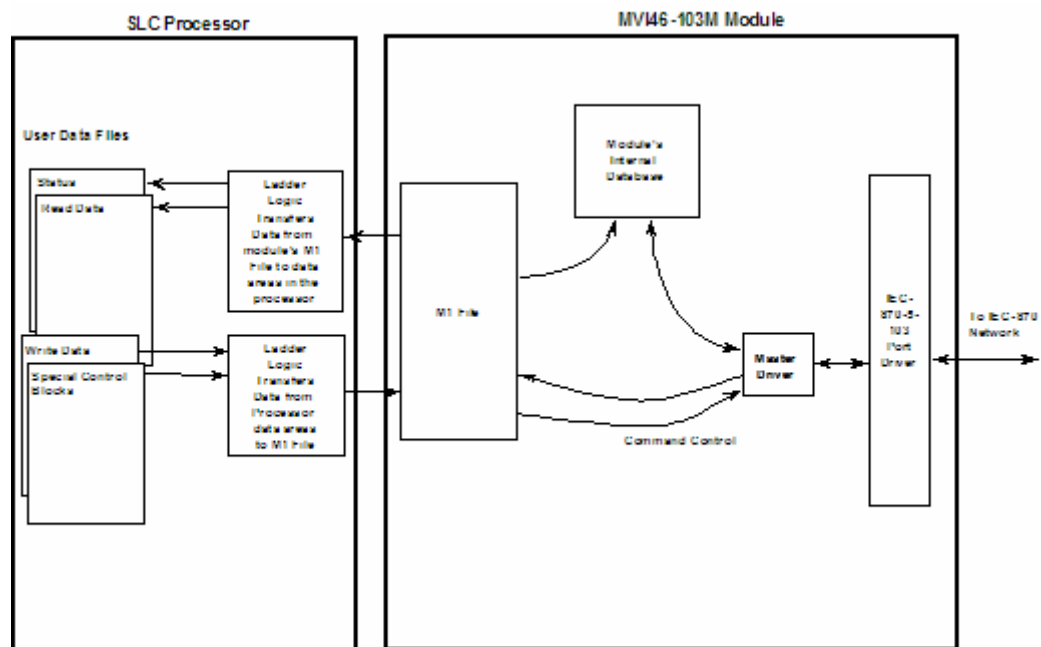
Whenever the module detects that the processor has gone out of the Run mode (that is, Fault or PGM), the protocol ports can be shut down as prescribed in the user configuration. When the processor is returned to a running state, the module will resume communications on the network.



### 3.1.4 Backplane Data Transfer

The MVI46-103M module is unique in the way it utilizes the SLC backplane. All data for the module is contained in the module's M1 file. Data is moved between the module and the SLC processor across the backplane using the module's M1 file. The SLC scan rate and the communication load on the module determine the update frequency of the M1 file. The COP instruction can be used to move data between user data files and the module's M1 file.

The following illustration shows the data transfer method used to move data between the SLC processor, the MVI46-103M module and the IEC 60870-5-103 network.



As shown in the previous diagram, all data transferred between the module and the processor over the backplane is through the M1 file. Ladder logic must be written in the SLC processor to interface the M-file data with data defined in the user-defined data files in the SLC. All data used by the module is stored in its internal database. Data contained in this database is constantly updated with the M1 file data by the module and requires no SLC ladder logic to implement. The user database resides in the M1 file at addresses 0 to 3999. Addresses above 3999 are used for special block control of the module.

### 3.1.5 Data Types and Mapping

When interfacing data in the processor to that of the IEC 60870-5-103 protocol, it is important that the user understand the mapping of the data types to their corresponding representation in the modules database. The table that follows

lists the data types supported by the module and their associated storage representation:

| Type ID | Description   | Data Representation  |
|---------|---|--|
| 1       | Time-tagged messages with each data point represented by two bits.  | Dual-bit status (7.2.6.5 with 00b<br>(0 decimal) = not used<br>01b (1 decimal) = Off,<br>10b (2 decimal) = On and<br>11b (3 decimal) = not used                                  |
| 2       | Time-tagged messages with relative time with each point represented by two bits.  | Dual-bit status (7.2.6.5 with 00b<br>(0 decimal) = not used<br>01b (1 decimal) = Off,<br>10b (2 decimal) = On and<br>11b (3 decimal) = not used                                  |
| 3       | Measurands with quality descriptor. The lower 3 bits of the values represented in this data type contain status information. The upper 13 bits of the value contained a signed, 12-bit number. This data type will return from 1 to 4 values. The number of words received is dependant on the information object number and the slave device.  | Measurand with quality descriptor (7.2.6.8)<br>Bit 0: 0 = No overflow, 1 = Overflow<br>Bit 1: 0 = Valid, 1 = Invalid<br>Bit 2: Reserved<br>Bits 3-25: Value from $-1..+1-2^{12}$ |
| 4       | Time-tagged measurands with relative time with the value in the packet represented by a single floating point number.*  | Short floating-point number stored in IEEE STD 754 format (Fraction, Exponent, Sign) (7.2.6.20)  |
| 5       | Identification data composed of 12 characters of data. Each point in defined of this data type should reserve 12 bytes (6-word addresses) in the database for the data received.  | Byte data as defined in 7.2.6.2. First 8 bytes are characters 1 to 8 and last 4 bytes are manufacture bytes either decimal (0-255) or as ASCII characters.                       |
| 9       | Measurands with quality descriptor. The lower 3 bits of the values represented in this data type contain status information. The upper 13 bits of the value contained a signed, 12-bit number. This data type will return from 1 to 9 values (some slaves may return up to 16 values). The number of words received is dependant on the information object number and the slave device. | Measurand with quality descriptor (7.2.6.8)<br>Bit 0: 0 = No overflow, 1 = Overflow<br>Bit 1: 0 = Valid, 1 = Invalid<br>Bit 2: Reserved<br>Bits 3-25: Value from $-1..+1-2^{12}$ |
| 20      | General command to control a dual-point object. Each command issued by the module uses the values of two adjacent bits in the database or an override value specified by the user command.  | Dual-bit status (7.2.6.4 with 00b<br>(0 decimal) = not used<br>01b (1 decimal) = Off,<br>10b (2 decimal) = On and<br>11b (3 decimal) = not used                                  |

\* The words should be swapped in the ladder logic.

As shown in the table above, all bit types are addressed as bits in the modules database.

### Addressing the Data Types

The following table shows an example of how to address the data types in the MVI46-103M:

| Data Type | Address Type | Length  | Example   |
|-----------|--------------|---------|---|
| 1         | Bit          | 2 bits  | Address 160 refers to first and second bits from word 10.                 |
| 2         | Bit          | 2 bits  | Address 160 refers to first and second bits from word 10.                 |
| 3         | Word         | 4 words | Address 50 refers to word 50, 51, 52, and 53.                             |
| 4         | Double-word  | 2 words | Address 40 refers to two consecutive words starting at word 80.           |
| 5         | Bit          | 12 bits | Address 180 refers to 12 consecutive bits starting at the LSB of word 90. |
| 9         | Word         | 5 words | Address 100 refers to 9 consecutive words starting at word 100.           |
| 20        | Bit          | 2 bits  | Address 160 refers to the first and second bits from word 10.             |

Therefore, address 16000 represents bit zero in word 1000 of the module's database. Short floating-point, 32-bit strings and integrated total values each occupy a double-word space in the database. Therefore, short float database address of 100 represents the two words, 200 and 201, in the modules database. Identification objects are stored as byte values in the modules database. Identification object address 1000 is stored in the module's database in word addresses 500 to 505.

When setting the monitored data and commands, each point is defined by its ASDU type, function code, and information number. Valid function codes are listed in the following table:

| Function Code                       | Symbol | Number |
|-------------------------------------|--------|--------|
| Distance Protection                 | T(z)   | 128    |
| Over-current Protection             | I>>    | 160    |
| Transformer Differential Protection | rlt    | 176    |
| Line Differential Protection        | rll    | 192    |
| Global Function Type                | GLB    | 255    |

Refer to Protocol Interoperability Documentation for a full listing of the protocol support offered by the module.

Ladder logic in the SLC can control the module or devices on the serial network using special control blocks. The following table lists the special control codes (block numbers) used by the module:

---

| <b>Block Code</b> | <b>Descriptions</b>                                       |
|-------------------|---|
| 9901              | User Constructed Command                                  |
| 9902              | Command Control Block (Add command to Command List Queue) |
| 9903              | Event Messages from Master port                           |
| 9950              | Command List Error data                                   |
| 9970              | Set SLC time using module's time                          |
| 9971              | Set module's time using SLC time                          |
| 9998              | Warm Boot Request from SLC (Block contains no data)       |
| 9999              | Cold Boot Request from SLC (Block contains no data)       |

---

Registers 4000 to 4099 report module status data. This data area should be copied to a user file for use in the SLC. **MVI46-103M Status Data Area** (on page 77) contains a listing of the contents of this data area.

Registers 4200 to 4299 are utilized for the transfer of event messages from the master driver to the SLC. These data are passed to the module when a value of 9903 is placed in register 4200. The ladder logic should copy the information contained in the block of data and then set register 4200 to 0 to inform the module that the messages have been accepted.

Registers 4100 to 4199 are used control of the module by the ladder logic. For example,

if the processor places a value of 9998 in register 4100, the module will perform a warm-boot operation. If the processor places a value of 9999 in this register, the module will perform a cold-boot operation. In this application module, both of these operations perform the same function. They exit the program and then restart the program. Many of the program parameters set in the user configuration must be set at program initialization and cannot be set while the program is running. Therefore, both functions operate the same.

The command functions supported by the module and there data formats are discussed in the following section.

### **3.1.6 Command Control Blocks**

Block identification codes greater than 9900 are utilized to perform special functions in the module when placed in the M-file's 4100 register. Each control block recognized and used by the module is defined in the following sections:

#### *User Constructed Command Block (9901)*

Block identification code 9901 issues one or more user constructed commands. When the module receives a block 9901 identification code, it will place the included commands into the command queue.

| Word Offset in Block | Data Field(s) | Description   |
|----------------------|---------------|---|
| 4100                 | Block ID      | This field contains the block identification code of 9901 for the block.  |
| 4101                 | Command Count | This field defines the number of user commands contained in the block. The valid range for the field is 1 to 8. |
| 4102 to 4111         | Command #1    | Data required to build the user defined command in the command queue.   |
| 4112 to 4121         | Command #2    | Data required to build the user defined command in the command queue.   |
| 4122 to 4131         | Command #3    | Data required to build the user defined command in the command queue.   |
| 4132 to 4141         | Command #4    | Data required to build the user defined command in the command queue.   |
| 4142 to 4151         | Command #5    | Data required to build the user defined command in the command queue.   |
| 4152 to 4161         | Command #6    | Data required to build the user defined command in the command queue.   |
| 4161 to 4171         | Command #7    | Data required to build the user defined command in the command queue.   |
| 4172 to 4181         | Command #8    | Data required to build the user defined command in the command queue.   |

The following fields are used for each 10-word record in the command list:

| Word Offset | Definitions                          | Description   |
|-------------|--------------------------------------|---|
| 0           | Database Index                       | Address in module to associate with the command.                        |
| 1           | Session Index                        | Session index defined in the module to associate with the command.      |
| 2           | Sector Index                         | Sector index for session as defined in the module.                      |
| 3           | Data Type                            | ASDU data type associated with the command.                             |
| 4           | Function Code                        | Function Code for the command.  |
| 5           | Point Index *<br>*Information Number | Information object address for the point on which the command operates. |
| 6           | Override Flag                        | Override flag for general command.                                      |
| 7           | Override Value                       | Override value for general command.                                     |
| 8           | Reserved                             | Reserved for future use.  |
| 9           | Reserved                             | Reserved for future use.  |

Refer to the command list section of this documentation for a detailed definition of the fields contained in this block. They are the same as those used in constructed the commands in the command list.

There is no response block built by the module to send back to the processor after the block is processed. The module will set register 4100 to a value of zero after the commands have been processed. The commands are placed in the command queue and issued at a high priority.

This block and block 9902 should be used when controlling double-point data points in remote units using general commands. This provides complete control of the slave devices under ladder logic control. Alternatively, the slaves can be controlled by changing data in the database and having the data be transferred using pre-constructed commands in the user's command list. Some points only accept value of on for control (that is, LED reset or activate characteristic). For these points, block 9901 and 9902 should only be utilized.

Command Control Block (9902)

The block 9902 identification code is used by the processor to send a list of commands to be placed in the command queue from the user configured command list. Commands placed in the queue with this method need not have their enable bit set in the command list.

| Word Offset in Block | Data Field(s)             | Description   |
|----------------------|---------------------------|---|
| 4100                 | Block ID                  | This field contains the value of 9902 identifying the enable command to the module.   |
| 4101                 | Command count             | This field contains the number of commands to enable in the command list. Valid values for this field are 1 to 60.  |
| 4102 to 4161         | Command Numbers to enable | These 60 words of data contain the command numbers in the command list to enable. The commands in the list will be placed in the command queue for immediate processing by the module. The first command in the list has an index of 0. |

There is no response to this block by the module. The module will place the selected commands into the command queue and set register 4100 to a value of 0. If the command references a unit that is not defined, the command will not be placed in the command queue. Normal processing of the command list will continue after the commands specified in this block are processed.

For digital output control, the use of block 9901 and 9902 is preferred to the use of the command list. The exact state of the output can be specified in the command list and then the command can be enabled through the use of block 9902. When the user wishes to execute this command (knowing the state of the command), can enable the command with the block 9902 request.

Event Message Block (9903)

Block identification code 9903 sends event messages received on the master port to the processor.

**Note:** Events are recognized when using a COT=SPONTANEOUS.

Block Format for Read

| Word Offset in Block | Data Field(s) | Description   |
|----------------------|---------------|---|
| 4200                 | Block ID      | This field contains the block identification code of 9903 for the block.                    |
| 4201                 | Event Count   | This field contains the number of events present in the block. Values of 1 to 20 are valid. |
| 4202-4211            | Event 1       | Event message   |
| 4212-4221            | Event 2       | Event message   |
| 4222-4231            | Event 3       | Event message   |
| 4232-4241            | Event 4       | Event message   |
| 4242-4251            | Event 5       | Event message   |
| 4252-4261            | Event 6       | Event message   |
| 4262-4271            | Event 7       | Event message   |
| 4272-4281            | Event 8       | Event message   |
| 4282-4291            | Event 9       | Event message   |

The format of each 10-word data region in the block is as follows:

| Word Offset | Definitions                | Description  |
|-------------|----------------------------|--|
| 0           | Session Index/Sector Index | This field contains the session and sector indices used to define the controlled unit in the module from which the event was generated. The MSB contains the session index and the LSB contains the sector index.  |
| 1           | ASDU Type                  | This field contains the ASDU type code for the data contained in the message.  |
| 2           | Function Code/Point Index* | This field contains the function code and the point index associated with the event message. The MSB contains the function code and the LSB contains the point index.  |
| 3           | Fault Number               | This is the fault number for the event if applicable. Only valid for ASDU types 2 and 4.   |
| 4           | Sec/mSec                   | This word contains the seconds and millisecond values with a range of 0 to 59999 time at which the message was generated by the slave device.  |
| 5           | Hr/Min.                    | This word contains the hour and minutes the message was generated by the slave. The MSB contains the hour and the LSB contains the minute value.   |
| 6           | Invalid/DST                | This word contains two bits that relate to the time value recorded in the slave device for the message. Bit 0 corresponds to the validity of the time (0=valid, 1=invalid) and Bit 1 defines if daylight savings time is used in the time (0=no, 1=yes). |
| 7           | Relative Time              | This field contains the relative time value if applicable to the object. Only valid for ASDU types 2 and 4.  |

| Word Offset | Definitions | Description  |
|-------------|-------------|--|
| 8 to 9      | Value       | This double-word value contains the value for the point index/function code in the event message. For ASDU types 1 and 2, this value is only 2 bits wide. For ASDU type 4, this double-word value contains the floating-point number (short circuit location). |

\* Point Index refers to the Data Information Number

In order for this feature to be activated, the event pass-through parameter must be set. When a master driver receives an event message from a controlled station, it will build an event message corresponding to the event in the event buffer of the module. This buffer is then sent to the processor when any messages are present. Therefore, these blocks are sent to the processor on a high priority. After the block is sent, the event message is removed from the module's event buffer. The ladder logic should set register 4200 to a value of zero after processing the event message data.

If too many events are present in the buffer (>200), the module will set the event message overflow flag in the error/status data area of the normal read data block. There is no response block to be received by the module from the processor.

#### Command List Error Data Block (9950)

Block 9950 identification code requests the Command List Error Table from the module for the 1000 user configurable commands. The format for the block is shown in the following table:

| Word Offset in Block | Data Field(s)                | Description   |
|----------------------|------------------------------|---|
| 4100                 | Block ID                     | This field contains the value of 9950 identifying the block type to the module.   |
| 4101                 | Number of Commands to report | This field contains the number of commands to report in the response message. The value has a range of 1 to 60.   |
| 4102                 | Start Index of First Command | This parameter sets the index in the command list where to start. The first command in the list has a value of 0. The last index in the list has a value of MaxCommands -1. |

The module will respond to a valid request with a block containing the requested error information. The format for the block is shown in the following table:

| Word Offset in Block | Data Field(s)               | Description   |
|----------------------|-----------------------------|---|
| 4100                 | Done Flag                   | A value of zero will be placed in this register to indicate the function is complete and the data is ready.                                   |
| 4101                 | Block ID                    | This field contains the value of 9950 identifying the block type to the PLC.  |
| 4102                 | Number of Commands reported | This field contains the number of commands contained in the block that must be processed by the PLC. This field will have a value of 1 to 60. |



| Word Offset in Block | Data Field(s)                | Description   |
|----------------------|------------------------------|---|
| 4103                 | Start Index of First Command | This field contains the index in the command list for the first value in the file. This field will have a value of 0 to MaxCommands-1.  |
| 4104 to 4163         | Command List Errors          | Each word of this area contains the last error value recorded for the command. The command index of the first value (offset 4) is specified in word 3 of the block. The number of valid command errors in the block is set in word 2 of the block. Refer to the command error list to interpret the error codes reported. |

### Set SLC Time Block (9970)

Block 9970 identification code requests the module's date and time. Use this data to set the PLC clock.

| Word Offset in Block | Data Field(s) | Description   |
|----------------------|---------------|---|
| 4100                 | Block ID      | This field contains the value of 9970 identifying the block type to the module. |

The module will respond to a valid block 9970 request with a block containing the requested date and time. The format for the block is shown in the following table:

| Word Offset in Block | Data Field(s) | Description   |
|----------------------|---------------|---|
| 4100                 | Done Flag     | A value of zero will be placed in this register to indicate the function is complete and the data is ready.         |
| 4101                 | Block ID      | This field contains the identification code of 9970 for the block block.  |
| 4102                 | Year          | This field contains the four-digit year to be used with the new time value.   |
| 4103                 | Month         | This field contains the month value for the new time. Valid entry for this field is in the range of 1 to 12.        |
| 4104                 | Day           | This field contains the day value for the new time. Valid entry for this field is in the range of 1 to 31.          |
| 4105                 | Hour          | This field contains the hour value for the new time. Valid entry for this field is in the range of 0 to 23.         |
| 4106                 | Minute        | This field contains the minute value for the new time. Valid entry for this field is in the range of 0 to 59.       |
| 4107                 | Seconds       | This field contains the second value for the new time. Valid entry for this field is in the range of 0 to 59.       |
| 4108                 | Milliseconds  | This field contains the millisecond value for the new time. Valid entry for this field is in the range of 0 to 999. |

### Set Module Time Block (9971)

Block identification code 9971 passes the clock time in the SLC to the module. The date and time provided will be used to set the module's clock.

| Word Offset in Block | Data Field(s) | Description   |
|----------------------|---------------|---|
| 4100                 | Block ID      | This field contains the block identification code of 9971 for the block.  |
| 4101                 | Year          | This field contains the four-digit year to be used with the new time value.   |
| 4102                 | Month         | This field contains the month value for the new time. Valid entry for this field is in the range of 1 to 12.        |
| 4103                 | Day           | This field contains the day value for the new time. Valid entry for this field is in the range of 1 to 31.          |
| 4104                 | Hour          | This field contains the hour value for the new time. Valid entry for this field is in the range of 0 to 23.         |
| 4105                 | Minute        | This field contains the minute value for the new time. Valid entry for this field is in the range of 0 to 59.       |
| 4106                 | Seconds       | This field contains the second value for the new time. Valid entry for this field is in the range of 0 to 59.       |
| 4107                 | Milliseconds  | This field contains the millisecond value for the new time. Valid entry for this field is in the range of 0 to 999. |

The module does not send a response block to the processor after receiving this block. The module will set register 4100 to zero after processing the data.

Warm Boot Block (9998)

Block 9998 performs a warm-boot operation on the module. The format of the block constructed by the processor is as follows:

| Offset | Description | Length |
|--------|-------------|--------|
| 4100   | 9998        | 1      |

In this version of the module, the warm and cold boot processes perform the same operation as many of the variables that must be initialized are fixed when the module first boots and cannot be changed after the application starts.

Cold Boot Block (9999)

Block 9999 performs a cold-boot operation on the module. The format of the block constructed by the processor is as follows:

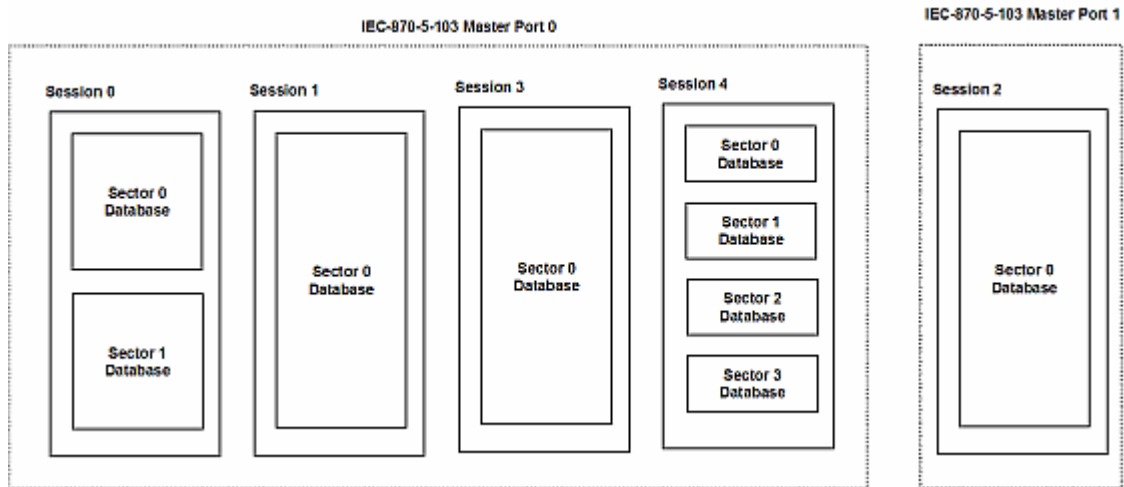
| Offset | Description | Length |
|--------|-------------|--------|
| 4100   | 9999        | 1      |

In this version of the module, the warm and cold boot processes perform the same operation as many of the variables that must be initialized are fixed when the module first boots and cannot be changed after the application starts.

### 3.2 Master Driver

The master driver supported on each application port of the module emulates an IEC 60870-5-103 Master device. Configuration of each port is independent and should be connected to different serial networks.

Each port on the module communicates with one or more controlled stations on what are referred to as sessions. A session represents a controlled device with a unique data link layer address. Each session (controlled device) contains one or more data sets (sectors) that are defined by the vendor of the device. The following illustration shows these relationships.



Port 0 on the module communicates with 4 sessions (0, 1, 3 and 4) each of which has their own data set(s). Session 1 only has one sector (all data for device contained in a single database). This sector is addressed by the master using the Common address of ASDU value set for the sector in the configuration file. Session 0 contains two sectors each with their own unique Common address of ASDU value to identify the sector.

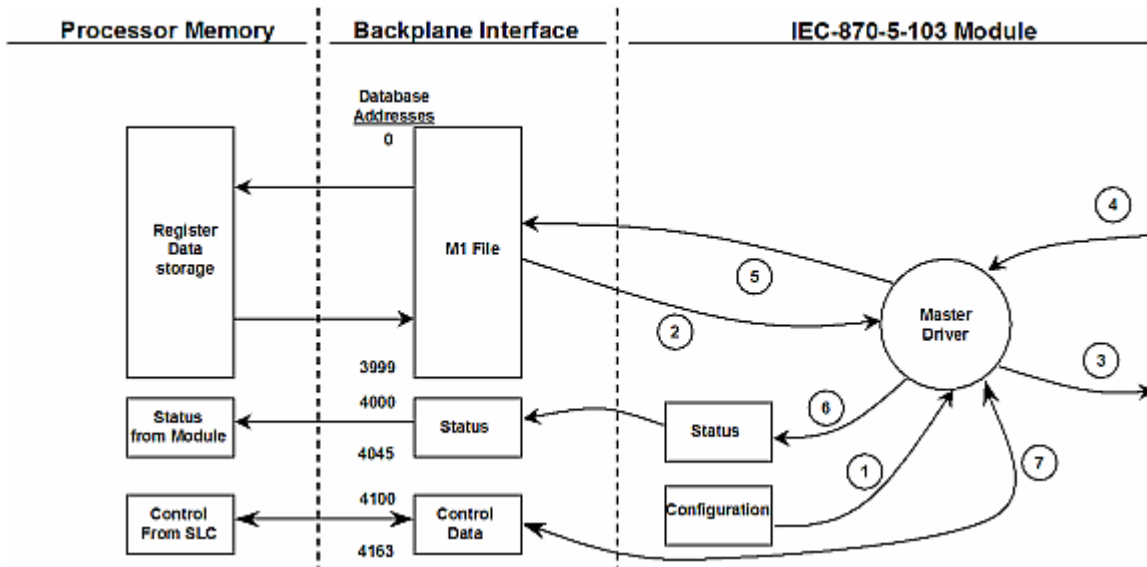
Port 1 is connected to one device on the network. This device is defined in the Session 2 section of the configuration file. In this example, all data of the device is stored in a single sector.

**Note:** The IEC 60870-5-103 specification only supports the unbalanced mode. No support is given in the protocol for the balanced mode and the module does not support this mode.

The module supports two application ports. Thirty-two sessions can be defined on the module with each session being assigned to an application port. Within each session, up to five sectors can be defined. This system permits a very flexible assignment of resources in the module. The definition of the data

associated with each sector in the system is defined by the user in the configuration file.

The following diagram illustrates the functionality of the master driver:



- 1 The master driver is configured as specified by the IEC103M.CFG file
- 2 The master will construct control commands using the data in the database
- 3 The master will send these commands and class polls out on the serial network
- 4 Response messages or spontaneous messages generated by controlled devices on the serial network are received by the master driver
- 5 Monitor data received by the master is passed to the module's database and passed to the processor
- 6 Additionally, status data for the module is passed to the processor

## 4 Module Configuration

### *In This Chapter*

- Installing and Configuring the Module ..... 29
- Module Data ..... 31
- Configuration File ..... 31
- Uploading and Downloading the Configuration File ..... 35

This section contains the setup procedure, data, and ladder logic for successful application of the MVI46-103M module. Each step in the setup procedure is defined in order to simplify the use of the module.

### 4.1 Installing and Configuring the Module

This chapter describes how to install and configure the module to work with your application. The configuration process consists of the following steps.

- 1 Use RSLogix to identify the module to the processor and add the module to a project.

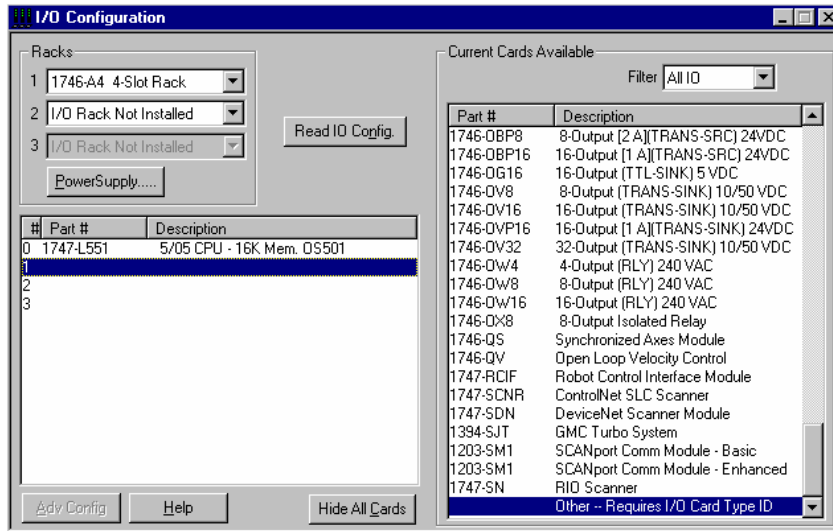
**NOTE:** The RSLogix software must be in "offline" mode to add the module to a project.

- 2 Modify the module's configuration files to meet the needs of your application, and copy the updated configuration to the module. Example configuration files are provided on the CD-ROM.
- 3 Modify the example ladder logic to meet the needs of your application, and copy the ladder logic to the processor. Example ladder logic files are provided on the CD-ROM.

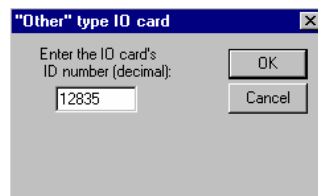
**Note:** If you are installing this module in an existing application, you can copy the necessary elements from the example ladder logic into your application.

The rest of this chapter describes these steps in more detail.

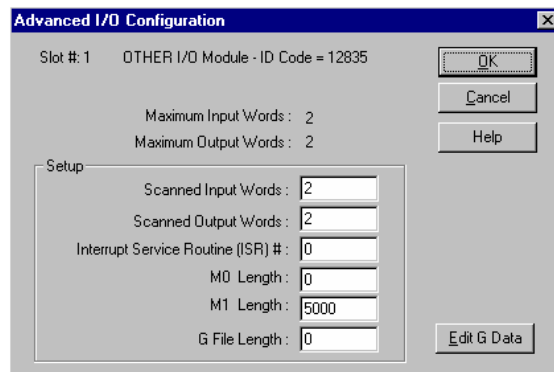
The first step in installing and configuring the module is to define the module to the system. Select the I/O Configuration option from the program screen. This displays the following dialog box:



Select the Other module from the list. This action opens the following dialog box.



Enter the module I/O card ID number as 12835, and then select the OK command button. Double-click the mouse on the module just added to the rack. Fill in the dialog box presented as shown in the following screen shot:



Select the OK command button to apply these settings to the module. Then, close the I/O Configuration dialog box.

The next step in the module's setup is to define the user defined data areas to hold the status and read and write database areas. Edit the IEC103M.CFG file now for the application to implement. Use any text editor to set the values in the file. You must retain the file name, IEC103M.CFG.

The last step in the module setup is to add the ladder logic. If the example ladder logic is used, adjust the ladder to fit the application. When the ladder example is not used, copy the example ladder logic to your application and alter as necessary.

The module is now set up and ready to be used with your application. Insert the module in the rack and attach the serial communication cables. Download the IEC103M.CFG file to the module. Download the new application to the controller and place the processor in run mode. If all the configuration parameters are set correctly and the module is attached to a network, the module's Application LED (APP LED) should remain off and the backplane activity LED (BP ACT) should blink very rapidly. Refer to the **Troubleshooting** section if you encounter errors. Attach a computer or terminal to Debug/Configuration port on the module and check the status of the module using the resident debugger in the module.

## 4.2 Module Data

All data related to the MVI46-103M module is stored in a user defined data files and the module's M1 file. Files should be defined for each data type to be used with the module. Additionally, a file should be defined to hold the module status data. The status data should be copied from the M1 file and placed in the assigned status file. Input (monitor) data should be copied from the user file to the M1 file and output (command) data should be copied from the user files to the M1 file.

## 4.3 Configuration File

The module requires a configuration file (IEC103M.CFG) to operate. This configuration file configures the module's master drivers, sets up the databases for the controlled devices and establishes a command list. Each parameter in the file must be set carefully in order for the application to be implemented successfully.

The configuration file contains at least the following seven sections with topic header names enclosed in [ ] characters.:

|                                  |   |
|----------------------------------|---|
| <b>[Backplane Configuration]</b> | <b>Backplane transfer parameter section</b> |
| [IEC-870-5-103 Master]           | General Configuration for driver            |
| [IEC-870-5-103 Master Port 0]    | Configuration for first application port    |

| <b>[Backplane Configuration]</b>    | <b>Backplane transfer parameter section</b>        |
|-------------------------------------|--|
| [IEC-870-5-103 Master Port 1]       | Configuration for second application port          |
| [IEC-103 Master Session x]          | Definition for each control unit                   |
| [IEC-103 Master Session x Sector y] | Definition for each sector in each controlled unit |
| [IEC-103 Master Commands]           | Command list to control slave units                |

A set of parameters follows each section header. Use unique labels under each section to specify a parameter. Each label in the file must be entered exactly as shown in the file for the parameter to be identified by the program. If the module is not considering a parameter, check the label for the data item. Each parameter's value is separated from the label with the ':' character. This character is used by the program to delimit the position in the data record where to start reading data. All data for a parameter must follow the ':' character. There must be at least one space character between the end of the parameter value and the following text. An example of a parameter entry is given below:

```
Baud Rate : 19200 #Baud rate for master port
```

The parameter label is "Baud Rate" and the parameter value is 19200. The characters after a numeric parameter value are ignored and can be used to document the configuration file.

Any record that begins with the '#' character is considered to be a comment. Comments can be placed anywhere in the file so long as the '#' character is the first column of the line. Liberal use of comments within the file can ease the use and interpretation of the data in the file.

Use any text editor to alter the supplied IEC103M.CFG file for the specific application. You must enter each parameter correctly for successful application of the module. ***MVI46-103M Configuration Data Definition*** (on page 82) contains a complete listing and definition of all parameters utilized by the module.

The session sections of the configuration file are determined by the number of sessions set in the configuration file. The sessions are referenced by a zero based index value. For example, if the module is configured for four sessions, the configuration file should contain sections for sessions 0 to 3 (that is, [IEC-103 Master Session 0] to [IEC-103 Master Session 3]). Each of these sections will define the characteristics of the specific controlled device to be interfaced. Within each session definition, is a parameter that specifies the number of sectors for the session. For each sector defined for a session, there must exist a [IEC-103 Master Session x Sector y] section. Where the x value represents the session index and the y value represents sector index. For example if session 0 contains 1 sector, there must be a section with the following name in the configuration file: [IEC-103 Master Session 0 Sector 0]. The specific sector parameter set and database is defined in this section.

The last section of the configuration file is the command list definition ([IEC-103 Master Commands]). This section can contain up to 1000 user defined commands to be executed by the module and sent to the controlled devices. There is no need to place Class 1 or Class 2 polls in the this list for the controlled



devices as the master driver for each port will execute these automatically when the port is idle. In order for the port to be idle, make sure that there is idle time available and that the commands do not constantly utilize the ports. The command list section starts with a reserved label **START** and ends with the label **END**. Each row in the file corresponds to an individual command with the first character position in each row left blank (white space). The contents of each command record is outlined in the following table:

| Col  | Field                           | Description   |      |             |               |   |                 |               |   |                      |               |   |                       |               |    |                                 |             |
|------|---------------------------------|---|------|-------------|---------------|---|-----------------|---------------|---|----------------------|---------------|---|-----------------------|---------------|----|---------------------------------|-------------|
| 1    | Enable Code                     | <p>This field determines when the command will be executed according to the following codes:</p> <p>0=Command is disabled and will only execute if enabled from PLC processor</p> <p>1=Command will execute no more frequently than the time set in the Poll interval</p> <p>2=Command will execute when the last value read in the database differs from the current value</p>   |      |             |               |   |                 |               |   |                      |               |   |                       |               |    |                                 |             |
| 2    | Database Index                  | <p>This field specifies the location in the module's internal database to associate with the command. The data type used in the command determines addressing of the index as follows:</p> <table border="1"> <thead> <tr> <th>Type</th> <th>Description</th> <th>DB Index type</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Special command</td> <td>*Word address</td> </tr> <tr> <td>6</td> <td>Time synchronization</td> <td>*Word address</td> </tr> <tr> <td>7</td> <td>General interrogation</td> <td>*Word address</td> </tr> <tr> <td>20</td> <td>General command (2 bit control)</td> <td>Bit address</td> </tr> </tbody> </table> <p>*Word address = Value only used to signal when to send event (Enable Code = 2)</p> | Type | Description | DB Index type | 0 | Special command | *Word address | 6 | Time synchronization | *Word address | 7 | General interrogation | *Word address | 20 | General command (2 bit control) | Bit address |
| Type | Description                     | DB Index type   |      |             |               |   |                 |               |   |                      |               |   |                       |               |    |                                 |             |
| 0    | Special command                 | *Word address   |      |             |               |   |                 |               |   |                      |               |   |                       |               |    |                                 |             |
| 6    | Time synchronization            | *Word address   |      |             |               |   |                 |               |   |                      |               |   |                       |               |    |                                 |             |
| 7    | General interrogation           | *Word address   |      |             |               |   |                 |               |   |                      |               |   |                       |               |    |                                 |             |
| 20   | General command (2 bit control) | Bit address   |      |             |               |   |                 |               |   |                      |               |   |                       |               |    |                                 |             |
| 3    | Poll Interval                   | This parameter is used if the Enable Code field is set to a value of 1. It sets the minimum number of seconds to delay between successive execution of the command.   |      |             |               |   |                 |               |   |                      |               |   |                       |               |    |                                 |             |
| 4    | Session                         | This parameter is utilized to associate the command with one of the sessions defined for the module.  |      |             |               |   |                 |               |   |                      |               |   |                       |               |    |                                 |             |
| 5    | Sector                          | This parameter associates the command with the proper sector of the selected session.   |      |             |               |   |                 |               |   |                      |               |   |                       |               |    |                                 |             |

| Col  | Field                           | Description   |      |             |   |                 |   |                      |   |                       |    |                                 |
|------|---------------------------------|---|------|-------------|---|-----------------|---|----------------------|---|-----------------------|----|---------------------------------|
| 1    | Enable Code                     | <p>This field determines when the command will be executed according to the following codes:</p> <p>0=Command is disabled and will only execute if enabled from PLC processor</p> <p>1=Command will execute no more frequently than the time set in the Poll interval</p> <p>2=Command will execute when the last value read in the database differs from the current value</p>   |      |             |   |                 |   |                      |   |                       |    |                                 |
| 6    | Data Type                       | <p>This parameter sets the ASDU data type to be used with the message. The codes specified are those defined for the IEC-870-5-103 protocol. The following is a listing of command control data types supported in this module:</p> <table border="1"> <thead> <tr> <th>Type</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Special command</td> </tr> <tr> <td>6</td> <td>Time synchronization</td> </tr> <tr> <td>7</td> <td>General interrogation</td> </tr> <tr> <td>20</td> <td>General command (2 bit control)</td> </tr> </tbody> </table> <p>The type 0 is not defined in the protocol specification but is added to the module application to perform special commands. The command to be issued is determined by the value of the function field. If the function is set to 1, the module will issue a reset process command to the session specified. If the function is set to 2, the module will a class 2 poll to the selected session.</p> | Type | Description | 0 | Special command | 6 | Time synchronization | 7 | General interrogation | 20 | General command (2 bit control) |
| Type | Description                     |   |      |             |   |                 |   |                      |   |                       |    |                                 |
| 0    | Special command                 |   |      |             |   |                 |   |                      |   |                       |    |                                 |
| 6    | Time synchronization            |   |      |             |   |                 |   |                      |   |                       |    |                                 |
| 7    | General interrogation           |   |      |             |   |                 |   |                      |   |                       |    |                                 |
| 20   | General command (2 bit control) |   |      |             |   |                 |   |                      |   |                       |    |                                 |
| 7    | Function Code                   | <p>This parameter specifies the function code as specified in the protocol specification except when Data Type code 0 is used. For the general commands, this field is used in conjunction with the Point Index to determine the point to control in the remote slave device. For the special commands, a value of 1 performs a reset process and 2 to issue a class 2 poll.</p>  |      |             |   |                 |   |                      |   |                       |    |                                 |
| 8    | Point Index                     | <p>This parameter along with the Function Code determines the point to control in the remote device when the general command is utilized. For all other commands this field is ignored.</p>   |      |             |   |                 |   |                      |   |                       |    |                                 |
| 9    | Override Flag                   | <p>This field is used with general commands to determine if the override or database value should be used with the command. If the flag is set to 0, the value in the module's database will be sent when the command is issued. If the flag is set to 1 (or non-zero value), the value set in the Override Value will be sent with the command. When the override flag and value are utilized, the Enable Code 2 and Database Index can be used to trigger the execution of the command.</p>   |      |             |   |                 |   |                      |   |                       |    |                                 |
| 10   | Override Value                  | <p>This parameter specifies the override value to use with the command if the Override Flag is set. This value should have one of the following values:<br/>1=Off and 2=On. Other values maybe valid for certain controlled</p>   |      |             |   |                 |   |                      |   |                       |    |                                 |

\* The Point Index is the Data Information Number.

As an alternative to using a command list, blocks with an identification code of 9901 can be used to issue commands from the ladder logic.

## 4.4 Uploading and Downloading the Configuration File

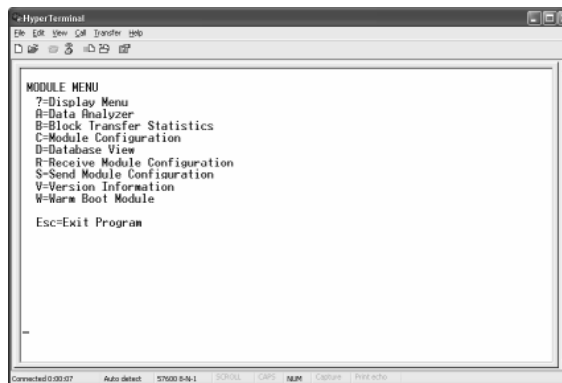
ProSoft modules are shipped with a pre-loaded configuration file. In order to edit this file, you must transfer the file from the module to your PC. After editing, you must transfer the file back to the module.

This section describes these procedures.

**Important:** The illustrations of configuration/debug menus in this section are intended as a general guide, and may or may not exactly match the configuration/debug menus in your own module. For specific details about the configuration/debug menus in your module, please refer to *The Configuration/Debug Menu* (on page 47).

### 4.4.1 Transferring the Configuration File to Your PC

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

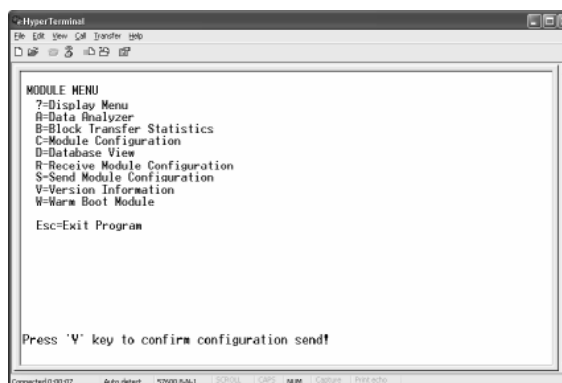


```
HyperTerminal
File Edit View Ctrl Transfer Help
[Icons]
MODULE MENU
?=Display Menu
@=Data Analyzer
B=Block Transfer Statistics
C=Module Configuration
D=Database View
R=Receive Module Configuration
S=Send Module Configuration
V=Version Information
W=Warm Boot Module

Esc=Exit Program

Connected 03/07 Auto detect 57600 8-N-1 SCROLL CAPS NUM Capture Print Active
```

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



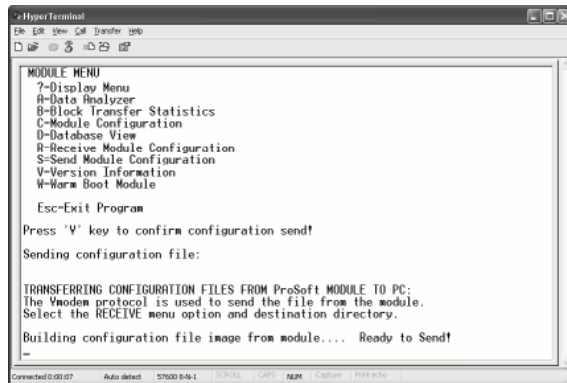
```
HyperTerminal
File Edit View Ctrl Transfer Help
[Icons]
MODULE MENU
?=Display Menu
@=Data Analyzer
B=Block Transfer Statistics
C=Module Configuration
D=Database View
R=Receive Module Configuration
S=Send Module Configuration
V=Version Information
W=Warm Boot Module

Esc=Exit Program

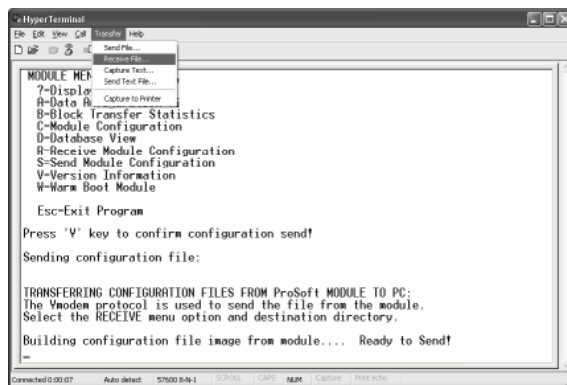
Press 'Y' key to confirm configuration send!

Connected 03/07 Auto detect 57600 8-N-1 SCROLL CAPS NUM Capture Print Active
```

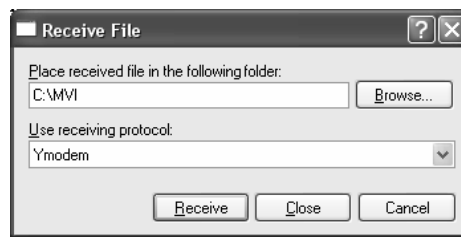
- Press [Y]. The screen now indicates that the module is ready to send.



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



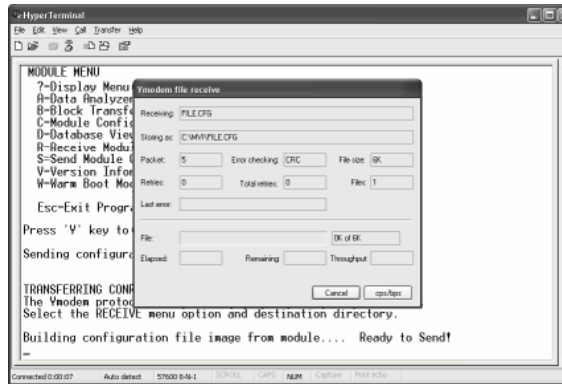
- Use the Browse button to choose a folder on your computer to save the file, and then click Receive.



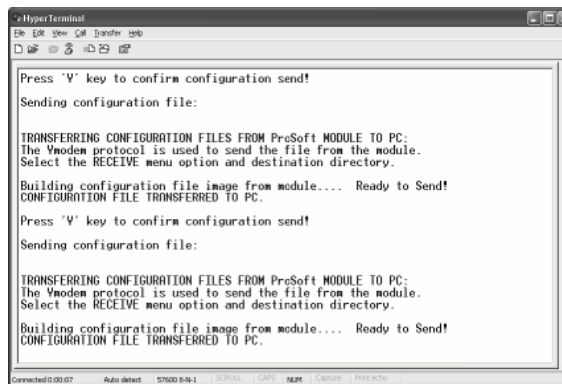
- Note:** ProSoft Technology suggests that you download the configuration file pre-loaded on your module. However, configuration files are also available on the ProSoft CD as well as the ProSoft Technology web site at <http://www.prosoft-technology.com>.

- Select Ymodem as the receiving protocol.

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



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



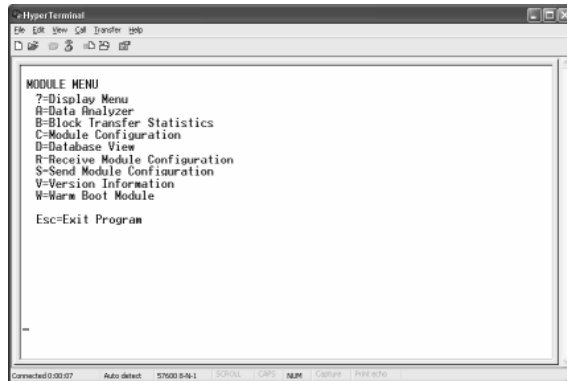
The configuration file is now on your PC at the location you specified.

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

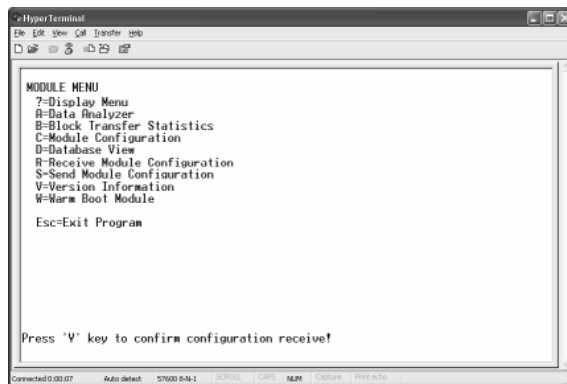
#### 4.4.2 **Transferring the Configuration File to the Module**

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

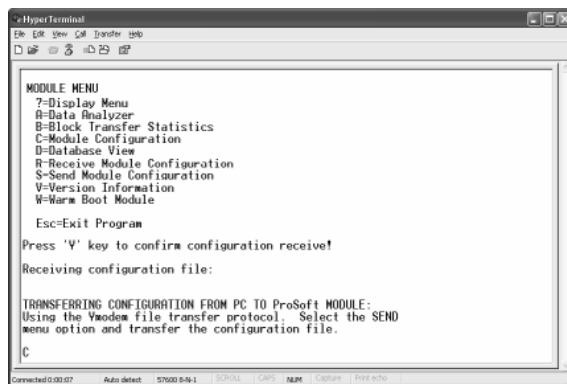
- 1 Connect your PC to the Configuration/Debug port of the module using a terminal program such as HyperTerminal. Press [?] to display the main menu.



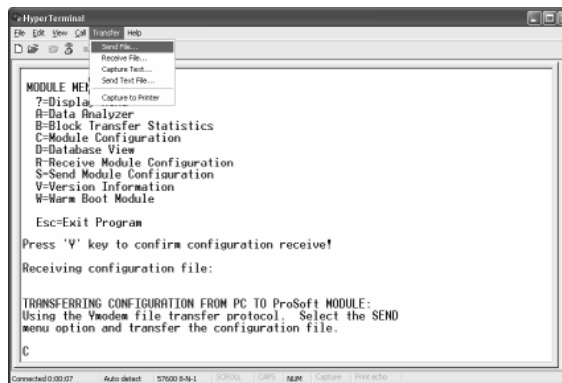
- 2 Press [S] (Receive Module Configuration). The message "Press Y key to confirm configuration receive!" is displayed at the bottom of the screen.



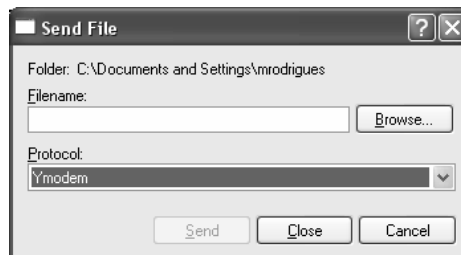
- 3 Press [Y]. The screen now indicates that the PC is ready to send.



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



The Send File dialog appears.

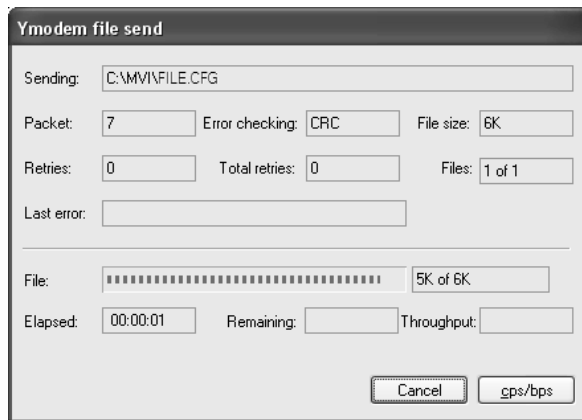


- Use the Browse button to locate the configuration file your computer.

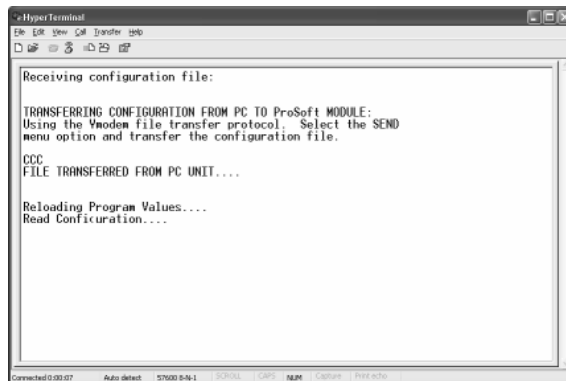
**Note:** This procedure assumes that you are uploading a newly edited configuration file from your PC to the module. However, configuration files are also available on the ProSoft CD as well as the ProSoft Technology web site at <http://www.prosoft-technology.com>.

- Select **Ymodem** as the protocol.

- Click the Send button. This action opens the Ymodem File Send dialog box.



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



- Your module now contains the new configuration.



## 5 Ladder Logic

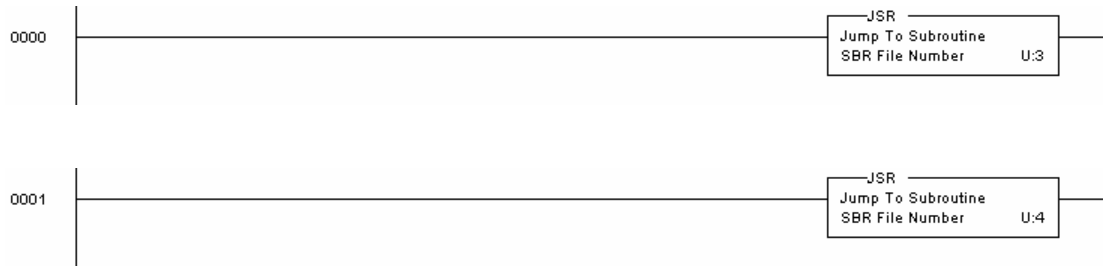
### *In This Chapter*

- Main Routine ..... 41
- Data Transfer (U:3) ..... 42
- Control Routine (U:4) ..... 43

Ladder logic is required for application of the MVI46-103M module. Tasks that must be handled by the ladder logic are data transfer, special block handling and status data receipt. This section discusses each aspect of the ladder logic as required by the module. Additionally, a power-up handler should be written to handle the initialization of the module's data and to clear any processor fault conditions.

### 5.1 Main Routine

The main routine is used to call the data transfer and control subroutines. An example of the main routine is shown in the following two rungs:



## 5.2 Data Transfer (U:3)

The data transfer routine is responsible for placing all the output data into the M1 file and for retrieving all the input data from the M1 file. The following two rungs transfer the data between the M1 file and the user data files. The first rung transfers monitored data received by the master driver into the user data files.



The following rung transfers the control data from the SLC to the master driver.



The next rung transfers the module's status data to a user data file. A rung to perform this operation is as follows:



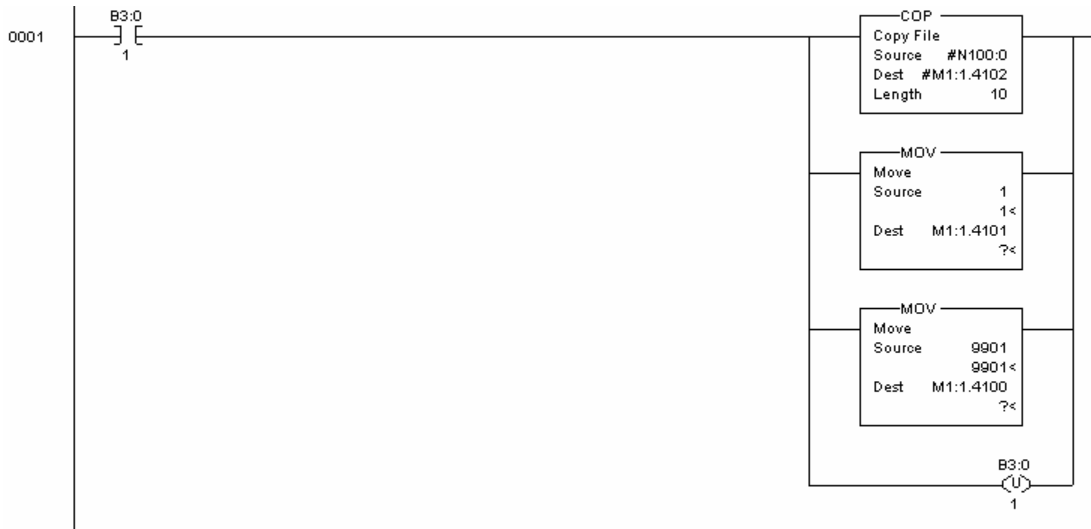
The last rung displays a simple example of retrieving event message data from the M1 file:



After the data is moved to the user file, word 4200 in the M1 file is set to zero. This informs the module that the message was received and that more



The next rung displays an example of building a 9901 block to send a user command to a controlled device:



This rung copies the command to issue from the user file into the M1 file and sets the command count to 1. The last operation on the M1 file is to set the command code 9901 in to the file. The module will place all valid commands received in 9901 blocks into the command queue for execution.

The next rung demonstrates the use of the 9902 block:



This rung requests that the single command number 1 in the user command list be placed into the command queue.

The next rung shows how to write date and time information to the MVI46-103M module using block 9971:



The last rung shows how to request date and time information from the module using block 9970. The module response is handled in the Data Transfer routine.





## 6 Diagnostics and Troubleshooting

### *In This Chapter*

|   |    |
|---|----|
| ➤ The Configuration/Debug Menu .....      | 47 |
| ➤ Required Hardware .....                 | 48 |
| ➤ Required Software .....                 | 49 |
| ➤ Using the Configuration/Debug Port..... | 49 |
| ➤ LED Status Indicators .....             | 69 |

This section provides information on diagnostics and troubleshooting in three forms:

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

### 6.1 The Configuration/Debug Menu

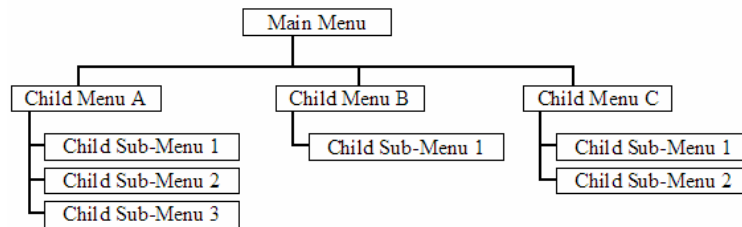
The Configuration and Debug menu for this module is arranged as a tree structure, with the Main Menu at the top of the tree, and one or more sub-menus for each menu command. The first menu you see when you connect to the module is the Main menu.

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

#### **6.1.1**                    *Navigation*

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

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



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

### Keystrokes

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

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

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

## 6.2 Required Hardware

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

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

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



### 6.3 Required Software

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

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

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

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

### 6.4 Using the Configuration/Debug Port

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

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

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

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

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

- 1 Verify that the null modem cable is connected properly between your computer's serial port and the module. A regular serial cable will not work.
- 2 Verify that your communication software is using the correct settings for baud rate, parity and handshaking.

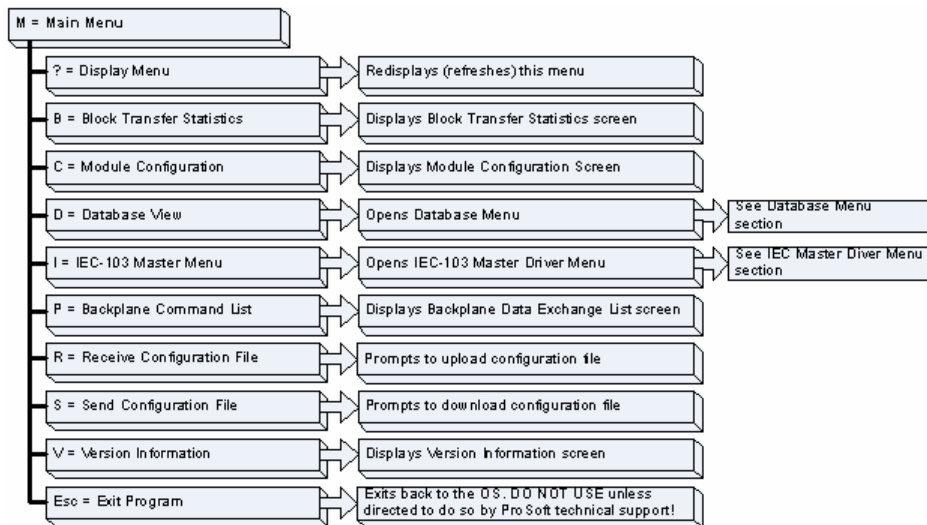
- On computers with more than one serial port, verify that your communication program is connected to the same port that is connected to the module.

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

### 6.4.1 Main Menu

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

```
IEC-870-5-103 MASTER COMMUNICATION MODULE
?=Display Menu
B=Block Transfer Statistics
C=Module Configuration
D=Database View
I=IEC-103 Master Menu
P=Backplane Command List
R=Receive Configuration File
S=Send Configuration File
V=Version Information
Esc=Exit Program
```



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

are active nevertheless. Please be careful when pressing keys so that you do not accidentally execute an unwanted command.

#### Viewing Block Transfer Statistics

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

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

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

#### Viewing Module Configuration

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

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

#### Opening the Database Menu

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

#### Opening the IEC-103 Master Menu

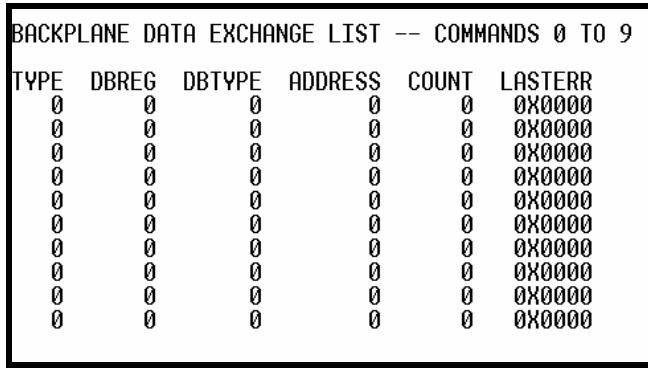
Press **[I]** from the Main Menu to open the IEC-870-5-103 Master Driver Menu. Use this menu command to view detailed configuration information for the module.

```
IEC-103 MASTER Menu Selected
IEC-870-5-103 MASTER DRIVER MENU
?=Display Menu
A=Data Analyzer
C=General Configuration
I=Command List Menu
P=Port Configuration Menu
Q=Port Status Menu
S=Session Menu
V=Version
Z=Previous Menu
```

For more information about the commands on this menu, refer to **IEC-103 Master Driver Menu** (on page 56).

Viewing the Backplane Command List

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



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

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

Receiving the Configuration File

Press **[R]** to download (receive) the current configuration file from the module. For more information on receiving and sending configuration files, please see **Uploading and Downloading the Configuration File** (on page 35).

Sending the Configuration File

Press **[S]** to upload (send) an updated configuration file to the module. For more information on receiving and sending configuration files, please see **Uploading and Downloading the Configuration File** (on page 35).

Viewing Version Information

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

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

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

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

### Exiting the Program

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

Press **[Esc]** to exit the program and display the operating system prompt. This command will cause the module to cease operation and stop transferring data between the ports and the module, and between the processor and the module. This could interrupt a currently running process. Only use this command if instructed to do so by the ProSoft Technical Support Group.

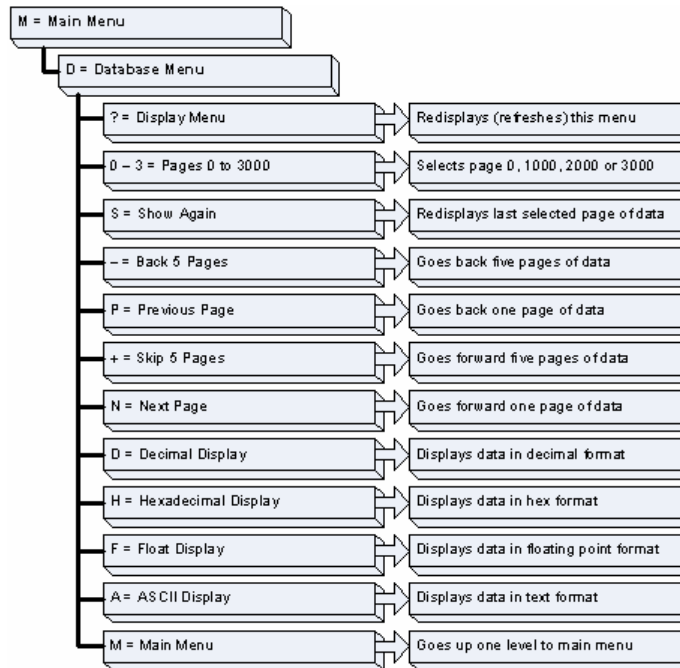
### Warm Booting the Module

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

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

## 6.4.2 Database View Menu

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



### Viewing Database Pages 0 to 3000

The database is divided into pages that correspond with a specific number of registers. The total number of database pages and registers depends on the memory capacity and configuration of the module.

Use the keyboard commands **[0]** through **[3]** to display database contents starting from 0 (zero), 1000, 2000 and 3000 respectively.

### Moving Back Through 5 Pages of Registers

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

### Viewing the Previous 100 Registers of Data

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

*Skipping 500 Registers of Data*

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

*Viewing the Next 100 Registers of Data*

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

*Viewing Data in Decimal Format*

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

*Viewing Data in Hexadecimal Format*

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

*Viewing Data in Floating Point Format*

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

*Viewing Data in ASCII (Text) Format*

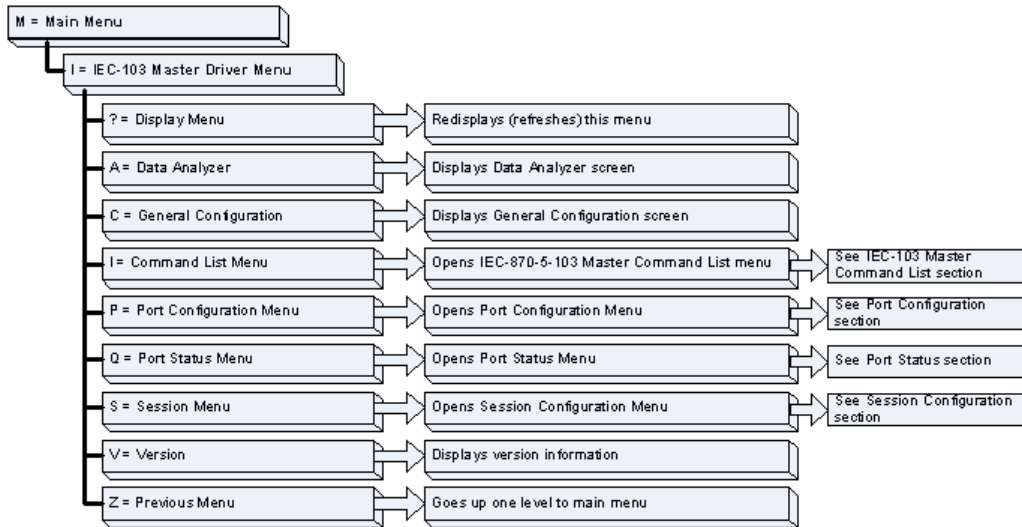
Press **[A]** to display the data on the current page in ASCII format. This is useful for regions of the database that contain ASCII data.

*Returning to the Main Menu*

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

### 6.4.3 IEC-103 Master Driver Menu

Press **[I]** from the Main Menu to open the IEC-870-5-103 Master Driver Menu. Use this menu command to view detailed configuration information for the module.



#### Opening the Data Analyzer Menu

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

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



Viewing General Configuration

Press **[C]** from the IEC-103 Master Driver Menu to display the general configuration for the protocol. The following illustration shows an example of the Module Configuration screen:

```

MODULE CONFIGURATION
Max Sessions      = 0 (of maximum 32)
Online States    = 0x00000000 (bit mapped in hex format)
Command Count    = 0 (of maximum 1000)
Current Command  = 0 (IDLE)
MEMORY COUNTERS:
applRec          = 0
applTrans        = 0
anything         = 0

```

The Busy/Idle message indicates the current activity state of the module. "Idle" means it is waiting to execute a command. "Busy" means it is executing a command and is waiting for the response to the request. This does not include the normal class 1 and 2 polls as these are automatically generated.

The counter data displays the number of memory areas allocated for the application layer. When no packets are pending, the counts should all be 0. If messages are waiting to be sent, the applRec count will indicate the number waiting to be sent. If many messages are received at the same time, the applRec count will indicate the number of packets that must be processed. The "anything" count indicates any other buffer area that is allocated and must be processed by the application.

Opening the IEC-870-Master Command List Menu

Press **[I]** from the IEC-103 Master Driver Menu to open the ICE-870 Master Command List menu. Use this command to view the configured command list for the module.

```

IEC 870 5 103 MASTER COMMAND LIST, COMMANDS 0 TO 9
END DDIDR POLLI SES SEC #SDU FUNC POINT LAST POLL LERROR OVR VALUE
0 U 10 B U 20 12W 16 14932 B 1 1
0 1000 41 B U 20 12W 17 14932 B 0 0
0 1002 41 B U 20 12W 18 14932 B 0 0
1 H 5 H H 7 255 W 4 H 0
~~~~~ END OF COMMAND LIST REACHED ~~~~~

```

For more information about the commands on this menu, refer to ***IEC-870-Master Command List Menu*** (on page 60).

Opening the Port Configuration Menu

Press **[P]** from the IEC-103 Master Driver Menu to open the Port Configuration menu. Use this command to view the port configuration information for each of the application ports.

```
Port Configuration Menu Selected
IEC-870-5-103 MASTER CHANNEL 0 CONFIGURATION
Baudrate           = 19200
Parity             = EVEN
RTS On            = 1
RTS Off           = 0
Minimum Delay     = 30
Receive Timeout   = 2000
Single ACK        = YES
Data Link Length  = 1
Use Balanced Mode = NO
```

The *Port Configuration Menu* section has more information about the commands on this menu.

Opening the Port Status Menu

Press **[Q]** from the IEC-103 Master Driver Menu to open the Port Status menu. Use this command to verify the status of the master commands sent through the port. If the display indicates a communication error, you should compare the generated error code with the command error codes listed in the Appendices of this manual.

```
Port Status Menu Selected
IEC-870-5-103 MASTER PORT 0 STATUS
Commands Executed = 0
Command Responses = 0
Command Err Count = 0
Request Count     = 0
Response Count    = 0
Error Sent Count  = 0
Error Rec Count   = 0
Cfg Error Word    = 0x0000
Current Error Code = 0
Last Error Code   = 0
```

The *Port Status Menu* section has more information about the commands on this menu.

Opening the Session Configuration Menu

Press **[S]** to open the Session Configuration menu. Use this command to view the session configuration data.

Refer to *Session Configuration Menu* (on page 66) for more information about the commands on this menu.

#### Opening the Sector Menu

Press **[1]** from the IEC-103 Master Driver Menu to open the Sector Configuration menu. Use this command to view the Sector Configuration data.

```
SECTOR CONFIGURATION MENU
?=Display Menu
D=Sector Database Menu
S=Show Again
P=Previous Page
N=Next Page
M=Return to Session Menu
```

The *Sector Configuration Menu* section has more information about the commands on this menu.

#### Viewing Master Driver Version Information

Press **[V]** from the IEC-103 Master Driver Menu to view the master driver version information.

```
DRIVER VERSION INFORMATION:

(c) 1999-2003, ProLinx Communication Gateways, Inc.

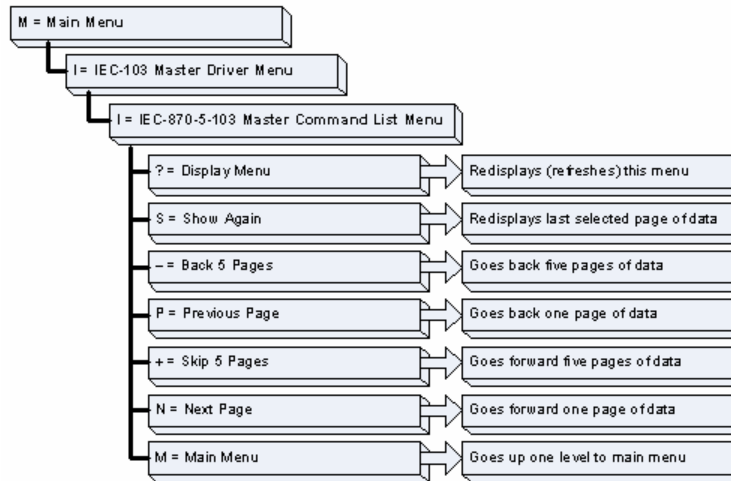
PRODUCT NAME CODE       : IEC-870-5-103 Master Driver
SOFTWARE REVISION LEVEL : 2.14
SOFTWARE REVISION DATE  : 04/17/2003
FAR CORE LEFT           : 322368
```

#### Returning to the Main Menu

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

#### 6.4.4 IEC-870-Master Command List Menu

Press **[I]** from the IEC-103 Master Driver Menu to open the ICE-870 Master Command List menu. Use this command to view the configured command list for the module.



##### Redisplaying the Menu

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

##### Redisplaying the Current Page

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

##### Moving Back Through 5 Pages of Registers

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

##### Viewing the Previous 100 Registers of Data

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

##### Skipping 500 Registers of Data

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

##### Viewing the Next 100 Registers of Data

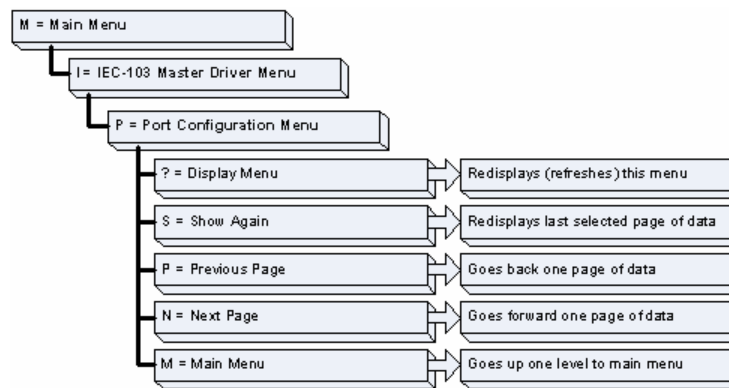
Press **[N]** from the Database View menu to select and display the next 100 registers of data.

Returning to the Main Menu

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

**6.4.5 Port Configuration Menu**

Press **[P]** from the IEC-103 Master Driver Menu to open the Port Configuration menu. Use this command to view the port configuration information for each of the application ports.

Redisplaying the Menu

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

Redisplaying the Current Page

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

Displaying the Next Page

Press **[N]** to display the next 100 registers. Use this command to step forward through the data a page at a time.

Displaying the Previous Page

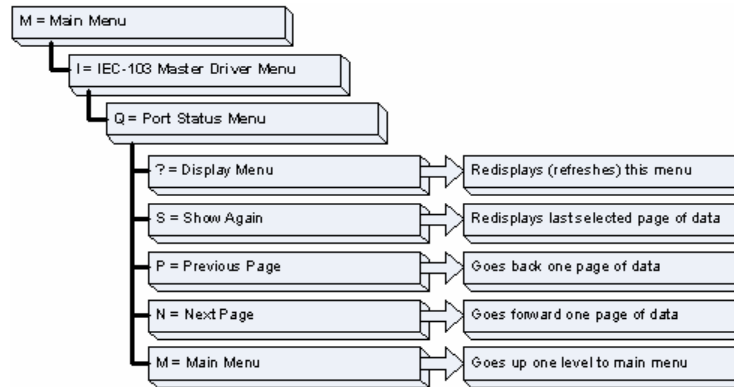
Press **[P]** to display the previous 100 registers. Use this command to step backward through the data a page at a time.

Returning to the Main Menu

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

### 6.4.6 Port Status Menu

Press **[Q]** from the IEC-103 Master Driver Menu to open the Port Status menu. Use this command to view the communication status information for each application port.



#### Redisplaying the Menu

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

#### Redisplaying the Current Page

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

#### Displaying the Previous Page

Press **[P]** to display the previous 100 registers. Use this command to step backward through the data a page at a time.

#### Displaying the Next Page

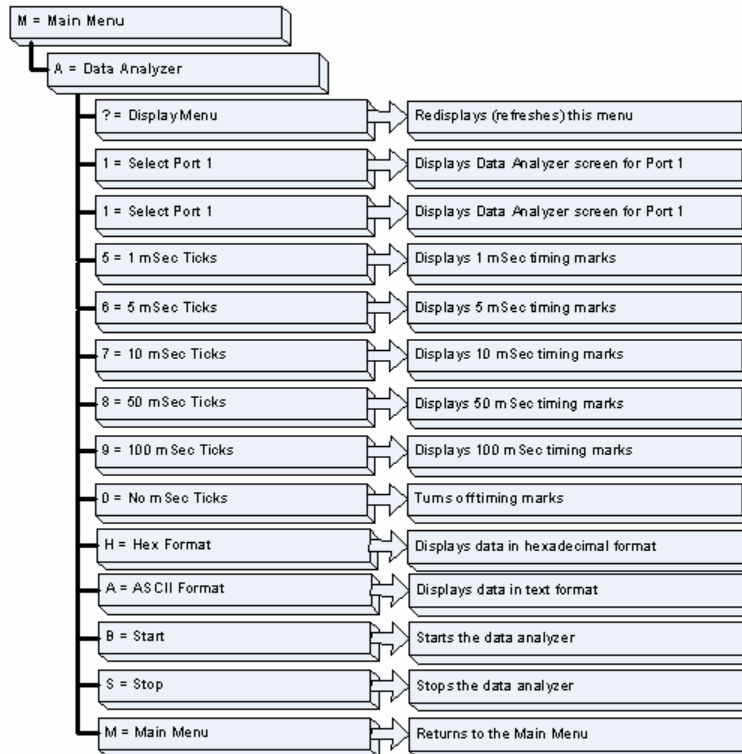
Press **[N]** to display the next 100 registers. Use this command to step forward through the data a page at a time.

#### Returning to the Main Menu

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

### 6.4.7 Data Analyzer

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



```

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

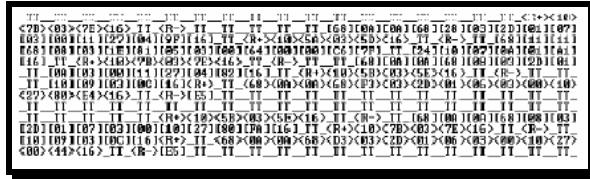
Port = 1, Format=HEX, Tick=10

```

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

Analyzing Data for Port 1

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



Analyzing Data for Port 2

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

Displaying Timing Marks in the Data Analyzer

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

| Key | Interval       |
|-----|----------------|
| [5] | 1 mSec ticks   |
| [6] | 5 mSec ticks   |
| [7] | 10 mSec ticks  |
| [8] | 50 mSec ticks  |
| [9] | 100 mSec ticks |

Removing Timing Marks in the Data Analyzer

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

Viewing Data in Hexadecimal Format

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

Viewing Data in ASCII (Text) Format

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

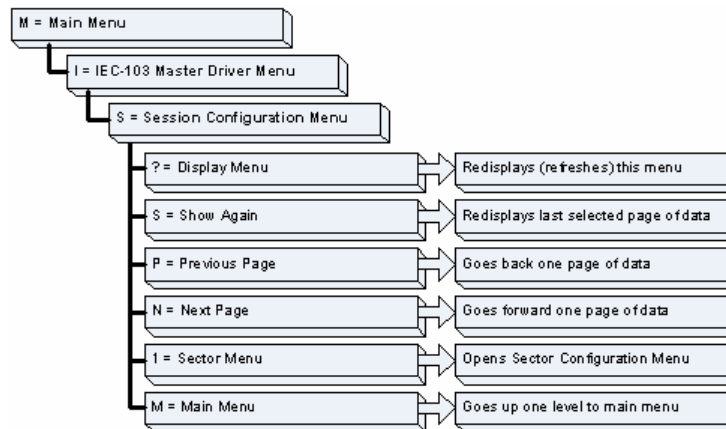




### 6.4.8 Session Configuration Menu

Press **[S]** from the IEC-103 Master Driver Menu to open the Session Configuration menu. Use this command to view the session configuration for each controlled device.

```
IEC-870-5-103 MASTER SESSION 0 CONFIGURATION
Online State           = 0
Session State          = 2 (0)
Communication Port     = 0
Sector Count           = 3
Data Link Address      = 0
Failure Delay          = 10
Confirm Timeout        = 1000
Retry Count            = 0
C1/C2 Poll Count Pend = 5
Class 1 Polls          = 0
Class 1 Pend Delay     = 0
Class 2 Pend Delay     = 0
Class 1 Poll Delay     = 0
Class 2 Poll Delay     = 0
Response Timeout       = 20000
```



#### Online State

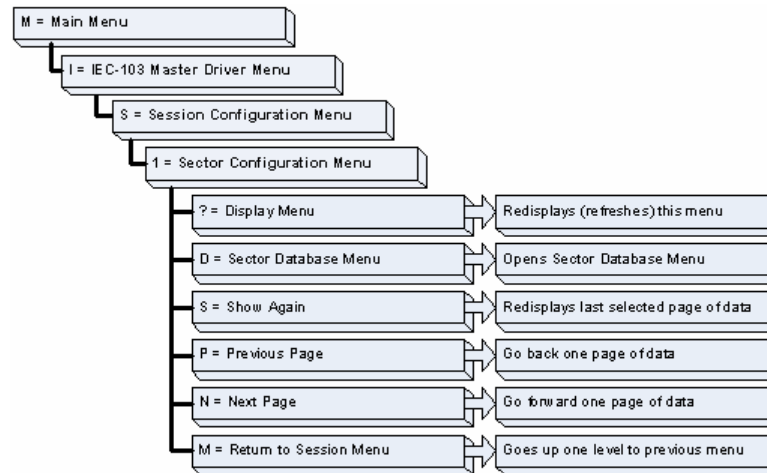
The Online State indicator displays 0 if the module is not online, 1 if the module is online.

#### Session State

The Session State indicator displays 1 if there is a configuration error, or 2 if the module is ready for communication. If the session is not in use, the Session State indicator displays 0.

### 6.4.9 Sector Configuration Menu

Press **[1]** from the IEC-103 Master Driver Menu to open the Sector Configuration menu. Use this command to view the contents of the Sector Configuration Databases for each session (controlled device). The module supports up to three sectors (databases) per session.



#### Redisplaying the Menu

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

#### Opening the Sector Database Menu

Press **[D]** from the Sector Configuration menu to open the Sector Database menu. Use this command to check the configuration and current value for each point.

```

SECTOR CONFIGURATION MENU
?=Display Menu
S=Show again
0=ASDU 1 and 2 data
1=ASDU 3 data
2=ASDU 4 data
3=ASDU 5 data
4=ASDU 9 data
5=List ASDU point counts
M=Return to Sector
  
```

The *IEC-870-Master Command List Menu* section has more information about the commands on this menu.

Redisplaying the Current Page

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

Displaying the Next Page

Press **[N]** to display the next 100 registers. Use this command to step forward through the data a page at a time.

Displaying the Previous Page

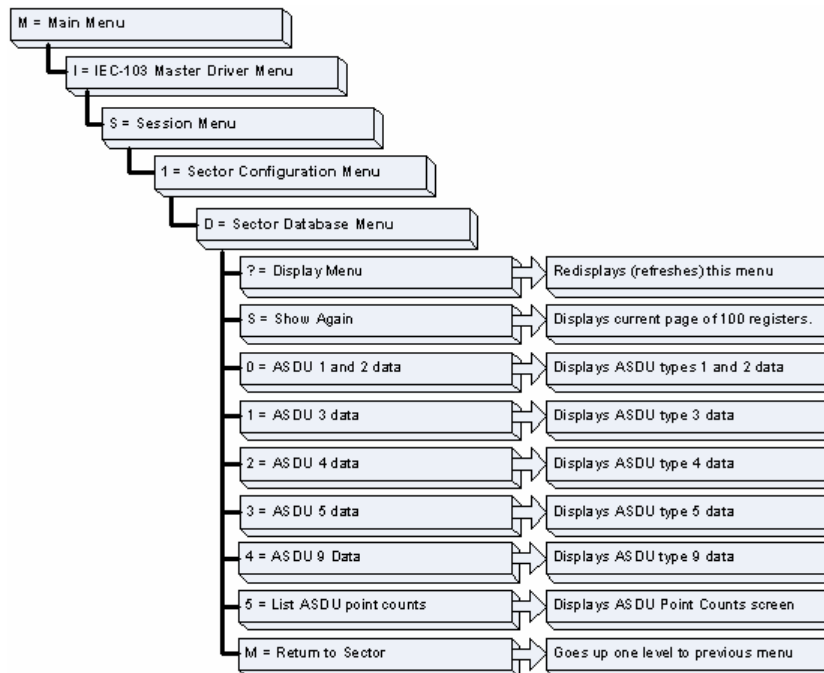
Press **[P]** to display the previous 100 registers. Use this command to step backward through the data a page at a time.

Returning to the Main Menu

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

### 6.4.10 Sector Database Menu

Press **[D]** from the Sector Configuration menu to open the Sector Database menu. Use this command to display the sector database values. Each session (controlled device) contains one or more data sets (sectors) that are defined by the vendor of the device.



Redisplaying the Menu

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

Redisplaying the Current Page

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

Viewing ASDU n Data

Press keys **[0]** (zero) through **[4]** to display ASDU (Application Data Service Unit) data for each of the supported data types. Refer to *[IEC-103 Master Session x Sector x]* for a list of ASDU types.

Listing ASDU point counts

Press **[5]** to display the ASDU point counts for each ASDU type.

| SESSION 0 | SECTOR 0 | DATABASE SIZES: |
|-----------|----------|-----------------|
| ASDU      | COUNT    |                 |
| 1&2       | 10       |                 |
| 3         | 0        |                 |
| 4         | 0        |                 |
| 5         | 0        |                 |
| 9         | 0        |                 |

Returning to the Main Menu

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

## 6.5 LED Status Indicators

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

| ProSoft Module | Color | Status | Indication   |
|----------------|-------|--------|--|
| P1             | Green | On     | Data is being transferred between the module and a remote terminal using the Configuration/Debug port. |
|                |       | Off    | No data is being transferred on the Configuration/Debug port.  |
| P2             | Green | On     | Data being transferred on the first application port   |
|                |       | Off    | No data being transferred on port  |
| P3             | Green | On     | Data being transferred on the second application port  |
|                |       | Off    | No data being transferred on port  |
| APP Status     | Amber | Off    | The MVI46-103M is working normally.  |
|                |       | On     | The MVI46-103M module program has recognized a communication error.                                    |

| ProSoft Module | Color         | Status | Indication   |
|----------------|---------------|--------|--|
| BP ACT         | Amber         | On     | The LED is on when the module is performing a write operation on the backplane.  |
|                |               | Off    | The LED is off when the module is performing a read operation on the backplane. Under normal operation, the LED should blink rapidly on and off.   |
| OK             | Red/<br>Green | Off    | The card is not receiving any power and is not securely plugged into the rack.   |
|                |               | Green  | The module is operating normally.  |
|                |               | Red    | The program has detected an error or is being configured. If the LED remains red for over 10 seconds, the program has probably halted. Remove the card from the rack and re-insert the card to restart the module's program. |
| BAT            | Red           | Off    | The battery voltage is OK and functioning.   |
|                |               | On     | The battery voltage is low or the battery is not present. Replace the battery on the module.   |

### 6.5.1 Clearing a Fault Condition

Typically, if the ACT/FAULT LED on the front of the module becomes illuminated red for over ten seconds, a hardware problem has been detected in the module or the program has exited. To attempt to clear the condition:

- 1 Turn the power to the rack off
- 2 Remove the card from the rack
- 3 Re-insert the card in the rack and turn the power back on
- 4 Verify the configuration data being transferred to the module from the SLC processor

If the module's ACT/FAULT LED does not turn green, make sure the module is inserted completely into the rack. If this does not cure the problem, contact the factory.

### 6.5.2 Troubleshooting

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

| Problem Description | Steps to take   |
|---------------------|---|
| Processor Fault     | Be sure that the module is plugged into the slot that has been configured for the MVI46-103M module. Be sure the ladder logic has been set up correctly |

---

| <b>Problem Description</b>              | <b>Steps to take</b>   |
|---|--|
| BP ACT LED remains off or blinks slowly | <p>This indicates that backplane transfer operations are failing. Use the Configuration/Debug port facility to check this. To establish backplane communications, verify the following items:</p> <ul style="list-style-type: none"><li>▪ The backplane driver is loaded in the module.</li><li>▪ The ladder logic handles all read and write block situations.</li><li>▪ The module is configured in the processor.</li></ul> |
| OK LED remains red                      | <p>The program has halted or a critical error has occurred. Connect to the Configuration/Debug port to see if the module is running. If the program has halted, remove the card from the rack, then re-insert.</p>   |

---





## 7 Reference

### *In This Chapter*

- Cable Connections ..... 73
- Setting Jumpers ..... 77
- MVI46-103M Status Data Area ..... 77
- MVI46-103M Configuration Data Definition ..... 82
- MVI46-103M Configuration File Example ..... 86
- Database Form ..... 158
- Command List Form ..... 160
- Protocol Support ..... 161
- Protocol Interoperability Documentation ..... 172

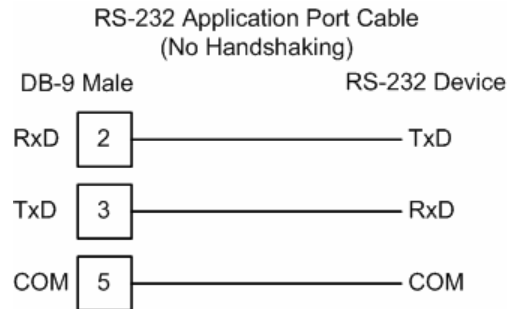
### 7.1 Cable Connections

The application ports on the MVI46-103M module support RS-232, RS-422, and RS-485 interfaces. Please check the module to ensure that the jumpers are set correctly to correspond with the type of interface you are using.

**Note:** When using RS-232 with radio modem applications, the module requires hardware handshaking.

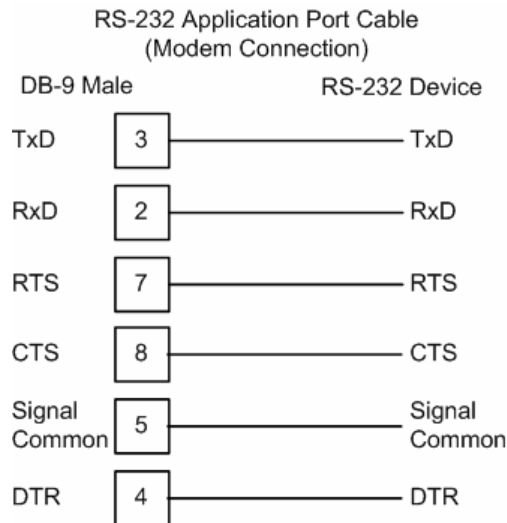
### 7.1.1 RS-232

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



#### RS-232 -- Modem Connection

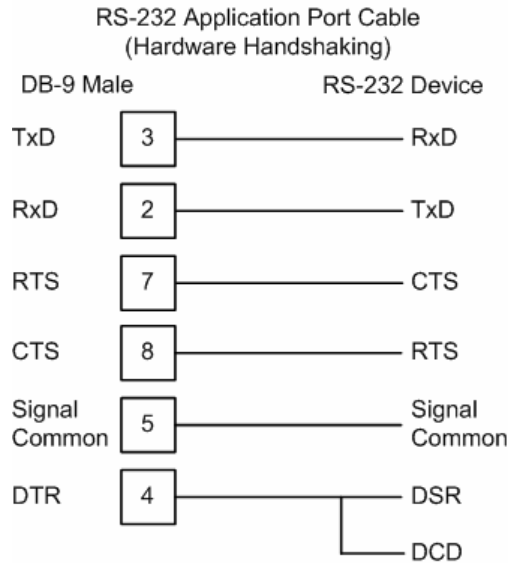
This type of connection is required between the module and a modem or other communication device.



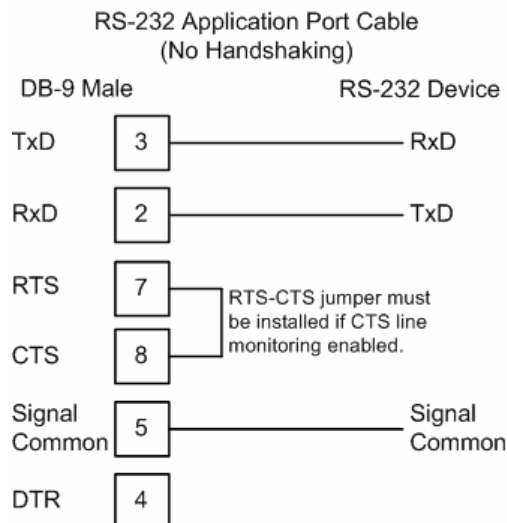
The "Use CTS Line" parameter for the port configuration should be set to 'Y' for most modem applications.

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

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

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

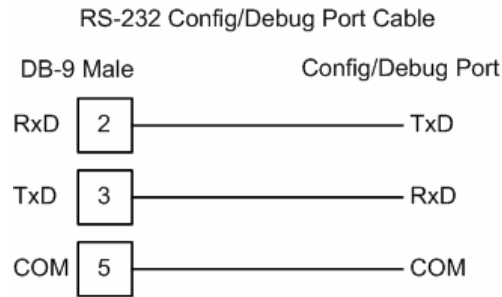
This type of connection can be used to connect the module to a computer or field device communication port.



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

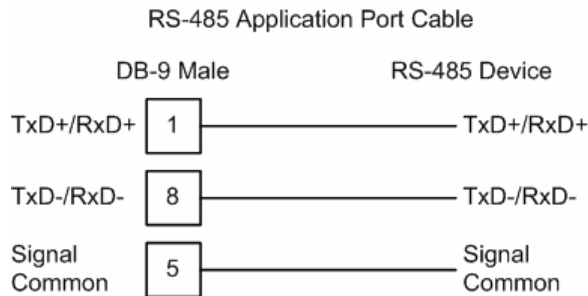
### 7.1.2 RS-232 Configuration/Debug Port

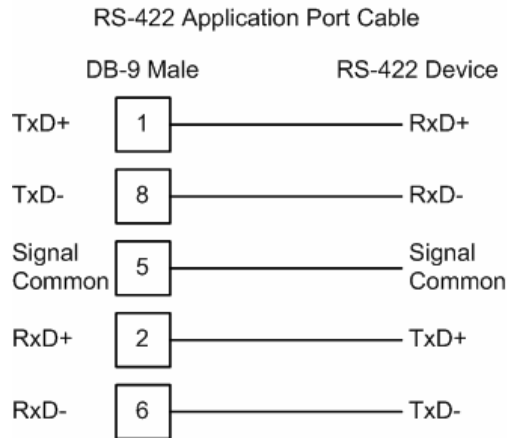
This port is physically an RJ45 connection. An RJ45 to DB-9 adapter cable is included with the module. This port permits a PC based terminal emulation program to view configuration and status data in the module and to control the module. The cable for communications on this port is shown in the following diagram:



### 7.1.3 RS-485

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

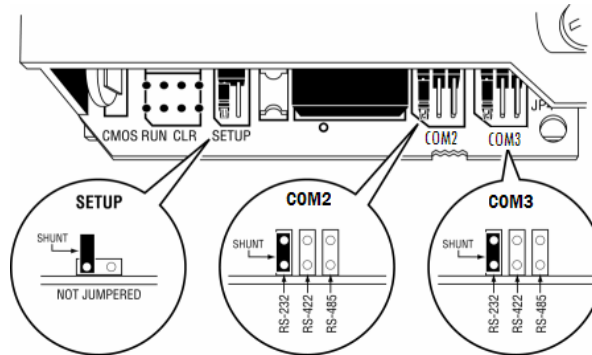


**7.1.4 RS-422****RS-485 and RS-422 Tip**

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

**7.2 Setting Jumpers**

If you use an interface other than RS-232 (default), you must change the jumper configuration to match the interface. The following illustration shows the MVI46-103M jumper configuration:



### 7.3 MVI46-103M Status Data Area

This section contains a listing of the data contained in the MVI46-103M status data object, configuration error word and module error codes.

#### 7.3.1 *MVI46-IEC 60870-5-103 Master Communication Module Error/Status Data Format*

| Offset       | Parameter          | Description  |
|--------------|--------------------|--|
| 4000         | Scan Count         | This status value contains a counter incremented on each scan of the module's main loop.   |
| 4001 to 4002 | Product Name       | This two-word data area contains the text values representing the product name. These words contain the text 'I3M4' for the MVI46 platform.                          |
| 4003 to 4004 | Revision           | This two-word data area contains the text values for the revision number.  |
| 4005 to 4006 | Op Sys #           | This two-word data area contains the text values for the operating system number.  |
| 4007 to 4008 | Run Number         | This two-word data area contains the text values for the run number.   |
| 4009         | Read Blk Cnt       | This word contains the total number of block read operations successfully executed.  |
| 4010         | Write Blk Cnt      | This word contains the total number of block write operations successfully executed.   |
| 4011         | Parse Blk Cnt      | This word contains the total number of write blocks successfully parsed.   |
| 4012         | Error Blk Cnt      | This word contains the total number of block transfer errors.  |
| 4013         | Event Msg Cnt      | This word contains the number of event messages waiting to send to the processor.  |
| 4014         | Event Msg Overflow | This word contains a value of 0 if the event message buffer has not overflowed. If the event buffer overflows, this word will be set to a value of 1.                |
| 4015         | Session Count      | This word contains the number of session configured in the module.   |
| 4016         | Current Cmd        | This word contains the index of the current command being executed in the command list.  |
| 4017         | Cmd Busy Flag      | This word is set to zero if no command is currently being executed and waiting on a response. If the word is set to 1, a command is currently executing.             |
| 4018         | Cmd Count          | This word contains the count of the number of commands configured for the module.  |
| 4019         | Cmd Delay          | This word contains the command delay counter preset. There is a fixed delay between each command to permit the module to perform class polls on controlled stations. |
| 4020         | Cmd Queue          | This word is set to zero if the command executing is from the command list. If the executing command is from the command queue, the word will be set to 1.           |

| Offset       | Parameter       | Description  |
|--------------|-----------------|--|
| 4021         | Cmd Queue Count | This word contains the number of active commands in the command queue for the module. Up to 100 commands can be buffered in this queue. These commands are transferred from the processor to the module using special command blocks.  |
| 4022 to 4023 | Online Status   | This double word value contains a bit for each of the 32 potential sessions in the module. If the bit is set for a session in the double word, the station is online. If the bit is clear, the station is offline. Use this value to determine if commands sent from the processor will have a chance of succeeding. |
| 4024         | CH 0 State      | This word contains the state machine value for channel 0.  |
| 4025         | Cmd Req         | This word contains the number of commands transferred out channel 0.   |
| 4026         | Cmd Resp        | This word contains the number of command response messages received on channel 0.  |
| 4027         | Cmd Err         | This word contains the number of command errors recognized on channel 0.   |
| 4028         | Requests        | This word contains the total number of messages transmitted on channel 0.  |
| 4029         | Responses       | This word contains the total number of messages received on channel 0.   |
| 4030         | Err Sent        | This word contains the number of error messages sent on channel 0.   |
| 4031         | Err Received    | This word contains the number of error messages received on channel 0.   |
| 4032         | Cfg Err         | This bit mapped word recognizes any configuration errors for channel 0. Refer to the configuration error word table for a definition of each bit.  |
| 4033         | Current Error   | This word contains the error code for the current command executing on channel 0.  |
| 4034         | Last Error      | This word contains the error code for the last error recognized on channel 0.  |
| 4035         | CH 1 State      | This word contains the state machine value for channel 1.  |
| 4036         | Cmd Req         | This word contains the number of commands transferred out channel 1.   |
| 4037         | Cmd Resp        | This word contains the number of command response messages received on channel 1.  |
| 4038         | Cmd Err         | This word contains the number of command errors recognized on channel 1.   |
| 4039         | Requests        | This word contains the total number of messages transmitted on channel 1.  |
| 4040         | Responses       | This word contains the total number of messages received on channel 1.   |
| 4041         | Err Sent        | This word contains the number of error messages sent on channel 1.   |
| 4042         | Err Received    | This word contains the number of error messages received on channel 1.   |
| 4043         | Cfg Err         | This bit mapped word recognizes any configuration errors for channel 1. Refer to the configuration error word table for a definition of each bit.  |
| 4044         | Current Error   | This word contains the error code for the current command executing on channel 1.  |

| Offset | Parameter  | Description   |
|--------|------------|---|
| 4045   | Last Error | This word contains the error code for the last error recognized on channel 1. |

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

| Bit | Code   | Description   |
|-----|--------|---|
| 0   | 0x0001 | Invalid baud rate selected                            |
| 1   | 0x0002 | Invalid parity selected                               |
| 2   | 0x0004 | Received timeout set to 0                             |
| 3   | 0x0008 | Invalid Port selected for a session                   |
| 4   | 0x0010 | Invalid sector count for session                      |
| 5   | 0x0020 | Could not allocate memory for sector of a session.    |
| 6   | 0x0040 |   |
| 7   | 0x0080 | Invalid failure delay or confirm timeout for session. |
| 8   | 0x0100 |   |
| 9   | 0x0200 |   |
| 10  | 0x0400 |   |
| 11  | 0x0800 |   |
| 12  | 0x1000 |   |
| 13  | 0x2000 |   |
| 14  | 0x4000 |   |
| 15  | 0x8000 |   |

The following table lists all potential errors that can be generated by the IEC 60870-5-103 Master driver:

### **7.3.2 MVI46-IEC 60870-5-103 Master Communication Module Error Codes**

| Error | Description   |
|-------|---|
| 51    | Physical layer error - Error transmitting message                                     |
| 52    | Physical layer error - Intercharacter timeout occurred before message fully received. |
| 53    | Physical layer error - Frame not entirely received before timeout condition.          |
| 54    | Physical layer error - Invalid frame length.  |
| 101   | Link layer error - Invalid checksum received  |
| 102   | Link layer error - Address unknown to module  |
| 103   | Link layer error - Link established   |
| 104   | Link layer error - Link failed  |
| 105   | Link layer error - Received primary   |
| 106   | Link layer error - FCB error discard  |



| <b>Error</b> | <b>Description</b>                                   |
|--------------|--|
| 107          | Link layer error - FCB error repeat                  |
| 108          | Link layer error - Invalid start character received  |
| 109          | Link layer error - Invalid second character received |
| 110          | Link layer error - Invalid ending character received |
| 111          | Link layer error - Length mismatch error             |
| 112          | Link layer error - Illegal function                  |
| 113          | Link layer error - No confirmation received          |
| 114          | Link layer error - No ACK received                   |
| 115          | Link layer error - Sequence unknown                  |
| 116          | Link layer error - Out of sequence                   |
| 117          | Link layer error - Remote close                      |
| 118          | Link layer error - Unexpected ACK                    |
| 119          | Link layer error - Request cancelled                 |
| 201          | Application layer error - Length mismatch            |
| 202          | Application layer error - Address unknown            |
| 203          | Application layer error - Response late              |
| 251          | RBE error - Clock event buffer overflow              |
| 252          | RBE error - Event buffer overflow                    |
| 271          | Data error - Address unknown                         |
| 281          | Control error - Illegal operation                    |
| 282          | Control error - Illegal value                        |
| 283          | Control error - Not selected                         |
| 301          | Initialization error - Database                      |
| 302          | Initialization error - Out of memory                 |
| 401          | Channel open error                                   |
| 501          | Session error - Database                             |
| 502          | Session error - Configuration                        |
| 601          | No memory to receive message                         |
| 602          | Session not reserved                                 |
| 603          | Illegal session                                      |
| 604          | Session is reserved                                  |
| 605          | Session is not available                             |
| 701          | No memory to transmit message                        |
| 702          | ASDU not supported                                   |
| 703          | Duplicate request                                    |
| 704          | Illegal sector                                       |
| 705          | Control mode is illegal                              |
| 801          | Partial stop request                                 |
| 802          | Stop request failed                                  |
| 901          | Response timeout                                     |

| Error | Description                                      |
|-------|--|
| 902   | Negative COT in response                         |
| 903   | Session is offline                               |
| 904   | Session is disabled                              |
| 905   | Select confirmation received, waiting to execute |
| 906   | Execute confirmation has not be received         |

## 7.4 MVI46-103M Configuration Data Definition

This section contains a listing of the parameters and their definitions for the MVI46-103M module configuration file definition (IEC103M.CFG).

### 7.4.1 MVI46 IEC 60870-5-103 Master Communication Module Configuration

| [Section]/Item            | Value | Range              | Description  |
|---------------------------|-------|--------------------|--|
| [Backplane Configuration] |       |                    | Backplane transfer parameters  |
| Module Name:              |       | 0 to 80 characters | This parameter assigns a name to the module that can be viewed using the configuration/debug port. It can be used to identify the module and the configuration file.   |
| Failure Flag Count:       |       | 0 to 65535         | This parameter specifies the number of successive transfer errors that must occur before the communication ports are shut down. If the parameter is set to 0, the communication ports will continue to operate under all conditions. If the value is set larger than 0 (1-65535), communications will cease if the specified number of failures occur. |
| Pass-Through Events:      |       | Y or N             | This parameter specifies if event messages received on the master ports will be passed to the processor. If the parameter is set to N, event messages will not be passed to the processor. If the parameter is set to Y, the module will pass all events received to the processor using block identifier 9903.  |

| [Section]/Item           | Value | Range   | Description  |
|--------------------------|-------|---------|--|
| [IEC 60870-5-103 MASTER] |       |         | Module level parameters  |
| Session Count:           |       | 1 to 32 | This parameter specifies the maximum number of session to establish on the module. This corresponds to the number of slaves to be interfaced with the module. This value represents the total number of slaves on both ports combined. |

| [Section]/Item                  | Value | Range               | Description   |
|---------------------------------|-------|---------------------|---|
| [IEC 60870-5-103 MASTER PORT x] |       |                     | Settings for each communication port on module  |
| Baud Rate:                      |       | Value for baud rate | This parameter specifies the baud rate to be used on the communication channel (port). Values from 110 to 19200K are permitted.   |
| Parity:                         |       | None, Odd, Even     | This parameter sets the parity to be used on the port. The values correspond to the following settings: None, Odd, Even.<br>NOTE: the specification only utilizes Even Parity.  |
| RTS On:                         |       | 0 to 65535          | The parameter sets the RTS present delay. The value entered represents the number of milliseconds the module will wait after setting the RTS modem line before sending the data.  |
| RTS Off:                        |       | 1 to 65535          | This parameter sets the RTS off delay. The value entered represents the number of milliseconds the module will wait after the data packet is sent before dropping the RTS modem line.   |
| Minimum Delay:                  |       | 1 to 60000          | This parameter specifies the minimum number of milliseconds to delay before sending the message (setting RTS high). This can be used when the serial network requires time for units to turn off their transmitters.  |
| Receive Timeout:                |       | 1 to 65535          | This value represents the number of milliseconds to wait on a port from the time the first character is received until the last character in the longest message received on the port. This parameter should be set dependent on the baud rate. A value of 2000 should work with most applications. |
| Single char ACK F0,1 or 3:      |       | Y or N              | This parameter specifies if the single E5 character will be used for ACK messages.  |
| [Section]/Item                  | Value | Range               | Description   |
| [IEC-103 MASTER SESSION x]      |       |                     | Settings for each session utilized  |
| Communication Port:             |       | 0 or 1              | This parameter sets the port to which the controlled device is connected. On this module, values of 0 and 1 are permitted.  |
| Sector Count:                   |       | 1 to 5              | This parameter sets the number of sectors contained in this controlled device. This version of the application supports 1 to 5 sectors for each session.  |

| [Section]/Item                      | Value        | Range                   | Description   |
|-------------------------------------|--------------|-------------------------|---|
| Data Link Address:                  |              | 0 to 254                | This parameter uniquely defines the data link address for this unit on the communication channel. The ranges of values are from 0 to 254. Address 255 is the broadcast address.                           |
| Failure Delay:                      |              | 0 to 2000               | This parameter sets the minimum number of seconds to delay before polling this session when it is not online. This parameter is only used in unbalanced mode.   |
| Confirm Timeout:                    |              | 0 to 2 <sup>32</sup> -1 | This parameter sets the number of milliseconds to wait for a confirm response from the controlled device.   |
| Retry Count:                        |              | 0 to 255                | This parameter sets the number of retries to be performed on the controlled device when a communication error occurs.   |
| C1/C2 Poll Count Pend:              |              | 0 to 100                | This parameter sets the maximum number of class 1 and class 2 polls performed on this session before trying the next session. This parameter prevents a session from monopolizing the communication port. |
| Class 1 Polls:                      |              | 0 to 65535              | This parameter sets the maximum number of class one polls performed on this session before switching to another session. This parameter prevents a session from monopolizing the communication port.      |
| Class 1 Pend Delay:                 |              | 0 to 2 <sup>32</sup> -1 | This parameter sets the minimum number of milliseconds to delay between class 1 polls for pending data.   |
| Class 2 Pend Delay:                 |              | 0 to 2 <sup>32</sup> -1 | This parameter sets the minimum number of milliseconds to delay between class 2 polls for pending data.   |
| Class 1 Poll Delay:                 |              | 0 to 2 <sup>32</sup> -1 | This parameter sets the minimum number of milliseconds to delay between each class 1 poll.  |
| Class 2 Poll Delay:                 |              | 0 to 2 <sup>32</sup> -1 | This parameter sets the minimum number of milliseconds to delay between each class 2 poll.  |
| Response Timeout:                   |              | 0 to 2 <sup>32</sup> -1 | This parameter sets the maximum number of milliseconds to wait for a confirmation from the controlled station to a request from this module.  |
| <b>[Section]/Item</b>               | <b>Value</b> | <b>Range</b>            | <b>Description</b>  |
| [IEC-103 MASTER SESSION x SECTOR y] |              |                         | This section sets the parameters for a specific sector of a session.  |

| [Section]/Item                 | Value | Range    | Description  |
|--------------------------------|-------|----------|--|
| Common ASDU Address:           |       | 0 to 255 | This parameter sets the common ASDU address to association with this sector of the specified session. This parameter is usually set the same as the data link address when only one sector is utilized.  |
| Online Time Sync:              |       | Y or N   | This parameter specifies if the sector in the controlled device will be sent a time synchronization command when the unit is first recognized as being online. This should only be utilized for devices that do not send an EOI message after initializing.  |
| Online General Int:            |       | Y or N   | This parameter specifies if the sector in the controlled device will be sent a general interrogation command when the unit is first recognized as being online. This should only be utilized for devices that do not send an EOI message after initializing. |
| EOI Time Sync:                 |       | Y or N   | This parameter specifies if the sector in the controlled device will be sent a time synchronization command after this module received an EOI message from the controlled unit.  |
| EOI General Int:               |       | Y or N   | This parameter specifies if the sector in the controlled device will be sent a general interrogation command after this module received an EOI message from the controlled unit.   |
| # ASDU Point Function Database |       |          |  |
| # Type Index Code Point        |       |          |  |
| START                          |       |          |  |
|                                |       |          |  |
| END                            |       |          |  |
|                                |       |          |  |

| [Section]/Item   |  | Description                                       |
|--|--|---|
| [IEC-103 MASTER COMMANDS]                                  |  | This section contains the commands for the module |
| # Enable DB Poll Session Sector Data Func Point Ovrdr      |  |   |
| # Code Index Interval Index Index Type Code Index Flag Val |  |   |
| START  |  |   |
|  |  |   |
| END  |  |   |

Please refer to the **Module Configuration** section for command field definitions.

## 7.5 MVI46-103M Configuration File Example

This section contains a listing of an example IEC103M.CFG file for the MVI46-103M module.

```
# IEC103M.CFG
#
# Example file for use with the IEC-870-5-103 Master protocol driver on the
# MVI46 platform (two master ports).
#
[Backplane Configuration]

Module Name : MVI46-103M DEMO MODULE

Failure Flag Count : 0 #Determines if BP failure will cause protocol to be
                    #disabled (0=Ignore, >0 = failure count to disable)
Pass-Through Events : Y #Pass event messages to processor

# This is the configuration for the IEC-870-5-103 master port emulated on the
# module.
[IEC-870-5-103 Master]
Session Count : 1 #1 to 32 - maximum number of slaves on all channels

[IEC-870-5-103 Master Port 0]
# Communication Parameters
Baud Rate : 19200 #Baud rate for port 110-115200
Parity : E #N=None, O=Odd, E=Even, M=Mark, S=Space
RTS On : 0 #0-65536 mSec before message
RTS Off : 0 #0-65536 mSec after message
Minimum Delay : 10 #Minimum # of mSec before response sent
Receive Timeout : 2000 #Maximum mSec from first char to last to wait

# These parameters are protocol specific
Single char ACK F0,1 or 3 : Y #Single E5 resp to ACK func 0, 1 & 3 req (Y/N)

[IEC-870-5-103 Master Port 1]
# Communication Parameters
Baud Rate : 19200 #Baud rate for port 110-115200
Parity : E #N=None, O=Odd, E=Even, M=Mark, S=Space
RTS On : 0 #0-65536 mSec before message
RTS Off : 0 #0-65536 mSec after message
Minimum Delay : 5 #Minimum # of mSec before response sent
Receive Timeout : 1000 #Maximum mSec from first char to last to wait

# These parameters are protocol specific
Single char ACK F0,1 or 3 : Y #Single E5 resp to ACK func 0, 1 & 3 req (Y/N)

# This section is used to define session 0 which runs on Port 0
[IEC-103 Master Session 0]
Communication Port : 0 #Index of COM port for session (0 or 1)
Sector Count : 5 #5 is max for this version of app
Data Link Address : 0 #Range is 0 to 65535 DL address of slave

Failure Delay : 3 #Min Sec to delay before poll of offline slave
              #(0 to 2000 seconds)
```

```

Confirm Timeout      : 20000 #0 to 2^32-1 mSec to wait for DL confirm
Retry Count         :      5 #0 to 255 retries for if no confirm
Response Timeout    : 5000 #Timeout for confirm of req (0 to 2^32-1)
C1/C2 Poll Count Pend :      6 #class 1 or 2 polls before next slave tried (0-
65535)
Class 1 Polls       :      10 #Max class 1 polls to this session
Class 1 Pend delay   : 1000 #Min mSec delay between call (0 to 2^32-1)
Class 2 Pend delay   : 1000 #Min mSec delay between call (0 to 2^32-1)
Class 1 Poll delay   : 1000 #Min mSec delay between call (0 to 2^32-1)
Class 2 Poll delay   : 1000 #Min mSec delay between call (0 to 2^32-1)

[IEC-103 Master Session 0 Sector 0]
Common ASDU Address :      0 #Range 0 to 255 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync     :      Y #Send time sync message when first online
Online General Int   :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync        :      Y #Send time sync message when first online
EOI General Int      :      Y #Send general interrogation

# Database definition for this session/sector
# Types are as follows:
#
# Monitored ASDU Types
# 1 = Time-tagged message (bit addressed with 2 bits/point)
# 2 = Time-tagged message with relative time (bit addressed with 2 bits/point)
# 3 = Measurands I (4 word values using word address using double-word address)
# 4 = Time-tagged measurands with relative time (1 float value)
# 5 = Identification (12 characters using a byte address)
# 9 = Measurands II (9 word values using word address)
#
# Point Index is the information object number in the slave unit.
# ASDU Index Description
#   1 Bit address with each point occupying 2 bits
#   2 Bit address with each point occupying 2 bits
#   3 Word address with each point occupying 4 words
#   4 Double-word address for the single float value
#   5 Byte address with each point occupying 12 bytes
#   9 Word address with each point occupying 9 words
#
# Function Code Values
# Code Definition
# 128 Distance protection
# 160 Overcurrent protection
# 176 Transformer differential protection
# 192 Line Differential protection
# 255 Global function type
#
# Database Index = Database address in module
#
# ASDU Database Function Point
# Type Index Code Index

```

```
START
  1      0      128      16
  1     16     128     18
  1     32     128     19
  1     48     128     20
  1     64     128     21
  1     80     128     22
  1     96     128     23
  1    112     128     24
  1    128     128     25
  1    144     128     26
END
```

```
[IEC-103 Master Session 0 Sector 1]
```

```
Common ASDU Address      :      1 #Range 0 to 255 Sector address
```

```
#Req init requests when session first online (not req if slave sends  
# EOI sequence)
```

```
Online Time sync        :      Y #Send time sync message when first online
```

```
Online General Int      :      Y #Send general interrogation
```

```
#Req init requests when EOI (end of initialization) received from slave
```

```
EOI Time sync          :      Y #Send time sync message when first online
```

```
EOI General Int        :      Y #Send general interrogation
```

```
# Database definition for this session/sector
```

```
# Types are as follows:
```

```
#
```

```
# Monitored ASDU Types
```

```
# 1 = Time-tagged message (bit addressed with 2 bits/point)
```

```
# 2 = Time-tagged message with relative time (bit addressed with 2 bits/point)
```

```
# 3 = Measurands I (4 word values using word address using double-word address)
```

```
# 4 = Time-tagged measurands with relative time (1 float value)
```

```
# 5 = Identification (12 characters using a byte address)
```

```
# 9 = Measurands II (9 word values using word address)
```

```
#
```

```
# Point Index is the information object number in the slave unit.
```

```
# ASDU Index Description
```

```
# 1 Bit address with each point occupying 2 bits
```

```
# 2 Bit address with each point occupying 2 bits
```

```
# 3 Word address with each point occupying 4 words
```

```
# 4 Double-word address for the single float value
```

```
# 5 Byte address with each point occupying 12 bytes
```

```
# 9 Word address with each point occupying 9 words
```

```
#
```

```
# Function Code Values
```

```
# Code Definition
```

```
# 128 Distance protection
```

```
# 160 Overcurrent protection
```

```
# 176 Transformer differential protection
```

```
# 192 Line Differential protection
```

```
# 255 Global function type
```

```
#
```

```
# Database Index = Database address in module
```

```
#
```



| # ASDU<br># Type | Database<br>Index | Function<br>Code | Point<br>Index |
|------------------|-------------------|------------------|----------------|
| START            |                   |                  |                |
| 1                | 160               | 128              | 27             |
| 1                | 176               | 128              | 28             |
| 1                | 192               | 128              | 29             |
| 1                | 208               | 128              | 30             |
| 1                | 224               | 128              | 32             |
| 1                | 240               | 128              | 33             |
| 1                | 256               | 128              | 35             |
| 1                | 272               | 128              | 36             |
| 1                | 288               | 128              | 37             |
| 1                | 304               | 128              | 38             |

END

[IEC-103 Master Session 0 Sector 2]

Common ASDU Address : 2 #Range 0 to 255 Sector address

#Req init requests when session first online (not req if slave sends

# EOI sequence)

Online Time sync : Y #Send time sync message when first online

Online General Int : Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave

EOI Time sync : Y #Send time sync message when first online

EOI General Int : Y #Send general interrogation

# Database definition for this session/sector

# Types are as follows:

#

# Monitored ASDU Types

# 1 = Time-tagged message (bit addressed with 2 bits/point)

# 2 = Time-tagged message with relative time (bit addressed with 2 bits/point)

# 3 = Measurands I (4 word values using word address using double-word address)

# 4 = Time-tagged measurands with relative time (1 float value)

# 5 = Identification (12 characters using a byte address)

# 9 = Measurands II (9 word values using word address)

#

# Point Index is the information object number in the slave unit.

# ASDU Index Description

# 1 Bit address with each point occupying 2 bits

# 2 Bit address with each point occupying 2 bits

# 3 Word address with each point occupying 4 words

# 4 Double-word address for the single float value

# 5 Byte address with each point occupying 12 bytes

# 9 Word address with each point occupying 9 words

#

# Function Code Values

# Code Definition

# 128 Distance protection

# 160 Overcurrent protection

# 176 Transformer differential protection

# 192 Line Differential protection

# 255 Global function type

#

```
# Database Index = Database address in module
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
    1     320       128       39
    1     336       128       46
    1     352       128       47
    1     368       128       48
    1     384       128       49
    1     400       128       50
    1     416       128       51
    1     432       128       52
    1     448       128      128
    1     464       128      129
END

[IEC-103 Master Session 0 Sector 3]
Common ASDU Address   :      3 #Range 0 to 255 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync         :      Y #Send time sync message when first online
EOI General Int       :      Y #Send general interrogation

# Database definition for this session/sector
# Types are as follows:
#
# Monitored ASDU Types
# 1 = Time-tagged message (bit addressed with 2 bits/point)
# 2 = Time-tagged message with relative time (bit addressed with 2 bits/point)
# 3 = Measurands I (4 word values using word address using double-word address)
# 4 = Time-tagged measurands with relative time (1 float value)
# 5 = Identification (12 characters using a byte address)
# 9 = Measurands II (9 word values using word address)
#
# Point Index is the information object number in the slave unit.
# ASDU Index Description
#   1 Bit address with each point occupying 2 bits
#   2 Bit address with each point occupying 2 bits
#   3 Word address with each point occupying 4 words
#   4 Double-word address for the single float value
#   5 Byte address with each point occupying 12 bytes
#   9 Word address with each point occupying 9 words
#
# Function Code Values
# Code Definition
# 128 Distance protection
# 160 Overcurrent protection
# 176 Transformer differential protection
# 192 Line Differential protection
```

```

# 255 Global function type
#
# Database Index = Database address in module
#
# ASDU      Database  Function   Point
# Type      Index     Code       Index
START
    2        480      128        64
    2        496      128        65
    2        512      128        66
    2        528      128        67
    2        544      128        68
    2        560      128        69
    2        576      128        70
    2        592      128        71
    2        608      128        72
    2        624      128        73
END

[IEC-103 Master Session 0 Sector 4]
Common ASDU Address   :      4 #Range 0 to 255 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync         :      Y #Send time sync message when first online
EOI General Int       :      Y #Send general interrogation

# Database definition for this session/sector
# Types are as follows:
#
# Monitored ASDU Types
# 1 = Time-tagged message (bit addressed with 2 bits/point)
# 2 = Time-tagged message with relative time (bit addressed with 2 bits/point)
# 3 = Measurands I (4 word values using word address using double-word address)
# 4 = Time-tagged measurands with relative time (1 float value)
# 5 = Identification (12 characters using a byte address)
# 9 = Measurands II (9 word values using word address)
#
# Point Index is the information object number in the slave unit.
# ASDU  Index Description
#   1   Bit address with each point occupying 2 bits
#   2   Bit address with each point occupying 2 bits
#   3   Word address with each point occupying 4 words
#   4   Double-word address for the single float value
#   5   Byte address with each point occupying 12 bytes
#   9   Word address with each point occupying 9 words
#
# Function Code Values
# Code  Definition
#  128  Distance protection
#  160  Overcurrent protection

```

```

# 176 Transformer differential protection
# 192 Line Differential protection
# 255 Global function type
#
# Database Index = Database address in module
#
# ASDU      Database  Function  Point
# Type      Index      Code      Index
START
    2        640       128       74
    2        656       128       75
    2        672       128       76
    2        688       128       77
    2        704       128       78
    2        720       128       79
    2        736       128       80
    2        752       128       81
    2        768       128       82
    2        784       128       83
    2        800       128       84
    2        816       128       85
    2        832       128       86
    2        848       128       87
    2        864       128       88
    2        880       128       89
    2        896       128       90
    2        912       128       91
    2        880       128       89
    2        896       128       90
    3         60       128      144
    3         64       128      145
    3         68       128      146
    3         72       128      147
END

# This section is used to define session 0 which runs on Port 1
[IEC-103 Master Session 1]
Communication Port      :    0 #Index of COM port for session (0 or 1)
Sector Count           :    5 #1 is max value for this version of app
Data Link Address      :    1 #Range is 0 to 65535 DL address of slave

Failure Delay          :   10 #Min Sec to delay before poll of offline slave
                        #: (0 to 2000 seconds)

Confirm Timeout        :   1000 #0 to 2^32-1 mSec to wait for DL confirm
Retry Count            :    0 #0 to 255 retries for if no confirm
Response Timeout       :  20000 #Timeout for confirm of req (0 to 2^32-1)
C1/C2 Poll Count Pend :    5 #class 1 or 2 polls before next slave tried (0-
65535)
Class 1 Polls          :    0 #Max class 1 polls to this session
Class 1 Pend delay     :    0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Pend delay     :    0 #Min mSec delay between call (0 to 2^32-1)
Class 1 Poll delay     :    0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Poll delay     :    0 #Min mSec delay between call (0 to 2^32-1)

```

```
[IEC-103 Master Session 1 Sector 0]
Common ASDU Address      :      0 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync        :      Y #Send time sync message when first online
Online General Int      :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync           :      Y #Send time sync message when first online
EOI General Int         :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

[IEC-103 Master Session 1 Sector 1]
Common ASDU Address      :      1 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync        :      Y #Send time sync message when first online
Online General Int      :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync           :      Y #Send time sync message when first online
EOI General Int         :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

[IEC-103 Master Session 1 Sector 2]
Common ASDU Address      :      2 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync        :      Y #Send time sync message when first online
Online General Int      :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync           :      Y #Send time sync message when first online
EOI General Int         :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
```

```
END

[IEC-103 Master Session 1 Sector 3]
Common ASDU Address      :      3 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync        :      Y #Send time sync message when first online
Online General Int      :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync          :      Y #Send time sync message when first online
EOI General Int        :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

[IEC-103 Master Session 1 Sector 4]
Common ASDU Address      :      4 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync        :      Y #Send time sync message when first online
Online General Int      :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync          :      Y #Send time sync message when first online
EOI General Int        :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

# This section is used to define session 0 which runs on Port 1
[IEC-103 Master Session 2]
Communication Port      :      0 #Index of COM port for session (0 or 1)
Sector Count           :      5 #1 is max value for this version of app
Data Link Address      :      2 #Range is 0 to 65535 DL address of slave

Failure Delay          :      10 #Min Sec to delay before poll of offline slave
                        #: (0 to 2000 seconds)
Confirm Timeout        :      1000 #0 to 2^32-1 mSec to wait for DL confirm
Retry Count            :      2 #0 to 255 retries for if no confirm
Response Timeout       :      10000 #Timeout for confirm of req (0 to 2^32-1)
C1/C2 Poll Count Pend :      0 #class 1 or 2 polls before next slave tried (0-
65535)
Class 1 Polls          :      0 #Max class 1 polls to this session
Class 1 Pend delay     :      0 #Min mSec delay between call (0 to 2^32-1)
```

```

Class 2 Pend delay      :      0 #Min mSec delay between call (0 to 2^32-1)
Class 1 Poll delay      :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Poll delay      :      0 #Min mSec delay between call (0 to 2^32-1)

```

```
[IEC-103 Master Session 2 Sector 0]
```

```
Common ASDU Address    :      0 #Range 0 to 65534 Sector address
```

```
#Req init requests when session first online (not req if slave sends
# EOI sequence)
```

```
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation
```

```
#Req init requests when EOI (end of initialization) received from slave
EOI Time sync         :      Y #Send time sync message when first online
EOI General Int       :      Y #Send general interrogation
```

```
# Database definition for this session/sector
```

```
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END
```

```
[IEC-103 Master Session 2 Sector 1]
```

```
Common ASDU Address    :      1 #Range 0 to 65534 Sector address
```

```
#Req init requests when session first online (not req if slave sends
# EOI sequence)
```

```
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation
```

```
#Req init requests when EOI (end of initialization) received from slave
EOI Time sync         :      Y #Send time sync message when first online
EOI General Int       :      Y #Send general interrogation
```

```
# Database definition for this session/sector
```

```
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END
```

```
[IEC-103 Master Session 2 Sector 2]
```

```
Common ASDU Address    :      2 #Range 0 to 65534 Sector address
```

```
#Req init requests when session first online (not req if slave sends
# EOI sequence)
```

```
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation
```

```
#Req init requests when EOI (end of initialization) received from slave
EOI Time sync         :      Y #Send time sync message when first online
EOI General Int       :      Y #Send general interrogation
```

```
# Database definition for this session/sector
```

```
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

[IEC-103 Master Session 2 Sector 3]
Common ASDU Address   :      3 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync         :      Y #Send time sync message when first online
EOI General Int       :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

[IEC-103 Master Session 2 Sector 4]
Common ASDU Address   :      4 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync         :      Y #Send time sync message when first online
EOI General Int       :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

# This section is used to define session 0 which runs on Port 1
[IEC-103 Master Session 3]
Communication Port    :      0 #Index of COM port for session (0 or 1)
Sector Count          :      5 #1 is max value for this version of app
Data Link Address     :      3 #Range is 0 to 65535 DL address of slave

Failure Delay         :      10 #Min Sec to delay before poll of offline slave
                        #(0 to 2000 seconds)
Confirm Timeout       :      1000 #0 to 2^32-1 mSec to wait for DL confirm
Retry Count           :      2 #0 to 255 retries for if no confirm
Response Timeout      :      10000 #Timeout for confirm of req (0 to 2^32-1)
```



```

C1/C2 Poll Count Pend :      0 #class 1 or 2 polls before next slave tried (0-
65535)
Class 1 Polls          :      0 #Max class 1 polls to this session
Class 1 Pend delay     :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Pend delay     :      0 #Min mSec delay between call (0 to 2^32-1)
Class 1 Poll delay     :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Poll delay     :      0 #Min mSec delay between call (0 to 2^32-1)

```

```
[IEC-103 Master Session 3 Sector 0]
```

```
Common ASDU Address   :      0 #Range 0 to 65534 Sector address
```

```
#Req init requests when session first online (not req if slave sends
# EOI sequence)
```

```
Online Time sync      :      Y #Send time sync message when first online
```

```
Online General Int    :      Y #Send general interrogation
```

```
#Req init requests when EOI (end of initialization) received from slave
```

```
EOI Time sync         :      Y #Send time sync message when first online
```

```
EOI General Int       :      Y #Send general interrogation
```

```
# Database definition for this session/sector
```

```
#
```

```
# ASDU   Database   Function   Point
```

```
# Type   Index      Code       Index
```

```
START
```

```
END
```

```
[IEC-103 Master Session 3 Sector 1]
```

```
Common ASDU Address   :      1 #Range 0 to 65534 Sector address
```

```
#Req init requests when session first online (not req if slave sends
```

```
# EOI sequence)
```

```
Online Time sync      :      Y #Send time sync message when first online
```

```
Online General Int    :      Y #Send general interrogation
```

```
#Req init requests when EOI (end of initialization) received from slave
```

```
EOI Time sync         :      Y #Send time sync message when first online
```

```
EOI General Int       :      Y #Send general interrogation
```

```
# Database definition for this session/sector
```

```
#
```

```
# ASDU   Database   Function   Point
```

```
# Type   Index      Code       Index
```

```
START
```

```
END
```

```
[IEC-103 Master Session 3 Sector 2]
```

```
Common ASDU Address   :      2 #Range 0 to 65534 Sector address
```

```
#Req init requests when session first online (not req if slave sends
```

```
# EOI sequence)
```

```
Online Time sync      :      Y #Send time sync message when first online
```

```
Online General Int    :      Y #Send general interrogation
```

```
#Req init requests when EOI (end of initialization) received from slave
```

```
EOI Time sync      :      Y #Send time sync message when first online
EOI General Int    :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

[IEC-103 Master Session 3 Sector 3]
Common ASDU Address :      3 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync    :      Y #Send time sync message when first online
Online General Int  :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync      :      Y #Send time sync message when first online
EOI General Int    :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

[IEC-103 Master Session 3 Sector 4]
Common ASDU Address :      4 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync    :      Y #Send time sync message when first online
Online General Int  :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync      :      Y #Send time sync message when first online
EOI General Int    :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

# This section is used to define session 0 which runs on Port 1
[IEC-103 Master Session 4]
Communication Port  :      0 #Index of COM port for session (0 or 1)
Sector Count        :      5 #1 is max value for this version of app
Data Link Address   :      4 #Range is 0 to 65535 DL address of slave

Failure Delay       :      10 #Min Sec to delay before poll of offline slave
```

```

#(0 to 2000 seconds)
Confirm Timeout      : 1000 #0 to 2^32-1 mSec to wait for DL confirm
Retry Count         : 2 #0 to 255 retries for if no confirm
Response Timeout    : 10000 #Timeout for confirm of req (0 to 2^32-1)
C1/C2 Poll Count Pend : 0 #class 1 or 2 polls before next slave tried (0-65535)
Class 1 Polls       : 0 #Max class 1 polls to this session
Class 1 Pend delay  : 0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Pend delay  : 0 #Min mSec delay between call (0 to 2^32-1)
Class 1 Poll delay  : 0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Poll delay  : 0 #Min mSec delay between call (0 to 2^32-1)

```

```
[IEC-103 Master Session 4 Sector 0]
```

```
Common ASDU Address : 0 #Range 0 to 65534 Sector address
```

```
#Req init requests when session first online (not req if slave sends
# EOI sequence)
```

```
Online Time sync      : Y #Send time sync message when first online
Online General Int    : Y #Send general interrogation
```

```
#Req init requests when EOI (end of initialization) received from slave
```

```
EOI Time sync        : Y #Send time sync message when first online
EOI General Int      : Y #Send general interrogation
```

```
# Database definition for this session/sector
```

```
#
# ASDU Database Function Point
# Type Index Code Index
START
END
```

```
[IEC-103 Master Session 4 Sector 1]
```

```
Common ASDU Address : 1 #Range 0 to 65534 Sector address
```

```
#Req init requests when session first online (not req if slave sends
# EOI sequence)
```

```
Online Time sync      : Y #Send time sync message when first online
Online General Int    : Y #Send general interrogation
```

```
#Req init requests when EOI (end of initialization) received from slave
```

```
EOI Time sync        : Y #Send time sync message when first online
EOI General Int      : Y #Send general interrogation
```

```
# Database definition for this session/sector
```

```
#
# ASDU Database Function Point
# Type Index Code Index
START
END
```

```
[IEC-103 Master Session 4 Sector 2]
```

```
Common ASDU Address : 2 #Range 0 to 65534 Sector address
```

```
#Req init requests when session first online (not req if slave sends
# EOI sequence)
```

```
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync        :      Y #Send time sync message when first online
EOI General Int      :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

[IEC-103 Master Session 4 Sector 3]
Common ASDU Address  :      3 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync     :      Y #Send time sync message when first online
Online General Int   :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync        :      Y #Send time sync message when first online
EOI General Int      :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

[IEC-103 Master Session 4 Sector 4]
Common ASDU Address  :      4 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync     :      Y #Send time sync message when first online
Online General Int   :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync        :      Y #Send time sync message when first online
EOI General Int      :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

# This section is used to define session 0 which runs on Port 1
[IEC-103 Master Session 5]
Communication Port    :      0 #Index of COM port for session (0 or 1)
```

```

Sector Count      :      5 #1 is max value for this version of app
Data Link Address :      5 #Range is 0 to 65535 DL address of slave

Failure Delay     :      10 #Min Sec to delay before poll of offline slave
                  :      #(0 to 2000 seconds)
Confirm Timeout   :      1000 #0 to 2^32-1 mSec to wait for DL confirm
Retry Count       :      2 #0 to 255 retries for if no confirm
Response Timeout  :      10000 #Timeout for confirm of req (0 to 2^32-1)
C1/C2 Poll Count Pend :      0 #class 1 or 2 polls before next slave tried (0-
65535)
Class 1 Polls     :      0 #Max class 1 polls to this session
Class 1 Pend delay :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Pend delay :      0 #Min mSec delay between call (0 to 2^32-1)
Class 1 Poll delay :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Poll delay :      0 #Min mSec delay between call (0 to 2^32-1)

```

```
[IEC-103 Master Session 5 Sector 0]
```

```
Common ASDU Address :      0 #Range 0 to 65534 Sector address
```

```
#Req init requests when session first online (not req if slave sends
# EOI sequence)
```

```
Online Time sync    :      Y #Send time sync message when first online
```

```
Online General Int  :      Y #Send general interrogation
```

```
#Req init requests when EOI (end of initialization) received from slave
```

```
EOI Time sync       :      Y #Send time sync message when first online
```

```
EOI General Int     :      Y #Send general interrogation
```

```
# Database definition for this session/sector
```

```
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END
```

```
[IEC-103 Master Session 5 Sector 1]
```

```
Common ASDU Address :      1 #Range 0 to 65534 Sector address
```

```
#Req init requests when session first online (not req if slave sends
# EOI sequence)
```

```
Online Time sync    :      Y #Send time sync message when first online
```

```
Online General Int  :      Y #Send general interrogation
```

```
#Req init requests when EOI (end of initialization) received from slave
```

```
EOI Time sync       :      Y #Send time sync message when first online
```

```
EOI General Int     :      Y #Send general interrogation
```

```
# Database definition for this session/sector
```

```
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END
```

```
[IEC-103 Master Session 5 Sector 2]
```

```
Common ASDU Address      :      2 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync         :      Y #Send time sync message when first online
Online General Int       :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync           :      Y #Send time sync message when first online
EOI General Int         :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

[IEC-103 Master Session 5 Sector 3]
Common ASDU Address      :      3 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync         :      Y #Send time sync message when first online
Online General Int       :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync           :      Y #Send time sync message when first online
EOI General Int         :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

[IEC-103 Master Session 5 Sector 4]
Common ASDU Address      :      4 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync         :      Y #Send time sync message when first online
Online General Int       :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync           :      Y #Send time sync message when first online
EOI General Int         :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END
```

```

# This section is used to define session 0 which runs on Port 1
[IEC-103 Master Session 6]
Communication Port      :    0 #Index of COM port for session (0 or 1)
Sector Count           :    5 #1 is max value for this version of app
Data Link Address      :    6 #Range is 0 to 65535 DL address of slave

Failure Delay          :    10 #Min Sec to delay before poll of offline slave
                        :          #(0 to 2000 seconds)
Confirm Timeout        :   1000 #0 to 2^32-1 mSec to wait for DL confirm
Retry Count            :    2 #0 to 255 retries for if no confirm
Response Timeout       :  10000 #Timeout for confirm of req (0 to 2^32-1)
C1/C2 Poll Count Pend :    0 #class 1 or 2 polls before next slave tried (0-
65535)
Class 1 Polls          :    0 #Max class 1 polls to this session
Class 1 Pend delay     :    0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Pend delay     :    0 #Min mSec delay between call (0 to 2^32-1)
Class 1 Poll delay     :    0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Poll delay     :    0 #Min mSec delay between call (0 to 2^32-1)

[IEC-103 Master Session 6 Sector 0]
Common ASDU Address    :    0 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync       :    Y #Send time sync message when first online
Online General Int     :    Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync          :    Y #Send time sync message when first online
EOI General Int        :    Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

[IEC-103 Master Session 6 Sector 1]
Common ASDU Address    :    1 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync       :    Y #Send time sync message when first online
Online General Int     :    Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync          :    Y #Send time sync message when first online
EOI General Int        :    Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index

```

START  
END

[IEC-103 Master Session 6 Sector 2]

Common ASDU Address : 2 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends  
# EOI sequence)

Online Time sync : Y #Send time sync message when first online

Online General Int : Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave

EOI Time sync : Y #Send time sync message when first online

EOI General Int : Y #Send general interrogation

# Database definition for this session/sector

#

| # ASDU | Database | Function | Point |
|--------|----------|----------|-------|
|--------|----------|----------|-------|

| # Type | Index | Code | Index |
|--------|-------|------|-------|
|--------|-------|------|-------|

START

END

[IEC-103 Master Session 6 Sector 3]

Common ASDU Address : 3 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends  
# EOI sequence)

Online Time sync : Y #Send time sync message when first online

Online General Int : Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave

EOI Time sync : Y #Send time sync message when first online

EOI General Int : Y #Send general interrogation

# Database definition for this session/sector

#

| # ASDU | Database | Function | Point |
|--------|----------|----------|-------|
|--------|----------|----------|-------|

| # Type | Index | Code | Index |
|--------|-------|------|-------|
|--------|-------|------|-------|

START

END

[IEC-103 Master Session 6 Sector 4]

Common ASDU Address : 4 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends  
# EOI sequence)

Online Time sync : Y #Send time sync message when first online

Online General Int : Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave

EOI Time sync : Y #Send time sync message when first online

EOI General Int : Y #Send general interrogation

# Database definition for this session/sector

#



```

# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

# This section is used to define session 0 which runs on Port 1
[IEC-103 Master Session 7]
Communication Port      :      0 #Index of COM port for session (0 or 1)
Sector Count           :      5 #1 is max value for this version of app
Data Link Address      :      7 #Range is 0 to 65535 DL address of slave

Failure Delay          :      10 #Min Sec to delay before poll of offline slave
                        :          #(0 to 2000 seconds)
Confirm Timeout        :     1000 #0 to 2^32-1 mSec to wait for DL confirm
Retry Count            :      2 #0 to 255 retries for if no confirm
Response Timeout       :    10000 #Timeout for confirm of req (0 to 2^32-1)
C1/C2 Poll Count Pend :      0 #class 1 or 2 polls before next slave tried (0-
65535)
Class 1 Polls          :      0 #Max class 1 polls to this session
Class 1 Pend delay     :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Pend delay     :      0 #Min mSec delay between call (0 to 2^32-1)
Class 1 Poll delay     :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Poll delay     :      0 #Min mSec delay between call (0 to 2^32-1)

[IEC-103 Master Session 7 Sector 0]
Common ASDU Address    :      0 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync       :      Y #Send time sync message when first online
Online General Int     :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync          :      Y #Send time sync message when first online
EOI General Int        :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

[IEC-103 Master Session 7 Sector 1]
Common ASDU Address    :      1 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync       :      Y #Send time sync message when first online
Online General Int     :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync          :      Y #Send time sync message when first online
EOI General Int        :      Y #Send general interrogation

```

```
# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

[IEC-103 Master Session 7 Sector 2]
Common ASDU Address   :      2 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync         :      Y #Send time sync message when first online
EOI General Int       :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

[IEC-103 Master Session 7 Sector 3]
Common ASDU Address   :      3 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync         :      Y #Send time sync message when first online
EOI General Int       :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

[IEC-103 Master Session 7 Sector 4]
Common ASDU Address   :      4 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync         :      Y #Send time sync message when first online
```

```

EOI General Int      :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

# This section is used to define session 0 which runs on Port 1
[IEC-103 Master Session 8]
Communication Port   :      0 #Index of COM port for session (0 or 1)
Sector Count        :      5 #1 is max value for this version of app
Data Link Address    :      8 #Range is 0 to 65535 DL address of slave

Failure Delay        :      10 #Min Sec to delay before poll of offline slave
                        #(0 to 2000 seconds)
Confirm Timeout      :      1000 #0 to 2^32-1 mSec to wait for DL confirm
Retry Count          :      2 #0 to 255 retries for if no confirm
Response Timeout     :      10000 #Timeout for confirm of req (0 to 2^32-1)
C1/C2 Poll Count Pend :      0 #class 1 or 2 polls before next slave tried (0-
65535)
Class 1 Polls        :      0 #Max class 1 polls to this session
Class 1 Pend delay   :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Pend delay   :      0 #Min mSec delay between call (0 to 2^32-1)
Class 1 Poll delay   :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Poll delay   :      0 #Min mSec delay between call (0 to 2^32-1)

[IEC-103 Master Session 8 Sector 0]
Common ASDU Address :      0 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync     :      Y #Send time sync message when first online
Online General Int   :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync        :      Y #Send time sync message when first online
EOI General Int      :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

[IEC-103 Master Session 8 Sector 1]
Common ASDU Address :      1 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync     :      Y #Send time sync message when first online
Online General Int   :      Y #Send general interrogation

```

```
#Req init requests when EOI (end of initialization) received from slave
EOI Time sync      :      Y #Send time sync message when first online
EOI General Int    :      Y #Send general interrogation
```

```
# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END
```

```
[IEC-103 Master Session 8 Sector 2]
Common ASDU Address :      2 #Range 0 to 65534 Sector address
```

```
#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync    :      Y #Send time sync message when first online
Online General Int  :      Y #Send general interrogation
```

```
#Req init requests when EOI (end of initialization) received from slave
EOI Time sync      :      Y #Send time sync message when first online
EOI General Int    :      Y #Send general interrogation
```

```
# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END
```

```
[IEC-103 Master Session 8 Sector 3]
Common ASDU Address :      3 #Range 0 to 65534 Sector address
```

```
#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync    :      Y #Send time sync message when first online
Online General Int  :      Y #Send general interrogation
```

```
#Req init requests when EOI (end of initialization) received from slave
EOI Time sync      :      Y #Send time sync message when first online
EOI General Int    :      Y #Send general interrogation
```

```
# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END
```

```
[IEC-103 Master Session 8 Sector 4]
Common ASDU Address :      4 #Range 0 to 65534 Sector address
```

```
#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync    :      Y #Send time sync message when first online
```

```

Online General Int      :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync          :      Y #Send time sync message when first online
EOI General Int        :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

# This section is used to define session 0 which runs on Port 1
[IEC-103 Master Session 9]
Communication Port      :      0 #Index of COM port for session (0 or 1)
Sector Count            :      5 #1 is max value for this version of app
Data Link Address       :      9 #Range is 0 to 65535 DL address of slave

Failure Delay           :      10 #Min Sec to delay before poll of offline slave
                        :      #(0 to 2000 seconds)
Confirm Timeout         :      1000 #0 to 2^32-1 mSec to wait for DL confirm
Retry Count             :      2 #0 to 255 retries for if no confirm
Response Timeout        :      10000 #Timeout for confirm of req (0 to 2^32-1)
C1/C2 Poll Count Pend  :      0 #class 1 or 2 polls before next slave tried (0-
65535)
Class 1 Polls           :      0 #Max class 1 polls to this session
Class 1 Pend delay      :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Pend delay      :      0 #Min mSec delay between call (0 to 2^32-1)
Class 1 Poll delay      :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Poll delay      :      0 #Min mSec delay between call (0 to 2^32-1)

[IEC-103 Master Session 9 Sector 0]
Common ASDU Address     :      0 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync        :      Y #Send time sync message when first online
Online General Int      :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync          :      Y #Send time sync message when first online
EOI General Int        :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

[IEC-103 Master Session 9 Sector 1]
Common ASDU Address     :      1 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends

```

```
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int   :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync        :      Y #Send time sync message when first online
EOI General Int      :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

[IEC-103 Master Session 9 Sector 2]
Common ASDU Address  :      2 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int   :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync        :      Y #Send time sync message when first online
EOI General Int      :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

[IEC-103 Master Session 9 Sector 3]
Common ASDU Address  :      3 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int   :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync        :      Y #Send time sync message when first online
EOI General Int      :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

[IEC-103 Master Session 9 Sector 4]
Common ASDU Address  :      4 #Range 0 to 65534 Sector address
```

```

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync        :      Y #Send time sync message when first online
EOI General Int      :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

# This section is used to define session 0 which runs on Port 1
[IEC-103 Master Session 10]
Communication Port    :      0 #Index of COM port for session (0 or 1)
Sector Count         :      5 #1 is max value for this version of app
Data Link Address    :      10 #Range is 0 to 65535 DL address of slave

Failure Delay        :      10 #Min Sec to delay before poll of offline slave
                        #(0 to 2000 seconds)
Confirm Timeout      :      1000 #0 to 2^32-1 mSec to wait for DL confirm
Retry Count          :      2 #0 to 255 retries for if no confirm
Response Timeout     :      10000 #Timeout for confirm of req (0 to 2^32-1)
C1/C2 Poll Count Pend :      0 #class 1 or 2 polls before next slave tried (0-
65535)
Class 1 Polls        :      0 #Max class 1 polls to this session
Class 1 Pend delay   :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Pend delay   :      0 #Min mSec delay between call (0 to 2^32-1)
Class 1 Poll delay   :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Poll delay   :      0 #Min mSec delay between call (0 to 2^32-1)

[IEC-103 Master Session 10 Sector 0]
Common ASDU Address  :      0 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync        :      Y #Send time sync message when first online
EOI General Int      :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

```

```
[IEC-103 Master Session 10 Sector 1]
Common ASDU Address      :      1 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync         :      Y #Send time sync message when first online
Online General Int       :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync            :      Y #Send time sync message when first online
EOI General Int          :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

[IEC-103 Master Session 10 Sector 2]
Common ASDU Address      :      2 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync         :      Y #Send time sync message when first online
Online General Int       :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync            :      Y #Send time sync message when first online
EOI General Int          :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

[IEC-103 Master Session 10 Sector 3]
Common ASDU Address      :      3 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync         :      Y #Send time sync message when first online
Online General Int       :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync            :      Y #Send time sync message when first online
EOI General Int          :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
```



```

END

[IEC-103 Master Session 10 Sector 4]
Common ASDU Address      :      4 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync        :      Y #Send time sync message when first online
Online General Int      :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync          :      Y #Send time sync message when first online
EOI General Int        :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

# This section is used to define session 0 which runs on Port 1
[IEC-103 Master Session 11]
Communication Port      :      0 #Index of COM port for session (0 or 1)
Sector Count           :      5 #1 is max value for this version of app
Data Link Address      :      11 #Range is 0 to 65535 DL address of slave

Failure Delay          :      10 #Min Sec to delay before poll of offline slave
                        :      #(0 to 2000 seconds)
Confirm Timeout        :      1000 #0 to 2^32-1 mSec to wait for DL confirm
Retry Count            :      2 #0 to 255 retries for if no confirm
Response Timeout      :      10000 #Timeout for confirm of req (0 to 2^32-1)
C1/C2 Poll Count Pend :      0 #class 1 or 2 polls before next slave tried (0-
65535)
Class 1 Polls         :      0 #Max class 1 polls to this session
Class 1 Pend delay    :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Pend delay    :      0 #Min mSec delay between call (0 to 2^32-1)
Class 1 Poll delay    :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Poll delay    :      0 #Min mSec delay between call (0 to 2^32-1)

[IEC-103 Master Session 11 Sector 0]
Common ASDU Address      :      0 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync        :      Y #Send time sync message when first online
Online General Int      :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync          :      Y #Send time sync message when first online
EOI General Int        :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point

```

```
# Type      Index      Code      Index
START
END

[IEC-103 Master Session 11 Sector 1]
Common ASDU Address :      1 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync         :      Y #Send time sync message when first online
EOI General Int       :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU      Database  Function  Point
# Type      Index     Code      Index
START
END

[IEC-103 Master Session 11 Sector 2]
Common ASDU Address :      2 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync         :      Y #Send time sync message when first online
EOI General Int       :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU      Database  Function  Point
# Type      Index     Code      Index
START
END

[IEC-103 Master Session 11 Sector 3]
Common ASDU Address :      3 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync         :      Y #Send time sync message when first online
EOI General Int       :      Y #Send general interrogation

# Database definition for this session/sector
```

```

#
# ASDU   Database   Function   Point
# Type   Index     Code      Index
START
END

[IEC-103 Master Session 11 Sector 4]
Common ASDU Address   :      4 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync         :      Y #Send time sync message when first online
EOI General Int       :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code      Index
START
END

# This section is used to define session 0 which runs on Port 1
[IEC-103 Master Session 12]
Communication Port    :      0 #Index of COM port for session (0 or 1)
Sector Count          :      5 #1 is max value for this version of app
Data Link Address     :      12 #Range is 0 to 65535 DL address of slave

Failure Delay         :      10 #Min Sec to delay before poll of offline slave
                        #(0 to 2000 seconds)
Confirm Timeout       :      1000 #0 to 2^32-1 mSec to wait for DL confirm
Retry Count           :      2 #0 to 255 retries for if no confirm
Response Timeout      :      10000 #Timeout for confirm of req (0 to 2^32-1)
C1/C2 Poll Count Pend :      0 #class 1 or 2 polls before next slave tried (0-
65535)
Class 1 Polls         :      0 #Max class 1 polls to this session
Class 1 Pend delay    :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Pend delay    :      0 #Min mSec delay between call (0 to 2^32-1)
Class 1 Poll delay    :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Poll delay    :      0 #Min mSec delay between call (0 to 2^32-1)

[IEC-103 Master Session 12 Sector 0]
Common ASDU Address   :      0 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync         :      Y #Send time sync message when first online
EOI General Int       :      Y #Send general interrogation

```

```
# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

[IEC-103 Master Session 12 Sector 1]
Common ASDU Address   :      1 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync         :      Y #Send time sync message when first online
EOI General Int       :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

[IEC-103 Master Session 12 Sector 2]
Common ASDU Address   :      2 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync         :      Y #Send time sync message when first online
EOI General Int       :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

[IEC-103 Master Session 12 Sector 3]
Common ASDU Address   :      3 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
```

```

EOI Time sync      :      Y #Send time sync message when first online
EOI General Int   :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

[IEC-103 Master Session 12 Sector 4]
Common ASDU Address :      4 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync   :      Y #Send time sync message when first online
Online General Int :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync      :      Y #Send time sync message when first online
EOI General Int    :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

# This section is used to define session 0 which runs on Port 1
[IEC-103 Master Session 13]
Communication Port :      0 #Index of COM port for session (0 or 1)
Sector Count       :      5 #1 is max value for this version of app
Data Link Address  :      13 #Range is 0 to 65535 DL address of slave

Failure Delay      :      10 #Min Sec to delay before poll of offline slave
                    # (0 to 2000 seconds)
Confirm Timeout    :      1000 #0 to 2^32-1 mSec to wait for DL confirm
Retry Count        :      2 #0 to 255 retries for if no confirm
Response Timeout   :      10000 #Timeout for confirm of req (0 to 2^32-1)
C1/C2 Poll Count Pend :      0 #class 1 or 2 polls before next slave tried (0-
65535)
Class 1 Polls      :      0 #Max class 1 polls to this session
Class 1 Pend delay :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Pend delay :      0 #Min mSec delay between call (0 to 2^32-1)
Class 1 Poll delay :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Poll delay :      0 #Min mSec delay between call (0 to 2^32-1)

[IEC-103 Master Session 13 Sector 0]
Common ASDU Address :      0 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync   :      Y #Send time sync message when first online
Online General Int :      Y #Send general interrogation

```

```
#Req init requests when EOI (end of initialization) received from slave
EOI Time sync      :      Y #Send time sync message when first online
EOI General Int   :      Y #Send general interrogation
```

```
# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END
```

```
[IEC-103 Master Session 13 Sector 1]
Common ASDU Address :      1 #Range 0 to 65534 Sector address
```

```
#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync   :      Y #Send time sync message when first online
Online General Int :      Y #Send general interrogation
```

```
#Req init requests when EOI (end of initialization) received from slave
EOI Time sync      :      Y #Send time sync message when first online
EOI General Int   :      Y #Send general interrogation
```

```
# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END
```

```
[IEC-103 Master Session 13 Sector 2]
Common ASDU Address :      2 #Range 0 to 65534 Sector address
```

```
#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync   :      Y #Send time sync message when first online
Online General Int :      Y #Send general interrogation
```

```
#Req init requests when EOI (end of initialization) received from slave
EOI Time sync      :      Y #Send time sync message when first online
EOI General Int   :      Y #Send general interrogation
```

```
# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END
```

```
[IEC-103 Master Session 13 Sector 3]
Common ASDU Address :      3 #Range 0 to 65534 Sector address
```

```
#Req init requests when session first online (not req if slave sends
# EOI sequence)
```

```

Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

```

```

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync        :      Y #Send time sync message when first online
EOI General Int      :      Y #Send general interrogation

```

```

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

```

```

[IEC-103 Master Session 13 Sector 4]
Common ASDU Address  :      4 #Range 0 to 65534 Sector address

```

```

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

```

```

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync        :      Y #Send time sync message when first online
EOI General Int      :      Y #Send general interrogation

```

```

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

```

```

# This section is used to define session 0 which runs on Port 1
[IEC-103 Master Session 14]
Communication Port    :      0 #Index of COM port for session (0 or 1)
Sector Count         :      5 #1 is max value for this version of app
Data Link Address     :      14 #Range is 0 to 65535 DL address of slave

Failure Delay        :      10 #Min Sec to delay before poll of offline slave
                        #(0 to 2000 seconds)
Confirm Timeout      :      1000 #0 to 2^32-1 mSec to wait for DL confirm
Retry Count          :      2 #0 to 255 retries for if no confirm
Response Timeout     :      10000 #Timeout for confirm of req (0 to 2^32-1)
C1/C2 Poll Count Pend :      0 #class 1 or 2 polls before next slave tried (0-
65535)
Class 1 Polls        :      0 #Max class 1 polls to this session
Class 1 Pend delay   :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Pend delay   :      0 #Min mSec delay between call (0 to 2^32-1)
Class 1 Poll delay   :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Poll delay   :      0 #Min mSec delay between call (0 to 2^32-1)

```

```

[IEC-103 Master Session 14 Sector 0]
Common ASDU Address  :      0 #Range 0 to 65534 Sector address

```

```
#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync        :      Y #Send time sync message when first online
EOI General Int      :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

[IEC-103 Master Session 14 Sector 1]
Common ASDU Address  :      1 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync        :      Y #Send time sync message when first online
EOI General Int      :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

[IEC-103 Master Session 14 Sector 2]
Common ASDU Address  :      2 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync        :      Y #Send time sync message when first online
EOI General Int      :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

[IEC-103 Master Session 14 Sector 3]
```



```

Common ASDU Address      :      3 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync        :      Y #Send time sync message when first online
Online General Int      :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync           :      Y #Send time sync message when first online
EOI General Int         :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

[IEC-103 Master Session 14 Sector 4]
Common ASDU Address      :      4 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync        :      Y #Send time sync message when first online
Online General Int      :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync           :      Y #Send time sync message when first online
EOI General Int         :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

# This section is used to define session 0 which runs on Port 1
[IEC-103 Master Session 15]
Communication Port      :      0 #Index of COM port for session (0 or 1)
Sector Count            :      5 #1 is max value for this version of app
Data Link Address       :      15 #Range is 0 to 65535 DL address of slave

Failure Delay           :      10 #Min Sec to delay before poll of offline slave
                        :      #(0 to 2000 seconds)
Confirm Timeout         :      1000 #0 to 2^32-1 mSec to wait for DL confirm
Retry Count             :      2 #0 to 255 retries for if no confirm
Response Timeout       :      10000 #Timeout for confirm of req (0 to 2^32-1)
C1/C2 Poll Count Pend  :      0 #class 1 or 2 polls before next slave tried (0-
65535)
Class 1 Polls          :      0 #Max class 1 polls to this session
Class 1 Pend delay     :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Pend delay     :      0 #Min mSec delay between call (0 to 2^32-1)
Class 1 Poll delay     :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Poll delay     :      0 #Min mSec delay between call (0 to 2^32-1)

```

```
[IEC-103 Master Session 15 Sector 0]
Common ASDU Address      :      0 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync        :      Y #Send time sync message when first online
Online General Int      :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync           :      Y #Send time sync message when first online
EOI General Int         :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

[IEC-103 Master Session 15 Sector 1]
Common ASDU Address      :      1 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync        :      Y #Send time sync message when first online
Online General Int      :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync           :      Y #Send time sync message when first online
EOI General Int         :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

[IEC-103 Master Session 15 Sector 2]
Common ASDU Address      :      2 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync        :      Y #Send time sync message when first online
Online General Int      :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync           :      Y #Send time sync message when first online
EOI General Int         :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
```

```
START
END
```

```
[IEC-103 Master Session 15 Sector 3]
```

```
Common ASDU Address : 3 #Range 0 to 65534 Sector address
```

```
#Req init requests when session first online (not req if slave sends
# EOI sequence)
```

```
Online Time sync : Y #Send time sync message when first online
```

```
Online General Int : Y #Send general interrogation
```

```
#Req init requests when EOI (end of initialization) received from slave
```

```
EOI Time sync : Y #Send time sync message when first online
```

```
EOI General Int : Y #Send general interrogation
```

```
# Database definition for this session/sector
```

```
#
```

```
# ASDU Database Function Point
```

```
# Type Index Code Index
```

```
START
```

```
END
```

```
[IEC-103 Master Session 15 Sector 4]
```

```
Common ASDU Address : 4 #Range 0 to 65534 Sector address
```

```
#Req init requests when session first online (not req if slave sends
```

```
# EOI sequence)
```

```
Online Time sync : Y #Send time sync message when first online
```

```
Online General Int : Y #Send general interrogation
```

```
#Req init requests when EOI (end of initialization) received from slave
```

```
EOI Time sync : Y #Send time sync message when first online
```

```
EOI General Int : Y #Send general interrogation
```

```
# Database definition for this session/sector
```

```
#
```

```
# ASDU Database Function Point
```

```
# Type Index Code Index
```

```
START
```

```
END
```

```
# This section is used to define session 0 which runs on Port 1
```

```
[IEC-103 Master Session 16]
```

```
Communication Port : 0 #Index of COM port for session (0 or 1)
```

```
Sector Count : 5 #1 is max value for this version of app
```

```
Data Link Address : 16 #Range is 0 to 65535 DL address of slave
```

```
Failure Delay : 10 #Min Sec to delay before poll of offline slave
#(0 to 2000 seconds)
```

```
Confirm Timeout : 1000 #0 to 2^32-1 mSec to wait for DL confirm
```

```
Retry Count : 2 #0 to 255 retries for if no confirm
```

```
Response Timeout : 10000 #Timeout for confirm of req (0 to 2^32-1)
```

```
C1/C2 Poll Count Pend : 0 #class 1 or 2 polls before next slave tried (0-
65535)
```

```
Class 1 Polls : 0 #Max class 1 polls to this session
```

```
Class 1 Pend delay      :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Pend delay      :      0 #Min mSec delay between call (0 to 2^32-1)
Class 1 Poll delay      :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Poll delay      :      0 #Min mSec delay between call (0 to 2^32-1)
```

```
[IEC-103 Master Session 16 Sector 0]
```

```
Common ASDU Address    :      0 #Range 0 to 65534 Sector address
```

```
#Req init requests when session first online (not req if slave sends
# EOI sequence)
```

```
Online Time sync       :      Y #Send time sync message when first online
Online General Int      :      Y #Send general interrogation
```

```
#Req init requests when EOI (end of initialization) received from slave
EOI Time sync          :      Y #Send time sync message when first online
EOI General Int        :      Y #Send general interrogation
```

```
# Database definition for this session/sector
```

```
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END
```

```
[IEC-103 Master Session 16 Sector 1]
```

```
Common ASDU Address    :      1 #Range 0 to 65534 Sector address
```

```
#Req init requests when session first online (not req if slave sends
# EOI sequence)
```

```
Online Time sync       :      Y #Send time sync message when first online
Online General Int      :      Y #Send general interrogation
```

```
#Req init requests when EOI (end of initialization) received from slave
EOI Time sync          :      Y #Send time sync message when first online
EOI General Int        :      Y #Send general interrogation
```

```
# Database definition for this session/sector
```

```
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END
```

```
[IEC-103 Master Session 16 Sector 2]
```

```
Common ASDU Address    :      2 #Range 0 to 65534 Sector address
```

```
#Req init requests when session first online (not req if slave sends
# EOI sequence)
```

```
Online Time sync       :      Y #Send time sync message when first online
Online General Int      :      Y #Send general interrogation
```

```
#Req init requests when EOI (end of initialization) received from slave
EOI Time sync          :      Y #Send time sync message when first online
EOI General Int        :      Y #Send general interrogation
```

```

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

[IEC-103 Master Session 16 Sector 3]
Common ASDU Address   :      3 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync         :      Y #Send time sync message when first online
EOI General Int       :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

[IEC-103 Master Session 16 Sector 4]
Common ASDU Address   :      4 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync         :      Y #Send time sync message when first online
EOI General Int       :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

# This section is used to define session 0 which runs on Port 1
[IEC-103 Master Session 17]
Communication Port    :      0 #Index of COM port for session (0 or 1)
Sector Count         :      5 #1 is max value for this version of app
Data Link Address    :      17 #Range is 0 to 65535 DL address of slave

Failure Delay        :      10 #Min Sec to delay before poll of offline slave
                        #(0 to 2000 seconds)
Confirm Timeout      :      1000 #0 to 2^32-1 mSec to wait for DL confirm
Retry Count          :      2 #0 to 255 retries for if no confirm

```

```
Response Timeout      : 10000 #Timeout for confirm of req (0 to 2^32-1)
Cl/C2 Poll Count Pend : 0 #class 1 or 2 polls before next slave tried (0-
65535)
Class 1 Polls        : 0 #Max class 1 polls to this session
Class 1 Pend delay   : 0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Pend delay   : 0 #Min mSec delay between call (0 to 2^32-1)
Class 1 Poll delay   : 0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Poll delay   : 0 #Min mSec delay between call (0 to 2^32-1)
```

```
[IEC-103 Master Session 17 Sector 0]
```

```
Common ASDU Address : 0 #Range 0 to 65534 Sector address
```

```
#Req init requests when session first online (not req if slave sends
# EOI sequence)
```

```
Online Time sync      : Y #Send time sync message when first online
Online General Int    : Y #Send general interrogation
```

```
#Req init requests when EOI (end of initialization) received from slave
EOI Time sync        : Y #Send time sync message when first online
EOI General Int      : Y #Send general interrogation
```

```
# Database definition for this session/sector
```

```
#
# ASDU Database Function Point
# Type Index Code Index
START
END
```

```
[IEC-103 Master Session 17 Sector 1]
```

```
Common ASDU Address : 1 #Range 0 to 65534 Sector address
```

```
#Req init requests when session first online (not req if slave sends
# EOI sequence)
```

```
Online Time sync      : Y #Send time sync message when first online
Online General Int    : Y #Send general interrogation
```

```
#Req init requests when EOI (end of initialization) received from slave
EOI Time sync        : Y #Send time sync message when first online
EOI General Int      : Y #Send general interrogation
```

```
# Database definition for this session/sector
```

```
#
# ASDU Database Function Point
# Type Index Code Index
START
END
```

```
[IEC-103 Master Session 17 Sector 2]
```

```
Common ASDU Address : 2 #Range 0 to 65534 Sector address
```

```
#Req init requests when session first online (not req if slave sends
# EOI sequence)
```

```
Online Time sync      : Y #Send time sync message when first online
Online General Int    : Y #Send general interrogation
```

```
#Req init requests when EOI (end of initialization) received from slave
EOI Time sync      :      Y #Send time sync message when first online
EOI General Int    :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

[IEC-103 Master Session 17 Sector 3]
Common ASDU Address :      3 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync   :      Y #Send time sync message when first online
Online General Int :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync      :      Y #Send time sync message when first online
EOI General Int    :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

[IEC-103 Master Session 17 Sector 4]
Common ASDU Address :      4 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync   :      Y #Send time sync message when first online
Online General Int :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync      :      Y #Send time sync message when first online
EOI General Int    :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

# This section is used to define session 0 which runs on Port 1
[IEC-103 Master Session 18]
Communication Port :      0 #Index of COM port for session (0 or 1)
Sector Count       :      5 #1 is max value for this version of app
Data Link Address  :      18 #Range is 0 to 65535 DL address of slave
```

```
Failure Delay      :    10 #Min Sec to delay before poll of offline slave
                   :          #(0 to 2000 seconds)
Confirm Timeout    :   1000 #0 to 2^32-1 mSec to wait for DL confirm
Retry Count        :     2 #0 to 255 retries for if no confirm
Response Timeout   :  10000 #Timeout for confirm of req (0 to 2^32-1)
C1/C2 Poll Count Pend :    0 #class 1 or 2 polls before next slave tried (0-
65535)
Class 1 Polls      :     0 #Max class 1 polls to this session
Class 1 Pend delay :     0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Pend delay :     0 #Min mSec delay between call (0 to 2^32-1)
Class 1 Poll delay :     0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Poll delay :     0 #Min mSec delay between call (0 to 2^32-1)
```

[IEC-103 Master Session 18 Sector 0]

```
Common ASDU Address :    0 #Range 0 to 65534 Sector address
```

```
#Req init requests when session first online (not req if slave sends
# EOI sequence)
```

```
Online Time sync    :     Y #Send time sync message when first online
```

```
Online General Int  :     Y #Send general interrogation
```

```
#Req init requests when EOI (end of initialization) received from slave
```

```
EOI Time sync       :     Y #Send time sync message when first online
```

```
EOI General Int     :     Y #Send general interrogation
```

```
# Database definition for this session/sector
```

```
#
```

```
# ASDU   Database   Function   Point
```

```
# Type   Index      Code       Index
```

```
START
```

```
END
```

[IEC-103 Master Session 18 Sector 1]

```
Common ASDU Address :    1 #Range 0 to 65534 Sector address
```

```
#Req init requests when session first online (not req if slave sends
```

```
# EOI sequence)
```

```
Online Time sync    :     Y #Send time sync message when first online
```

```
Online General Int  :     Y #Send general interrogation
```

```
#Req init requests when EOI (end of initialization) received from slave
```

```
EOI Time sync       :     Y #Send time sync message when first online
```

```
EOI General Int     :     Y #Send general interrogation
```

```
# Database definition for this session/sector
```

```
#
```

```
# ASDU   Database   Function   Point
```

```
# Type   Index      Code       Index
```

```
START
```

```
END
```

[IEC-103 Master Session 18 Sector 2]

```
Common ASDU Address :    2 #Range 0 to 65534 Sector address
```

```
#Req init requests when session first online (not req if slave sends
```



```
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync        :      Y #Send time sync message when first online
EOI General Int      :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

[IEC-103 Master Session 18 Sector 3]
Common ASDU Address  :      3 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync        :      Y #Send time sync message when first online
EOI General Int      :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

[IEC-103 Master Session 18 Sector 4]
Common ASDU Address  :      4 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync        :      Y #Send time sync message when first online
EOI General Int      :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

# This section is used to define session 0 which runs on Port 1
[IEC-103 Master Session 19]
```

```
Communication Port      :      0 #Index of COM port for session (0 or 1)
Sector Count           :      5 #1 is max value for this version of app
Data Link Address      :     19 #Range is 0 to 65535 DL address of slave

Failure Delay          :     10 #Min Sec to delay before poll of offline slave
                        :      #(0 to 2000 seconds)
Confirm Timeout        :    1000 #0 to 2^32-1 mSec to wait for DL confirm
Retry Count            :      2 #0 to 255 retries for if no confirm
Response Timeout       :   10000 #Timeout for confirm of req (0 to 2^32-1)
C1/C2 Poll Count Pend :      0 #class 1 or 2 polls before next slave tried (0-
65535)
Class 1 Polls          :      0 #Max class 1 polls to this session
Class 1 Pend delay     :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Pend delay     :      0 #Min mSec delay between call (0 to 2^32-1)
Class 1 Poll delay     :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Poll delay     :      0 #Min mSec delay between call (0 to 2^32-1)
```

[IEC-103 Master Session 19 Sector 0]

```
Common ASDU Address    :      0 #Range 0 to 65534 Sector address
```

```
#Req init requests when session first online (not req if slave sends
# EOI sequence)
```

```
Online Time sync       :      Y #Send time sync message when first online
Online General Int     :      Y #Send general interrogation
```

```
#Req init requests when EOI (end of initialization) received from slave
EOI Time sync          :      Y #Send time sync message when first online
EOI General Int        :      Y #Send general interrogation
```

```
# Database definition for this session/sector
```

```
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END
```

[IEC-103 Master Session 19 Sector 1]

```
Common ASDU Address    :      1 #Range 0 to 65534 Sector address
```

```
#Req init requests when session first online (not req if slave sends
# EOI sequence)
```

```
Online Time sync       :      Y #Send time sync message when first online
Online General Int     :      Y #Send general interrogation
```

```
#Req init requests when EOI (end of initialization) received from slave
EOI Time sync          :      Y #Send time sync message when first online
EOI General Int        :      Y #Send general interrogation
```

```
# Database definition for this session/sector
```

```
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END
```

```

[IEC-103 Master Session 19 Sector 2]
Common ASDU Address      :      2 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync        :      Y #Send time sync message when first online
Online General Int      :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync           :      Y #Send time sync message when first online
EOI General Int         :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

[IEC-103 Master Session 19 Sector 3]
Common ASDU Address      :      3 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync        :      Y #Send time sync message when first online
Online General Int      :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync           :      Y #Send time sync message when first online
EOI General Int         :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

[IEC-103 Master Session 19 Sector 4]
Common ASDU Address      :      4 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync        :      Y #Send time sync message when first online
Online General Int      :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync           :      Y #Send time sync message when first online
EOI General Int         :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START

```

```
END

# This section is used to define session 0 which runs on Port 1
[IEC-103 Master Session 20]
Communication Port      :      0 #Index of COM port for session (0 or 1)
Sector Count           :      5 #1 is max value for this version of app
Data Link Address      :      20 #Range is 0 to 65535 DL address of slave

Failure Delay          :      10 #Min Sec to delay before poll of offline slave
                        :          #(0 to 2000 seconds)
Confirm Timeout        :     1000 #0 to 2^32-1 mSec to wait for DL confirm
Retry Count            :      2 #0 to 255 retries for if no confirm
Response Timeout       :     10000 #Timeout for confirm of req (0 to 2^32-1)
C1/C2 Poll Count Pend :      0 #class 1 or 2 polls before next slave tried (0-
65535)
Class 1 Polls          :      0 #Max class 1 polls to this session
Class 1 Pend delay     :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Pend delay     :      0 #Min mSec delay between call (0 to 2^32-1)
Class 1 Poll delay     :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Poll delay     :      0 #Min mSec delay between call (0 to 2^32-1)

[IEC-103 Master Session 20 Sector 0]
Common ASDU Address    :      0 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync       :      Y #Send time sync message when first online
Online General Int     :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync          :      Y #Send time sync message when first online
EOI General Int        :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

[IEC-103 Master Session 20 Sector 1]
Common ASDU Address    :      1 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync       :      Y #Send time sync message when first online
Online General Int     :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync          :      Y #Send time sync message when first online
EOI General Int        :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
```

```

# Type      Index      Code      Index
START
END

[IEC-103 Master Session 20 Sector 2]
Common ASDU Address :      2 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync         :      Y #Send time sync message when first online
EOI General Int       :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU      Database  Function  Point
# Type      Index     Code      Index
START
END

[IEC-103 Master Session 20 Sector 3]
Common ASDU Address :      3 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync         :      Y #Send time sync message when first online
EOI General Int       :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU      Database  Function  Point
# Type      Index     Code      Index
START
END

[IEC-103 Master Session 20 Sector 4]
Common ASDU Address :      4 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync         :      Y #Send time sync message when first online
EOI General Int       :      Y #Send general interrogation

# Database definition for this session/sector

```

```
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

# This section is used to define session 0 which runs on Port 1
[IEC-103 Master Session 21]
Communication Port      :      0 #Index of COM port for session (0 or 1)
Sector Count           :      5 #1 is max value for this version of app
Data Link Address      :     21 #Range is 0 to 65535 DL address of slave

Failure Delay          :     10 #Min Sec to delay before poll of offline slave
                        :          #(0 to 2000 seconds)
Confirm Timeout        :    1000 #0 to 2^32-1 mSec to wait for DL confirm
Retry Count            :      2 #0 to 255 retries for if no confirm
Response Timeout       :   10000 #Timeout for confirm of req (0 to 2^32-1)
C1/C2 Poll Count Pend :      0 #class 1 or 2 polls before next slave tried (0-
65535)
Class 1 Polls          :      0 #Max class 1 polls to this session
Class 1 Pend delay     :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Pend delay     :      0 #Min mSec delay between call (0 to 2^32-1)
Class 1 Poll delay     :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Poll delay     :      0 #Min mSec delay between call (0 to 2^32-1)

[IEC-103 Master Session 21 Sector 0]
Common ASDU Address    :      0 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync       :      Y #Send time sync message when first online
Online General Int     :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync          :      Y #Send time sync message when first online
EOI General Int        :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

[IEC-103 Master Session 21 Sector 1]
Common ASDU Address    :      1 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync       :      Y #Send time sync message when first online
Online General Int     :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync          :      Y #Send time sync message when first online
EOI General Int        :      Y #Send general interrogation
```

```
# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

[IEC-103 Master Session 21 Sector 2]
Common ASDU Address   :      2 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync         :      Y #Send time sync message when first online
EOI General Int       :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

[IEC-103 Master Session 21 Sector 3]
Common ASDU Address   :      3 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync         :      Y #Send time sync message when first online
EOI General Int       :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

[IEC-103 Master Session 21 Sector 4]
Common ASDU Address   :      4 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
```

```
EOI Time sync      :      Y #Send time sync message when first online
EOI General Int   :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

# This section is used to define session 0 which runs on Port 1
[IEC-103 Master Session 22]
Communication Port :      0 #Index of COM port for session (0 or 1)
Sector Count      :      5 #1 is max value for this version of app
Data Link Address :      22 #Range is 0 to 65535 DL address of slave

Failure Delay     :      10 #Min Sec to delay before poll of offline slave
                    #(0 to 2000 seconds)
Confirm Timeout   :      1000 #0 to 2^32-1 mSec to wait for DL confirm
Retry Count       :      2 #0 to 255 retries for if no confirm
Response Timeout  :      10000 #Timeout for confirm of req (0 to 2^32-1)
C1/C2 Poll Count Pend :      0 #class 1 or 2 polls before next slave tried (0-
65535)
Class 1 Polls     :      0 #Max class 1 polls to this session
Class 1 Pend delay :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Pend delay :      0 #Min mSec delay between call (0 to 2^32-1)
Class 1 Poll delay :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Poll delay :      0 #Min mSec delay between call (0 to 2^32-1)

[IEC-103 Master Session 22 Sector 0]
Common ASDU Address :      0 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync   :      Y #Send time sync message when first online
Online General Int :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync      :      Y #Send time sync message when first online
EOI General Int    :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

[IEC-103 Master Session 22 Sector 1]
Common ASDU Address :      1 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync   :      Y #Send time sync message when first online
Online General Int :      Y #Send general interrogation
```



```
#Req init requests when EOI (end of initialization) received from slave
EOI Time sync      :      Y #Send time sync message when first online
EOI General Int    :      Y #Send general interrogation
```

```
# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END
```

```
[IEC-103 Master Session 22 Sector 2]
Common ASDU Address :      2 #Range 0 to 65534 Sector address
```

```
#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync    :      Y #Send time sync message when first online
Online General Int  :      Y #Send general interrogation
```

```
#Req init requests when EOI (end of initialization) received from slave
EOI Time sync      :      Y #Send time sync message when first online
EOI General Int    :      Y #Send general interrogation
```

```
# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END
```

```
[IEC-103 Master Session 22 Sector 3]
Common ASDU Address :      3 #Range 0 to 65534 Sector address
```

```
#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync    :      Y #Send time sync message when first online
Online General Int  :      Y #Send general interrogation
```

```
#Req init requests when EOI (end of initialization) received from slave
EOI Time sync      :      Y #Send time sync message when first online
EOI General Int    :      Y #Send general interrogation
```

```
# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END
```

```
[IEC-103 Master Session 22 Sector 4]
Common ASDU Address :      4 #Range 0 to 65534 Sector address
```

```
#Req init requests when session first online (not req if slave sends
# EOI sequence)
```

```
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync        :      Y #Send time sync message when first online
EOI General Int      :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

# This section is used to define session 0 which runs on Port 1
[IEC-103 Master Session 23]
Communication Port    :      0 #Index of COM port for session (0 or 1)
Sector Count         :      5 #1 is max value for this version of app
Data Link Address    :      23 #Range is 0 to 65535 DL address of slave

Failure Delay        :      10 #Min Sec to delay before poll of offline slave
                        #(0 to 2000 seconds)
Confirm Timeout      :      1000 #0 to 2^32-1 mSec to wait for DL confirm
Retry Count          :      2 #0 to 255 retries for if no confirm
Response Timeout     :      10000 #Timeout for confirm of req (0 to 2^32-1)
C1/C2 Poll Count Pend :      0 #class 1 or 2 polls before next slave tried (0-
65535)
Class 1 Polls        :      0 #Max class 1 polls to this session
Class 1 Pend delay   :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Pend delay   :      0 #Min mSec delay between call (0 to 2^32-1)
Class 1 Poll delay   :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Poll delay   :      0 #Min mSec delay between call (0 to 2^32-1)

[IEC-103 Master Session 23 Sector 0]
Common ASDU Address  :      0 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync        :      Y #Send time sync message when first online
EOI General Int      :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

[IEC-103 Master Session 23 Sector 1]
Common ASDU Address  :      1 #Range 0 to 65534 Sector address
```

```

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync        :      Y #Send time sync message when first online
EOI General Int      :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

[IEC-103 Master Session 23 Sector 2]
Common ASDU Address  :      2 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync        :      Y #Send time sync message when first online
EOI General Int      :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

[IEC-103 Master Session 23 Sector 3]
Common ASDU Address  :      3 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync        :      Y #Send time sync message when first online
EOI General Int      :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

[IEC-103 Master Session 23 Sector 4]

```

```
Common ASDU Address      :      4 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync         :      Y #Send time sync message when first online
Online General Int       :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync           :      Y #Send time sync message when first online
EOI General Int         :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

# This section is used to define session 0 which runs on Port 1
[IEC-103 Master Session 24]
Communication Port       :      0 #Index of COM port for session (0 or 1)
Sector Count            :      5 #1 is max value for this version of app
Data Link Address       :      24 #Range is 0 to 65535 DL address of slave

Failure Delay           :      10 #Min Sec to delay before poll of offline slave
                        :      #(0 to 2000 seconds)
Confirm Timeout         :      1000 #0 to 2^32-1 mSec to wait for DL confirm
Retry Count             :      2 #0 to 255 retries for if no confirm
Response Timeout       :      10000 #Timeout for confirm of req (0 to 2^32-1)
C1/C2 Poll Count Pend  :      0 #class 1 or 2 polls before next slave tried (0-
65535)
Class 1 Polls          :      0 #Max class 1 polls to this session
Class 1 Pend delay     :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Pend delay     :      0 #Min mSec delay between call (0 to 2^32-1)
Class 1 Poll delay     :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Poll delay     :      0 #Min mSec delay between call (0 to 2^32-1)

[IEC-103 Master Session 24 Sector 0]
Common ASDU Address     :      0 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync        :      Y #Send time sync message when first online
Online General Int      :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync          :      Y #Send time sync message when first online
EOI General Int        :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
```

```
END

[IEC-103 Master Session 24 Sector 1]
Common ASDU Address      :      1 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync        :      Y #Send time sync message when first online
Online General Int      :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync           :      Y #Send time sync message when first online
EOI General Int         :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

[IEC-103 Master Session 24 Sector 2]
Common ASDU Address      :      2 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync        :      Y #Send time sync message when first online
Online General Int      :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync           :      Y #Send time sync message when first online
EOI General Int         :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

[IEC-103 Master Session 24 Sector 3]
Common ASDU Address      :      3 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync        :      Y #Send time sync message when first online
Online General Int      :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync           :      Y #Send time sync message when first online
EOI General Int         :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
```

```
# Type      Index      Code      Index
START
END

[IEC-103 Master Session 24 Sector 4]
Common ASDU Address :      4 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync         :      Y #Send time sync message when first online
EOI General Int       :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU      Database  Function  Point
# Type      Index      Code      Index
START
END

# This section is used to define session 0 which runs on Port 1
[IEC-103 Master Session 25]
Communication Port    :      0 #Index of COM port for session (0 or 1)
Sector Count         :      5 #1 is max value for this version of app
Data Link Address    :      25 #Range is 0 to 65535 DL address of slave

Failure Delay        :      10 #Min Sec to delay before poll of offline slave
                        #(0 to 2000 seconds)
Confirm Timeout      :      1000 #0 to 2^32-1 mSec to wait for DL confirm
Retry Count          :      2 #0 to 255 retries for if no confirm
Response Timeout     :      10000 #Timeout for confirm of req (0 to 2^32-1)
C1/C2 Poll Count Pend :      0 #class 1 or 2 polls before next slave tried (0-
65535)
Class 1 Polls        :      0 #Max class 1 polls to this session
Class 1 Pend delay   :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Pend delay   :      0 #Min mSec delay between call (0 to 2^32-1)
Class 1 Poll delay   :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Poll delay   :      0 #Min mSec delay between call (0 to 2^32-1)

[IEC-103 Master Session 25 Sector 0]
Common ASDU Address :      0 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync         :      Y #Send time sync message when first online
EOI General Int       :      Y #Send general interrogation

# Database definition for this session/sector
```

```
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

[IEC-103 Master Session 25 Sector 1]
Common ASDU Address   :      1 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync         :      Y #Send time sync message when first online
EOI General Int       :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

[IEC-103 Master Session 25 Sector 2]
Common ASDU Address   :      2 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync         :      Y #Send time sync message when first online
EOI General Int       :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

[IEC-103 Master Session 25 Sector 3]
Common ASDU Address   :      3 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync         :      Y #Send time sync message when first online
EOI General Int       :      Y #Send general interrogation
```

```
# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

[IEC-103 Master Session 25 Sector 4]
Common ASDU Address   :      4 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync         :      Y #Send time sync message when first online
EOI General Int       :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

# This section is used to define session 0 which runs on Port 1
[IEC-103 Master Session 26]
Communication Port    :      0 #Index of COM port for session (0 or 1)
Sector Count          :      5 #1 is max value for this version of app
Data Link Address     :      26 #Range is 0 to 65535 DL address of slave

Failure Delay         :      10 #Min Sec to delay before poll of offline slave
                        #(0 to 2000 seconds)
Confirm Timeout       :      1000 #0 to 2^32-1 mSec to wait for DL confirm
Retry Count           :      2 #0 to 255 retries for if no confirm
Response Timeout      :      10000 #Timeout for confirm of req (0 to 2^32-1)
C1/C2 Poll Count Pend :      0 #class 1 or 2 polls before next slave tried (0-
65535)
Class 1 Polls         :      0 #Max class 1 polls to this session
Class 1 Pend delay    :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Pend delay    :      0 #Min mSec delay between call (0 to 2^32-1)
Class 1 Poll delay    :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Poll delay    :      0 #Min mSec delay between call (0 to 2^32-1)

[IEC-103 Master Session 26 Sector 0]
Common ASDU Address   :      0 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
```



```
EOI Time sync      :      Y #Send time sync message when first online
EOI General Int    :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

[IEC-103 Master Session 26 Sector 1]
Common ASDU Address :      1 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync    :      Y #Send time sync message when first online
Online General Int  :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync      :      Y #Send time sync message when first online
EOI General Int    :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

[IEC-103 Master Session 26 Sector 2]
Common ASDU Address :      2 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync    :      Y #Send time sync message when first online
Online General Int  :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync      :      Y #Send time sync message when first online
EOI General Int    :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

[IEC-103 Master Session 26 Sector 3]
Common ASDU Address :      3 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync    :      Y #Send time sync message when first online
Online General Int  :      Y #Send general interrogation
```

```
#Req init requests when EOI (end of initialization) received from slave
EOI Time sync      :      Y #Send time sync message when first online
EOI General Int    :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

[IEC-103 Master Session 26 Sector 4]
Common ASDU Address :      4 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync   :      Y #Send time sync message when first online
Online General Int :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync      :      Y #Send time sync message when first online
EOI General Int    :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

# This section is used to define session 0 which runs on Port 1
[IEC-103 Master Session 27]
Communication Port  :      0 #Index of COM port for session (0 or 1)
Sector Count       :      5 #1 is max value for this version of app
Data Link Address  :      27 #Range is 0 to 65535 DL address of slave

Failure Delay      :      10 #Min Sec to delay before poll of offline slave
                    #:      (0 to 2000 seconds)
Confirm Timeout    :      1000 #0 to 2^32-1 mSec to wait for DL confirm
Retry Count        :      2 #0 to 255 retries for if no confirm
Response Timeout   :      10000 #Timeout for confirm of req (0 to 2^32-1)
C1/C2 Poll Count Pend :      0 #class 1 or 2 polls before next slave tried (0-
65535)
Class 1 Polls      :      0 #Max class 1 polls to this session
Class 1 Pend delay :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Pend delay :      0 #Min mSec delay between call (0 to 2^32-1)
Class 1 Poll delay :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Poll delay :      0 #Min mSec delay between call (0 to 2^32-1)

[IEC-103 Master Session 27 Sector 0]
Common ASDU Address :      0 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
```

```
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync        :      Y #Send time sync message when first online
EOI General Int      :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

[IEC-103 Master Session 27 Sector 1]
Common ASDU Address  :      1 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync     :      Y #Send time sync message when first online
Online General Int   :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync        :      Y #Send time sync message when first online
EOI General Int      :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

[IEC-103 Master Session 27 Sector 2]
Common ASDU Address  :      2 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync     :      Y #Send time sync message when first online
Online General Int   :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync        :      Y #Send time sync message when first online
EOI General Int      :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

[IEC-103 Master Session 27 Sector 3]
Common ASDU Address  :      3 #Range 0 to 65534 Sector address
```

```
#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync        :      Y #Send time sync message when first online
EOI General Int      :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

[IEC-103 Master Session 27 Sector 4]
Common ASDU Address  :      4 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync        :      Y #Send time sync message when first online
EOI General Int      :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

# This section is used to define session 0 which runs on Port 1
[IEC-103 Master Session 28]
Communication Port    :      0 #Index of COM port for session (0 or 1)
Sector Count          :      5 #1 is max value for this version of app
Data Link Address     :      28 #Range is 0 to 65535 DL address of slave

Failure Delay         :      10 #Min Sec to delay before poll of offline slave
                        #(0 to 2000 seconds)
Confirm Timeout       :      1000 #0 to 2^32-1 mSec to wait for DL confirm
Retry Count           :      2 #0 to 255 retries for if no confirm
Response Timeout      :      10000 #Timeout for confirm of req (0 to 2^32-1)
C1/C2 Poll Count Pend :      0 #class 1 or 2 polls before next slave tried (0-
65535)
Class 1 Polls         :      0 #Max class 1 polls to this session
Class 1 Pend delay    :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Pend delay    :      0 #Min mSec delay between call (0 to 2^32-1)
Class 1 Poll delay    :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Poll delay    :      0 #Min mSec delay between call (0 to 2^32-1)

[IEC-103 Master Session 28 Sector 0]
```

```
Common ASDU Address      :      0 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync        :      Y #Send time sync message when first online
Online General Int      :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync           :      Y #Send time sync message when first online
EOI General Int         :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

[IEC-103 Master Session 28 Sector 1]
Common ASDU Address      :      1 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync        :      Y #Send time sync message when first online
Online General Int      :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync           :      Y #Send time sync message when first online
EOI General Int         :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

[IEC-103 Master Session 28 Sector 2]
Common ASDU Address      :      2 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync        :      Y #Send time sync message when first online
Online General Int      :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync           :      Y #Send time sync message when first online
EOI General Int         :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END
```

```
[IEC-103 Master Session 28 Sector 3]
Common ASDU Address      :      3 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync         :      Y #Send time sync message when first online
Online General Int       :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync            :      Y #Send time sync message when first online
EOI General Int          :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

[IEC-103 Master Session 28 Sector 4]
Common ASDU Address      :      4 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync         :      Y #Send time sync message when first online
Online General Int       :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync            :      Y #Send time sync message when first online
EOI General Int          :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

[IEC-103 Master Session 29]
Communication Port       :      0 #Index of COM port for session (0 or 1)
Sector Count             :      5 #1 is max value for this version of app
Data Link Address        :      29 #Range is 0 to 65535 DL address of slave

Failure Delay            :      10 #Min Sec to delay before poll of offline slave
                          #(0 to 2000 seconds)
Confirm Timeout          :      1000 #0 to 2^32-1 mSec to wait for DL confirm
Retry Count              :      2 #0 to 255 retries for if no confirm
Response Timeout         :      10000 #Timeout for confirm of req (0 to 2^32-1)
Cl/C2 Poll Count Pend   :      0 #class 1 or 2 polls before next slave tried (0-
65535)
Class 1 Polls            :      0 #Max class 1 polls to this session
Class 1 Pend delay       :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Pend delay       :      0 #Min mSec delay between call (0 to 2^32-1)
Class 1 Poll delay       :      0 #Min mSec delay between call (0 to 2^32-1)
```

```

Class 2 Poll delay      :      0 #Min mSec delay between call (0 to 2^32-1)

[IEC-103 Master Session 29 Sector 0]
Common ASDU Address    :      0 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync       :      Y #Send time sync message when first online
Online General Int     :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync         :      Y #Send time sync message when first online
EOI General Int       :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

[IEC-103 Master Session 29 Sector 1]
Common ASDU Address    :      1 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync       :      Y #Send time sync message when first online
Online General Int     :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync         :      Y #Send time sync message when first online
EOI General Int       :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

[IEC-103 Master Session 29 Sector 2]
Common ASDU Address    :      2 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync       :      Y #Send time sync message when first online
Online General Int     :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync         :      Y #Send time sync message when first online
EOI General Int       :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point

```

```
# Type      Index      Code      Index
START
END

[IEC-103 Master Session 29 Sector 3]
Common ASDU Address      :      3 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync        :      Y #Send time sync message when first online
Online General Int      :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync          :      Y #Send time sync message when first online
EOI General Int        :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU      Database  Function  Point
# Type      Index     Code      Index
START
END

[IEC-103 Master Session 29 Sector 4]
Common ASDU Address      :      4 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync        :      Y #Send time sync message when first online
Online General Int      :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync          :      Y #Send time sync message when first online
EOI General Int        :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU      Database  Function  Point
# Type      Index     Code      Index
START
END

[IEC-103 Master Session 30]
Communication Port      :      0 #Index of COM port for session (0 or 1)
Sector Count           :      5 #1 is max value for this version of app
Data Link Address      :      30 #Range is 0 to 65535 DL address of slave

Failure Delay          :      10 #Min Sec to delay before poll of offline slave
                        # (0 to 2000 seconds)
Confirm Timeout        :      1000 #0 to 2^32-1 mSec to wait for DL confirm
Retry Count            :      2 #0 to 255 retries for if no confirm
Response Timeout       :      10000 #Timeout for confirm of req (0 to 2^32-1)
C1/C2 Poll Count Pend :      0 #class 1 or 2 polls before next slave tried (0-
65535)
Class 1 Polls          :      0 #Max class 1 polls to this session
```



```

Class 1 Pend delay      :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Pend delay      :      0 #Min mSec delay between call (0 to 2^32-1)
Class 1 Poll delay      :      0 #Min mSec delay between call (0 to 2^32-1)
Class 2 Poll delay      :      0 #Min mSec delay between call (0 to 2^32-1)

```

```
[IEC-103 Master Session 30 Sector 0]
```

```
Common ASDU Address    :      0 #Range 0 to 65534 Sector address
```

```
#Req init requests when session first online (not req if slave sends
# EOI sequence)
```

```
Online Time sync       :      Y #Send time sync message when first online
Online General Int     :      Y #Send general interrogation
```

```
#Req init requests when EOI (end of initialization) received from slave
EOI Time sync          :      Y #Send time sync message when first online
EOI General Int        :      Y #Send general interrogation
```

```
# Database definition for this session/sector
```

```
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END
```

```
[IEC-103 Master Session 30 Sector 1]
```

```
Common ASDU Address    :      1 #Range 0 to 65534 Sector address
```

```
#Req init requests when session first online (not req if slave sends
# EOI sequence)
```

```
Online Time sync       :      Y #Send time sync message when first online
Online General Int     :      Y #Send general interrogation
```

```
#Req init requests when EOI (end of initialization) received from slave
EOI Time sync          :      Y #Send time sync message when first online
EOI General Int        :      Y #Send general interrogation
```

```
# Database definition for this session/sector
```

```
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END
```

```
[IEC-103 Master Session 30 Sector 2]
```

```
Common ASDU Address    :      2 #Range 0 to 65534 Sector address
```

```
#Req init requests when session first online (not req if slave sends
# EOI sequence)
```

```
Online Time sync       :      Y #Send time sync message when first online
Online General Int     :      Y #Send general interrogation
```

```
#Req init requests when EOI (end of initialization) received from slave
EOI Time sync          :      Y #Send time sync message when first online
EOI General Int        :      Y #Send general interrogation
```

```
# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

[IEC-103 Master Session 30 Sector 3]
Common ASDU Address   :      3 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync         :      Y #Send time sync message when first online
EOI General Int       :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

[IEC-103 Master Session 30 Sector 4]
Common ASDU Address   :      4 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync      :      Y #Send time sync message when first online
Online General Int    :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync         :      Y #Send time sync message when first online
EOI General Int       :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index      Code       Index
START
END

[IEC-103 Master Session 31]
Communication Port    :      1 #Index of COM port for session (0 or 1)
Sector Count          :      5 #1 is max value for this version of app
Data Link Address     :      31 #Range is 0 to 65535 DL address of slave

Failure Delay         :      10 #Min Sec to delay before poll of offline slave
                        #(0 to 2000 seconds)
Confirm Timeout       :      1000 #0 to 2^32-1 mSec to wait for DL confirm
Retry Count           :      2 #0 to 255 retries for if no confirm
Response Timeout      :      10000 #Timeout for confirm of req (0 to 2^32-1)
```

```

C1/C2 Poll Count Pend :      5 #class 1 or 2 polls before next slave tried (0-
65535)
Class 1 Polls          :      10 #Max class 1 polls to this session
Class 1 Pend delay    :     1000 #Min mSec delay between call (0 to 2^32-1)
Class 2 Pend delay    :     1000 #Min mSec delay between call (0 to 2^32-1)
Class 1 Poll delay    :     1000 #Min mSec delay between call (0 to 2^32-1)
Class 2 Poll delay    :     1000 #Min mSec delay between call (0 to 2^32-1)

```

```
[IEC-103 Master Session 31 Sector 0]
```

```
Common ASDU Address   :      0 #Range 0 to 65534 Sector address
```

```
#Req init requests when session first online (not req if slave sends
# EOI sequence)
```

```
Online Time sync      :      Y #Send time sync message when first online
```

```
Online General Int    :      Y #Send general interrogation
```

```
#Req init requests when EOI (end of initialization) received from slave
```

```
EOI Time sync         :      Y #Send time sync message when first online
```

```
EOI General Int       :      Y #Send general interrogation
```

```
# Database definition for this session/sector
```

```
#
```

```
# ASDU   Database   Function   Point
```

```
# Type   Index      Code       Index
```

```
START
```

```
END
```

```
[IEC-103 Master Session 31 Sector 1]
```

```
Common ASDU Address   :      1 #Range 0 to 65534 Sector address
```

```
#Req init requests when session first online (not req if slave sends
```

```
# EOI sequence)
```

```
Online Time sync      :      Y #Send time sync message when first online
```

```
Online General Int    :      Y #Send general interrogation
```

```
#Req init requests when EOI (end of initialization) received from slave
```

```
EOI Time sync         :      Y #Send time sync message when first online
```

```
EOI General Int       :      Y #Send general interrogation
```

```
# Database definition for this session/sector
```

```
#
```

```
# ASDU   Database   Function   Point
```

```
# Type   Index      Code       Index
```

```
START
```

```
END
```

```
[IEC-103 Master Session 31 Sector 2]
```

```
Common ASDU Address   :      2 #Range 0 to 65534 Sector address
```

```
#Req init requests when session first online (not req if slave sends
```

```
# EOI sequence)
```

```
Online Time sync      :      Y #Send time sync message when first online
```

```
Online General Int    :      Y #Send general interrogation
```

```
#Req init requests when EOI (end of initialization) received from slave
```

```
EOI Time sync      :      Y #Send time sync message when first online
EOI General Int    :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

[IEC-103 Master Session 31 Sector 3]
Common ASDU Address :      3 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync    :      Y #Send time sync message when first online
Online General Int  :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync      :      Y #Send time sync message when first online
EOI General Int    :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

[IEC-103 Master Session 31 Sector 4]
Common ASDU Address :      4 #Range 0 to 65534 Sector address

#Req init requests when session first online (not req if slave sends
# EOI sequence)
Online Time sync    :      Y #Send time sync message when first online
Online General Int  :      Y #Send general interrogation

#Req init requests when EOI (end of initialization) received from slave
EOI Time sync      :      Y #Send time sync message when first online
EOI General Int    :      Y #Send general interrogation

# Database definition for this session/sector
#
# ASDU   Database   Function   Point
# Type   Index     Code       Index
START
END

[IEC-103 Master Commands]
#
# Enable code field is as follows:
#   0 = Disabled
#   1 = Enabled with Poll Interval (seconds) utilized
#   2 = Conditional (executed when point in database changes)
#
```

```

# Database Index is the location in the module's database to use as the source
#       for the data in the command.  The data type field determines
#       the meaning of the index as follows:
#  Type   Description                               DB Index type
#  ----   -
#    6    Clock synchronization                     NA
#    7    General interrogataion                     NA
#   20    General Command                           Bit address
#
# Poll Interval field is utilized when the Enable Code is set to 1.  This field
#       determines the minimum number of seconds to wait between each
#       execution of the command.
#
# Session Index represents the session index in the module to associate with the
#       command.  This index is set when the session is read in from
#       this file.  The range of values for this field is 0 to 31.
#
# Sector Index represents the sector index for the specific session.  The range
#       of values for this field is 0 to 4.
#
# Data type file represents the ASDU type as follows:
#    6 = Time synchronization
#    7 = General Interrogation
#   20 = General Command
#
# Function code
# Code  Definition
# 128  Distance protection
# 160  Overcurrent protection
# 176  Transformer differential protection
# 192  Line Differential protection
# 255  Global function type
#
# Point Index field is used to specify the address in the remote slave device
#       of the point to interact with.
#
# Override Flag field is used for general commands to determine the value
#       to be written.  If the override flag is clear (0), the value
#       in the database will be utilized.  If the override flag is set (1),
#       the value specified in the override value field will be used.
#
# Override Value field is utilized if the override flag is set.  This can be
#       used to always force a control parameter to a fixed value.  The
#       Enable code 2 and the database value for the command can be used
#       to determine when the value should be written.
#
# Enable   DB      Poll  Session  Sector  Data  Func  Point  Ovrld  Ovrld
# Code    Index  Interval  Index   Index  Type  Code  Index  Flag   Val
START
      1   3200      5      0      0    20   160    16    0     0
      0     0      5      0      0     6   255    18    0     0
      0     0      5      0      0     7   128    20    0     0
END

```











## 7.8 Protocol Support

This section contains a listing of the IEC 60870-5-103 protocol support provided by the module.

**Note:** Shaded areas are not supported by the module.

### 7.8.1 List of Type Identification Codes

#### In Monitor Direction

| Type | Description                                |
|------|--|
| 1    | time-tagged message                        |
| 2    | time-tagged message with relative time     |
| 3    | measurands I                               |
| 4    | time-tagged measurands with relative time  |
| 5    | identification                             |
| 6    | time synchronization                       |
| 8    | general interrogation termination          |
| 9    | measurands II                              |
| 10   | generic data                               |
| 11   | generic identification                     |
| 23   | list of recorded disturbances              |
| 26   | ready for transmission of disturbance data |
| 27   | ready for transmission of channel          |
| 28   | ready for transmission of tags             |
| 29   | transmission of tags                       |
| 30   | transmission of disturbance values         |
| 31   | end of transmission                        |

#### In Control Direction

| Type | Description           |
|------|-----------------------|
| 6    | time synchronization  |
| 7    | general interrogation |
| 10   | generic data          |
| 20   | general command       |

| Type | Description                             |
|------|---|
| 21   | generic command                         |
| 24   | order for disturbance data transmission |
| 25   | ack for disturbance data transmission   |

## 7.8.2 List of Cause of Transmission Codes

### In Monitor Direction

| COT | Description                                   |
|-----|---|
| 1   | spontaneous                                   |
| 2   | cyclic  |
| 3   | reset frame count bit (FCB)                   |
| 4   | reset communication unit (CU)                 |
| 5   | start/restart                                 |
| 6   | power on                                      |
| 7   | test mode                                     |
| 8   | time synchronization                          |
| 9   | general interrogation                         |
| 10  | termination of general interrogation          |
| 11  | local operation                               |
| 12  | remote operation                              |
| 20  | positive ack of command                       |
| 21  | negative ack of command                       |
| 31  | transmission of disturbance data              |
| 40  | positive ack of generic write command         |
| 41  | negative ack of generic write command         |
| 42  | valid data response to generic read command   |
| 43  | invalid data response to generic read command |
| 44  | generic write confirmation                    |

### In Control Direction

| COT | Description                         |
|-----|-------------------------------------|
| 8   | time synchronization                |
| 9   | initiation of general interrogation |
| 20  | general command                     |

| COT | Description                      |
|-----|----------------------------------|
| 31  | transmission of disturbance data |
| 40  | generic write command            |
| 42  | generic read command             |

### 7.8.3 List of Function Types

| Fun | Description                         | SYMBOL       |
|-----|-------------------------------------|--------------|
| 128 | Distance protection                 | t(z)         |
| 160 | Overcurrent protection              | I>>          |
| 176 | Transformer differential protection | $\Delta I_T$ |
| 192 | Line differential protection        | $\Delta I_L$ |
| 254 | Generic function type               | GEN          |
| 255 | Global function type                | GLB          |

### 7.8.4 Information Numbers Used In Monitor Direction

#### System functions

| Inf | Description                  | GI | Type | Fun  |     |              |              |     |     |   |
|-----|------------------------------|----|------|------|-----|--------------|--------------|-----|-----|---|
|     |                              |    |      | 128  | 160 | 176          | 192          | 254 | 255 |   |
|     |                              |    |      | t(z) | I>> | $\Delta I_T$ | $\Delta I_L$ | GEN | GLB |   |
| 0   | end of general interrogation |    | 8    |      |     |              |              |     |     | X |
| 0   | time synchronization         |    | 6    |      |     |              |              |     |     | X |
| 2   | reset FCB                    |    | 5    |      |     |              |              |     |     | * |
| 3   | reset CU                     |    | 5    |      |     |              |              |     |     | * |
| 4   | start/restart                |    | 5    |      |     |              |              |     |     | * |
| 5   | power on                     |    | 5    |      |     |              |              |     |     | * |

\* - According to main function

#### Status Indications

| Inf | Description           | GI | Type | Fun  |     |              |              |     |     |  |
|-----|-----------------------|----|------|------|-----|--------------|--------------|-----|-----|--|
|     |                       |    |      | 128  | 160 | 176          | 192          | 254 | 255 |  |
|     |                       |    |      | t(z) | I>> | $\Delta I_T$ | $\Delta I_L$ | GEN | GLB |  |
| 16  | auto-recloser active  | X  | 1    | X    | X   |              | X            |     |     |  |
| 17  | teleprotection active | X  | 1    | X    | X   |              |              |     |     |  |
| 18  | protection active     | X  | 1    | X    | X   | X            | X            |     |     |  |
| 19  | LED reset             |    | 1    | X    | X   | X            | X            |     |     |  |

| Inf | Description               | GI | Type | Fun         |            |                     |                     |            |            |
|-----|---------------------------|----|------|-------------|------------|---------------------|---------------------|------------|------------|
|     |                           |    |      | 128<br>t(z) | 160<br>I>> | 176<br>$\Delta I_T$ | 192<br>$\Delta I_L$ | 254<br>GEN | 255<br>GLB |
| 20  | monitor direction blocked | X  | 1    | X           | X          | X                   | X                   |            |            |
| 21  | test mode                 | X  | 1    | X           | X          | X                   | X                   |            |            |
| 22  | local parameter setting   | X  | 1    | X           | X          | X                   | X                   |            |            |
| 23  | characteristic 1          | X  | 1    | X           |            |                     |                     |            |            |
| 24  | characteristic 2          | X  | 1    | X           |            |                     |                     |            |            |
| 25  | characteristic 3          | X  | 1    | X           |            |                     |                     |            |            |
| 26  | characteristic 4          | X  | 1    | X           |            |                     |                     |            |            |
| 27  | auxiliary input 1         | X  | 1    | X           | X          | X                   | X                   |            |            |
| 28  | auxiliary input 2         | X  | 1    | X           | X          | X                   | X                   |            |            |
| 29  | auxiliary input 3         | X  | 1    | X           | X          | X                   | X                   |            |            |
| 30  | auxiliary input 4         | X  | 1    | X           | X          | X                   | X                   |            |            |

Supervision indications

| Inf | Description                | GI | Type | Fun         |            |                     |                     |            |            |
|-----|----------------------------|----|------|-------------|------------|---------------------|---------------------|------------|------------|
|     |                            |    |      | 128<br>t(z) | 160<br>I>> | 176<br>$\Delta I_T$ | 192<br>$\Delta I_L$ | 254<br>GEN | 255<br>GLB |
| 32  | measured supervision I     | X  | 1    | X           | X          |                     |                     |            |            |
| 33  | measured supervision V     | X  | 1    | X           | X          |                     |                     |            |            |
| 35  | phase sequence supervision | X  | 1    | X           | X          |                     |                     |            |            |
| 36  | trip circuit supervision   | X  | 1    | X           | X          | X                   | X                   |            |            |
| 37  | I>>back-up operation       | X  | 1    | X           |            |                     |                     |            |            |
| 38  | VT fuse failure            | X  | 1    | X           | X          |                     |                     |            |            |
| 39  | teleprotection disturbed   | X  | 1    | X           | X          |                     | X                   |            |            |
| 46  | group warning              | X  | 1    | X           | X          | X                   | X                   |            |            |
| 47  | group alarm                | X  | 1    | X           | X          | X                   | X                   |            |            |

Earth fault indications

| Inf | Description                           | GI | Type | Fun         |            |                     |                     |            |            |
|-----|---------------------------------------|----|------|-------------|------------|---------------------|---------------------|------------|------------|
|     |                                       |    |      | 128<br>t(z) | 160<br>I>> | 176<br>$\Delta I_T$ | 192<br>$\Delta I_L$ | 254<br>GEN | 255<br>GLB |
| 48  | earth fault L <sub>1</sub>            | X  | 1    | X           | X          |                     |                     |            |            |
| 49  | earth fault L <sub>2</sub>            | X  | 1    | X           | X          |                     |                     |            |            |
| 50  | earth fault L <sub>3</sub>            | X  | 1    | X           | X          |                     |                     |            |            |
| 51  | earth fault forward (that is, line)   | X  | 1    | X           | X          |                     |                     |            |            |
| 52  | earth fault reverse (that is, busbar) | X  | 1    | X           | X          |                     |                     |            |            |

Fault indications

| Inf | Description                          | GI | Type | Fun         |            |                     |                     |            |            |
|-----|--------------------------------------|----|------|-------------|------------|---------------------|---------------------|------------|------------|
|     |                                      |    |      | 128<br>t(z) | 160<br>I>> | 176<br>$\Delta I_T$ | 192<br>$\Delta I_L$ | 254<br>GEN | 255<br>GLB |
| 64  | start/pick-up L <sub>1</sub>         | X  | 2    | X           | X          |                     | X                   |            |            |
| 65  | start/pick-up L <sub>2</sub>         | X  | 2    | X           | X          |                     | X                   |            |            |
| 66  | start/pick-up L <sub>3</sub>         | X  | 2    | X           | X          |                     | X                   |            |            |
| 67  | start/pick-up N                      | X  | 2    | X           | X          |                     | X                   |            |            |
| 68  | general trip                         |    | 2    | X           | X          | X                   | X                   |            |            |
| 69  | trip L <sub>1</sub>                  |    | 2    | X           | X          | X                   | X                   |            |            |
| 70  | trip L <sub>2</sub>                  |    | 2    | X           | X          | X                   | X                   |            |            |
| 71  | trip L <sub>3</sub>                  |    | 2    | X           | X          | X                   | X                   |            |            |
| 72  | trip I>> (back-up operation)         |    | 4    | X           |            |                     |                     |            |            |
| 73  | fault location X in ohms             |    | 2    | X           | X          |                     |                     |            |            |
| 74  | fault forward/line                   |    | 2    | X           | X          |                     |                     |            |            |
| 75  | fault reverse/busbar                 |    | 2    | X           | X          |                     |                     |            |            |
| 76  | teleprotection signal transmitted    |    | 2    | X           | X          |                     |                     |            |            |
| 77  | teleprotection signal received       |    | 2    | X           | X          |                     |                     |            |            |
| 78  | zone 1                               |    | 2    | X           |            |                     |                     |            |            |
| 79  | zone 2                               |    | 2    | X           |            |                     |                     |            |            |
| 80  | zone 3                               |    | 2    | X           |            |                     |                     |            |            |
| 81  | zone 4                               |    | 2    | X           |            |                     |                     |            |            |
| 82  | zone 5                               |    | 2    | X           |            |                     |                     |            |            |
| 83  | zone 6                               |    | 2    | X           |            |                     |                     |            |            |
| 84  | general start/pick-up                | X  | 2    | X           | X          | X                   | X                   |            |            |
| 85  | breaker failure                      |    | 2    | X           | X          |                     |                     |            |            |
| 86  | trip measuring system L <sub>1</sub> |    | 2    |             |            | X                   |                     |            |            |
| 87  | trip measuring system L <sub>2</sub> |    | 2    |             |            | X                   |                     |            |            |
| 88  | trip measuring system L <sub>3</sub> |    | 2    |             |            | X                   |                     |            |            |
| 89  | trip measuring system E              |    | 2    |             |            | X                   |                     |            |            |
| 90  | trip I>                              |    | 2    |             | X          |                     |                     |            |            |
| 91  | trip I>>                             |    | 2    |             | X          |                     |                     |            |            |
| 92  | trip IN>                             |    | 2    |             | X          |                     |                     |            |            |
| 93  | trip IN>>                            |    | 2    |             | X          |                     |                     |            |            |

Auto-reclosure indications

| Inf | Description             | GI | Type | Fun         |            |                     |                     |            |            |
|-----|-------------------------|----|------|-------------|------------|---------------------|---------------------|------------|------------|
|     |                         |    |      | 128<br>t(z) | 160<br>I>> | 176<br>$\Delta I_T$ | 192<br>$\Delta I_L$ | 254<br>GEN | 255<br>GLB |
| 128 | CB 'on' by AR           |    | 1    | X           | X          |                     | X                   |            |            |
| 129 | CB 'on' by long-time AR |    | 1    | X           | X          |                     | X                   |            |            |

| Inf | Description | GI | Type | Fun  |     |              |              |     |     |
|-----|-------------|----|------|------|-----|--------------|--------------|-----|-----|
|     |             |    |      | 128  | 160 | 176          | 192          | 254 | 255 |
|     |             |    |      | t(z) | I>> | $\Delta I_T$ | $\Delta I_L$ | GEN | GLB |
| 130 | AR blocked  | X  | 1    | X    | X   |              | X            |     |     |

Measurands

| Inf | Description                                      | GI | Type | Fun  |     |              |              |     |     |
|-----|--|----|------|------|-----|--------------|--------------|-----|-----|
|     |  |    |      | 128  | 160 | 176          | 192          | 254 | 255 |
|     |  |    |      | t(z) | I>> | $\Delta I_T$ | $\Delta I_L$ | GEN | GLB |
| 144 | measurand I                                      |    | 3.1  | X    | X   |              |              |     |     |
| 145 | measurands I, V                                  |    | 3.2  | X    | X   |              |              |     |     |
| 146 | measurands I, V, P, Q                            |    | 3.3  | X    |     |              |              |     |     |
| 147 | measurands $I_N$ , $V_{EN}$                      |    | 3.4  | X    | X   |              |              |     |     |
| 148 | measurands $I_{L1,2,3}$ , $V_{L1,2,3}$ , P, Q, f |    | 9    | X    |     |              |              |     |     |

Generic functions

| Inf | Description   | GI  | Type | Fun  |     |              |              |     |     |
|-----|---|-----|------|------|-----|--------------|--------------|-----|-----|
|     |   |     |      | 128  | 160 | 176          | 192          | 254 | 255 |
|     |   |     |      | t(z) | I>> | $\Delta I_T$ | $\Delta I_L$ | GEN | GLB |
| 240 | read headings of all defined groups                   |     | 10   |      |     |              |              | X   |     |
| 241 | read values or attributes of all entries of one group |     | 10   |      |     |              |              | X   |     |
| 243 | read directory of a single entry                      |     | 11   |      |     |              |              | X   |     |
| 244 | read value or attribute of a single entry             | (x) | 10   |      |     |              |              | X   |     |
| 245 | end of general interrogation of generic data          |     | 10   |      |     |              |              | X   |     |
| 249 | write entry with confirmation                         |     | 10   |      |     |              |              | X   |     |
| 250 | write entry with execution                            |     | 10   |      |     |              |              | X   |     |
| 251 | write entry aborted                                   |     | 10   |      |     |              |              | X   |     |

**7.8.5 Information Numbers Used In Control Direction**System functions

| Inf | Description                         | Type | Fun  |     |              |              |     |     |
|-----|-------------------------------------|------|------|-----|--------------|--------------|-----|-----|
|     |                                     |      | 128  | 160 | 176          | 192          | 254 | 255 |
|     |                                     |      | t(z) | I>> | $\Delta I_T$ | $\Delta I_L$ | GEN | GLB |
| 0   | initiation of general interrogation | 7    |      |     |              |              |     | X   |
| 0   | time synchronization                | 6    |      |     |              |              |     | X   |

General commands

| Inf | Description               | Com    | Type | Fun         |            |                     |                     |            |            |
|-----|---------------------------|--------|------|-------------|------------|---------------------|---------------------|------------|------------|
|     |                           |        |      | 128<br>t(z) | 160<br>I>> | 176<br>$\Delta I_T$ | 192<br>$\Delta I_L$ | 254<br>GEN | 255<br>GLB |
| 16  | auto-recloser on/off      | On/Off | 20   | X           | X          |                     | X                   |            |            |
| 17  | teleprotection on/off     | On/Off | 20   | X           | X          |                     |                     |            |            |
| 18  | protection on/off         | On/Off | 20   | X           | X          | X                   | X                   |            |            |
| 19  | LED reset                 | On     | 20   | X           | X          | X                   | X                   |            |            |
| 23  | activate characteristic 1 | On     | 20   | X           |            |                     |                     |            |            |
| 24  | activate characteristic 2 | On     | 20   | X           |            |                     |                     |            |            |
| 25  | activate characteristic 3 | On     | 20   | X           |            |                     |                     |            |            |
| 26  | activate characteristic 4 | On     | 20   | X           |            |                     |                     |            |            |

Generic functions

| Inf | Description   | Type | Fun         |            |                     |                     |            |            |  |
|-----|---|------|-------------|------------|---------------------|---------------------|------------|------------|--|
|     |   |      | 128<br>t(z) | 160<br>I>> | 176<br>$\Delta I_T$ | 192<br>$\Delta I_L$ | 254<br>GEN | 255<br>GLB |  |
| 240 | read headings of all defined groups                   | 21   |             |            |                     |                     |            | X          |  |
| 241 | read values or attributes of all entries of one group | 21   |             |            |                     |                     |            | X          |  |
| 243 | read directory of a single entry                      | 21   |             |            |                     |                     |            | X          |  |
| 244 | read value or attribute of a single entry             | 21   |             |            |                     |                     |            | X          |  |
| 245 | general interrogation of generic data                 | 21   |             |            |                     |                     |            | X          |  |
| 248 | write entry   | 10   |             |            |                     |                     |            | X          |  |
| 249 | write entry with confirmation                         | 10   |             |            |                     |                     |            | X          |  |
| 250 | write entry with execution                            | 10   |             |            |                     |                     |            | X          |  |
| 251 | write entry abort                                     | 10   |             |            |                     |                     |            | X          |  |

**7.8.6 Definition and Presentation of ASDU'S In Monitor Direction**

Type 1 - Time-tagged

| Bytes | Description            | Spec |
|-------|------------------------|------|
| 1     | Value of 1             |      |
| 1     | 0x81                   |      |
| 1     | COT                    |      |
| 1     | Common address of ASDU |      |
| 1     | Function Type          |      |



| Bytes | Description         | Spec     |
|-------|---------------------|----------|
| 1     | Information number  |          |
| 1     | DPI (0 to 3)        | 7.2.6.5  |
| 4     | 4-octet binary time | 7.2.6.28 |
| 1     | SIN                 | 7.2.6.23 |

*Type 2 - Time-tagged with relative time*

| Bytes | Description            | Spec     |
|-------|------------------------|----------|
| 1     | Value of 2             |          |
| 1     | 0x81                   |          |
| 1     | COT                    |          |
| 1     | Common address of ASDU |          |
| 1     | Function Type          |          |
| 1     | Information number     |          |
| 1     | DPI (0 to 3)           | 7.2.6.5  |
| 2     | RET (relative time)    | 7.2.6.15 |
| 2     | FAN (fault number)     | 7.2.6.6  |
| 4     | 4-octet binary time    | 7.2.6.28 |
| 1     | SIN                    | 7.2.6.23 |

*Type 3 - Measurands I*

| Bytes | Description  | Spec    |
|-------|--|---------|
| 1     | Value of 3   |         |
| 1     | i = value of 1, 2 or 4                                     |         |
| 1     | COT  |         |
| 1     | Common address of ASDU                                     |         |
| 1     | Function Type  |         |
| 1     | Information number   |         |
| 2     | Current L <sub>2</sub>                                     | 7.2.6.8 |
| 2     | Voltage L <sub>1</sub> -L <sub>2</sub>                     | 7.2.6.8 |
| 2     | Active power P   | 7.2.6.8 |
| 2     | Reactive power Q   | 7.2.6.8 |
|       | ASD U3.1: i=1  |         |
|       | ASDU 3.2: i=2  |         |
|       | ASDU 3.3: i=4  |         |
|       | ASDU 3.4: i=2 val1=I <sub>N</sub> and val2=V <sub>EN</sub> |         |

*Type 4 - Time-tagged measurands with relative time*

| Bytes | Description | Spec |
|-------|-------------|------|
| 1     | Value of 4  |      |

| Bytes | Description                         | Spec     |
|-------|-------------------------------------|----------|
| 1     | 0x81                                |          |
| 1     | COT                                 |          |
| 1     | Common address of ASDU              |          |
| 1     | Function Type                       |          |
| 1     | Information number                  |          |
| 4     | SCL (short-circuit location (real)) | 7.2.6.20 |
| 2     | RET (relative time)                 | 7.2.6.15 |
| 2     | FAN (fault number)                  | 7.2.6.6  |
| 4     | 4-octet binary time                 | 7.2.6.28 |

Type 5 - Identification

| Bytes | Description                      | Spec            |
|-------|----------------------------------|-----------------|
| 1     | Value of 5                       |                 |
| 1     | 0x81                             |                 |
| 1     | COT                              |                 |
| 1     | Common address of ASDU           |                 |
| 1     | Function Type                    |                 |
| 1     | Information number               |                 |
| 1     | COL (compatibility level 2 or 3) | 7.2.6.3         |
| 1     | Char 1                           | 7.2.6.2         |
| 1     | Char 2                           | 7.2.6.2         |
| 1     | Char 3                           | 7.2.6.2         |
| 1     | Char 4                           | 7.2.6.2         |
| 1     | Char 5                           | 7.2.6.2         |
| 1     | Char 6                           | 7.2.6.2         |
| 1     | Char 7                           | 7.2.6.2         |
| 1     | Char 8                           | 7.2.6.2         |
| 1     | Manufacture byte                 | Free assignment |
| 1     | Manufacture byte                 | Free assignment |
| 1     | Manufacture byte                 | Free assignment |
| 1     | Manufacture byte                 | Free assignment |

Type 6 - Time synchronization

| Bytes | Description            | Spec |
|-------|------------------------|------|
| 1     | Value of 6             |      |
| 1     | 0x81                   |      |
| 1     | COT                    |      |
| 1     | Common address of ASDU |      |
| 1     | Function Type          |      |
| 1     | Information number     |      |

| Bytes | Description         | Spec     |
|-------|---------------------|----------|
| 7     | 7-Octet binary time | 7.2.6.29 |

Type 8 - Termination of general interrogation

| Bytes | Description             | Spec     |
|-------|-------------------------|----------|
| 1     | Value of 8              |          |
| 1     | 0x81                    |          |
| 1     | COT                     |          |
| 1     | Common address of ASDU  |          |
| 1     | Function Type           |          |
| 1     | Information number      |          |
| 1     | SCN (scan number 0-255) | 7.2.6.21 |

Type 9 - Measurands II

| Bytes | Description                     | Spec    |
|-------|---------------------------------|---------|
| 1     | Value of 9                      |         |
| 1     | I = 1 to 9 for number of values |         |
| 1     | COT                             |         |
| 1     | Common address of ASDU          |         |
| 1     | Function Type                   |         |
| 1     | Information number              |         |
| 2     | Current L <sub>1</sub>          | 7.2.6.8 |
| 2     | Current L <sub>2</sub>          | 7.2.6.8 |
| 2     | Current L <sub>3</sub>          | 7.2.6.8 |
| 2     | Voltage L <sub>1-E</sub>        | 7.2.6.8 |
| 2     | Voltage L <sub>2-E</sub>        | 7.2.6.8 |
| 2     | Voltage L <sub>3-E</sub>        | 7.2.6.8 |
| 2     | Active power P                  | 7.2.6.8 |
| 2     | Reactive power Q                | 7.2.6.8 |
| 2     | Frequency f                     | 7.2.6.8 |

Value of I determines number of parameters passed in the message with a range of 1 to 9 but always starts with the L<sub>1</sub> value.

### 7.8.7 Definition and Presentation Of ASDU'S In Control Direction

Type 6 - Time synchronization

| Bytes | Description | Spec |
|-------|-------------|------|
| 1     | Value of 6  |      |
| 1     | 0x81        |      |
| 1     | COT         |      |

| Bytes | Description               | Spec     |
|-------|---------------------------|----------|
| 1     | Common address of ASDU    |          |
| 1     | Function Type = GLB (255) |          |
| 1     | Information number        |          |
| 7     | 7-Octet binary time       | 7.2.6.29 |

Type 7 - Initiation of general interrogation

| Bytes | Description               | Spec     |
|-------|---------------------------|----------|
| 1     | Value of 7                |          |
| 1     | 0x81                      |          |
| 1     | COT                       |          |
| 1     | Common address of ASDU    |          |
| 1     | Function Type = GLB (255) |          |
| 1     | Information number        |          |
| 1     | SCN (scan number 0-255)   | 7.2.6.21 |

Type 20 - General command

| Bytes | Description            | Spec     |
|-------|------------------------|----------|
| 1     | Value of 20            |          |
| 1     | 0x81                   |          |
| 1     | COT                    |          |
| 1     | Common address of ASDU |          |
| 1     | Function Type          |          |
| 1     | Information number     |          |
| 1     | DCO (1=Off, 2=On)      | 7.2.6.4  |
| 1     | RII                    | 7.2.6.19 |

RII is not to be processed within the protection equipment, but to be used as a SIN of the return message.

## 7.9 Protocol Interoperability Documentation

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 stemming 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)
- R** Function or ASDU is used in reverse mode
- B** Function or ASDU is used in standard and reverse mode

The possible selection (blank, X, R, B) is specified for each specific clause or parameter.

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

The pages in this section have been extracted from the 60870-5-103 © IEC:1997, pages 159 to 171.

## INTEROPERABILITY

### 7.9.1 *Physical Layer*

### 7.9.2 *Electrical Interface*

- EIA RS-485
- Number of loads 32. For one protection equipment

**NOTE** - EIA RS-485 standard defines unit loads so that 32 of them can be operated on one line. For detailed information refer to clause 3 of EIA RS-485 standard.

### 7.9.3 *Optical Interface*

- Glass fiber
- Plastic fiber
- F-SMA type connector
- BFOC/2,5 type connector

#### **7.9.4            *Transmission speed***

- 9 600 bit/s
- 19 200 bit/s

#### **7.9.5            *Link Layer***

There are no choices for the link layer.

#### **7.9.6            *Application Layer***

#### **7.9.7            *Transmission mode for application data***

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

#### **7.9.8            *Common address of ASDU***

- One Common Address of ASDU (identical with station address)
- More than one Common Address of ASDU

#### **7.9.9            *Selection of standard information numbers in monitor direction***

#### **7.9.10          *System functions in monitor direction***

- | <b>INF</b>                              | <b>Semantics</b>             |
|---|------------------------------|
| <input checked="" type="checkbox"/> <0> | End of general interrogation |
| <input checked="" type="checkbox"/> <0> | Time synchronization         |
| <input checked="" type="checkbox"/> <2> | Reset FCB                    |
| <input checked="" type="checkbox"/> <3> | Reset CU                     |
| <input checked="" type="checkbox"/> <4> | Start/restart                |
| <input checked="" type="checkbox"/> <5> | Power on                     |

#### **7.9.11          *Status indications in monitor direction***

- | <b>INF</b>                               | <b>Semantics</b>     |
|--|----------------------|
| <input checked="" type="checkbox"/> <16> | Auto-recloser active |

- 
- ☒ <17> Teleprotection active
  - ☒ <18> Protection active
  - ☒ <19> LED reset
  - ☒ <20> Monitor direction blocked
  - ☒ <21> Test mode
  - ☒ <22> Local parameter setting
  - ☒ <23> Characteristic 1
  - ☒ <24> Characteristic 2
  - ☒ <25> Characteristic 3
  - ☒ <26> Characteristic 4
  - ☒ <27> Auxiliary input 1
  - ☒ <28> Auxiliary input 2
  - ☒ <29> Auxiliary input 3
  - ☒ <30> Auxiliary input 4

### **7.9.12**                    ***Supervision indications in monitor direction***

#### **INF**    **Semantics**

- ☒ <32> Measurand supervision I
- ☒ <33> Measurand supervision V
- ☒ <35> Phase sequence supervision
- ☒ <36> Trip circuit supervision
- ☒ <37> I>> back-up operation
- ☒ <38> VT fuse failure
- ☒ <39> Teleprotection disturbed
- ☒ <46> Group warning
- ☒ <47> Group alarm

---

### **7.9.13**            ***Earth fault indications in monitor direction***

| <b>INF</b>                          | <b>Semantics</b>                          |
|-------------------------------------|---|
| <input checked="" type="checkbox"/> | <48> Earth fault L <sub>1</sub>           |
| <input checked="" type="checkbox"/> | <49> Earth fault L <sub>2</sub>           |
| <input checked="" type="checkbox"/> | <50> Earth fault L <sub>3</sub>           |
| <input checked="" type="checkbox"/> | <51> Earth fault forward, that is, line   |
| <input checked="" type="checkbox"/> | <52> Earth fault reverse, that is, busbar |

### **7.9.14**            ***Fault indications in monitor direction***

| <b>INF</b>                          | <b>Semantics</b>                       |
|-------------------------------------|--|
| <input checked="" type="checkbox"/> | <64> Start /pick-up L <sub>1</sub>     |
| <input checked="" type="checkbox"/> | <65> Start /pick-up L <sub>2</sub>     |
| <input checked="" type="checkbox"/> | <66> Start /pick-up L <sub>3</sub>     |
| <input checked="" type="checkbox"/> | <67> Start /pick-up N                  |
| <input checked="" type="checkbox"/> | <68> General trip                      |
| <input checked="" type="checkbox"/> | <69> Trip L <sub>1</sub>               |
| <input checked="" type="checkbox"/> | <70> Trip L <sub>2</sub>               |
| <input checked="" type="checkbox"/> | <71> Trip L <sub>3</sub>               |
| <input checked="" type="checkbox"/> | <72> Trip I>> (back-up operation)      |
| <input checked="" type="checkbox"/> | <73> Fault location X in ohms          |
| <input checked="" type="checkbox"/> | <74> Fault forward/line                |
| <input checked="" type="checkbox"/> | <75> Fault reverse/busbar              |
| <input checked="" type="checkbox"/> | <76> Teleprotection signal transmitted |
| <input checked="" type="checkbox"/> | <77> Teleprotection signal received    |
| <input checked="" type="checkbox"/> | <78> Zone 1                            |
| <input checked="" type="checkbox"/> | <79> Zone 2                            |
| <input checked="" type="checkbox"/> | <80> Zone 3                            |
| <input checked="" type="checkbox"/> | <81> Zone 4                            |



- ☒ <82> Zone 5
- ☒ <83> Zone 6
- ☒ <84> General start/pick-up
- ☒ <85> Breaker failure
- ☒ <86> Trip measuring system L<sub>1</sub>
- ☒ <87> Trip measuring system L<sub>2</sub>
- ☒ <88> Trip measuring system L<sub>3</sub>
- ☒ <89> Trip measuring system E
- ☒ <90> Trip I>
- ☒ <91> Trip I>>
- ☒ <92> Trip IN>
- ☒ <93> Trip IN>>

### **7.9.15**            *Auto-reclosure indications in monitor direction*

#### **INF**    **Semantics**

- ☒ <128> CB 'on' by AR
- ☒ <129> CB 'on' by long-time AR
- ☒ <130> AR blocked

### **7.9.16**            *Measurands in monitor direction*

#### **INF**    **Semantics**

- ☒ <144> Measurand I
- ☒ <145> Measurands I, V
- ☒ <146> Measurands I, V, P, Q
- ☒ <147> Measurands I<sub>N</sub>, V<sub>EN</sub>
- ☒ <148> Measurands I<sub>L1,2,3</sub>, V<sub>L1,2,3</sub>, P, Q, f

### **7.9.17**            *Generic functions in monitor direction*

#### **INF**    **Semantics**

- <240> Read headings of all defined groups
- <241> Read values or attributes of all entries of one group
- <243> Read directory of a single entry
- <244> Read value or attribute of a single entry
- <245> End of general interrogation of generic data
- <249> Write entry with confirmation
- <250> Write entry with execution
- <251> Write entry aborted

**7.9.18**                    ***Selection of standard information numbers in control direction***

**7.9.19**                    ***System functions in control direction***

**INF**    **Semantics**

- <0>    Initiation of general interrogation
- <0>    Time synchronization

**7.9.20**                    ***General commands in control direction***

**INF**    **Semantics**

- <16>    Auto-recloser on/off
- <17>    Teleprotection on/off
- <18>    Protection on/off
- <19>    LED reset
- <23>    Activate characteristic 1
- <24>    Activate characteristic 2
- <25>    Activate characteristic 3
- <26>    Activate characteristic 4

**7.9.21**                    ***Generic functions in control direction***

**INF**    **Semantics**

- <240> Read headings of all defined groups
- <241> Read values or attributes of all entries in one group
- <243> Read directory of a single entry
- <244> Read value or attribute of a single entry
- <245> General interrogation of generic data
- <248> Write entry
- <249> Write entry with confirmation
- <250> Write entry with execution
- <251> Write entry abort

### 7.9.22 **Basic application functions**

- Test mode
- Blocking of monitor direction
- Disturbance data
- Generic services
- Private data (*if ASDU type is supported by module*)

### 7.9.23 **Miscellaneous**

Measurands are transmitted with ASDU 3 as well as with ASDU 9. As defined in 7.2.6.8, the maximum MVAL can either be 1,2 or 2,4 times the rated value. No different rating shall be used in ASDU 3 and ASDU 9, that is, for each measurand there is only one choice.

#### **Measurand    Max. MVAL = rated value times**

|                          | <b>1,2</b>                          | <b>or</b> | <b>2,4</b>                          |
|--------------------------|-------------------------------------|-----------|-------------------------------------|
| Current L <sub>1</sub>   | <input checked="" type="checkbox"/> |           | <input checked="" type="checkbox"/> |
| Current L <sub>2</sub>   | <input checked="" type="checkbox"/> |           | <input checked="" type="checkbox"/> |
| Current L <sub>3</sub>   | <input checked="" type="checkbox"/> |           | <input checked="" type="checkbox"/> |
| Voltage L <sub>1-E</sub> | <input checked="" type="checkbox"/> |           | <input checked="" type="checkbox"/> |
| Voltage L <sub>2-E</sub> | <input checked="" type="checkbox"/> |           | <input checked="" type="checkbox"/> |
| Voltage L <sub>3-E</sub> | <input checked="" type="checkbox"/> |           | <input checked="" type="checkbox"/> |

|   |                                     |                                     |
|---|-------------------------------------|-------------------------------------|
| Active power P                          | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| Reactive power Q                        | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| Frequency f                             | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| Voltage L <sub>1</sub> - L <sub>2</sub> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |

---

## Support, Service & Warranty

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

---

|                    |   |
|--------------------|---|
| <b>Internet</b>    | Web Site: <a href="http://www.prosoft-technology.com/support">http://www.prosoft-technology.com/support</a><br>E-mail address: <a href="mailto:support@prosoft-technology.com">support@prosoft-technology.com</a> |
| <b>Phone</b>       | (661) 716-5100<br>(661) 716-5101 (Fax)  |
| <b>Postal Mail</b> | ProSoft Technology, Inc.<br>1675 Chester Avenue, Fourth Floor<br>Bakersfield, CA 93301  |

---

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

- 1 Product Version Number
- 2 System architecture
- 3 Module configuration and contents of IEC103M.CFG file
- 4 Module Operation
  - Configuration/Debug status information
  - LED patterns
- 5 Information about the processor and user data files as viewed through RSLogix 500 and LED patterns on the processor
- 6 Details about the serial devices interfaced

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

### Module Service and Repair

The MVI46-103M device is an electronic product, designed and manufactured to function under somewhat adverse conditions. As with any product, through age, misapplication, or any one of many possible problems the device may require repair.

When purchased from ProSoft Technology, Inc., the device has a 1 year parts and labor warranty (3 years for RadioLinx) according to the limits specified in the

warranty. Replacement and/or returns should be directed to the distributor from whom the product was purchased. If you must return the device for repair, obtain an RMA (Returned Material Authorization) number from ProSoft Technology, Inc. Please call the factory for this number, and print the number prominently on the outside of the shipping carton used to return the device.

## General Warranty Policy – Terms and Conditions

ProSoft Technology, Inc. (hereinafter referred to as ProSoft) warrants that the Product shall conform to and perform in accordance with published technical specifications and the accompanying written materials, and shall be free of defects in materials and workmanship, for the period of time herein indicated, such warranty period commencing upon receipt of the Product. Limited warranty service may be obtained by delivering the Product to ProSoft in accordance with our **product return procedures** (on page 183) and providing proof of purchase and receipt date. Customer agrees to insure the Product or assume the risk of loss or damage in transit, to prepay shipping charges to ProSoft, and to use the original shipping container or equivalent. Contact ProSoft Customer Service for further information.

This warranty is limited to the repair and/or replacement, at ProSoft's election, of defective or non-conforming Product, and ProSoft shall not be responsible for the failure of the Product to perform specified functions, or any other non-conformance caused by or attributable to: (a) any misuse, misapplication, accidental damage, abnormal or unusually heavy use, neglect, abuse, alteration (b) failure of Customer to adhere to ProSoft's specifications or instructions, (c) any associated or complementary equipment, software, or user-created programming including, but not limited to, programs developed with any EC1131-3 programming languages, "C" for example, and not furnished by ProSoft, (d) improper installation, unauthorized repair or modification (e) improper testing, or causes external to the product such as, but not limited to, excessive heat or humidity, power failure, power surges or natural disaster, compatibility with other hardware and software products introduced after the time of purchase, or products or accessories not manufactured by ProSoft; all of which components, software and products are provided as-is. In no event will ProSoft be held liable for any direct or indirect, incidental consequential damage, loss of data, or other malady arising from the purchase or use of ProSoft products.

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

ProSoft warrants its products to be free from defects in material and workmanship and shall conform to and perform in accordance with published technical specifications and the accompanying written materials for up to one year (12 months) from the date of original purchase (3 years for RadioLinx

products) from ProSoft. If you need to return the device for repair, obtain an RMA (Returned Material Authorization) number from ProSoft Technology, Inc. in accordance with the RMA instructions below. Please call the factory for this number, and print the number prominently on the outside of the shipping carton used to return the device.

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

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

### **Limitation of Liability**

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

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

### **RMA Procedures**

In the event that repairs are required for any reason, contact ProSoft Technical Support at +1 661.716.5100. A Technical Support Engineer will ask you to

perform several tests in an attempt to diagnose the problem. Simply calling and asking for a RMA without following our diagnostic instructions or suggestions will lead to the return request being denied. If, after these tests are completed, the module is found to be defective, we will provide the necessary RMA number with instructions on returning the module for repair.



## Index

### A

Addressing the Data Types • 19  
Analyzing Data for Port 1 • 64  
Analyzing Data for Port 2 • 64  
Application Layer • 174  
Auto-reclosure indications • 166  
Auto-reclosure indications in monitor direction • 177

### B

Backplane Data Transfer • 17  
Basic application functions • 179  
Block Format for Read • 23

### C

Cable Connections • 73  
Clearing a Fault Condition • 70  
Cold Boot Block (9999) • 26  
Command Control Block (9902) • 22  
Command Control Blocks • 20  
Command List Error Data Block (9950) • 24  
Command List Form • 160  
Common address of ASDU • 174  
Configuration File • 31  
Control Routine (U4) • 43

### D

Data Analyzer • 63  
Data Transfer (U3) • 42  
Data Types and Mapping • 17  
Database Form • 158  
Database View Menu • 54  
Definition and Presentation Of ASDU'S In Control Direction • 171  
Definition and Presentation of ASDU'S In Monitor Direction • 168  
Diagnostics and Troubleshooting • 47  
Displaying the Next Page • 61, 62, 68  
Displaying the Previous Page • 61, 62, 68  
Displaying Timing Marks in the Data Analyzer • 64

### E

Earth fault indications • 165  
Earth fault indications in monitor direction • 176  
Electrical Interface • 173  
Event Message Block (9903) • 22  
Exiting the Program • 53

### F

Fault indications • 166  
Fault indications in monitor direction • 176  
Functional Overview • 15

### G

General commands • 168  
General commands in control direction • 178  
General Concepts • 15  
General Specifications • 7  
General Warranty Policy – Terms and Conditions • 182  
Generic functions • 167, 168  
Generic functions in control direction • 178  
Generic functions in monitor direction • 177

### H

Hardware Specifications • 8

### I

IEC-103 Master Driver Menu • 51, 56  
IEC-870-Master Command List Menu • 57, 60  
In Control Direction • 162, 163  
In Monitor Direction • 162, 163  
Information Numbers Used In Control Direction • 167  
Information Numbers Used In Monitor Direction • 164  
Installing and Configuring the Module • 29

### K

Keystrokes • 48

### L

Ladder Logic • 41  
LED Status Indicators • 69  
Limitation of Liability • 183  
Link Layer • 174  
List of Cause of Transmission Codes • 163  
List of Function Types • 164  
List of Type Identification Codes • 162

Listing ASDU point counts • 69

## M

Main Logic Loop • 16  
Main Menu • 50  
Main Routine • 41  
Master Driver • 27  
Measurands • 167  
Measurands in monitor direction • 177  
Miscellaneous • 179  
Module Configuration • 29  
Module Data • 31  
Module Power Up • 15  
Module Service and Repair • 181  
Moving Back Through 5 Pages of Registers •  
54, 60  
MVI46 IEC 60870-5-103 Master  
Communication Module Configuration • 82  
MVI46-103M Configuration Data Definition •  
32, 82  
MVI46-103M Configuration File Example •  
86  
MVI46-103M Status Data Area • 20, 78  
MVI46-IEC 60870-5-103 Master  
Communication Module Error Codes • 80  
MVI46-IEC 60870-5-103 Master  
Communication Module Error/Status Data  
Format • 78

## N

Navigation • 47

## O

Online State • 66  
Opening the Data Analyzer Menu • 56  
Opening the Database Menu • 51  
Opening the IEC-103 Master Menu • 51  
Opening the IEC-870-Master Command List  
Menu • 57  
Opening the Port Configuration Menu • 58  
Opening the Port Status Menu • 58  
Opening the Sector Database Menu • 67  
Opening the Sector Menu • 59  
Opening the Session Configuration Menu •  
58  
Optical Interface • 173

## P

Physical • 7  
Physical Layer • 173  
Please Read This Notice • 2  
Port Configuration Menu • 61  
Port Status Menu • 62  
Product Specifications • 7

Protocol Interoperability Documentation •  
172  
Protocol Support • 162

## Q

Quick Start • 9

## R

Receiving the Configuration File • 52  
Redisplaying the Current Page • 60, 61, 62,  
68, 69  
Redisplaying the Menu • 60, 61, 62, 67, 69  
Reference • 73  
Removing Timing Marks in the Data  
Analyzer • 64  
Required Hardware • 48  
Required Software • 49  
Returning to the Main Menu • 55, 59, 61, 62,  
65, 68, 69  
RMA Procedures • 182, 183  
RS-232 • 74  
RS-232 -- Modem Connection • 74  
RS-232 -- Null Modem Connection  
(Hardware Handshaking) • 75  
RS-232 -- Null Modem Connection (No  
Hardware Handshaking) • 75  
RS-232 Configuration/Debug Port • 76  
RS-422 • 77  
RS-485 • 76  
RS-485 and RS-422 Tip • 77

## S

Sector Configuration Menu • 67  
Sector Database Menu • 68  
Selection of standard information numbers in  
control direction • 178  
Selection of standard information numbers in  
monitor direction • 174  
Sending the Configuration File • 52  
Session Configuration Menu • 59, 66  
Session State • 66  
Set Module Time Block (9971) • 25  
Set SLC Time Block (9970) • 25  
Setting Jumpers • 77  
Skipping 500 Registers of Data • 55, 60  
SLC Interface • 8  
SLC Processor Not in Run • 16  
Starting the Data Analyzer • 65  
Status Indications • 164  
Status indications in monitor direction • 174  
Stopping the Data Analyzer • 65  
Supervision indications • 165  
Supervision indications in monitor direction •  
175  
Support, Service & Warranty • 181

System functions • 164, 167  
System functions in control direction • 178  
System functions in monitor direction • 174

## T

The Configuration/Debug Menu • 35, 47  
Transferring the Configuration File to the Module • 14, 37  
Transferring the Configuration File to Your PC • 35  
Transmission mode for application data • 174  
Transmission speed • 174  
Troubleshooting • 70  
Type 1 - Time-tagged • 168  
Type 2 - Time-tagged with relative time • 169  
Type 20 - General command • 172  
Type 3 - Measurands I • 169  
Type 4 - Time-tagged measurands with relative time • 169  
Type 5 - Identification • 170  
Type 6 - Time synchronization • 170, 171  
Type 7 - Initiation of general interrogation • 172  
Type 8 - Termination of general interrogation • 171  
Type 9 - Measurands II • 171

## U

Uploading and Downloading the Configuration File • 35, 52  
User Constructed Command Block (9901) • 20  
Using the Configuration/Debug Port • 49

## V

Viewing ASDU n Data • 69  
Viewing Block Transfer Statistics • 51  
Viewing Data in ASCII (Text) Format • 55, 64  
Viewing Data in Decimal Format • 55  
Viewing Data in Floating Point Format • 55  
Viewing Data in Hexadecimal Format • 55, 64  
Viewing Database Pages 0 to 3000 • 54  
Viewing General Configuration • 57  
Viewing Master Driver Version Information • 59  
Viewing Module Configuration • 51  
Viewing the Backplane Command List • 52  
Viewing the Next 100 Registers of Data • 55, 60  
Viewing the Previous 100 Registers of Data • 54, 60  
Viewing Version Information • 52

## W

Warm Boot Block (9998) • 26  
Warm Booting the Module • 53

## Y

Your Feedback Please • 2