

## XYR 300G Wireless Gateway

### User's Manual

34-XY-25-24

Version 1

5/1/2008



# Notices and Trademarks

**Copyright 2008 by Honeywell International Inc.  
Version 1 May 1, 2008**

While this information is presented in good faith and believed to be accurate, Honeywell disclaims the implied warranties of merchantability and fitness for a particular purpose and makes no express warranties except as may be stated in its written agreement with and for its customers.

In no event is Honeywell liable to anyone for any indirect, special or consequential damages. The information and specifications in this document are subject to change without notice.

Honeywell, PlantScape, Experion PKS, and **TotalPlant** are registered trademarks of Honeywell International Inc.

Other brand or product names are trademarks of their respective owners.

Honeywell International  
Process Solutions  
2500 West Union Hills  
Phoenix, AZ 85027  
**1-800 343-0228**

# About This Document

This document describes preparation, operation and maintenance of the XYR 300G Wireless I/O extensions. Mounting, installation and wiring are covered in other documents.

Honeywell does not recommend using devices for critical control where there is a single point of failure or where single points of failure result in unsafe conditions. XYR3000 is targeted at open loop control, supervisory control, and controls that do not have environmental or safety consequences. As with any process control solution, the end-user must weigh the risks and benefits to determine if the products used are the right match for the application based on security, safety, and performance. Additionally, it is up to the end-user to ensure that the control strategy sheds to a safe operating condition if any crucial segment of the control solution fails.

## Revision Information

Document Name	Document ID	Version Number	Publication Date
XYR 300G Wireless Gateway	34-XY-25-24	1	5/1/2008

## References

The following list identifies all documents that may be sources of reference for material discussed in this publication.

Document Title
Getting Started with Honeywell OneWireless solutions
XYR3000 MSG
How to build an XYR3000 system

## Terminology

The following acronyms are used in the document.

Acronyms	Model numbers
XYR300X	XYR3001, XYR3002
XYR301X	XYR3011, XYR3012, XYR3013
XYR300G-X	XYR300G-M, XYR300G-E

## **Support and Contact Information**

### **United States and Canada**

Contact: Honeywell Solution Support Center  
Phone: 1-800 822-7673. In Arizona: 602- 313-5558  
Calls are answered by dispatcher between 6:00 am and 4:00 pm Mountain Standard Time. Emergency calls outside normal working hours are received by an answering service and returned within one hour.  
Facsimile: (602) 313-3293  
Mail: Honeywell TAC, MS P13  
2500 West Union Hills Drive  
Phoenix, AZ, 85027

### **World Wide Web**

Honeywell Solution Support Online:  
<http://www.honeywell.com/ps>

### **Elsewhere**











Call your nearest Honeywell office.

### **Training Classes**

Honeywell Automation College:  
<http://www.automationcollege.com>

## Symbol Definitions

The following table lists those symbols used in this document to denote certain conditions.

Symbol	Definition
	<b>ATTENTION:</b> Identifies information that requires special consideration.
	<b>TIP:</b> Identifies advice or hints for the user, often in terms of performing a task.
<b>CAUTION</b>	Indicates a situation which, if not avoided, may result in equipment or work (data) on the system being damaged or lost, or may result in the inability to properly operate the process.
	<p><b>CAUTION:</b> Indicates a potentially hazardous situation which, if not avoided, may result in minor or moderate injury. It may also be used to alert against unsafe practices.</p> <p><b>CAUTION</b> symbol on the equipment refers the user to the product manual for additional information. The symbol appears next to required information in the manual.</p>
	<p><b>WARNING:</b> Indicates a potentially hazardous situation, which, if not avoided, could result in serious injury or death.</p> <p><b>WARNING</b> symbol on the equipment refers the user to the product manual for additional information. The symbol appears next to required information in the manual.</p>
	<b>WARNING, Risk of electrical shock:</b> Potential shock hazard where HAZARDOUS LIVE voltages greater than 30 Vrms, 42.4 Vpeak, or 60 VDC may be accessible.
	<b>ESD HAZARD:</b> Danger of an electro-static discharge to which equipment may be sensitive. Observe precautions for handling electrostatic sensitive devices.
	<b>Protective Earth (PE) terminal:</b> Provided for connection of the protective earth (green or green/yellow) supply system conductor.
	<b>Functional earth terminal:</b> Used for non-safety purposes such as noise immunity improvement. NOTE: This connection shall be bonded to Protective Earth at the source of supply in accordance with national local electrical code requirements.
	<b>Earth Ground: Functional earth connection.</b> NOTE: This connection shall be bonded to Protective Earth at the source of supply in accordance with national and local electrical code requirements.
	<b>Chassis Ground:</b> Identifies a connection to the chassis or frame of the equipment shall be bonded to Protective Earth at the source of supply in accordance with national and local electrical code requirements.

## How to Use This Manual

To receive the maximum benefit from your XYR 300G product, please read the **Introduction**, **Installation** and **Operation** chapters of this manual thoroughly before using the XYR 300G.

Chapter Four **Configuration** explains how to configure the modules using the Configuration Software available.

Chapter Six **Troubleshooting** will help if your system has problems.

The foldout sheet *XYR 300G Installation Guide* is an installation drawing appropriate for most applications.

# CONTENTS

<b>CHAPTER 1</b>	<b>INTRODUCTION</b>	<b>1</b>
1.1	OVERVIEW	1
1.1.1	Modbus XYR 300G	2
1.1.2	Ethernet XYR 300G	2
1.2	THE XYR 300G STRUCTURE	3
1.2.1	On-board I/O	4
1.2.2	I/O Expansion - XYR 301X modules	4
1.3	THE WIRELESS NETWORK	4
1.3.1	XYR 300X to XYR 300G Network	5
1.3.2	XYR 300G to XYR 300G Network	6
1.3.3	“Data Concentrator” Networks	7
1.3.4	XYR 300G Repeaters	7
<b>CHAPTER 2</b>	<b>OPERATION</b>	<b>8</b>
2.1	START-UP	8
2.2	OPERATION	8
2.3	DATABASE	10
2.4	THE HOST - XYR 300G LINK	11
2.4.1	Modbus	11
2.4.2	Ethernet	12
2.5	RADIO SYSTEM DESIGN	12
2.5.1	Radio Signal Strength	12
2.5.2	Repeaters	13
2.6	RADIO COMMS FAILURE	13
2.6.1	Monitoring Communications Failure	14
2.7	SECURITY CONSIDERATIONS	15
<b>CHAPTER 3</b>	<b>INSTALLATION</b>	<b>16</b>
3.1	GENERAL	16
3.2	ANTENNA INSTALLATION	16
3.2.1	Dipole and Collinear antennas.	18
3.2.2	Yagi antennas.	18
3.3	POWER SUPPLY	19
3.3.1	AC Supply	20
3.3.2	DC Supply	20
3.3.3	Solar Supply	21
3.4	INPUT / OUTPUT	22
3.4.1	Digital Inputs / Outputs	22
3.5	SERIAL PORT	23
3.5.1	RS232 Serial Port	23
3.5.2	RS485 Serial Port	24
3.6	ETHERNET PORT	25
<b>CHAPTER 4</b>	<b>CONFIGURATION</b>	<b>26</b>
4.1	INTRODUCTION	26
4.2	CONFIGURATION PROGRAM	27
4.2.1	Program Operation	27
4.2.2	Security	30

4.3	UPLOADING AND DOWNLOADING	32
4.3.1	Loading from a XYR 300G	33
4.4	MAPPINGS XYR 300G TO XYR 300X I/O MODULES	34
4.4.1	Mappings from Inputs at Remote XYR 300X I/O Modules	34
4.4.2	Mappings from XYR 300G to Outputs at Remote XYR 300X I/O Modules	37
4.4.3	Don't Send if in Comm Fail	39
4.4.4	Startup Polls	39
4.4.5	Polls to Remote Modules	39
4.5	MAPPINGS FROM XYR 300G TO OTHER XYR 300G MODULES	40
4.5.1	Entering a Block Mapping	42
4.5.2	Host Device Trigger	43
4.5.3	Time Period	44
4.5.4	Real-Time	44
4.5.5	Change-of-State	45
4.5.6	Block Read Mapping's	46
4.5.7	Mixing Normal Mappings and Block Mappings	47
4.5.8	Block Mappings to internal I/O Registers.	47
4.5.9	Comms Fail for Block Mappings	47
4.5.10	"Repeater-only" Configuration	47
4.6	CHANGE SENSITIVITY & I/O VALUE SCALING	47
4.6.1	Change Sensitivity	47
4.6.2	I/O Value Scaling	49
4.6.3	Unit Details	52
4.7	SERIAL CONFIGURATION - MODBUS	54
4.7.1	MODBUS Slave	54
4.7.2	MODBUS Master	56
4.8	FIELDBUS CONFIGURATION	60
4.8.1	Fieldbus Mappings	61
4.9	FIELDBUS CONFIGURATION - ETHERNET	66
4.9.1	Setting IP Address	67
4.9.2	Modbus TCP	68
4.10	CONNECTING SERIAL I/O	73
4.11	ACCESS TO MESSAGE BUFFER COUNT	74
<b>CHAPTER 5</b>	<b>SPECIFICATIONS</b>	<b>75</b>
<b>CHAPTER 6</b>	<b>DIAGNOSTICS</b>	<b>78</b>
6.1	DIAGNOSTICS CHART	78
6.2	DIAGNOSTICS MENU	79
6.3	ETHERNET DIAGNOSTICS	86
6.4	FIELDBUS INDICATING LEDS	88
6.4.1	Ethernet Indicating LED's	88
6.5	RADIO PATH TESTING	89
<b>CHAPTER 7</b>	<b>WARRANTY</b>	<b>91</b>
<b>APPENDIX 1</b>	<b>STATUS REGISTERS</b>	<b>92</b>
<b>APPENDIX 2</b>	<b>IT FUNCTIONALITY</b>	<b>94</b>
<b>APPENDIX 3</b>	<b>COMMS LOGGING</b>	<b>110</b>



# Chapter 1

# INTRODUCTION

## 1.1 Overview

The Wireless Gateway products provide a wireless interface between various fieldbus protocols used in process and automation applications. The XYR 300G includes an integral radio transceiver, and transfers transducer and control signals (I/O) using a highly secure and highly reliable radio protocol.

Models are available with license free wireless (900MHz FHSS, 869MHz fixed frequency and 433MHz fixed frequency) or licensed band wireless (150-230MHz or 380-512MHz).

### ***Functionality discussed in this manual for the XYR 300G range.***

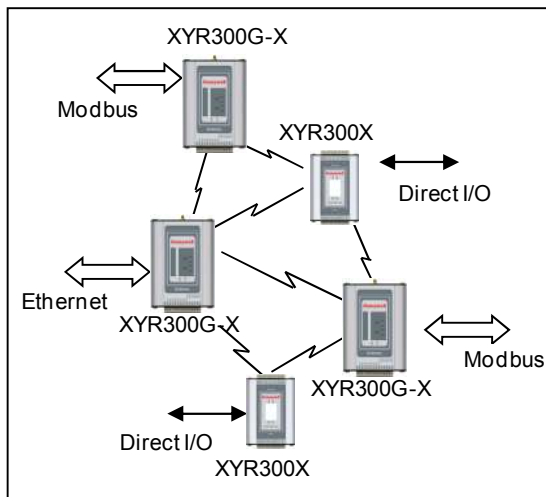
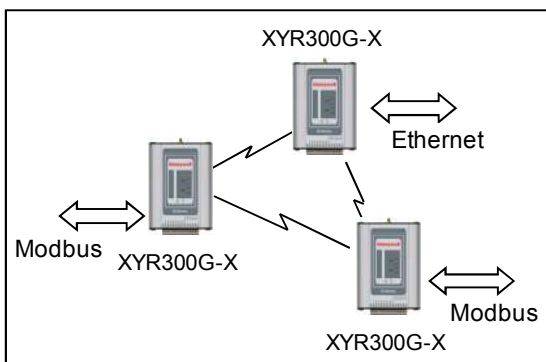
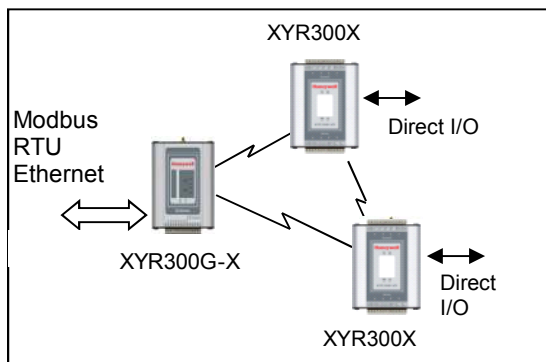
The XYR 300X radio protocol is designed for very efficient radio band usage, with event reporting communications, automatic acknowledgment and error-correction, peer to peer addressing, multiple path routing, and data encryption for system security.

Application types include:

- The XYR 300G interfaces between XYR 300X wireless I/O and various fieldbus protocols. Connect wireless I/O to PLC's, DCS, SCADA or Internet.
- Wireless interconnectivity between different fieldbuses - Ethernet to Modbus .
- Combined networks of the above.

The XYR 300G has eight on-board discrete I/O. Each I/O point can be configured individually as a contact input signal, or a discrete output signal. Input signals can sent via its fieldbus connection to a host device (PLC, DCS etc) or be transmitted by radio to other XYR 300X units. The output signals can be driven by a host device, or linked to inputs on remote XYR 300X units.

This document assumes the reader is familiar with the operation of the XYR 300X I/O modules - for further information, please refer to the User Manuals for these products.



**Ordering information:**

XYR 300G-M	Modbus Master & Slave
XYR 300G-E	Ethernet interface - Modbus TCP, FTP, HTML, Email

### 1.1.1 Modbus XYR 300G

The XYR 300G-M can be configured for Modbus master interface or Modbus slave.

Modbus is a Master-Slave protocol originally developed by Modicon (now part of the Schneider group). It became a popular interconnect protocol with many equipment manufacturers. One Modbus master controls the Modbus network communications, which can comprise up to 250 Modbus slave devices. The Modbus master can read or write I/O values to/from Modbus slaves. The XYR 300G can be configured as either Modbus Master or Modbus Slave. The variation of Modbus supported by the XYR 300G is “Modbus RTU” (also known as “Modbus binary”).

The XYR 300G-M has two serial connections - RS232 and RS485, on the bottom end plate of the module. The serial port provides both RS232 and RS485 hardware connections, however both connections are paralleled internally - both connections cannot be used at the same time. Either RS232 or RS485 can be used for Modbus communications. The serial port must be configured to suit the host device. Serial data rates between 1200 and 19200 baud may be selected, and character types with 7 or 8 data bits, even/odd/none parity, and 1 or 2 stop bits may be selected.

The Modbus XYR 300G has 4300 general-purpose I/O registers. Each discrete, analog and pulse I/O point takes up one register.

### 1.1.2 Ethernet XYR 300G

The Ethernet XYR 300G provides several different types of Ethernet functionality:

- ◆ Modbus TCP. Modbus TCP uses Modbus as a base protocol within an Ethernet communications structure. The XYR 300G provides class 0, 1 and partially class 2 slave functionality.
- ◆ Internet functionality. The XYR 300G has 1.4Mbyte of non-volatile “flash” memory for embedded web “pages” (dynamic HTTP), on-board file system, user downloadable web pages through FTP server, and email functionality (SMTP).

The Ethernet connection is a transformer isolated RJ45 connector, 10/100 Mbit/sec.

The Ethernet XYR 300G I/O database has 4300 registers (each of 16 bit value), however the Ethernet interface only supports 2048 input bytes and maximum 2048 output bytes. Each byte can be 8 discrete inputs or outputs, but analog or pulse I/O take up 1 byte for low resolution values (8-bit) or 2 bytes for high resolution values (16-bit).

An “output” is a value coming into the XYR 300G via the fieldbus. An input is a value going out from the XYR 300G via the fieldbus.

So an Ethernet XYR 300G can handle up to 4300 I/O total, but analog or pulse inputs are limited to 2048 x 8-bit values or 1024 x 16-bit values. The same limit applies to outputs.

For example, an Ethernet XYR 300G can handle 2000 discrete inputs and 500 analog inputs (assume analogs are 16-bit). The number of input bytes is 1250 (2000/8 + 500\*2). The same

unit could handle 4000 discrete outputs and 750 analog outputs. The number of output bytes is 2000 ( $4000/8 + 750*2$ ). The total number of I/O is 3250 which is less than the total limit of 4300.

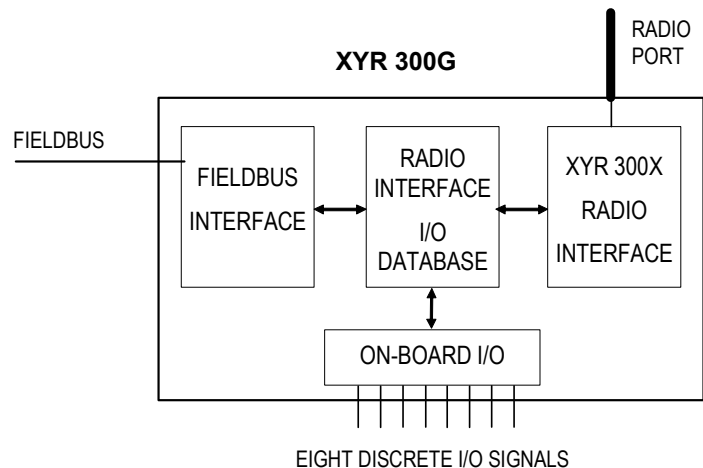
## 1.2 The XYR 300G Structure

The XYR 300G has three functional sections:

- The Radio Interface consists of an I/O database (or "Process Image") that maintains the latest values of all I/O in the wireless I/O system. The I/O database comprises 4300 x 16 bit I/O registers and 4300 x 16 bit status registers. There are also other registers in the database that can be used for system management - they are discussed later in this manual. NOTE – the terms 'Radio Interface' and 'I/O database' are used interchangeably throughout the manual.
- The radio port allows the XYR 300G to communicate with other XYR 300G and/or XYR 300X modules using the XYR 300X protocol (called "Honeywell XYR 300X"). Messages from the XYR 300X modules are received by the radio port and used to update the input values in the XYR 300G Radio Interface. The radio port also creates the correct radio message to set outputs on the remote XYR 300X modules.

The Honeywell XYR 300X protocol is an extremely efficient protocol for radio communications. Radio messages can be sent using exception reporting - that is, when there is a change of an input signal - or by read/write messages. Each message can comprise a single I/O value, or multiple I/O values (termed a "block" of I/O).

There are also update messages, which are sent for integrity purposes. Messages include error checking, with the destination address sending a return acknowledgment. Up to five attempts are made to transmit the message if an acknowledgment is not received. The Honeywell protocol is designed to provide reliable radio communications on an open license-free radio channel.



- The Fieldbus port enables communications between a host device, which could be a PLC, DCS, HMI, intelligent transducer, etc), and the XYR 300G Radio Interface database. A "host device" may be one or several devices connected to the same fieldbus or network (for example, an Ethernet LAN) - in this manual, the LAN is considered as a "host device".

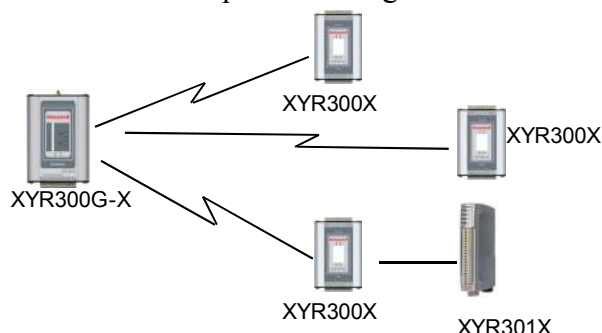
The fieldbus port decodes messages from the host device and reads or writes I/O values to the database. The fieldbus port can also generate messages to the host device.

The XYR 300G I/O database effectively isolates the fieldbus and the radio network. This provides a high level of system performance. The XYR 300X radio protocol is very efficient and reliable for radio communications. It minimizes radio channel usage by "change-of-state"

reporting, and allows the use of intermediate repeater addresses. It also allows peer-to-peer (XYR 300X to XYR 300X, XYR 300G to XYR 300G) and peer-to-master (XYR 300X to XYR 300G) communications. PLC protocols, by comparison, are designed to provide transfer of large I/O files by "wire" link. The XYR 300G retains the advantage of both protocols in their respective communications media.

### 1.2.1 On-board I/O

The XYR 300G has eight on-board discrete I/O. Each I/O point can be used as either a discrete input (voltage free contact input) or discrete output (transistor output) - an I/O point cannot be used as both input and output. Each I/O point is linked to two separate I/O registers in the database - one for the "input" function and one for the "output" function.. If the output register is set "on" by the fieldbus or by a radio message from a remote module, then the XYR 300G will automatically set the input register for the same I/O point to "off". This means that the output register has priority over the input register - if there is a conflict, the input value is ignored.



The XYR 300G also has three internal inputs linked to I/O registers:

- ◆ Supply voltage status - if the normal supply fails, this status is set on.
- ◆ Low battery voltage. The XYR 300G has an internal battery charger to trickle charge a back-up battery. If the battery voltage is low, this status is set.
- ◆ Battery voltage - the actual value of the connected battery voltage.

### 1.2.2 I/O Expansion - XYR 301X modules

The XYR 300G provides eight on-board discrete I/O. Where additional I/O is required, XYR 301X modules can be connected to the RS485 port of the XYR 300G modules.

**Note:** The XYR 301X modules can communicate using the Honeywell Protocol or via Modbus RTU protocol. The XYR 301X can act as a Modbus Slave device with a Modbus RTU address range of 1-99, which is selectable via the rotary switches on end plate of module.

If using a XYR 300G-M utilizing Modbus Protocol and additional I/O is required then the XYR 301X module can be added via RS485 communications onto the Modbus network with a unique Modbus RTU address.

## 1.3 The Wireless Network

The XYR 300G can communicate with up to 490 other addresses - this could be 490 other XYR 300X modules, or in the case of XYR300L modules, it could be many thousands of modules. XYR 300G modules may take up more than one address under some circumstances.

Any XYR 300G or XYR 300X module can act as a radio repeater for other modules - that is, radio messages can be passed onto other modules. Up to five repeater addresses can be configured for messages transmitted to a XYR 300G module.

Each module can have a unit address between 1 – 95, but the XYR 300G also recognizes repeater addresses in conjunction with the unit address as the module “identifier”. Hence module #2 is recognized as different to #2 via #57 - #57 being a repeater.

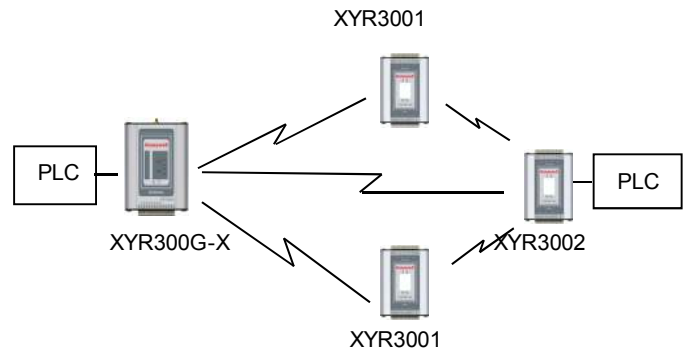
### 1.3.1 XYR 300X to XYR 300G Network

In the wireless I/O system, the XYR 300G acts as a normal XYR 300X module (this covers XYR 300X I/O and XYR300L modules).

XYR 300X modules transmit messages to the XYR 300G address and the XYR 300G acknowledges these messages like a normal XYR 300X module. When a XYR 300G transmits messages to change remote outputs, it will "re-try" if it does not receive an acknowledgment, like a normal XYR 300X module.

Remote XYR 300X modules can connect to XYR 301X modules in the normal way. The XYR 300G host can access I/O on XYR 301X modules by using the intermediate XYR 300X as a repeater.

XYR 300X modules can transmit input messages directly to outputs on other XYR 300X module, as well as the XYR 300G. The same input can be transmitted to different addresses by entering two "mapping" configurations at the remote module.



### Normal XYR 300X Messages

I/O registers in a XYR 300G can be configured (mapped) to outputs at remote XYR 300X modules, or I/O registers in XYR 300G modules. The XYR 300G will transmit an I/O message when a “change-of-state” occurs for that I/O register. Registers have a configurable “sensitivity” value - this determines how much the register value has to change to trigger a change message. A change-of-state occurs when the register value has changed by more than the sensitivity value since the last transmission.

The XYR 300G also transmits periodic update messages if there has been no change - if an I/O register is mapped to a remote output or another XYR 300G, then that register can be configured with an update time.

### Poll Messages

A XYR 300G can also generate poll messages to remote XYR 300X modules. These poll messages act in the same way as a start-up poll - the remote module immediately responds with update messages for any I/O mappings configured to the XYR 300G.

Poll messages can be triggered by:

- ◆ time period, configurable 1 – 4096 sec (1.1 hour), or
- ◆ real time clock, or
- ◆ on demand by the host device, by writing to a “trigger register” in the XYR 300G

### 1.3.2 XYR 300G to XYR 300G Network

Different types of XYR 300G modules can communicate - for example, a Modbus XYR 300G can communicate with an Ethernet XYR 300G. I/O registers in one XYR 300G can be transmitted to I/O registers in another XYR 300G. When the XYR 300G is configured, “mappings” can be entered linking I/O registers to registers in another XYR 300G.

As well as the normal “I/O change” messages and update messages, the XYR 300G has “block read” and “block write” messages for use with other XYR 300G modules. These messages will transmit multiple register values instead of only one as in the normal XYR 300X message. The block read/write messages increase the efficiency of radio communications where a XYR 300G “sees” a large number of changes in its database at the one time. For example, if a host writes a block of 100 signal values to a XYR 300G, and 20 of these values have changed since the last write-operation. If the block is mapped to another XYR 300G, then the XYR 300G can transmit all 20 values in one radio message, instead of 20 messages.

Normal I/O messages can be repeated by any type of XYR 300X I/O module, however block read/write messages can only be repeated by other XYR 300G modules.

#### Block Read Message

A block read message is a request to another XYR 300G to transmit the values of a consecutive block of registers. The destination XYR 300G will respond with the values, which will be stored in a corresponding block of registers in the originating XYR 300G. A block read message can be triggered by:

- ◆ Time period, configurable 1 – 4096 sec (1.1 hour), or
- ◆ Real time clock, or
- ◆ On demand by the host device, by writing to a “trigger register” in the XYR 300G.

#### Block Write Message

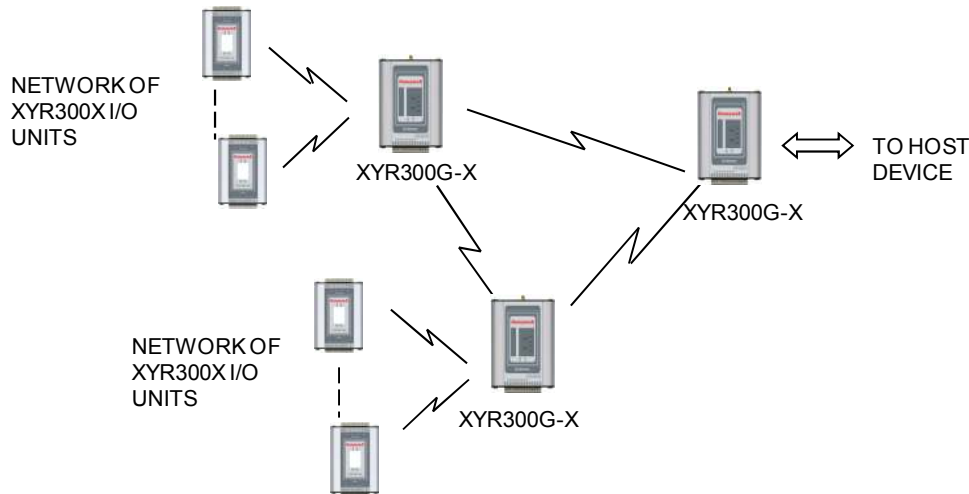
A block write message transmits a consecutive block of register values from one XYR 300G to a destination XYR 300G. It can be triggered by:

- ◆ Time period, configurable 1 – 4096 sec (1.1 hour), or
- ◆ Real time clock, or
- ◆ On demand by the host device, by writing to a “trigger register” in the XYR 300G, or
- ◆ A change-of-state event occurring within the block of I/O registers.

If a block write message has been configured to be transmitted on change-of-state, a “time window” is configured. When a change-of-state occurs in one of the registers in the block, the time window will be activated. All changes during the time window will be grouped together and transmitted as one block write message. That is, the block write message will not be sent immediately the first change-of-state occurs (unless the time window is configured to zero), but will be sent at the end of the time window - any other registers in the block that change during the time window will be sent as part of the same message. The time window can be configured from 0 – 255 seconds.

### 1.3.3 “Data Concentrator” Networks

XYR 300G units can act as “data concentrator” units to collect I/O from a local network of XYR 300X wireless I/O modules and pass the I/O on to another XYR 300G as a block.



This type of network reduces the amount of radio traffic and is suitable for systems with a large number of I/O modules. The system is divided into local sub-networks, each with a XYR 300G unit. The XYR 300X modules transmit their I/O values to the XYR 300G. The XYR 300G then transfers these values to the “central” XYR 300G using a block transfer which is very efficient compared to a lot of individual I/O transmissions.

The data concentrator network is different than using the XYR 300G as a repeater. A repeater re-transmits each message in the same format. A data concentrator collects the I/O values as a block, and transmits the complete block in one transmission.

### 1.3.4 XYR 300G Repeaters

Any XYR 300X module can repeat a normal radio message, however only XYR 300G modules can repeat a block message. XYR 300G units connected to a host device can also act as a repeater for other modules.

Where a XYR 300G is being used without a host device as a repeater or data-concentrator, it can be configured as “Repeater-only”. This allows the RS232/485 port to be used for on-line diagnostics. If the unit is a XYR 300G-M, the “Repeater-only” configuration also allows this module to connect to XYR 301X serial I/O modules.

## 2.1 Start-up

---

The XYR 300G operating software and the database configuration are stored in non-volatile memory, however the database I/O register values are lost on power failure (in the same way as a PLC).

On start-up, the XYR 300G sends "start-up poll" messages to remote modules based on the source address of inputs configured in the database (the start-up messages can be disabled by configuration). The remote modules respond with update messages for their inputs, which sets initial values in the XYR 300G I/O database registers. The XYR 300G provides a delay of 5 seconds between each start-up poll, to allow the remote module to respond and to avoid overloading the radio channel.

If there are a lot of remote modules, then this start-up stage may take a significant time, and this should be allowed for in the system design. The XYR 300G has an internal battery charger feature and the use of a back-up battery should be considered if this start-up delay presents a constraint to system reliability. Start-up polls may be disabled for individual remote modules in the database configuration.

For the host device, the XYR 300G provides an "Active" signal on the RS232 port (DCD pin 1). Its purpose is to indicate to the host that the XYR 300G is now processing output messages for the remote modules. When the XYR 300G powers down (or should an internal fault occur), the "Active" signal resets (turn "off" or "0"). When the XYR 300G starts-up, it holds the "Active" signal in a reset condition ("off" or "0") for a time equal to the number of remote addresses (or modules) configured times 5 seconds plus any delay if remote addresses are offline. For example, if there are 20 remote addresses configured in the XYR 300G database, then the "active" signal will be held in the reset state for 100 seconds (20 x 5). During this period, the XYR 300G will not change any output values in its database. After this time, the XYR 300G will set the "Active" signal (to "on" or "1") - the host can then send messages to the XYR 300G to update the output values in the database.

## 2.2 Operation

---

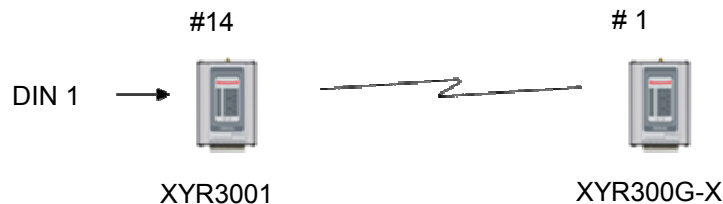
The XYR 300G database can hold values for 4300 I/O signals plus the 8 on-board I/O. The database registers (also called I/O registers) can be accessed by both the radio port and the fieldbus port. The host device can change values in the database via the fieldbus, and the XYR 300G can transmit radio messages out with the new values. Radio messages can be received with new values for database registers, and these new values can be written to the host device or read by the host device, via the fieldbus.

The XYR 300G operation must be configured before the XYR 300G will function. Configuration is achieved by creating a configuration file on a PC and downloading this file to the XYR 300G. The XYR 300G configuration may also be "uploaded" to a PC for viewing and modification. For more information, refer to the **Configuration** section of this document.



Each I/O register in the XYR 300G database has a 16-bit value. It doesn't matter if the remote I/O is digital (discrete), analog or pulse. The host protocol driver in the XYR 300G will convert the 16 bit value into a value that the host will understand. For example, if the host device requests a binary/digital read command, the XYR 300G will convert the 16 bit value into a binary (1 bit) value before it responds.

The XYR 300G is able to scale the I/O value between the I/O database and the host device - this is a user-configurable function.



An example of normal operation - assume that a remote module has address 14 and the XYR 300G is address 1. Module #14 is configured with a mapping DI1 → I/O Reg 76 at #1. When DI1 turns "on", module #14 transmits a message. If the XYR 300G can hear this message, it will transmit an acknowledgment back to module #14, and updates the value of I/O register 76 in the XYR 300G database. The host device can read I/O register 76 via the data-bus, or the XYR 300G may write the value of I/O register 76 to the host device.

I/O registers that receive values from other XYR 300X or G modules via radio are configured with a "Communications fail time". If the XYR 300G does not receive a message for this I/O register within the comms-fail time, then the I/O register is given a "comms fail" status which the host device can read. The I/O value can also be configured to reset to zero on comms-fail.

I/O registers that transmit out to other XYR 300X or G modules are configured with an "update time" and a "sensitivity". The XYR 300G will transmit a message to the configured remote output whenever the I/O register value changes by the sensitivity amount – if it has not changed within the update time, the XYR 300G will send a message anyway. The XYR 300G will make five attempts to send a message - if it does not receive an acknowledgment from the remote module, then the I/O register is given a "comms fail" status which the host device can read.

Each I/O register has an associated "status" register, which includes information such as comms-fail status. As well as each I/O register having an individual comms-fail status, each remote module has an overall comms fail status. This status is "set" (on) whenever a comms-fail occurs for an individual I/O register, and is "reset" (off) whenever a message is received from the remote module. The XYR 300G can be configured to not send any update messages to a remote module if it senses that the remote module is in "comms fail" - that is, if any I/O register associated with the remote module is in "comms fail". It will start sending update messages again when the XYR 300G receives a message from the remote module. The default configuration is that output updates ARE sent during comms fail conditions.

## 2.3 Database

---

The XYR 300G database (Radio Interface) has 10 000 registers, each of 16 bit size. The structure of the database is:

Registers	Purpose
0 - 4299	I/O registers
4300 - 4399	On-board I/O
4401 - 4499	Comms-fail status and radio strengths for remote modules
5000 - 9499	Status registers - 16 bit status for each I/O signal
9500 - 9999	Status registers for block read/write messages

The register numbers may be used by the Host Protocol Driver to access I/O values and I/O status information. Each configured I/O point has a 16 bit value (in registers 0000 - 4299), and a 16 bit status value. The status register is located at 5000 plus the I/O value register. For example, an I/O point in register number 2560 has a status value in register number 7560 (5000 + 2560).

Details of the status register are provided in Appendix A. The most important part of the status register is the 15<sup>th</sup> or most significant bit - this indicates comm-fail status for the I/O register. If the most significant bit is set, then the I/O register is in comms-fail.

The host device can read the status registers. For example, the communications status of an output configured at register number 3001 can be examined by reading register number 8001 (5000 + 3001). If the register value is greater than 32767, then the 15th bit is set, indicating that the output has a communications failure.

### 2.3.1 On-board I/O and Internal I/O

The XYR 300G has eight discrete I/O points. These may be used as inputs or as outputs. Inputs are linked to registers 4300-4307. That is, if a contact connected to DIO1 is “on”, then register 4300 is given an “on” value. The inverse of the input values are stored in registers 4370-4377.

Outputs are controlled from registers 4320-4327; that is, if register 4327 is set to an “on” value, then output DIO8 is activated.

Whenever an output register is set “on”, the corresponding input register is automatically set “off”. For example, if register 4321 is set to “1”, the XYR 300G will also set 4301 to “0”. This means that if both the input and output registers corresponding to the same I/O point are used in the configuration, then the output register has priority.

Outputs may be written to by either the host device or by a remote XYR 300X via the radio port. Input values can be sent to the host device or to a remote module via the radio port.

The XYR 300G also monitors its battery voltage and supply voltage. These are stored in registers 4310 and 4311 respectively, as 16 bit values, scaled so that a value of 16384 decimal (hex 4000) corresponds to 8 V, and a value of 49152 (hex C000) corresponds to 40V.

A low battery alarm is available at register 4308. This becomes active when the battery voltage falls below 11.3V, and clears when the battery voltage rises above 11.8V. Supply voltage is also monitored, and an alarm is available at register 4309. This becomes active if the supply voltage falls below 8.0V, and clears when the supply voltage rises above 9.0V.

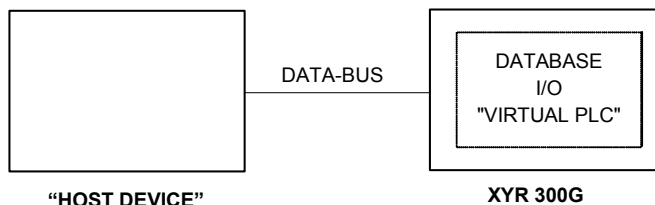
I/O Register	Description	I/O Register	Description
4300	Input value DIO 1	4320	Output value DIO 1
4301	Input value DIO 2	4321	Output value DIO 2
4302	Input value DIO 3	4322	Output value DIO 3
4303	Input value DIO 4	4323	Output value DIO 4
4304	Input value DIO 5	4324	Output value DIO 5
4305	Input value DIO 6	4325	Output value DIO 6
4306	Input value DIO 7	4326	Output value DIO 7
4307	Input value DIO 8	4327	Output value DIO 8
4308	Low battery voltage status		
4309	Supply voltage fail status		
4310	Battery voltage value		
4311	Supply voltage value		
4370 - 4379	Inverse values of 4300 - 4309		

## 2.4 The Host - XYR 300G Link

For the host device, the XYR 300G "looks" like a single device (or a "virtual PLC"), containing the I/O for the complete wireless I/O system.

### 2.4.1 Modbus

The user selects whether the XYR 300G-M should act as a Modbus Master or Modbus Slave .



The data type and baud rate of the serial communications must be configured at the XYR 300G to match the host. Data types can be 7 or 8 bit, even/odd/no parity, with 1 or 2 stop bits. Data rates can be 300 - 19200 baud.

The full XYR 300G database (4300 registers) can be accessed by the Host Device.

### **2.4.2 Ethernet**

The Ethernet port automatically handles Ethernet communications at 10 or 100 Mbit/sec. An IP address is entered so that other Ethernet devices can recognize the XYR 300G.

The Ethernet units have internal hardware comprising the Ethernet Interface. The Ethernet Interface handles all Ethernet Network communications. The internal Radio Interface is separate to the Ethernet Interface, and handles all radio communications. I/O in the Radio Interface is linked to I/O in the Ethernet Interface in a flexible way via XYR300X Configuration Software.

The Ethernet Interface provides a total of 2048 input bytes and 2048 output bytes. An Ethernet byte can contain 8 discrete (binary) values, or two bytes can be used for a 16-bit analog or pulse register. So the Ethernet Interface is limited to 4300 discrete inputs (the limit of the XYR 300G database) or 1024 analog inputs (the limit of the Ethernet interface) or a combination. The same applies for outputs.

For example, an Ethernet host wants to read 500 analog inputs (1000 bytes). The remaining input bytes (1548) could be used for 12,384 discrete inputs - but the XYR 300G database is not this big. Provided there are no outputs required, there could be 3800 discrete inputs (4300 – 500 analogs). If there are outputs required, then the number of discrete inputs available will be further limited.

## **2.5 Radio System Design**

---

Each wireless I/O system can have up to 95 unit addresses. Each XYR 300X module can have up to 31 XYR301X serial modules connected to it. These modules are addressed 96 - 127. More than one XYR301X module can have the same address, provided they are not connected to the same XYR 300X module - that is, #100 via #16 is identified as a different module to #100 via #65.

A constraint that needs to be considered is the capacity of the radio channel. If there is too much traffic on the radio channel, then the system quickly becomes unreliable. The recommended maximum average traffic density is 100 messages per minute provided all radio paths are reliable. If there are marginal radio paths, resulting in re-tries of transmitted messages, then the maximum traffic density is reduced considerably. Each block read/write messages should be counted as two messages because of the length of these messages.

A XYR 300G can be used as a repeater module for messages between other modules.

### **2.5.1 Radio Signal Strength**

The XYR 300G records the radio signal strength of remote modules that communicate directly (that is, not via repeaters). There are 95 database registers (4401 – 4495) which store the radio strengths – corresponding to remote addresses #1 - #95. The radio strength (RSSI) is measured in dBm (relative to 1mW of RF power). The RSSI value is stored in the 8 least significant bits of each register - a value of –84 dBm would be stored as decimal 84.

These database registers will hold the strength of the last message received from the address. If a message is received from a remote module via a repeater, then the measurement is recorded in the address of the last repeater. For example, if a message is received from #24 directly, then the

RSSI will be recorded in register 4424. If a message is received from #24 via #25, then the RSSI is recorded in register 4425. The XYR 300G will not know what the radio strength of the message from #24 to #25 is. If #25 is another XYR 300G, then it can record this RSSI and this register could be mapped to an I/O register in the first XYR 300G.

The RSSI registers can be read by the host device, or mapped to I/O registers in other XYR 300G modules.

The first half of the register (8 most significant bits) will be decimal 0 (hex 00) if the remote module has active communications. If a comms fail status to this address occurs, the most significant bit will be set. For example, if the last message received from #38 is -99dBm, then the 16 bit value of register 4438 will be decimal 99 or hex 0063. If the "comms fail" status for #38 is set, the 16 bit value of register 4438 will become decimal 32,867 (32768 + 99) or hex 8063.

### **2.5.2 Repeaters**

Radio paths may be extended by using intermediate modules as repeaters. A repeater will receive and re-transmit the radio message. Up to five repeater addresses can be configured - that is, a radio message can pass through five intermediate modules. For normal I/O messages, any XYR 300X module can be used as a repeater, however for block read/write messages, only XYR 300G modules can act as repeaters.

## **2.6 Radio Comms Failure**

---

The XYR 300G has an internal "communications failure" (comms fail) status for each I/O point in its database. There is also a comms fail status for each module with direct communications - see 2.5.1 above.

For I/O registers which are mapped to a remote output or another XYR 300G, the comms fail status is set if the XYR 300G does not receive an acknowledgment for a message being sent to that remote output. The comms fail status resets when a successful transmission occurs.

For I/O registers which have been mapped, from a remote input or another XYR 300G, a comms fail time period may be configured. If a radio message for this I/O register has not been received within this time, then this registers comms fail status is set. The comms fail status will reset when a message is received for this register. If the comms fail time is configured as zero, then the comms fail status will never be activated. A reset of registers if in comms fail option is available for values to reset back to 0.

The communications failure status is bit 15 of the status register for each I/O point. If the host device reads a register as a digital or binary value, then the XYR 300G returns bit 15 of the register (0 or 1) - this is the comms fail bit of a status register.

It is important to use the comms fail status in the overall system design, as any system can fail.

The XYR 300G also provides an additional comms failure feature to stop the XYR 300G transmitting output messages to an individual remote address if the XYR 300G already knows that this remote address is in communication failure. This prevents the XYR 300G from congesting the radio channel with a lot of unnecessary transmissions (and re-transmissions). This function is called "Don't Send if In Comm Fail" and is configurable by the user for each

individual remote address. The XYR 300G retains a "remote address comms fail" status for the remote addresses configured for this function. If any output with this remote address goes into communications failure, then the remote address comms fail status is set ("on" or 1) - every time an input with this remote address receives a radio message, then the remote address comms fail status is reset ("off" or 0). While the remote address comms fail status is set, the XYR 300G disables any output messages being sent to this remote address.

When this feature is configured, all output transmissions are stopped if communications with a remote module fails for a short period. They will start again when an input message from this module is received. If the XYR 300G determines that a output message should be sent to an output which is disabled because of this feature, then the output message will not be sent and the comms fail status of that output is set ("on" or 1).

If it is desired to use this function with a remote XYR 300X module, but there are no inputs from this module being used, then it is easy to configure an unused input or an internal input (mains fail or low battery voltage etc). It is the comms fail status for the input, which is used, not the input itself.

### **2.6.1 Monitoring Communications Failure**

The host device can monitor the communications status of an I/O point by reading the status register for this point as a binary/discrete register. Modbus, and many other protocols, will convert a 16 bit register value to a binary/discrete value by returning the most significant bit - for the status register, this corresponds to the comms status bit.

For example, to monitor the comms status of I/O register 1045, perform a binary/discrete read on register 6045 (the status register for 1045). A value of "1" will be returned if this I/O point is in comms fail, and a "0" returned if the status is normal.

If it is desired to monitor the comms status of all I/O points, it is more efficient to only monitor the comms status of one I/O point at each remote module (if this point is in comms fail, then all points at the remote module will be in comms fail). If this point is an input, then the comms fail time for this input can be made short, to give an early warning of a comms problem (this means that the corresponding update time for the input at the XYR 300X will need to be short). If the point is an output, then the update time for the output should be made short.

## 2.7 Security Considerations

---

There are three dimensions of security considerations:

1. Failure to operate when required - or “operational reliability”.

The features discussed above optimize operating reliability. Using an acknowledgment and re-try protocol ensures that the transmitting module is aware whether the transmitted message has been transmitted reliably. The “comms fail” alarms provide indication if the radio link has failed to operate.

2. Mal-operation, or operating when not requested.

This problem occurs when an output is “triggered” by the wrong radio device. The XYR 300G modules use frequency encoding and a very secure addressing system to ensure this does not occur. An additional security level using data encryption can also be selected.

3. Malicious operation, or “hacking”

This is the problem most associated with security concerns - the ability for someone to access information from a radio system by “listening-in”, or to cause damage by transmitting radio messages to force outputs.

A security option can be selected during the module configuration to protect against this. The security option (if selected) adds data encryption to radio messages. Modules in the same system are automatically configured with the encryption key, such that only these modules can understand each other. “Foreign” modules will hear the messages, but cannot decrypt the messages. For more information, refer to section 4.2.2.

# Chapter 3

# INSTALLATION

## 3.1 General

---

The XYR 300G module is housed in a rugged aluminum case, suitable for DIN-rail mounting. Terminals will accept wires up to 12 gauge (2.5 sqmm) in size.

**All connections to the module must be low voltage (SELV). Normal 110-240V mains supply should not be connected to any terminal of the XYR 300G module.** Refer to Section 3.3 Power Supply.

Before installing a new system, it is preferable to bench test the complete system. Configuration problems are easier to recognize when the system units are adjacent. Following installation, the most common problem is poor communications caused by incorrectly installed aerials, or radio interference on the same channel, or the radio path being inadequate. If the radio path is a problem (i.e. path too long, or obstructions in the way), then higher performance aerials or a higher mounting point for the aerial may rectify the problem. Alternately, use an intermediate XYR 300X Module as a repeater.

The foldout sheet XYR 300G *Installation Guide* provides an installation drawing appropriate to most applications. Further information is detailed below.

Each XYR 300G module should be effectively earthed/grounded via the "GND" terminal on the XYR 300X module - this is to ensure that the surge protection circuits inside the module are effective.

## 3.2 Antenna Installation

---

The XYR 300G and XYR 300X modules will operate reliably over large distances. The distance which may be reliably achieved will vary with each application - depending on the type and location of antennas, the degree of radio interference, and obstructions (such as hills or trees) to the radio path.

*The following information refers to the license-free 900MHz radio. For antenna information on other radio options, please refer to Honeywell Solution Support.*

Typical reliable distances are :

USA/Canada	15 miles	6dB net gain antenna configuration permitted (4W ERP)
Australia/NZ	12 km	unity gain antenna configuration (1W ERP)

Longer distances can be achieved if one antenna is mounted on top of a hill.

To achieve the maximum transmission distance, the antennas should be raised above intermediate obstructions so the radio path is true "line of sight". Because of the curvature of the earth, the antennas will need to be elevated at least 15 feet (5 metres) above ground for paths greater than 3 miles (5 km). The modules will operate reliably with some obstruction of the radio path, although the reliable distance will be reduced. Obstructions that are close to either antenna will have more of a blocking effect than obstructions in the middle of the radio path. For example, a group of trees around the antenna is a larger obstruction than a group of trees



further away from the antenna. The XYR 300G modules provide a test feature that displays the radio signal strength.

Line-of-sight paths are only necessary to obtain the maximum range. Obstructions will reduce the range, however may not prevent a reliable path. A larger amount of obstruction can be tolerated for shorter distances. For very short distances, it is possible to mount the antennas inside buildings. An obstructed path requires testing to determine if the path will be reliable - refer the section 6 of this manual.

Where it is not possible to achieve reliable communications between two modules, then another XYR 300X or XYR 300G module may be used to receive the message and re-transmit it. This module is referred to as a repeater.

An antenna should be connected to the module via 50 ohm coaxial cable (eg RG58, RG213 or Cellfoil) terminated with a male SMA coaxial connector. The higher the antenna is mounted, the greater the transmission range will be, however as the length of coaxial cable increases so do cable losses. For use on unlicensed frequency channels, there are several types of antennas suitable for use. It is important antenna are chosen carefully to avoid contravening the maximum power limit on the unlicensed channel - if in doubt refer to an authorized service provider.

The net gain of an antenna/cable configuration is the gain of the antenna (in dBi) less the loss in the coaxial cable (in dB).

The maximum net gain of the antenna/cable configuration permitted is

<b>Country</b>	<b>Max. gain (dB)</b>
USA / Canada	6

The gains and losses of typical antennas are

<b>Antenna</b>	<b>Gain (dB)</b>	<b>Honeywell Part Nos.</b>
Dipole with integral 15' cable	0	WH3000-O
5dBi Collinear (3dBd)	5	WH3000-O-5
8dBi Collinear (6dBd)	8	SG900-6 WH3000-O-8
6 element Yagi	10	YU6/900 WH3000-Y-8

<b>Cable type</b>	<b>Loss (dB per 30 ft / 10 m)</b>	
Cellfoil	-3	WH-CC10/900 (33' or 10m) WH-CC20/900 (66' or 20m)

The net gain of the antenna/cable configuration is determined by adding the antenna gain and the cable loss. For example, a 6 element Yagi with 66 feet (20 meters) of Cellfoil has a net gain of 4dB (10dB – 6dB).

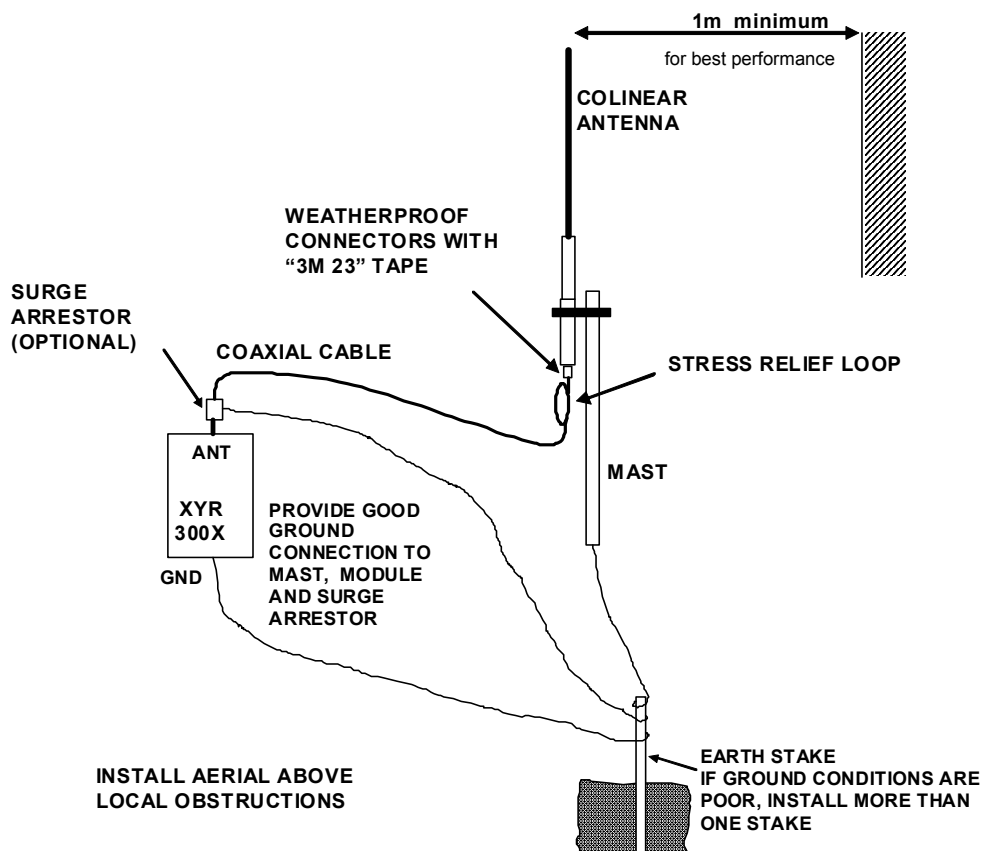
For information on antennas, please refer to Honeywell or an authorized distributor.

Connections between the antenna and coaxial cable should be carefully taped to prevent ingress of moisture. Moisture ingress in the coaxial cable is a common cause for problems with radio systems, as it greatly increases the radio losses. We recommend that the connection be taped, firstly with a layer of PVC Tape, then with a vulcanizing tape such as “3M 23 tape”, and finally

with another layer of PVC UV Stabilized insulating tape. The first layer of tape allows the joint to be easily inspected when trouble shooting as the vulcanizing seal can be easily removed.

Where antennas are mounted on elevated masts, the masts should be effectively earthed to avoid lightning surges. For high lightning risk areas, surge suppression devices between the module and the antenna are recommended. If the antenna is not already shielded from lightning strike by an adjacent earthed structure, a lightning rod should be installed above the antenna to provide shielding.

### 3.2.1 Dipole and Collinear antennas.



A collinear antenna transmits the same amount of radio power in all directions - it is easy to install and use. The dipole antenna with integral 15 ft (5m) cable does not require any additional coaxial cable, however the other collinear antennas do not have integral cable and an external cable length must be connected - such as the CC10 or CC20 cable kits..

Collinear and dipole antennas should be mounted vertically, preferably no less than 2 ft (0.6 metre) away from a wall or mast to obtain maximum range. The CFD890 dipole antenna is the preferred antenna for use in industrial plants and factories.

### 3.2.2 Yagi antennas.

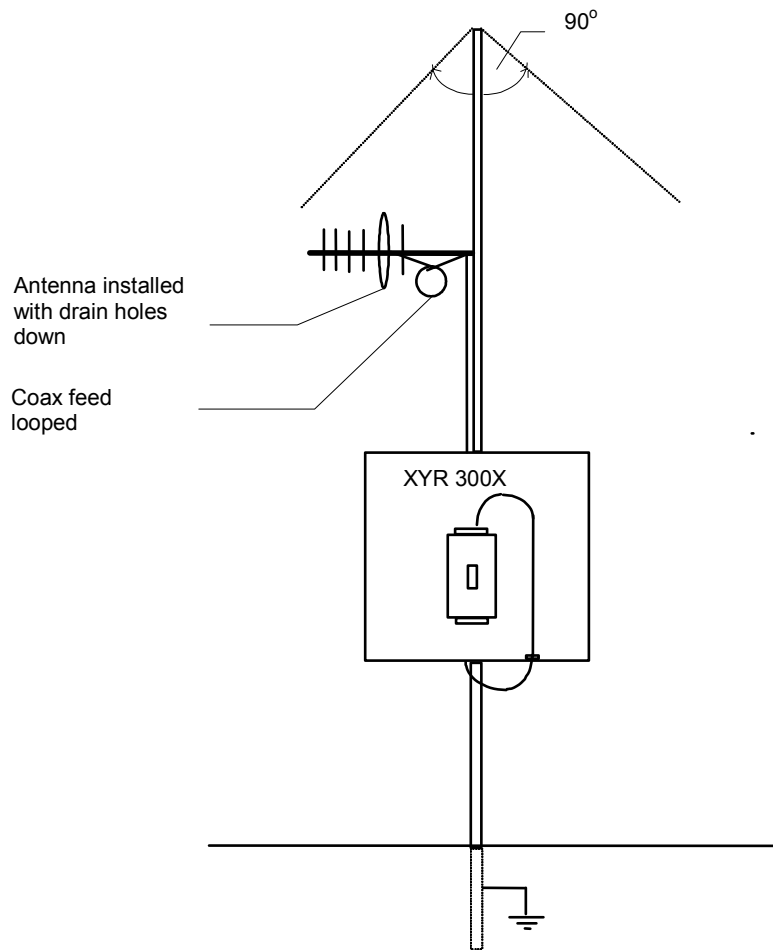
A Yagi antenna provides high gain in the forward direction, but lower gain in other directions. This may be used to compensate for coaxial cable loss for installations with marginal radio path.

The Yagi gain also acts on the receiver, so adding Yagi antennas at both ends of a link provides a double improvement.

Yagi antennas are directional. That is, they have positive gain to the front of the antenna, but negative gain in other directions. Hence Yagi antennas should be installed with the central beam horizontal and must be pointed exactly in the direction of transmission to benefit from the gain of the antenna. The Yagi antennas may be installed with the elements in a vertical plane (vertically polarized) or in a horizontal plane (horizontally polarized). For a two station installation, with both modules using Yagi antennas, horizontal polarization is recommended.

If there are more than two stations transmitting to a common station, then the Yagi antennas should have vertical polarization, and the common (or “central” station should have a collinear (non-directional) antenna.

Also note that Yagi antennas normally have a drain hole on the folded element - the drain hole should be located on the bottom of the installed antenna.



### 3.3 Power Supply

The XYR 300G power supply is a switch-mode design which will accept either AC or DC supply. The module includes an integral battery charger for a backup battery.

The module accepts supply voltages in the following ranges :

12 – 24 volts AC RMS or 9 – 30 volts DC at the “supply” terminals, or

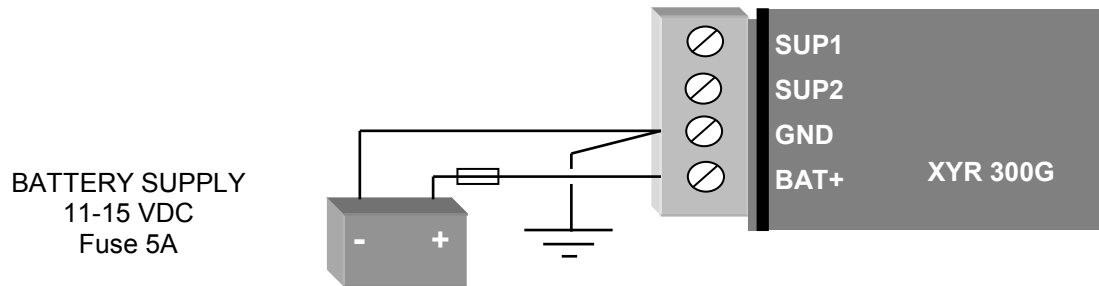
10.8 – 15 volts DC at the “battery” terminals.

The power supply should be rated at 1.5 Amps and be CSA Certified Class 2. For use in Class 1 Div 2 explosive areas (USA/Canada), the power supply must be approved for Class 1 Div 2 use.

**Note:** Connect module to the same ground/earth point as the antenna mounting to avoid differences in earth potential during voltage surges. The module needs an earth connection for the internal surge protection to be effective.

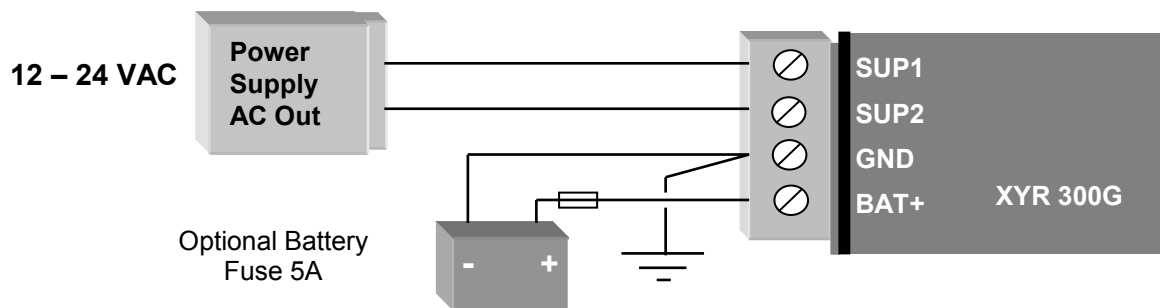
### 3.3.1 AC Supply

The AC supply is connected to the “SUP1” and “SUP2” terminals as shown below. The AC supply should be “floating” relative to earth.

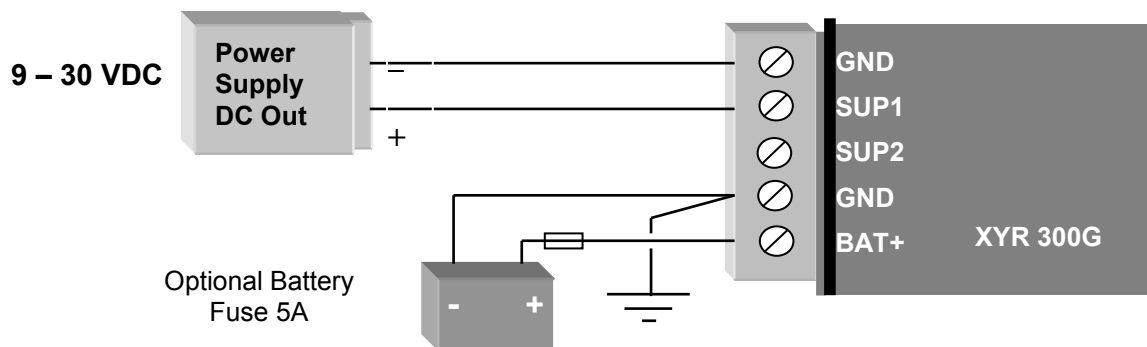


### 3.3.2 DC Supply

For DC supplies, the positive lead is connected to “SUP1” and the negative to “GND”. The positive side of the supply **must not be connected to earth**. The DC supply may be a floating supply or negatively grounded.



The module may also be powered from an external 11 – 15 VDC battery supply without the need for a “normal” supply connected to “SUP1”. This external battery supply is connected to “BAT+” and “GND” terminals. The positive lead of the external supply should be protected by a 5A fuse



.Upon failure of the normal supply, the module may continue to operate for several hours from a backup battery. The battery charger is designed for sealed or vented lead acid batteries between 5 and 24 amphours - **other types of batteries should not be used**. Typically, a 5 AHr battery will supply the XYR 300G for 1 – 2 days, depending on the type of XYR 300G.

On return of normal supply, the unit will recharge the battery. The maximum output of the battery charger is 0.7A when the supply voltage is greater than 12V, and 0.3A for less than 12V.

The XYR 300G monitors the power supply and provides the following internal values, which can be mapped as I/O values:

- Power failure (I/O Reg 4309) - if the supply voltage drops below 8V, this status value is set on, and set off again when the voltage is more than 9V. For AC Supplies, this indicates low voltage at approximately 10 VAC, and the status is cleared when the supply voltage rises above approximately 12VAC
- Low battery voltage (I/O Reg 4308) - this status value is set on if the battery voltage drops to 11.3, and resets off when the battery voltage is more than 11.8V.
- Battery voltage value (I/O Reg 4310) - 8 – 40VDC corresponds to hex 4000 – hex C000.
- Supply voltage (I/O Reg 4311) - 8 – 40VDC corresponds to hex 4000 – hex C000.

### **3.3.3 Solar Supply**

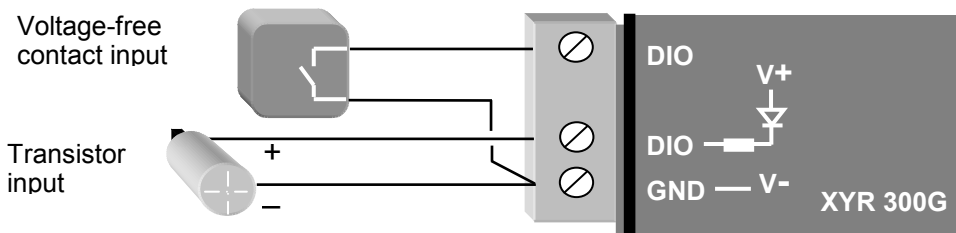
A XYR 300G can be powered from a solar supply using an external regulator. If a 12V solar supply is used, the 12V battery can be connected to the battery supply connections of the XYR 300G and the XYR 300G will monitor for low battery status and also battery voltage. If a 24V solar supply is used, the 24V battery should be connected as a DC supply (SUP1 and GND) - the supply voltage can be monitored however the “supply fail” voltage will activate too low to be used as a battery fail status.

## 3.4 Input / Output

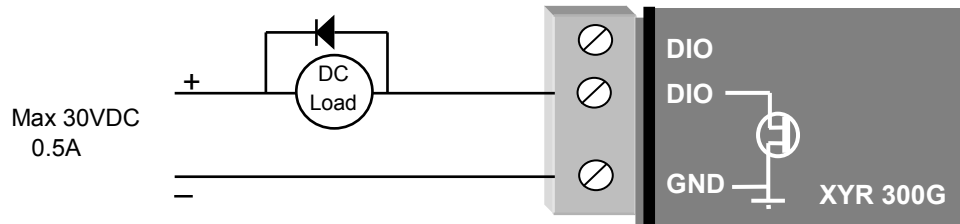
The XYR 300G has eight on-board discrete/digital I/O. These act as both discrete inputs and discrete outputs.

### 3.4.1 Digital Inputs / Outputs

All eight of the XYR 300G DIO terminals may be used as discrete inputs. These inputs are suitable for voltage free contacts (such as mechanical switches) or NPN transistor devices (such as electronic proximity switches). PNP transistor devices are not suitable. Contact wetting current of approximately 5mA is provided to maintain reliable operation of driving relays.



Each digital input is connected between the appropriate “DIO” terminal and common “COM”. Each digital input circuit includes a LED indicator which is lit when the digital input is active, that is, when the input circuit is closed. Provided the resistance of the switching device is less than 200 ohms, the device will be able to activate the digital input.



All eight of the XYR 300G DIO terminals may also be used as discrete outputs. The digital outputs are transistor switched DC signals, FET output to common rated at 30VDC 500 mA.

Digital outputs may be configured to individually turn off if no command message is received to that output for a certain period. This feature provides an intelligent watch dog for each output, so that a communications failure at a transmitting site causes the output to revert to a known state. See Chapter 4 **Configuration** for further details.

The output circuit is connected to the appropriate “DIO” terminal. Each digital output circuit includes a LED indicator which is lit when the digital output is active.

## 3.5 Serial Port

### 3.5.1 RS232 Serial Port

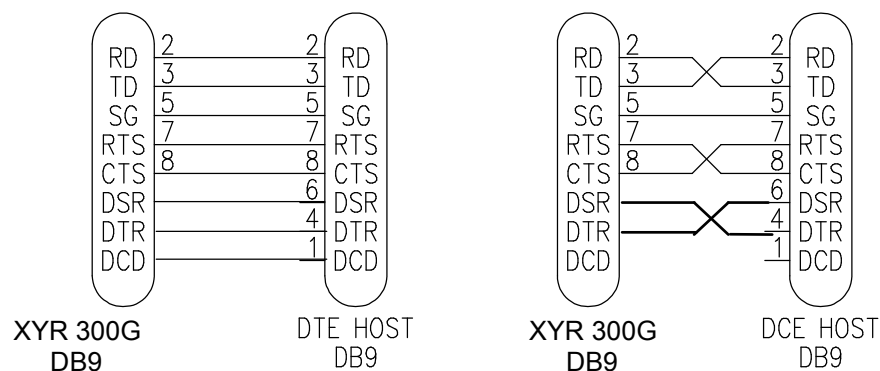
The serial port is a 9 pin DB9 female and provides for connection to a terminal or to a PC for configuration, field testing and for factory testing. It is also used by the Modbus/DF1 version for fieldbus connection.

This port is internally shared with the RS485 - ensure that the RS485 is disconnected before attempting to use the RS232 port. Communication is via standard RS232 signals. The XYR 300G is configured as DCE equipment with the pinout detailed below.

#### DB9 Connector Pinout:

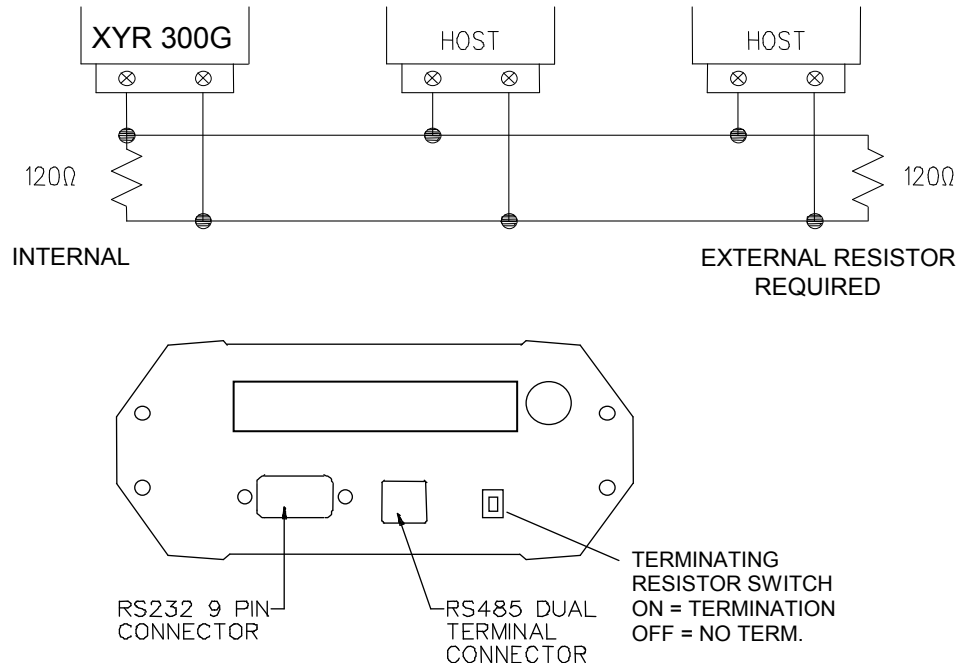
Pin	Name	Direction	Function
1	DCD	Out	Used for "active" signal.
2	RD	Out	Serial Data Output
3	TD	In	Serial Data Input
4	DTR	In	Data Terminal Ready - may be used by Host Protocol Driver
5	SG		Signal Ground
6	DSR	Out	Data Set Ready - always high when unit is powered on.
7	RTS	In	Request to Send - may be used by Host Protocol Driver
8	CTS	Out	Clear to send - may be used by Host Protocol Driver
9	RI		Ring indicate - not connected

Hardware handshaking using the CTS/RTS lines is provided, and are under the control of the Host Comms Driver. Example cable drawings for connection to a DTE host (a PC) or another DCE host are detailed below:



### 3.5.2 RS485 Serial Port

The RS485 port provides for communication between the XYR 300G unit and its host device using a multi-drop cable. Up to 32 devices may be connected in each multi-drop network. Note that the RS485 port is shared internally with the RS232 port - make sure that the RS232 port is disconnected before using the RS485 port.



RS485 is a balanced, differential standard but it is recommended that shielded, twisted pair cable be used to interconnect modules to reduce potential RFI. An RS485 network should be wired as indicated in the diagram below and terminated at each end of the network with a 120-ohm resistor. On-board 120 ohm resistors are provided and may be engaged by operating the single DIP switch in the end plate next to the RS485 terminals. The DIP switch should be in the “1” or “on” position to connect the resistor. If the module is not at one end of the RS485 cable, the switch should be off.

It is important to maintain the polarity of the two RS485 wires. On the XYR 300G, terminal A (the terminal on the right) is positive and terminal B is negative.

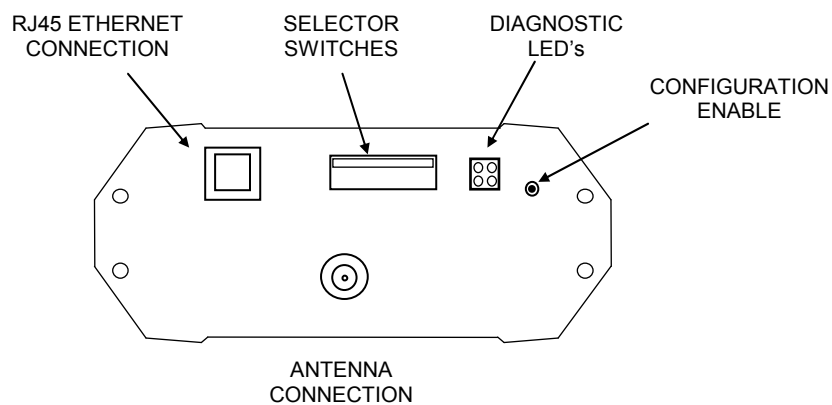


## 3.6 Ethernet Port

---

For XYR 300G-E modules only.

The Ethernet connection uses a standard RJ45 connector on the top end-plate of the module. The selector switches should all be “off” (in the diagram below, “off” is up).



# Chapter 4

# CONFIGURATION

## 4.1 Introduction

---

A Windows program is provided to configure the XYR 300X system. The configuration is done on a system basis - referred to as a “project” in the program. After the system configuration is entered, the configuration file can be loaded into each module via the RS232 port.

Each Project is configured with:

- A system address, which is common to every module in the same system, and is used to prevent "cross-talk" between modules in different systems. Separate networks with different system addresses may operate independently in the same area without affecting each other. The system address may be any number between 1 and 32 767. The actual value of the system address is not important, provided all modules in the same system have the same system address value. **A system address of zero should not be used.** The configuration program automatically offers a random number for the system address - you can change this to any number in the valid range but we recommend that you use the random number.
- A password for access protection. This is an optional feature. If selected, the project file can only be opened by entering the correct password.
- A security encryption key, used to encrypt and decrypt radio messages. This is an optional feature. If selected, the configuration program will offer a random security key, or this can be over-written with your own key. A key is a string of any 8 ASCII characters.

Each module in the project is configured with a unit address. Each module must have a unique unit address within the one system. A valid unit address for a XYR 300G is 1 to 95. A network may have up to 95 addresses communicating directly via radio (unit addresses 1 to 95).

XYR 300X I/O modules can have up to 31 modules communicating via RS485 depending on module type (unit addresses 96 to 127).

The configuration program may allocate more than one unit address to a XYR 300G if it is required because of the size of the system. If this is necessary, it will be done automatically by the configuration software.

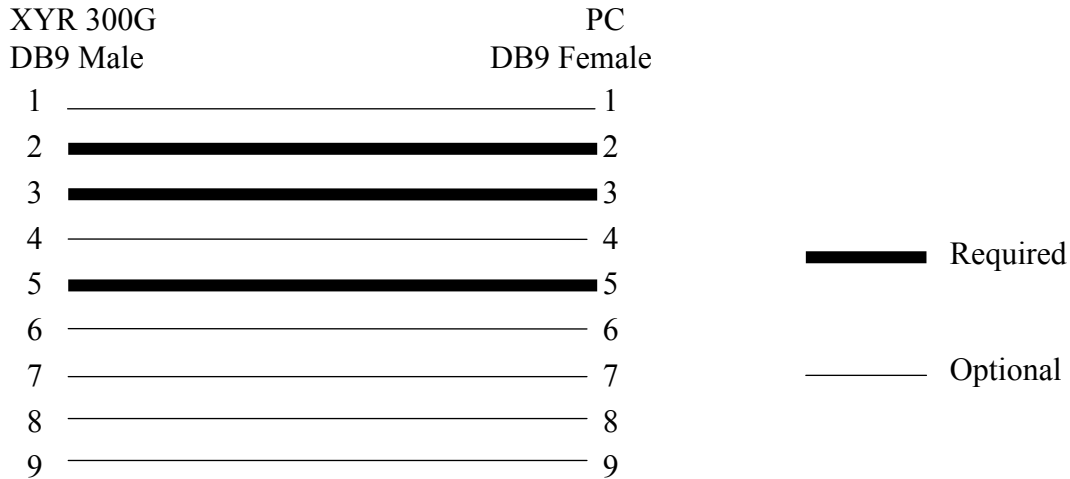
Configuration consists of:

1. Selecting the types of modules in the system and selecting address values
2. Linking (called “mapping”) I/O registers to remote I/O
3. Setting operating parameters such as change sensitivities and update times
4. Selecting “block mappings” - only for block transfer of I/O registers between XYR 300G modules
5. Selecting fieldbus addressing, and serial port configuration (Modbus & DF1 only)
6. Linking Radio Interface registers to Fieldbus Interface registers (All modules except MD1)

All of these steps must be performed to configure the XYR 300G module.

## 4.2 Configuration Program

The configuration software is available on a CD, and needs to be installed on your PC before you can use it. The CD contains a setup file called *setup.exe*. Select the configuration software window on the Product CD and an installation Wizard will guide you through the installation procedure. To upload and download configuration files to a module, you will need a RS-232 serial cable as shown below.

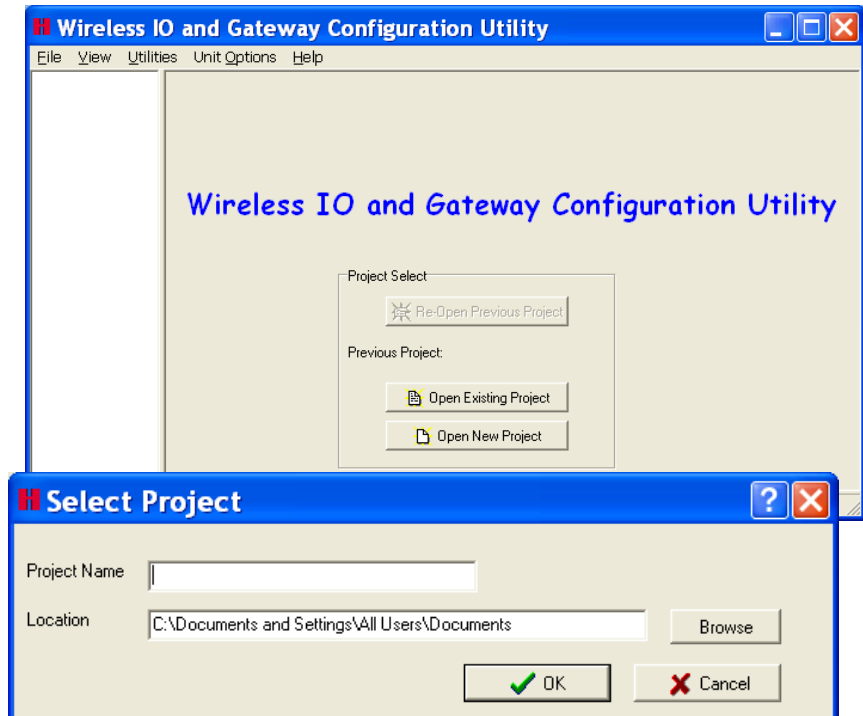


### 4.2.1 Program Operation

Start the software by either clicking on the start bar and navigating to the Configuration menu or by running the link in the directory selected in the setup stage.

The Initial screen will appear.

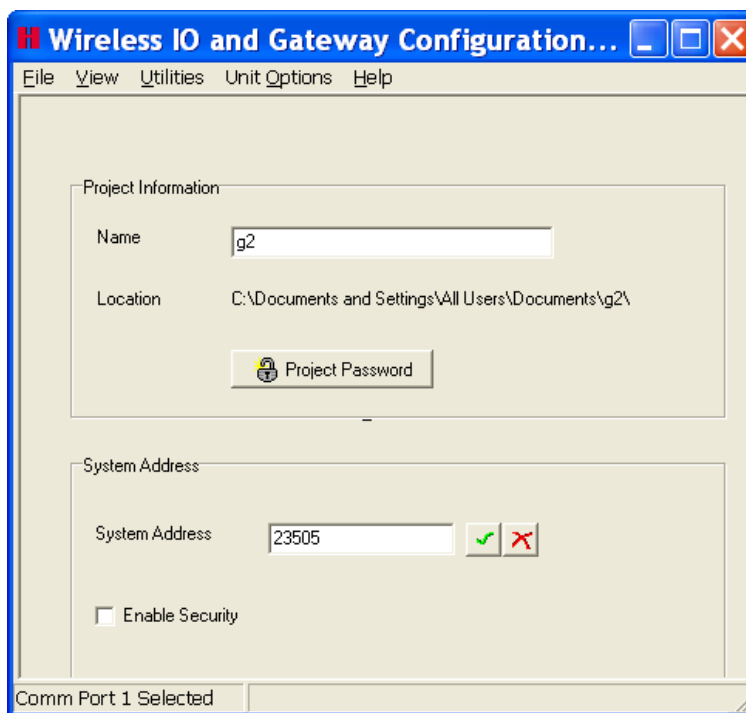
From the initial screen, you can select an existing project, or start a new project. The name of the project will create a new folder which will eventually contain the configuration files for the modules in this system. Project folders are located under the folder \Projects\ - for example, if you create a project called "Fire Pumps", then the files for this project will be found in the folder c:\.....\Projects\Fire Pumps\.



When you have selected the project, a screen will appear where you may enter the system address.

If you are editing an existing project, the system address will already have been entered. Do not change the system address unless you are going to re-program all of the modules in the system.

**Password.** You have the option of entering a password to protect the configuration files against unauthorized changes. When you open a new project, you will be asked to enter a password - if you do not enter any text - that is, press “ESC” or “Enter”, then password protection is disabled. If you do enter a password, then you will need to enter this password to access the project. Without the password, you are unable access the project



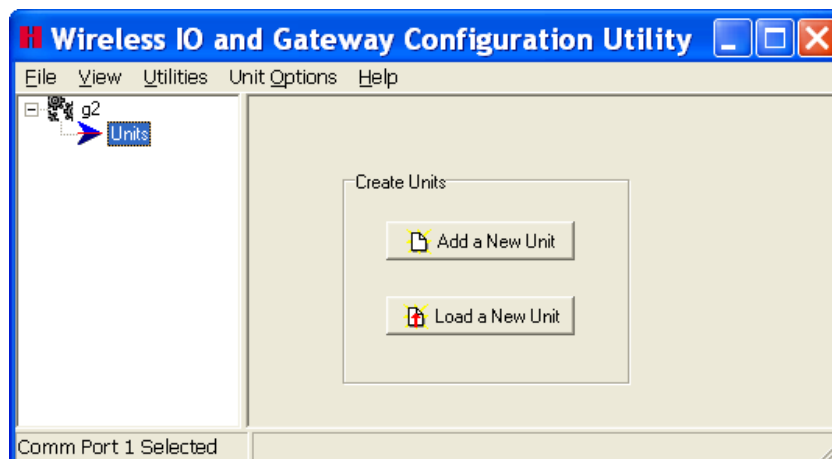
The password can be between 6 and 256 characters. You can also change password at any time by over-typing the password.

If you are starting a new project, you have the option of “Enabling Security”. This option enables encryption of the data sent over the radio. - please read Section 4.2.2 and the associated warnings before using this option.

To proceed with the configuration, double-click on the project name on the menu on the left side of the screen. “Units” will appear. You can now enter the types of units which will be used in the system. If you double-click on “Units” or select the “+” sign beside “Units”, then the modules that have already been created will be displayed.

### Loading configuration from an existing module

To load the configuration from a module, connect the module to the PC via the RS232 cable, put the module into “Configuration Mode” by pressing the configuration button on the top end-plate, and click on “Load Unit”. This will allow you to view the module configuration, change it, or copy it for



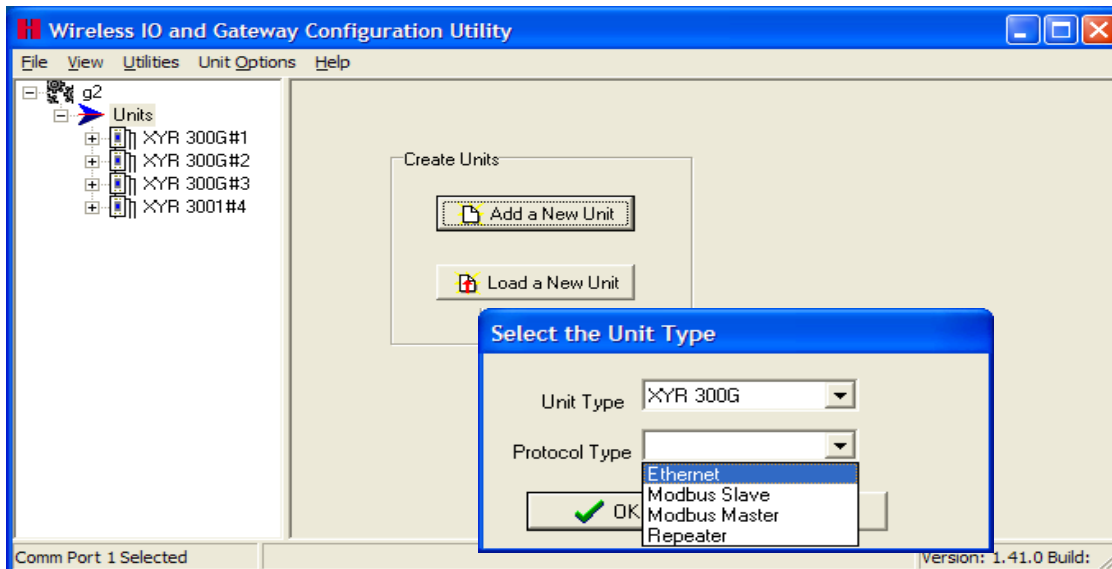
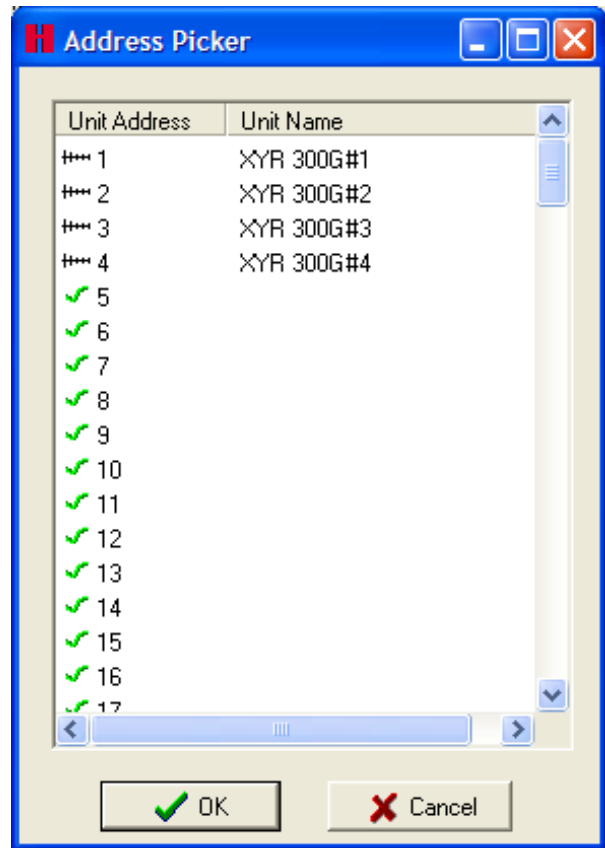
another module - refer to section 4.3 for full details.

### Adding a new module to the system configuration

To add a new module to the system configuration, click on “Units” on the left-hand menu and then “Add Unit”. Select the type of module from the list. For XYR 300G modules, you will be asked to select the bus protocol. This must match the XYR 300G module type you have installed.

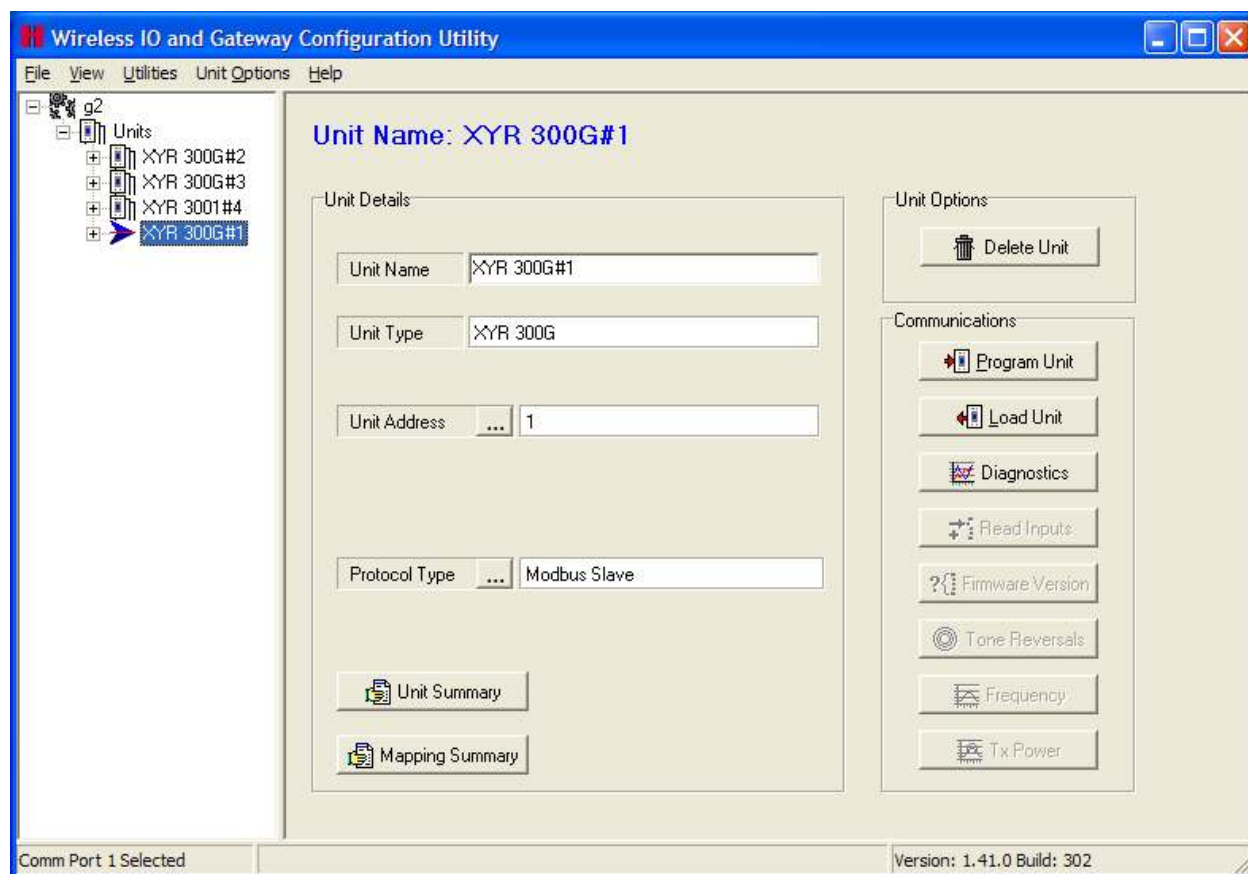
You have the option of selecting a unit address for the module, or allowing the program to select one automatically. If you choose to select the unit address the program will display the list of available addresses for you to select - valid addresses are 1 – 95.

The default name for a unit will include the unit address. For example, “XYR 300G #10” is a XYR 300G module with unit address 10. You can change the name of a unit - for example, you could replace the default name with “Pump Station 14”.



## Deleting a Unit

A module can be deleted from the configuration by highlighting the unit and selecting “Delete Unit”.



## 4.2.2 Security

There are two security features available. You can enter a password to protect the configuration files, and you can enable security encryption of the radio transmissions.

The password can be between 6 and 256 characters. The password is case sensitive and any ASCII characters can be used. If you have entered a password, then this password will need to be entered whenever the configuration is changed. You are able to change the password from the “Utilities” menu. If unauthorized access to the files is a concern, we recommend that you change the password regularly or whenever there is a change of staff.

Data Encryption is an additional level of security. The security option uses a 64 bit security key to provide data encryption of the radio messages. All modules in the same system will be configured with the same security key used to encrypt and decrypt the messages. This feature is available for modules with firmware version 2.1 and higher. If you are adding modules to an old system which does not have the security encryption feature, then you cannot use security encryption on the new modules.

Note that the security key is different than the password.

- To enable the security encryption, select the “Enable Security” box on the project display. An 8-character random security key is automatically generated. If desired, a different security key may be entered and you will be prompted to enter the security code a second time to

confirm. The security key can be any characters or numbers. Characters are case sensitive. The security key will never be displayed.

- If you do not enable security, there will be no data encryption of the radio messages. This is the default setting.
- If a security key has been entered, this key is downloaded into each module as part of the configuration download process. You can download another configuration at any time - if the security key is different, or if there is no security key in the new configuration, the old key will be over-written.
- You can change the security key in the configuration files simply by entering a new security key in the security key window. You will be prompted to confirm the new security key. Note that if you change the security key, it will not match the security key previously loaded into existing modules.
- If you want to change a configuration, we recommend that you change the archived configuration, and then download the configuration onto the module. The archived configuration already has the valid security key.
- If you lose the archived configuration, you can upload the configuration from a module, but you cannot upload a security key. That is, you can upload the module configuration, view it, change it - but if you don't know the original security key, the old key will be over-written when you download the new configuration. This module will no longer communicate with other modules in the system as the security key is different.

#### **Warning!!**

**These security options provide a high level of security, but no data-security system can provide “100% protection”. But it does make it very difficult for someone to interfere with the XYR 300G system - difficult to the point where there would be many easier alternate ways to cause malicious damage.**

**The password must be kept in a secure place. Security procedures need to be adopted. If staff with access to the password leave your organization, we recommend that the password be changed.**

**We recommend that you use a random 8-character string for the security key and that you do not record the key. It is not necessary to know what the security key is. The key will be recorded in the archived configuration files, and therefore the configuration files should be held in a secure place and backed up.**

**The security key does not prevent a hacker uploading a configuration from a module and downloading with a new security key. This module will no longer operate with other modules in the system. To prevent this, unauthorized access to modules must be prevented.**

The security options provide security against a “hacker” in the following way:

- ❑ A hacker cannot listen-in to radio messages without the security key to decrypt the radio messages. Similarly, a hacker cannot force outputs by transmitting a radio message to a module without the security key.

- ❑ A hacker cannot access the security key from an installed module or from the configuration files.
- ❑ The archived configuration files cannot be changed, downloaded or uploaded without the password.

If you lose the configuration files, you can regenerate these by uploading the configuration from every module in the system into a new project with a new security key. After uploading each module, download the configuration with the new security key.

If you wish to change the security key, simply enter a new key in the configuration program, and download the new configuration to all modules in the system.

**Note on Ethernet XYR 300G.** You are able to access the module configuration of an Ethernet XYR 300G via the Ethernet port. To prevent this access, do not select “Enable Ethernet Debug” on the Ethernet configuration display - see section 4.8.

## 4.3 Uploading and Downloading

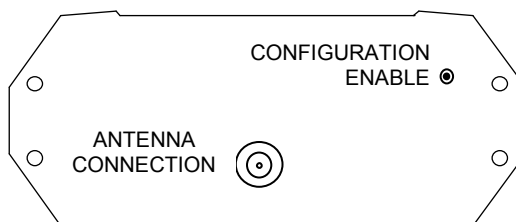
To upload or download a configuration file, the XYR 300G must be connected to the PC via a RS232 cable. For Modbus units, the host device must be disconnected, even if it is connected to the RS485 port. Other units do not need to disconnect the data bus. When the PC is connected, put the XYR 300G into configuration mode by pressing the small pushbutton switch in the end plate of the module for 5 seconds, until the ACT LED starts flashing.

In configuration mode, the XYR 300G will stop its normal functions.

Make sure the correct communications port is selected on the PC - if necessary, change the selection from the Utilities menu.

Connect the PC to the module using the configuration cable.

The configuration may be programmed into a XYR 300G, or a configuration may be loaded from a XYR 300G. After programming or loading is complete, disconnect the PC from the XYR 300G. Reset the XYR 300G by removing power and



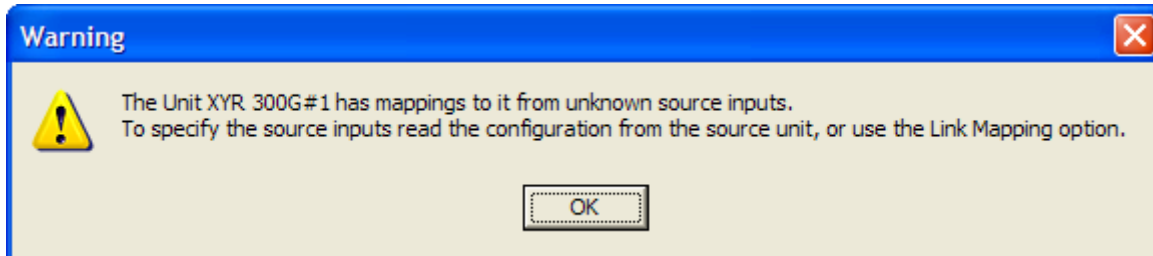
XYR 300X DB9 Male		PC End DB9 Female	
1	_____	1	
2	_____	2	
3	_____	3	_____
4	_____	4	Required
5	_____	5	
6	_____	6	_____
7	_____	7	Optional
8	_____	8	
9	_____	9	



re-connecting power. The XYR 300G will start up normally and the OK led will be on. The serial port will have its original set-up.

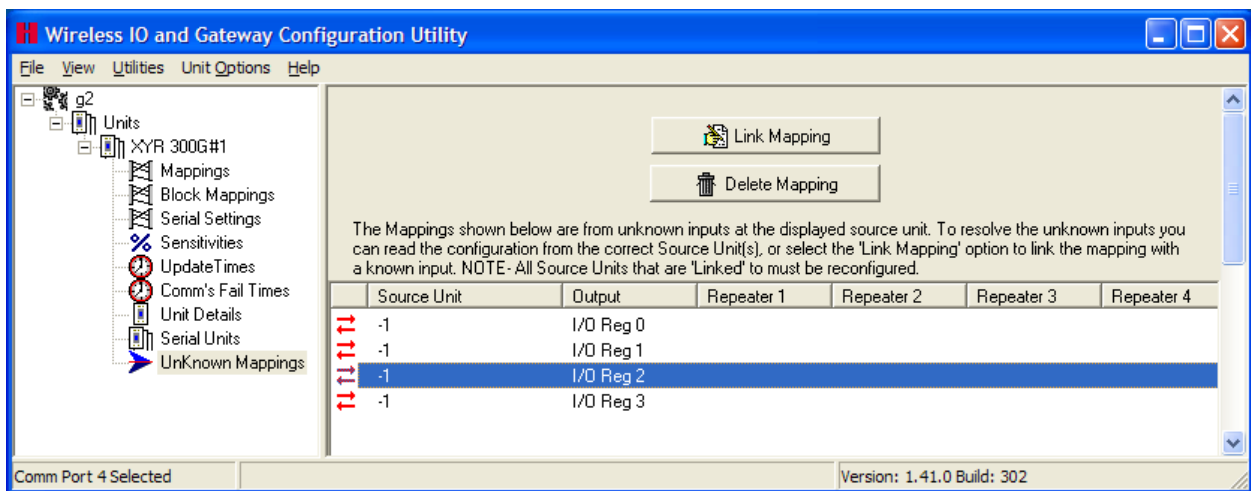
#### 4.3.1 Loading from a XYR 300G

If you load a configuration from a XYR 300G into a “blank” or new project, then the program will not be able to display the mappings from remote modules (as the program does not know what the remote modules are). You will get a warning message like this:



If you open the archived project first, and load into the archived project, then all mappings will display as normal - any mappings to/from the XYR 300G will be over-written on the PC display by the loading process.

If you are unable to load into the archived project, then mappings to remote modules will be displayed, but mappings from remote inputs will be shown as “Unknown Mappings”.



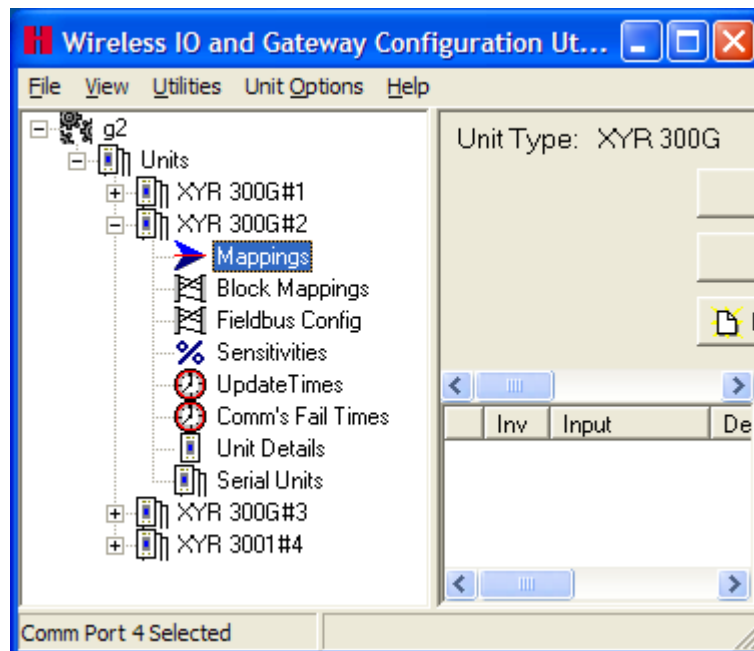
If you also load the configurations from the other remote modules in the system, then these unknown mappings will disappear as the program can determine where the remote inputs are. Alternately, you can select “Link Mapping” and manually enter the remote inputs.

## 4.4 Mappings XYR 300G to XYR 300X I/O Modules

To transfer remote input signals to a XYR 300G, or transfer a value to a remote output from a XYR 300G, you set up “I/O mappings”. You enter mappings into the source unit, not the destination unit. That is, you configure a mapping at the “input” module. If you want to transfer an input signal at a XYR 300X module to a XYR 300G register, you enter a mapping at the XYR 300X I/O module. If you want to transfer a XYR 300G register to an output signal at a XYR 300X module, you enter a mapping at the XYR 300G module.

To configure mappings, double-click on the module in the left-hand menu - the menu will expand with selections for that module. Select “Mappings”.

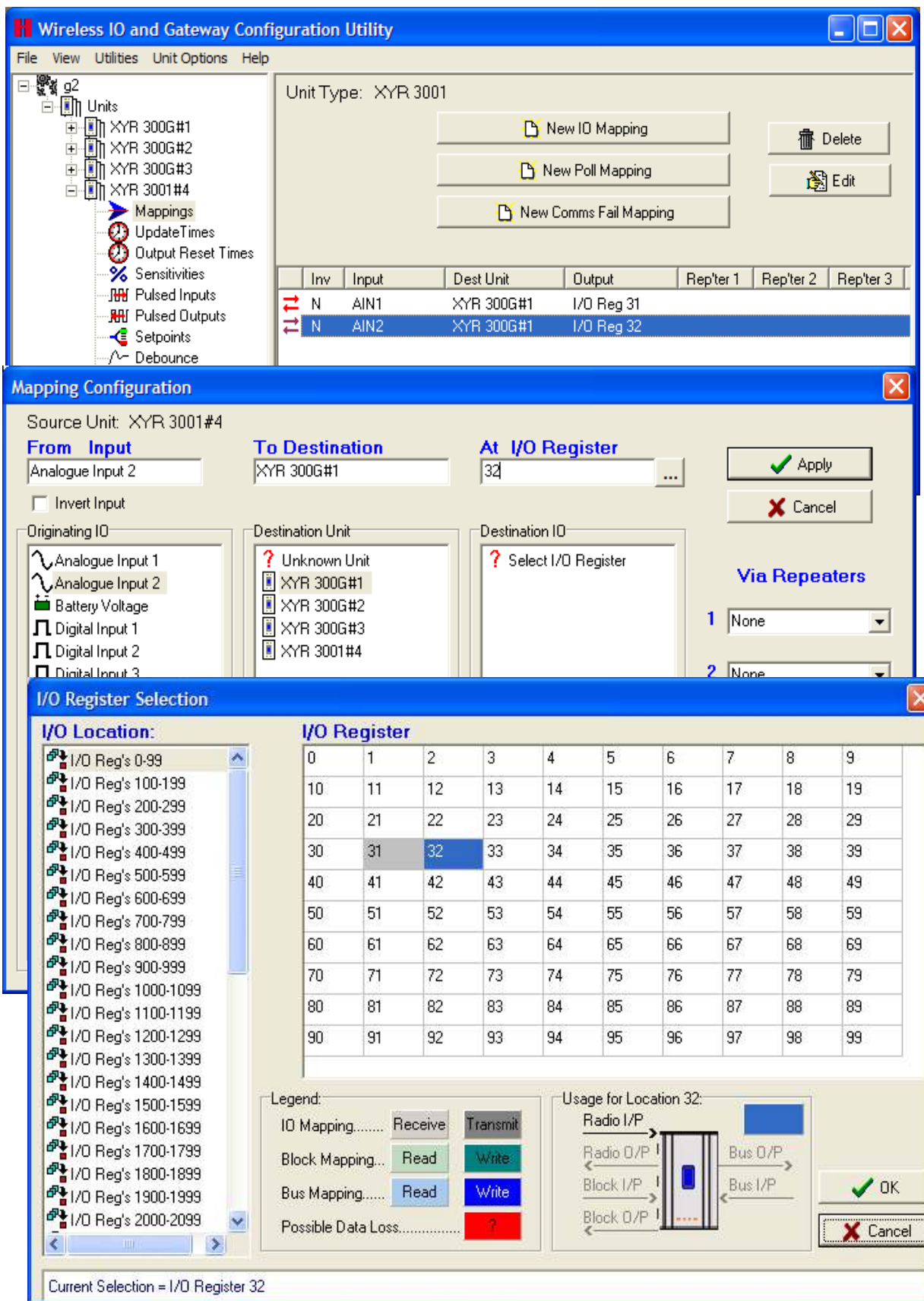
Each mapping comprises only one I/O point. “Block Mappings” provide more advanced communications between XYR 300G modules.



### 4.4.1 Mappings from Inputs at Remote XYR 300X I/O Modules

Refer to the XYR 300X I/O User Manual.

When mapping inputs to a XYR 300G, you will be asked to select an I/O Register. Select the “...” box beside the “At I/O Register” heading - this will allow you to select the I/O register between 0 and 4299.

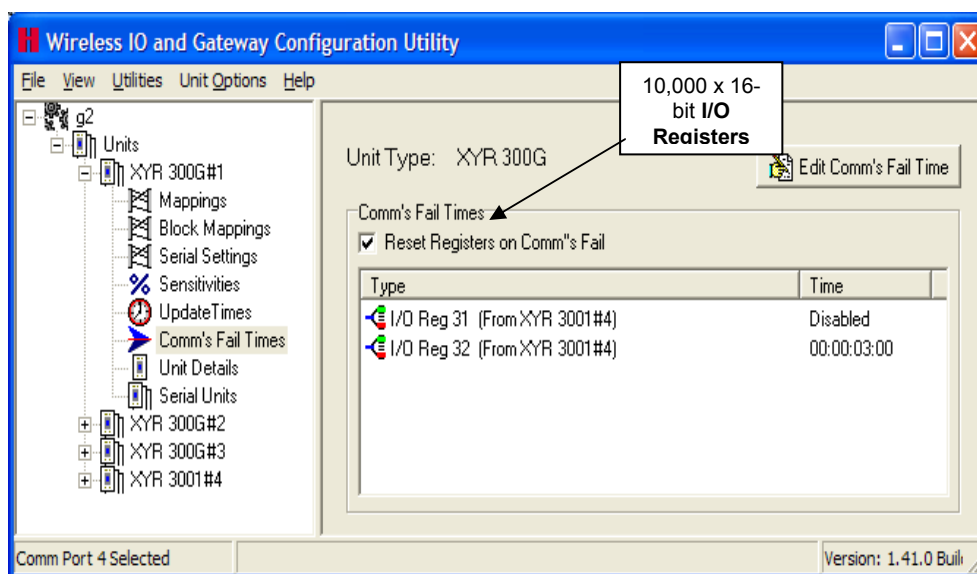


Any I/O registers that have already been selected will have a color shading.

The update times, analog sensitivities for these mappings can be set as per normal I/O mappings.

To map several inputs to consecutive I/O registers, use “Shift”-select or “Ctrl” - select to highlight the inputs, and select the first I/O register in the range. The selected mappings will be entered with consecutive I/O registers.

For each “remote input” configured to a XYR 300G, there is a comms-fail time parameter in the XYR 300G. If the XYR 300G does not receive a message destined to that I/O register within the “comms fail” time, then the “comms fail” status for that I/O register will be set - the most significant bit of the status register will be set to 1. The comms fail time should be more than the corresponding update time at the remote input.



To set the comms fail times, select the XYR 300G, and select the “Comms Fail Time” option. Each remote input already mapped to the XYR 300G will automatically be listed, including the remote module containing the mapping.

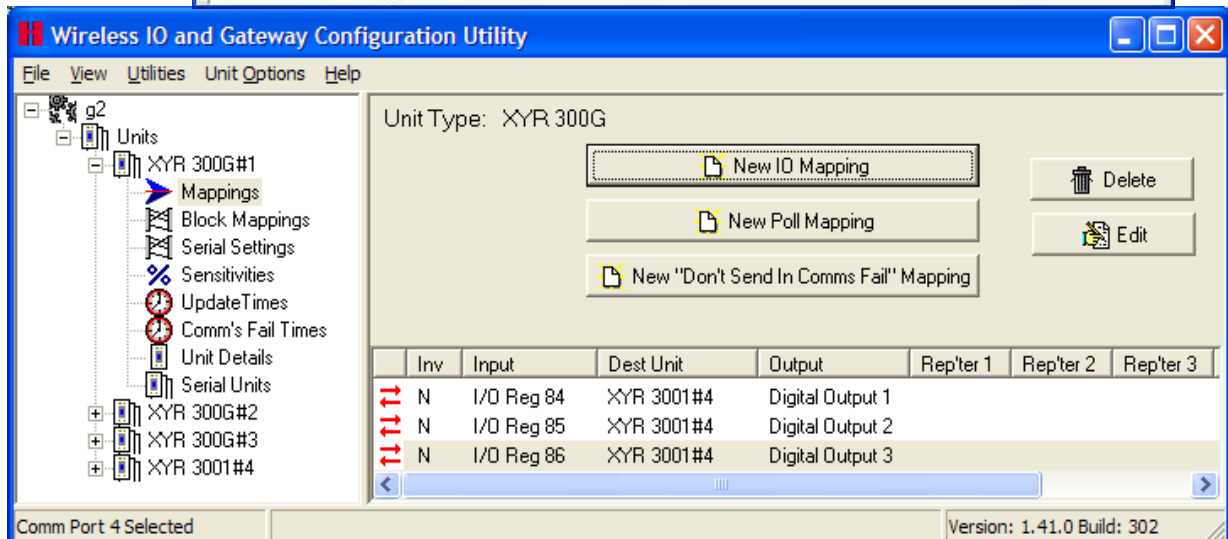
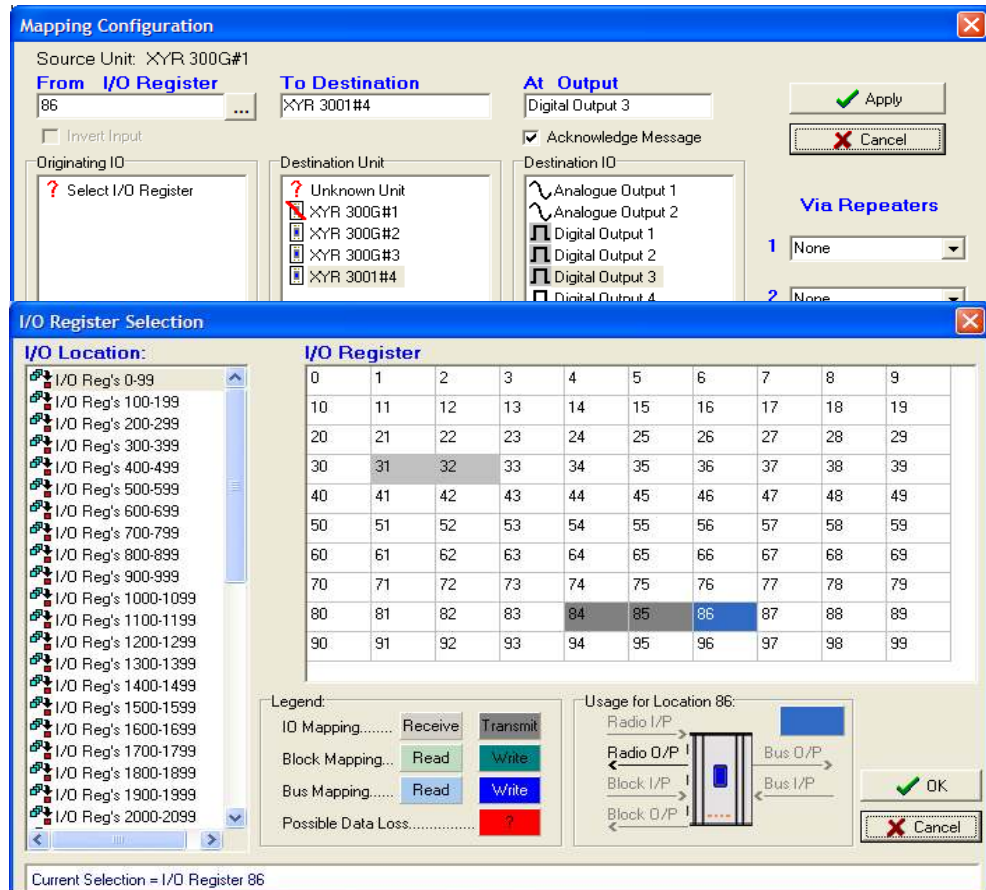
The default value for the comms-fail time is “disabled” or zero. To enter a time, select the I/O register from the list. The comms-fail time should be greater than the update time of the remote input.

The I/O value in the I/O registers can be reset to zero on comms-fail. To enable this, select the enable box in the “Comms Fail Times” configuration screen. Note that only the register that is in comms fail will be reset.

#### 4.4.2 Mappings from XYR 300G to Outputs at Remote XYR 300X I/O Modules

Mappings can be entered in the XYR 300G to remote outputs. Select the “Mappings” option under the XYR 300G. Select an I/O register and select the remote module and the output channel.

To map several consecutive I/O registers to several outputs, select the first I/O register in the range and use “Shift”-select or “Ctrl” - select to highlight the multiple outputs. The selected mappings will be entered with consecutive I/O registers.



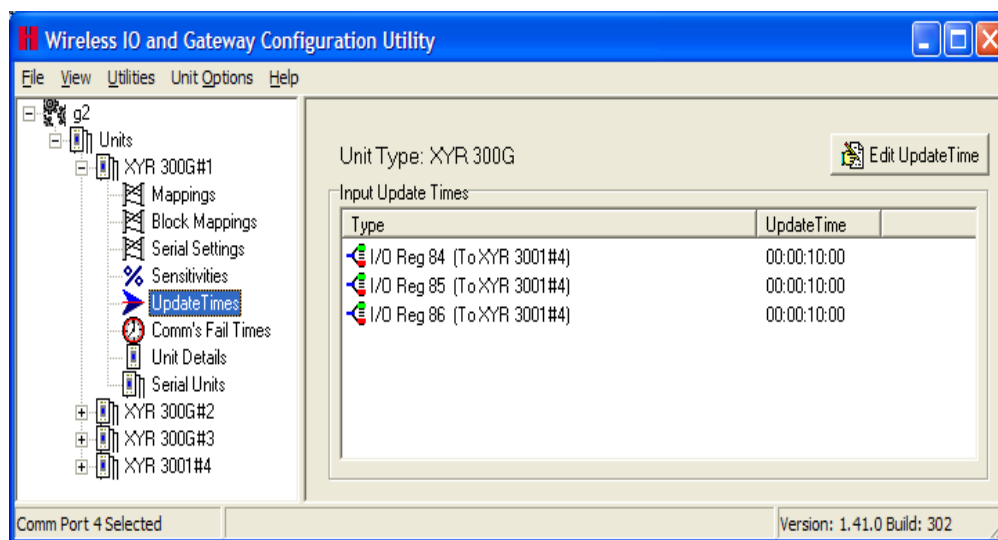
## Change Sensitivities

Radio messages to remote modules can be change messages (when the value of the I/O register changes) or update messages (when the update time has elapsed). If a change message is sent, the update period restarts.

You can configure the amount of change required to trigger a change message - this is called the change sensitivity. Sensitivities are configured for blocks of I/O registers - that is, each I/O register does not have a unique sensitivity. You can configure up to 50 sensitivity values - that is, there can be 50 blocks of registers with different sensitivities.

For more information on this, refer to section 4.6.

## Update Times



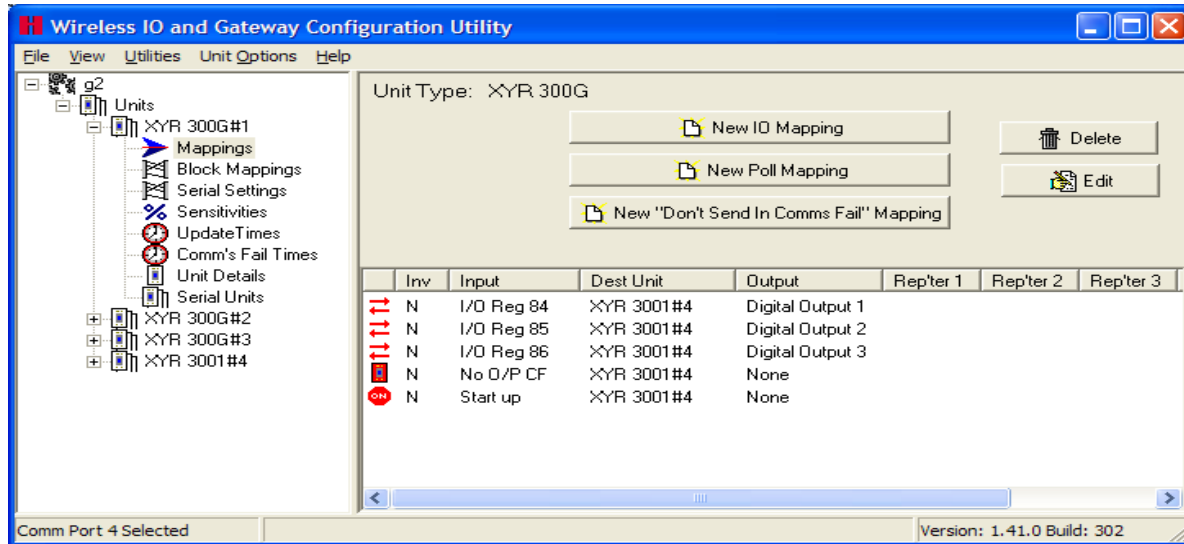
To change the update times of output mappings, select the Update Times option. Any I/O registers that have already been mapped to remote outputs will automatically be listed. The default update time is 10 minutes.

## Changing Multiple Settings

You can change the Comms Fail Times or Update Times of several I/O points simultaneously by using the <Shift> Select feature. For example, if you want to change all times to 1 minute, you could change each individually, or you could “block” all entries using the “Shift” Select feature and select “Edit”. You only need to enter the change once to change all of the inputs selected. This feature is also available with the other configurable parameters.

### 4.4.3 Don't Send if in Comm Fail

You can configure a special “Don’t Send if in Comms Fail” mapping. If this is configured for a particular remote module, the XYR 300G will not transmit output messages to this remote address, if there is a communications failure status on any input or output configured for the same remote address. Output messages will re-start when a message is received from the remote module. The use of this option can prevent the radio channel becoming congested if there are many outputs at that module.



To configure this special mapping, select the “New Don’t Send in Comms Fail Mapping” box. You will be asked to select which remote module this function applies to. You can enter more than one of these mappings if there are more than one modules.

### 4.4.4 Startup Polls

You can enter start up polls for remote modules by using the “New Poll Mapping” box. This function is the same as for the XYR 300X I/O modules. A start-up poll is a special message sent when the XYR 300G starts up. When the remote module receives a start-up poll, it will immediately respond with update messages for all its inputs that are mapped to the XYR 300G. This allows the XYR 300G to have correct values on start-up.

### 4.4.5 Polls to Remote Modules

It is possible for a XYR 300G to send a poll to a remote module at other times apart from start-up. A poll can be sent under the following events:

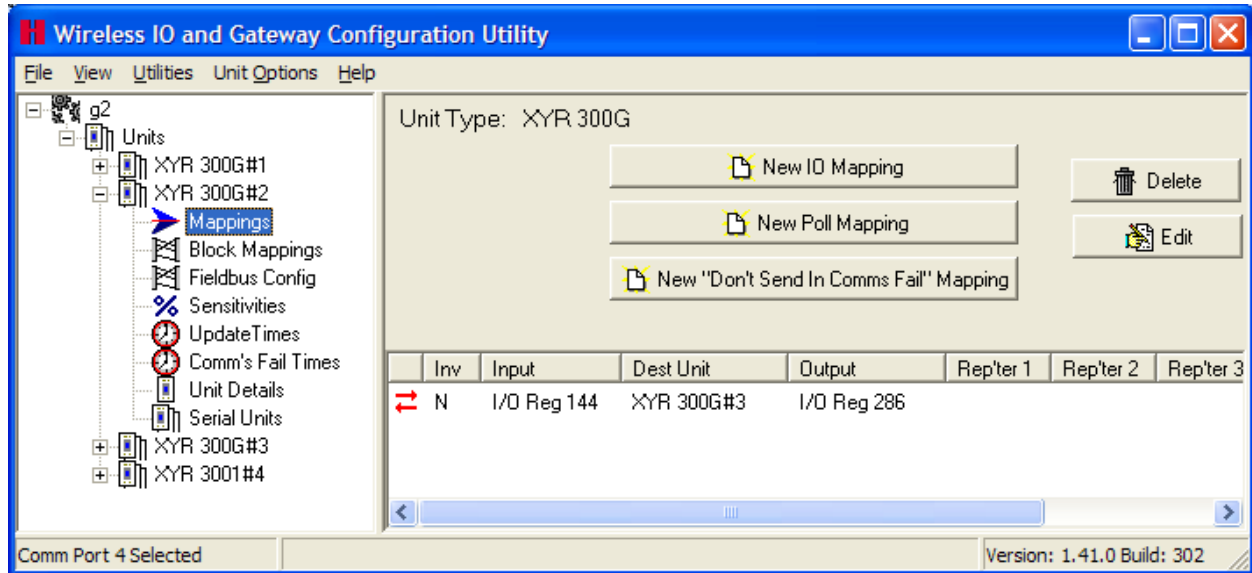
- based on a configurable time period
- based on real time clock
- on-demand by the host device.

For information on this configuration, refer to the next section on “Block Mappings”.



## 4.5 Mappings from XYR 300G to other XYR 300G Modules

Individual links between XYR 300G modules can be configured under the “Mappings” selection as described in the previous section. For example, if you want to transfer I/O Reg 144 in XYR 300G#2 to I/O Reg 286 in XYR 300G#3, you can enter the following mapping:



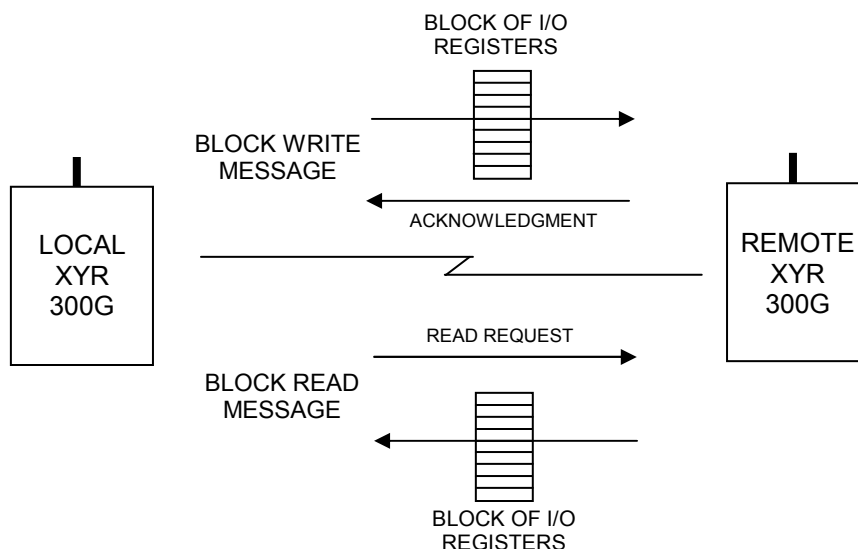
Whenever I/O Reg 144 changed by the sensitivity amount, XYR 300G#2 would send a message to XYR 300G#3 to write the value in I/O Reg 286. The problem arises if there are a lot of these mappings. Each radio message only relates to one register-register link. If you want to map 1000 registers from one XYR 300G to another, then this could generate a lot of radio messages.

To get around this problem, it is possible to configure “block mappings”. With a block mapping, multiple registers (a “block of registers”) can be transferred together in the one radio message. This improves the efficiency of the radio communications.

### Read/Write Mappings

The mappings can be “read” or “write” mappings. A Read mapping is a request sent to another XYR 300G to return a block of values. A Write mapping is a message sending a block of values to another XYR 300G. A Read mapping from XYR 300G#2 to XYR 300G #3 could be the same as a Write mapping from XYR 300G#3 to XYR 300G#2 (that is, in the reverse direction) - except the Read mapping is initiated from #2 and the Write mapping is initiated from #3.





### Word/Bit Mappings

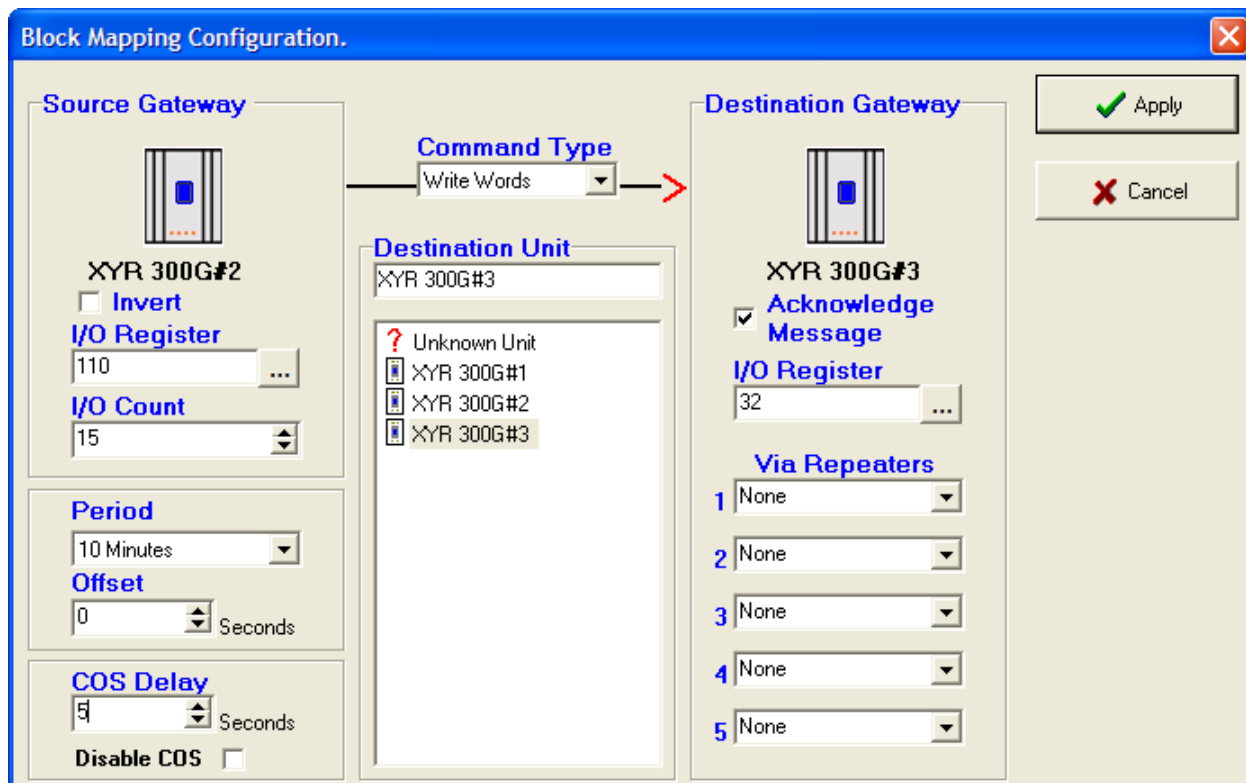
Read and Write mappings are also selected as Word or Bit mappings - that is, you can select a Read Word mapping or a Read Bit mapping and you can select a Write Word mapping or a Write Bit mapping. “Word” refers to a complete 16-bit register value; “Bit” refers to the value of the most significant bit of a register - this bit is the “binary value” or “digital value” of the register.

If you use a Word block mapping of 50 registers, you are transferring a block of 50 x 16-bit values. If you use a Bit block mapping of 50 registers, you are only transferring the digital value of each register - that is 50 x 1 bit values. This is a lot more efficient for a radio message, but bit mappings are only suitable for discrete or digital I/O. A Bit mapping will convert the 16-bit register to a single bit, transfer it and store the bit value in the most significant bit of the destination register.

**Note:** The maximum block size for each block mapping is 64 registers.

### 4.5.1 Entering a Block Mapping

Select the “source” XYR 300G on the left hand menu - select “Block Mappings” and then “New Block Mapping” from the right-hand display. The Block Mapping Configuration display will appear.



The image shows a 'Block Mapping Configuration' dialog box with a blue title bar and a close button (X) in the top right corner. The dialog is divided into several sections:

- Source Gateway:** Contains a module icon, the label 'XYR 300G#2', an 'Invert' checkbox, an 'I/O Register' field with the value '110' and a browse button (...), an 'I/O Count' field with the value '15' and a spin button, a 'Period' dropdown set to '10 Minutes', an 'Offset' field with the value '0' and a spin button with a 'Seconds' label, a 'COS Delay' field with the value '5' and a spin button with a 'Seconds' label, and a 'Disable COS' checkbox.
- Command Type:** A dropdown menu showing 'Write Words' with a red arrow pointing to the right.
- Destination Unit:** A list box containing 'XYR 300G#3' and a list of available units: 'Unknown Unit', 'XYR 300G#1', 'XYR 300G#2', and 'XYR 300G#3'.
- Destination Gateway:** Contains a module icon, the label 'XYR 300G#3', an 'Acknowledge Message' checkbox (checked), an 'I/O Register' field with the value '32' and a browse button (...), and five 'Via Repeaters' dropdown menus, all set to 'None'.
- Buttons:** 'Apply' (with a green checkmark) and 'Cancel' (with a red X) buttons are located on the right side.

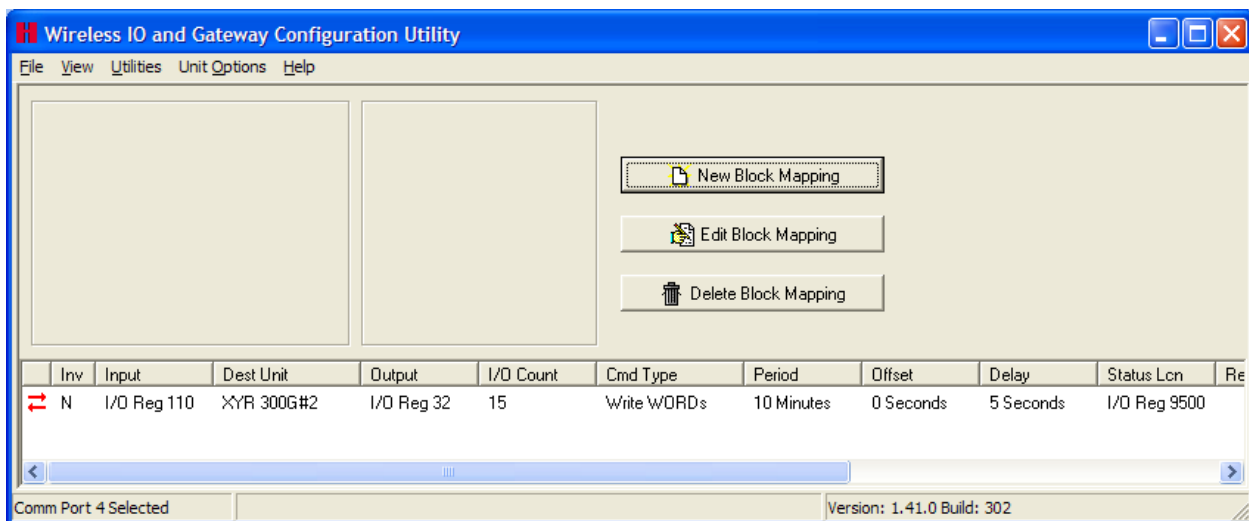
Select the “Command Type” from the pop-down window in the centre of the display. The red arrow will confirm the direction of the block transfer. Now select the destination module - only the XYR 300G modules already configured will be shown. If you need to use repeaters in the radio link, enter the repeater addresses, starting with the repeater closest to the source module.

Under “Source Gateway”, enter the I/O Register and I/O Count. The I/O Register is the first register in the block and the I/O Count is the number of registers - in the above example, the block of registers will be 110 – 124 (15 registers starting at I/O Reg 110).

If you are entering a Write mapping, then the values in this block will be sent to another XYR 300G. If it is a Read mapping, then values from another XYR 300G will be sent to this block.

Under “Destination Gateway”, enter the I/O Register - this is the first register in the block. You do not need to enter the block size as this will always be the same as the block size in the source XYR 300G. In the above example, the destination block will be I/O registers 32 – 46 (15 registers starting at register 32). So, in the above example, a block of 15 x 16-bit values will be written from I/O Reg 110 – 124 in XYR 300G#1 to I/O Reg 32 – 46 in XYR 300G#2.

Each mapping entered is allocated a status register - the register number appears on the right hand of the Block Mapping display. These registers store relevant status information about the block mapping - the structure of the Block Mapping status registers is shown in Appendix 1.



In the above example, the status register for the block mapping has been automatically assigned to register 9500.

The rest of the mapping configuration involves the mapping trigger - or what initiates the mapping message.

Block write mapping's have option to invert the I/O message that will be sent. This can be selected when adding a new Write block mapping.

### Mapping "Triggers"

A block mapping can be "triggered" or initiated by several different methods.

- By the host device writing to a "trigger register" in the source XYR 300G - the whole block mapping message is sent each time the host device writes to the trigger register.
- By configuring a time period - the XYR 300G will send the whole block mapping message if this time period has elapsed since the last message has been sent.
- By configuring a real-time clock - the XYR 300G will send the whole block mapping message at the configured times.
- By a change-of-state within the I/O block. This can only occur for Write mappings. If a value in the block changes by more than the sensitivity amount, then only this I/O that has changed in the block message will be sent. You can enter a delay period such that the message is sent after the delay period. All I/O that has not changed will not be sent until the real time clock, time period or host device trigger has been set.

Combinations of the above triggers can occur - for example, the block mapping message will be sent if a change-of-state occurs, AND at the configured real-time, AND when the host device writes to the trigger register.

#### 4.5.2 Host Device Trigger

Each block mapping that is configured is allocated a status register in the range 9500 – 9999 (i.e. one status register for a maximum of 500 possible block mappings). The status register for a given block mapping is shown on the right hand side of the Block Mapping display (under the heading "Status Lcn"). Bit 13 of the associated status register is the "Force bit" - if Bit 13 is turned "on", then the associated mapping is forced, or triggered. Depending on the module

version, a particular algorithm may apply to the setting of the force bit. This algorithm and details of the block status registers are given in Appendix 1.

### **4.5.3 Time Period**

On the Block Mapping display, there are two configuration windows - “Period” and “Offset” - these determine the time period trigger and real-time trigger.

For a time-period trigger, select “Continuous” in the “Period” pop-down window. Under “Offset” enter the time-period in seconds. In the above example, the whole block mapping will be sent every 300 seconds or 5 minutes.

The “Offset” value can be set from 0 – 4095 seconds (68 minutes). If you do not want the message to be sent on a time period, set the “Offset” value to zero.

If you want the block mapping to be sent only on time period (and not on change as well), select the “Disable” box in the bottom left hand corner - this disables change messages for this block mapping. If you want any changes sent within this Time period uncheck Disable box and enter in time to wait before sending only the I/O that has changed in the Block.

### **4.5.4 Real-Time**

The block mapping message can be sent at a real-time by setting the “Period” value. In this example, “period” is set to 6 minutes - the message will be sent every 6 minutes starting at the beginning of each hour. That is, the message will be sent at XX:00, XX:06, XX:12, XX:18, XX:24 .... XX:54 - where XX represents any hour of the day.

If “Period” was set to 1 minute, then the message would be sent every minute, on the minute.

The “Offset” value provides an offset to the specified time. In this example, if the “Offset” was set to 10 seconds, then the messages will be sent 10 seconds later - at XX:00:10, XX:06:10, XX:12:10 etc.

The reason for the offset is to stagger messages with the same time setting. For example, if you configure 5 block mappings all to be sent at 10 minutes, then the XYR 300G will try to send these messages at the same time - some of the messages will have to wait until the earlier messages have been sent. If you are sending Read messages as well as Write messages, then the return messages could clash with outgoing messages.

To avoid this, you can delay some messages using the Offset feature. For example, if you have 5 mappings to be sent at 10 minutes, then the first could have zero offset, the second 3 sec offset, the third 6 sec offset etc.

If you do not wish to have a real-time trigger, set “Period” to continuous.

If you want the block mapping to be sent only on real-time (and not on change as well), select the “Disable” box in the bottom left hand corner - this disables change messages for this block mapping.

### **Setting the Clock**

The clock within the XYR 300G can be set by the host device, and read by the host device. The XYR 300G provides four clock registers for days/hours/minutes/seconds - the registers are 4330 – 4333. On power-up, these registers are set to zero. Reg 4333 increments each second, Reg 4332 increments each minute, Reg 4331 each hour and Reg 4330 each day.

The clock registers are used by the XYR 300G for the real-time-clock trigger. The host device can read these registers. The host device can also set the XYR 300G clock at any time by writing to the appropriate *Set* register. The *Set* registers are : 4340 – 4343. The procedure for setting the real time clock via these registers depends on the module firmware version (to find out what firmware version the module contains, simply display the diagnostics menu – see section on diagnostics). The *set registers* can also be set via radio using appropriate I/O or block mappings.

Item	Clock Location	Set Location
Days	4330	4340
Hours	4331	4341
Minutes	4332	4342
Seconds	4333	4343

Registers 4340 – 4343 will only be transferred to the corresponding clock registers when their value changes from 0. For example to write a value of 7 to the hours register, first write the value 0 to the *Set hours* register 4341, then write the value 7 to the same register. (i.e. by always first writing the value 0 to the *Set* register this ensures that the change-of-state from 0 will be detected). Values must be held (i.e. not change) for approx 200msec to be detected.

#### 4.5.5 Change-of-State

The image shows a 'Block Mapping Configuration' dialog box. It is divided into three main sections: Source Gateway, Destination Gateway, and a central Command Type/Unit section.

- Source Gateway:**
  - Icon: XYR 300G#2
  - ☐ Invert
  - I/O Register: 110
  - I/O Count: 15
  - Period: Continuous
  - Offset: 300 Seconds
  - COS Delay: 20 Seconds
  - ☐ Disable COS
- Command Type:** Write Words
- Destination Unit:**
  - XYR 300G#3
  - Unknown Unit
  - XYR 300G#1
  - XYR 300G#2
  - XYR 300G#3
- Destination Gateway:**
  - Icon: XYR 300G#3
  - ☒ Acknowledge Message
  - I/O Register: 32
  - Via Repeaters:
    - 1: None
    - 2: None
    - 3: None
    - 4: None
    - 5: None

Buttons: Apply, Cancel

If a value in the block changes by more than the sensitivity amount, then only the I/O in the block message that has changed will be sent, not the whole block mapping. (this can only occur

for Write mappings). The sensitivity values are set under the “Sensitivity” option as per section 4.6.

A delay time can be entered to reduce the number of change triggers in active systems. For example, if 20 seconds is selected in the “Delay” window, then the change message will be sent 20 seconds after the change-of-state occurs - if other changes occur during the 20 second period, all of these changes are sent in the one message.

The delay time can be set from 0 – 254 seconds.

If you do not wish change messages to occur, select the “Disable” box.

#### 4.5.6 Block Read Mapping's

A Read mapping is a request sent to another XYR 300G to return a block of values. Like the Block Write mapping it can be triggered by a Real Time clock, Time period or by Host trigger however the main difference is that the COS Delay is now a Response Timeout as shown below.

The image shows a "Block Mapping Configuration" dialog box. It is divided into several sections. On the left, under "Source Gateway", there is a unit icon and "XYR 300G#4". Below this, "I/O Register" is set to 52 and "I/O Count" is set to 10. Further down, "Period" is set to 10 Minutes, "Offset" is 0 seconds, and "Response Timeout" is 5 seconds. A "Disable" checkbox is at the bottom left. In the center, "Command Type" is set to "Read Words" with a red arrow pointing left. Below it, "Destination Unit" is set to "XYR 300G#5". A list box shows "Unknown Unit" with a red question mark, and below it, "XYR 300G#5" and "XYR 300G#4" are listed with blue icons. On the right, under "Destination Gateway", there is a unit icon and "XYR 300G#5". Below this, "Acknowledge Message" is checked, "I/O Register" is set to 52, and "Via Repeaters" has five dropdown menus, all set to "None". At the top right, there are "Apply" and "Cancel" buttons.

In the above example XYR 300G-G#4 is performing a Block Read Request to XYR 300G-G# 5. XYR 300G-G#5 will send the values from its I/O registers 52 for a count of 10 to XYR 300G-G#4 I/O registers 52 – 61. This Block read will be performed using the real time clock at 10-minute intervals. A response timeout of 5 seconds is used to indicate that if the Block read values have not been received in 5 seconds then the Comms Fail bit for this block read mapping will be set.

It is not recommended to have the Response Timeout set to 0 seconds as a Comms fail bit will be set upon transmission.

If the Response Timeout is greater than the Block mapping time period and radio's are in a High traffic or poor radio path then instances could occur that received messages could be from previous block read mapping's hence giving incorrect values.

If a need for frequent communications between modules is required then Block Write mapping's would be more suitable.

#### 4.5.7 Mixing Normal Mappings and Block Mappings

Block mappings can include I/O Registers already used with normal I/O mappings.

For example, a remote XYR 300X I/O module could map a remote input to I/O Reg 743. At the XYR 300G, the host device could read I/O Reg 743, and you could also configure a block mapping including this register to another XYR 300G. You could write a block I/O Reg 700 – 800 to another XYR 300G.

#### 4.5.8 Block Mappings to internal I/O Registers.

Using the Block mapping feature this will allow to send block write mapping to self. This could be useful if wanting to have a global output to indicate a comms problem from any remote module by block mapping the internal status registers to a local output.

#### 4.5.9 Comms Fail for Block Mappings

Each block mapping has an associated mapping number. Up to 500 block mappings may be entered. A status register is maintained for each block mapping. The most significant bit of this register contains the comm fail status.

If a block mapping does not receive an acknowledgement from the remote module, then the comms fail status is set - this can be monitored by the host device.

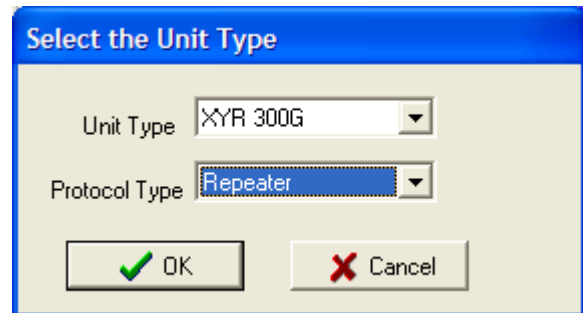
#### 4.5.10 “Repeater-only” Configuration

Any XYR 300G module can act as a repeater unit. However a XYR 300G may need to be installed as a repeater only (that is, there is no host device connected). In this case, the base XYR 300G, the

XYR 300G-M unit would normally be used as this is the lowest cost of the XYR 300G modules.

A repeater can be configured as a “Repeater-only” unit. The advantages are:

- ❑ the serial port will then provide on-line diagnostics (instead of off-line diagnostics), or
- ❑ XYR 301X serial I/O modules can be connected to the serial port - normally XYR 301X modules cannot be used with MD1 units.

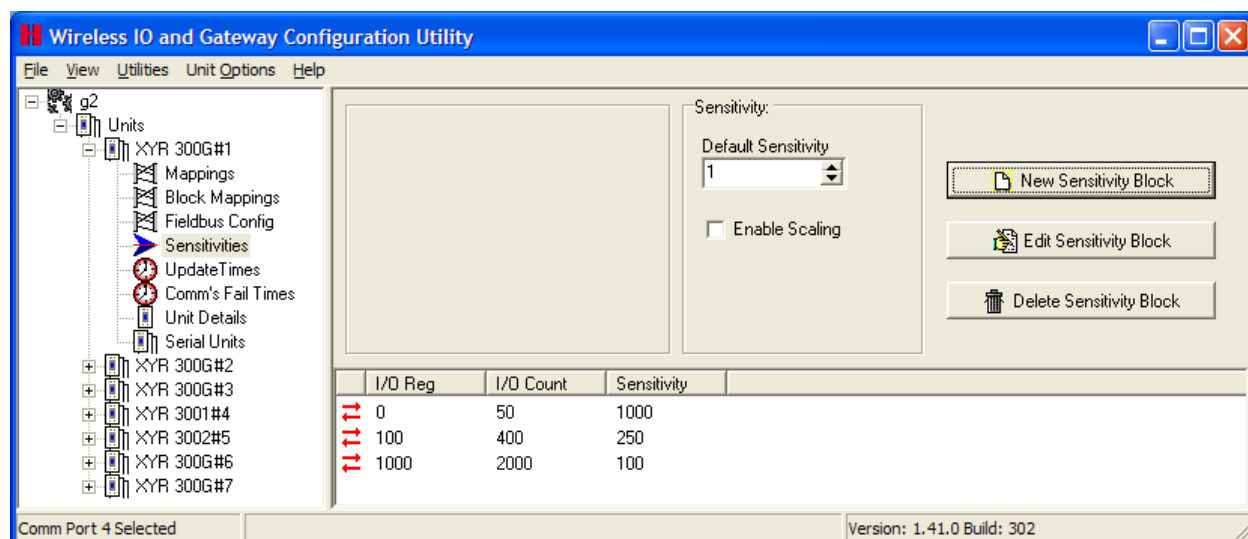


## 4.6 Change Sensitivity & I/O Value Scaling

### 4.6.1 Change Sensitivity

“Change” messages for both individual I/O mappings and block mappings use a sensitivity value to trigger the message. Sensitivities are configured for blocks of I/O registers - that is,

each I/O register does not have a unique sensitivity. You can configure up to 50 sensitivity values - that is, there can be 50 blocks of registers with different sensitivities.



In the above example, three sensitivity blocks have been configured:

1. I/O registers 0 – 49 have a sensitivity of 1000 (or 1.5% of the 16 bit range)
2. I/O registers 100 – 499 have a sensitivity of 250 (or 0.4% of the 16 bit range)
3. I/O registers 1000 – 2999 have a sensitivity of 100 (or 0.15% of the 16 bit range)

All of the registers between 0 and 49 have a sensitivity value of 1000. If register 34 has changed value by more than 1000 since the last transmission for that register, then a change trigger will occur for register 34. Sensitivity values are in decimal and can vary between 1 and 65535 (16-bit).

Up to 50 blocks of sensitivities can be configured. If a register is included in more than one block, then the first sensitivity value configured will be accepted and later values ignored. If Scaling is configured (refer next section), then the number of blocks is reduced to 25.

Registers which are not included in any block use the “default” sensitivity which is also user-configurable. In the above example, the default sensitivity is 1 and is the sensitivity for all I/O registers not included in the three blocks.

**Important Note.** Sensitivity values need to be selected carefully for analogue or counting registers as small values can result in a large number of change messages, which can overload the radio channel. A sensitivity value of 1 in 65535 is a change of 0.0015%. If the host device writes an analogue value to a XYR 300G every 100msec, it will change by at least 1 bit each time. A small sensitivity value will cause a change message to be sent every 100msec. If there are many analogue values in the same situation, then there would be many change messages every 100msec. Sensitivity values for analogue I/O should be set to be greater than the normal process noise of the signal. For example, if a flow signal has a normal process oscillation of 2.5%, then the sensitivity should be set to 3% (or a value of 2000) to avoid change transmissions from the process oscillations.



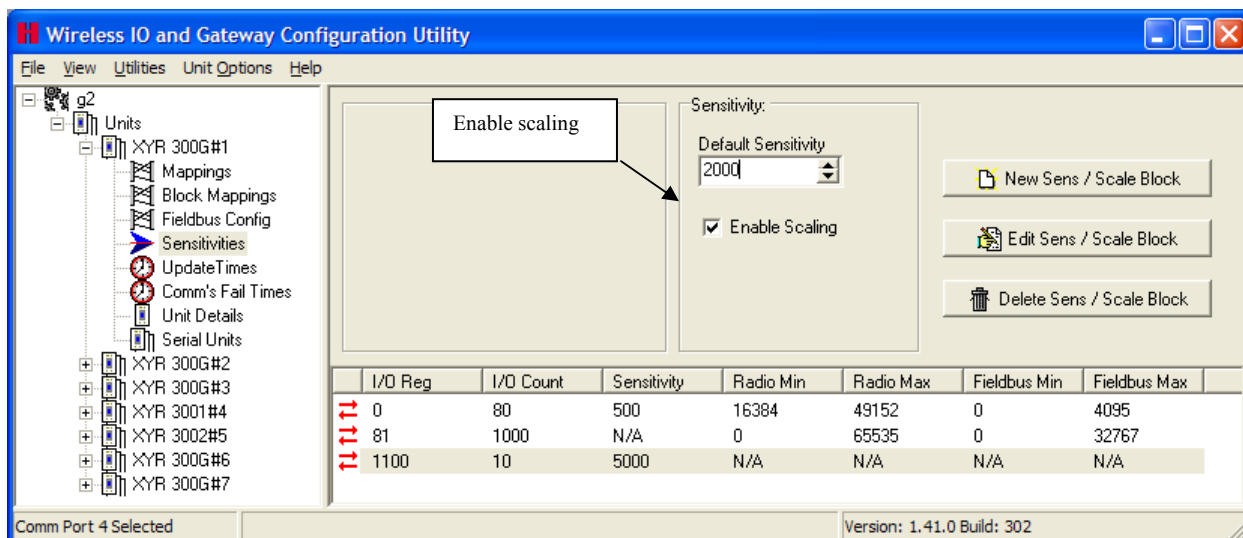
## 4.6.2 I/O Value Scaling

The values in I/O registers can be scaled as the values are transferred to the data bus, or from the data bus.

The I/O values in the XYR 300G database registers are stored as 16-bit values (between 0 and FFFF hexadecimal or 0 and 65,535 decimal). Analog inputs at a XYR 300X I/O module are scaled hex 4000 (dec 16,384) for 4mA and hex C000 (dec 49152) for 20mA. A 12 mA signal is half-way in this range at hex 8000 (dec 32,768).

The reason for adding additional scaling between the XYR 300G database (radio side) and the data bus is to cater for external host devices which do not handle normal 16-bit values. Two examples are:

- Honeywell Modbus gateways which only handle 12-bits values (0-4,095 decimal), and
- Sensor / analyzer devices with “signed 16-bit” values. A signed 16-bit value is a 15-bit value with an additional bit to signify plus (0) or minus (1).



Scaling of I/O registers can be configured in blocks. Different blocks can have different scaling.

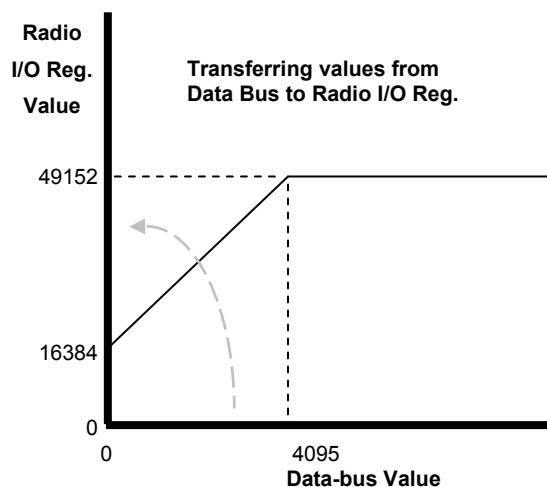
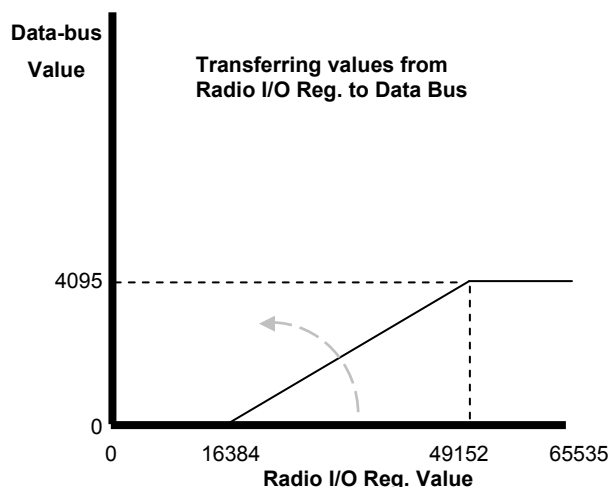
Note that scaling only affects values transferred in or out of the data bus port. It has no effect on the radio side.

Scaling is configured in the “Sensitivities” section of the configuration software. If you select a new sensitivity/scaling block, you can select/deselect sensitivity or scaling or both. There is no relationship between sensitivity and scaling - we use the same configuration area as it is convenient because both features use blocks of I/O registers.

In the first example, a block of I/O registers is configured for both sensitivity and scaling. I/O block 0 to 79 (total of 80 registers) is configured with a sensitivity value of 500. The same block has scaling configured converting the range 16384-49152 on the radio side to 0-4095 on the data bus side.

This is an example of converting an “Honeywell 4-20mA value” to a “Honeywell 12-bit value”. Note that the scaling works in both directions - for values being read from the I/O registers to the data bus, and values written from the data bus to the I/O registers.

Any values outside of the scaling range are set to the minimum or maximum value. For example, if the data bus read a value of 10,000 from a register in this block, as it is less than the minimum range on the radio side (the min. is 16,384) it will be transferred as 0 which is the minimum value on the data bus side. If a value of 65,535 is read from another register, then as it is more than the maximum value on the radio side (max. value is 49,152), then the value is transferred as 4095 which is the maximum on the data bus side. This works in both directions - if the data bus tries to write a value of 10,000 to an I/O register in this block, it will be written as value 49,152 (which is the max. value on the radio side).



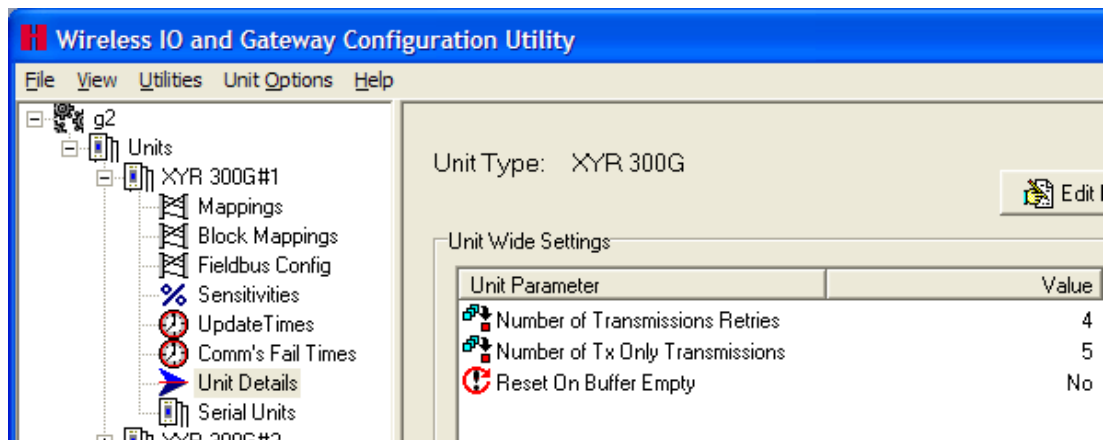
The second example shows another I/O block (registers 81 to 1080) that has been selected for scaling only - the sensitivity function has been disabled (these registers will use the default sensitivity of 2000 configured on the main Sensitivity configuration screen).

In this example, the full 16-bit range (0-65535) is scaled to “signed 16-bit values”. A value greater than 32767 (which will be seen as a negative value) can’t be written to the data bus.

In the last example, Scaling has been disabled for register block 1100 – 1109. Only sensitivity functionality is being used.

**Note:** If Scaling is not used at all, up to 50 blocks can be configured with different sensitivity values. However if Scaling is used, then only half this number of blocks is available.

### 4.6.3 Unit Details



**Number of Transmission Retries.** - Configurable value between 0 – 4, If XYR 300G does not receive an Acknowledgment from a message it will retry up to this configured amount.

**Note:** Setting to 0 will not allow for any retries. Not recommended for poor radio paths.

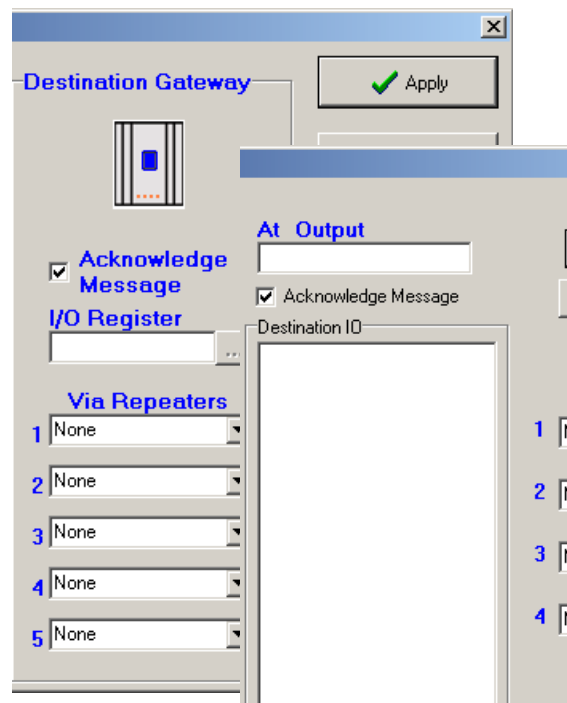
#### **Number of TX only Transmissions.** –

Under I/O mapping's and Block mapping's is option for the XYR 300G to send messages as a Transmit only. By default under each section all messages will be acknowledged. Uncheck the Acknowledge Message box to make the transmission TX only.

In the unit details section a Configurable number of TX only transmissions are available between 1 – 5.

As a Change of State occurs or Timed Update expires each message will be sent this number of times.

**Note:** If setting Number of TX only transmissions to 1 ensure you have a good radio path and/or use Output reset times at destination to indicate comms fails.



### **Reset on Buffer Empty**

The XYR 300G has a series of internal buffers that are used for moving I/O between the Radio Interface, I/O Database and Fieldbus Interfaces. There are a number of different buffers that the XYR 300G uses and a list of these can be found in section 4.16.

The option of Reset Buffer on Empty will allow for the XYR 300G to be fully reset to clear its buffers. By default this is disabled and can be used in applications of when a XYR 300G is used as a repeater where there is excessive traffic and also in marginal radio paths. As each message is passed via a repeater it will buffer this message in a queue and then forward on and wait for ACK before it clears the buffer. If system is experiencing high radio message count and poor radio path creating number of retries, cases could exist that the buffers will empty. Enabling the Reset on Buffer Empty feature will reset the module and then continue to operate.

## 4.7 Serial Configuration - MODBUS

---

The XYR 300G-M module provides interface for Modbus Slave, Modbus Master and Allen-Bradley DF1. This Modbus interface uses the Modbus RTU protocol - also known as the Modbus Binary protocol. This manual assumes that the reader has a good understanding of the Modbus or DF1 protocol.

### 4.7.1 MODBUS Slave

If you use the XYR 300G Modbus Slave interface, then the host device will be a Modbus Master device. The only configuration required for the Modbus slave interface is selecting the Modbus address and serial port parameters. This is done in the “Serial Settings” screen. A valid Modbus slave address is 1 to 255.

Each I/O register (and status register) in the XYR 300G can act as one of the following types of Modbus registers

00001-09999 = Output Coils (digital/single bit)  
10001-19999 = Input Bits (digital/single bit)  
30001-39999 = Input Registers (analog/16 bit)  
40001-49999 = Output Registers (analog/16 bit)

For example:

- If the Modbus Master sends the XYR 300G a “read” command for Modbus input 10457, then the XYR 300G will respond with the value in I/O register 457.
- If the Modbus Master sends the XYR 300G a “write” command for Modbus output 02650, then the XYR 300G will write the value to I/O register 2650.
- If the Modbus Master sends the XYR 300G a “read” command for Modbus input 30142, then the XYR 300G will respond with the value in I/O register 142.
- If the Modbus Master sends the XYR 300G a “write” command for Modbus output 40905, then the XYR 300G will write the value to I/O register 905.

The XYR 300G I/O register values are 16 bit (hexadecimal values ‘0000’ to ‘FFFF’, or decimal 0 to 65535), regardless of whether the register represents a discrete, analog or count point.

The value of a discrete (digital) I/O point is stored in the XYR 300G database as a hexadecimal '0000' (“off”) or hex 'FFFF' (“on”). However the XYR 300G will respond with either a ‘0’ (“off”) or ‘1’ (“on”) to a digital read command from the Modbus master - these are commands 01 and 02. Similarly, the XYR 300G will accept ‘0’ or ‘1’ from the Modbus master in a digital write command and store ‘0000’ or ‘FFFF’ in the database location - these commands are 05 and 15.

The Modbus function codes that the XYR 300G will respond to are shown in the table below.

**Supported Modbus Function Codes:**

Function Code	Meaning
01	Read the state of multiple digital output points
02	Read the state of multiple digital input points
03	Read the value of multiple output registers
04	Read the value of multiple input registers
05	Set a single digital output ON or OFF
06	Set the value of a single output register
07	Read Exception Status - compatibility - returns zero
08	Loopback test Supported codes   0   return query data 10   clear diagnostic counters 11   bus message count 12   CRC error count 14   slave message count
15	Set multiple digital output points ON or OFF
16	Set multiple output registers

Analog I/O are 16 bit register values. A value of decimal 8192 (hex 2000) represents 0mA. A value of 49152 (hex C000) represents 20mA. Each 1 mA has a value of 2048 (hex 0800) - a change of 4096 (hex 1000) is equivalent to a change of 2mA. A 4-20mA signal will vary between 16384 (hex 4000) and 49152 (hex C000). A 0-20mA signal will vary between 8192 (hex 2000) and 49152 (hex C000).

Pulse counts are stored as a 16-bit register. When the register rolls over, from 'FFFF' (hex), the next value will be '0001'. The register will only have a value of '0000' when the remote module starts up, and the previous count is lost. This value will indicate that the counter has reset.

**Modbus Errors**

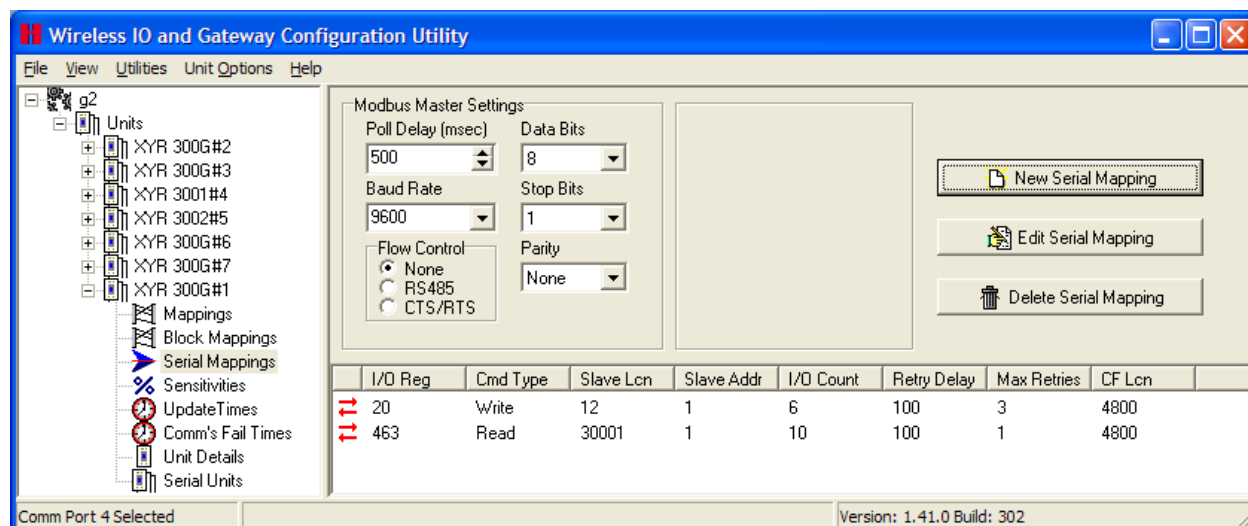
Four Modbus error messages are reported to the Modbus Master. An error response is indicated by the address of the return message being 128 plus the original slave address.

### Supported Exception Codes:

Exception Code	Name	Description
01	Illegal function	The module does not support the function code in the query
02	Illegal data address	The data address received in the query is outside the initialized memory area
03	Illegal data value	The data in the request is illegal
06	Busy	Unable to process message

## 4.7.2 MODBUS Master

If you use the XYR 300G as a Modbus Master, then the host device/s will be Modbus Slave device/s. If the RS485 port is used, then multiple Modbus Slave devices can be connected to the XYR 300G. The XYR 300G Modbus Master will generate Modbus read and write commands to the Modbus Slave devices.



First read the above section on Modbus Slave operation, for an understanding of how the XYR 300G handles Modbus registers, and the types of Modbus commands the XYR 300G Master can generate.

The Modbus Master commands are configured in the “Serial Mapping” screen. The serial port is configured in the same way as described in the above section on Modbus Slave.

To enter a Modbus command, select “New Serial Mapping”. The following example is a digital write command which writes XYR 300G I/O registers 20 – 25 (6 registers) to Modbus outputs 00012 – 00017, at Modbus Slave address 1.

The entry under “I/O Register” is the first I/O register in the XYR 300G to be transferred - the “I/O count” is the number of registers to be transferred. If the selected Modbus slave does not respond to the command, then the XYR 300G will write a ‘FFFF’ value to one of its own registers, configured under “CF Register” - in this case it is register 4800.



The “Command Type” selected is a write command (you can select read or write) - which means that the values are sent from the XYR 300G to the Modbus Slave. The type of write command is a “Digital” write, meaning that the register values will be written as digital/binary values”.

**Modbus Configuration**

Modbus Configuration for: XYR 300G#1

**Master**

I/O Register: 20

I/O Count: 6

CF Register: 4800

**Command Type**

Write

☒ Digital (Single Bit)

☐ Register (16 Bit)

Max Retries: 3

Retry Delay (ms): 100

**Slave**

Slave Address: 1

Slave Location: 12

0 = special "elpro" poll  
 1-9999 = Output Coils  
 10001-19999 = Input Bits  
 30001-39999 = Input Registers  
 40001-49999 = Output Registers

Apply

Cancel

If the Modbus Slave device does not respond to the Modbus command, the XYR 300G will try another 3 times (“Max Retries” = 3). The Modbus command will be sent to the Modbus Slave every 100msec. The address of the Modbus Slave is 1 (permissible addresses are 1 – 255). Because a digital write command has been selected, the destination register type will be digital outputs, with Modbus tag “0xxxxx”. The first destination Modbus location is 12 (or 00012) - as there are 6 registers transferred, the destination locations will be 00012 – 00017.

The second example is a register read command to the same Modbus Slave (address 1). The command requests the Modbus Slave to return the values of 10 registers which will be stored in I/O registers 463 - 473 in the XYR 300G. As the command is a “register read” command, the target Modbus locations will be of the type 3xxxx. The starting location is 30001. So the values of locations 30001 – 30010 in Modbus Slave 1 will be transferred to I/O registers 463 – 473 in the XYR 300G.

The CF Register (“comms fail” register) acts as a digital alarm – the value of the register will normally be 0, and will be set to FFFF (hex) if the slave device does not positively respond to the serial command within *Max Retries* attempts. In the examples, the same CF Register (4327 – i.e. DOT8) has been used for both serial mappings, such that the local digital output will be activated if the slave fails to respond to either serial command. Alternately, any other internal register could have been chosen and mapped via radio if desired.

**Modbus Configuration**

Modbus Configuration for: XYR 300G#1

**Master**

**I/O Register**  
463

**I/O Count**  
10

**CF Register**  
4800

**Slave**

**Slave Address**  
1

**Slave Location**  
30001

**Command Type**  
Read  
☐ Digital (Single Bit)  
☒ Register (16 Bit)

**Max Retries**  
1

**Retry Delay (ms)**  
100

0 = special "elpro" poll  
 1-9999 = Output Coils  
 10001-19999 = Input Bits  
 30001-39999 = Input Registers  
 40001-49999 = Output Registers

Apply

Cancel

To complete the Fieldbus Configuration, enter any other Modbus commands that may be required to transfer I/O points between the XYR 300G and the Modbus Slave devices.

### Digital I/O

The value of a digital I/O point is stored in the XYR 300G database as a hexadecimal '0000' ("off") or hex 'FFFF' ("on"). However the XYR 300G will generate either a '0' ("off") or '1' ("on") to a digital output point (Coil) when sending commands to a Modbus slave - these are commands 05 and 15. Similarly, the XYR 300G will accept '0' or '1' from the Modbus slave in response to a digital read command and store '0000' or 'FFFF' in the database location - these commands are 01 and 02.

### Analog I/O

Analog I/O from the remote XYR 300X modules are 16 bit register value. A value of 8192 (hex 2000) represents 0mA. A value of 49152 (hex C000) represents 20mA. Each mA has value of 2048 (hex 0800) - a change of 4096 (hex 1000) is equivalent to a change of 2mA. A 4-20mA signal will vary between 16384 (hex 4000) and 49152 (hex C000). A 0-20mA signal will vary between 8192 (hex 2000) and 49152 (hex C000).

### Pulse I/O

Pulse counts from the remote XYR 300X modules are shown as a 16-bit register. When the register rolls over, from 'FFFF' (hex), the next value will be '0001'. The register will only have a value of '0000' when the remote module starts up, and the previous count is lost. This value will indicate that the counter has reset.

## **Modbus Retry Delay**

The XYR 300G Modbus Master configuration includes a feature to limit the frequency at which slave devices are polled for data. The XYR 300G will poll each Modbus slave in order. If there is no delay time entered, the XYR 300G will poll as quickly as it is able to. If there is a delay time entered, then this delay time will occur between each poll message.

When updated values are received from the XYR 300X radio network, the current polling sequence is interrupted, and the new values are written immediately to the appropriate slaves.

## **Re-tries on the Serial Port**

When communicating with Modbus slaves, the XYR 300G may be configured to re-try (or re-send) a message zero or more times if no response is received from a slave. If all retries are used up, that slave is flagged as being in communication failure. Further attempts to communicate with the slave will have zero re-tries. When a successful response is received from the Modbus slave, the communication failure flag is reset and the configured number of re-tries will be used. This means that an off-line slave device will not unduly slow down the communications network.

## **Comms Fail**

A “Comms Fail” image location in the XYR 300G database. This image location should be in the range 4500 to 4999. If a response is not received from the Modbus slave after all re-tries have been sent, the XYR 300G will set this Comms Fail image location to hex(FFFF). When the XYR 300G sends the next poll for this I/O Command, it will not send any re-tries if a response is not received to the first message. When a response is eventually received, the XYR 300G will reset the value in Comms Fail image location to 0, and the normal re-try sequence will operate.

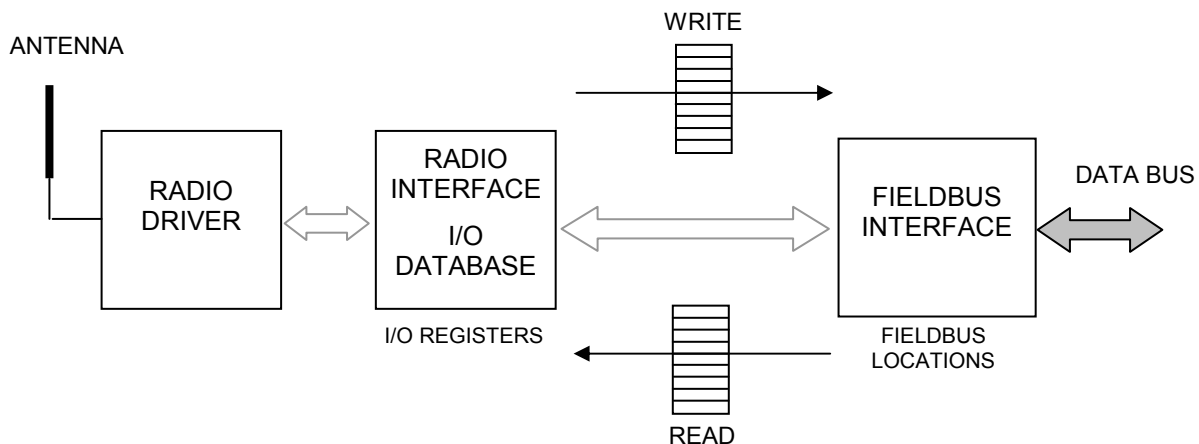
Different I/O Commands can use different Comms Fail image locations, however we recommend that you use the same image location for all I/O Commands to the same Modbus slave address.

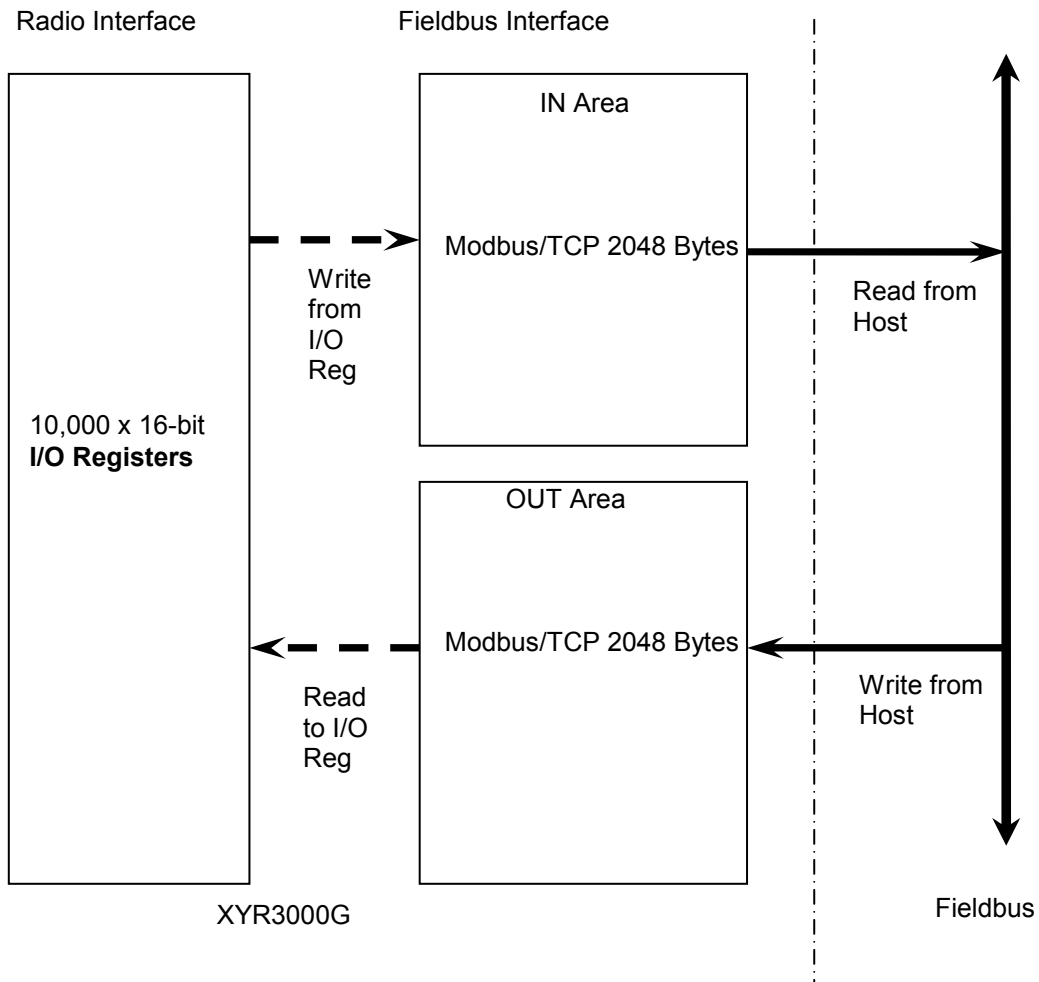
## 4.8 Fieldbus Configuration

All XYR 300G modules (*except XYR300G-M*) have separate internal hardware comprising the Fieldbus Interface, consisting of a separate microprocessor and appropriate hardware for the network connection. This Fieldbus Interface handles all fieldbus communications, and transfers I/O in the Fieldbus Interface Registers to/from the fieldbus. Conversely, the XYR 300G Radio Interface handles all radio communications, and transfers I/O in the Radio Interface Registers to/from the radio network. For I/O transfer between the radio network and the fieldbus network, I/O Registers in the Radio Interface must be linked with registers in the Fieldbus Interface using configuration software.

Depending on the fieldbus protocol, the size of the Fieldbus Interface may be limited (for example, the Profibus Slave interface supports only 416 bytes I/O). The Radio Interface supports 10,000 registers, of which 4300 are general-purpose I/O registers. Each Radio Interface register is 16-bit, even for discrete (or “digital”) input or output values. The Fieldbus Interface comprises a block of 8-bit bytes (referred to as “locations”). Digital I/O can be packed - each fieldbus location can hold 8 digital inputs or outputs. Analog or pulse values can be stored as a low resolution 8-bit value (a single fieldbus location) or as a high resolution 16-bit value (two consecutive fieldbus locations).

To optimize I/O usage, the XYR 300G provides a flexible method of data transfer between the Radio Interface and the Fieldbus Interface. The user configures *links* between the Radio Interface and Fieldbus Interface via *Fieldbus Mappings* in the XYR300X Configuration Software. The diagram shows in more detail the relationship between the Radio Interface and Fieldbus Interface.





#### 4.8.1 Fieldbus Mappings

The Fieldbus Interface is divided into two distinct areas. The IN Area contains input data that is made available to the host device. The OUT Area contains output data from the host device. This is in contrast to the Radio Interface, in which each 16-bit register can be used as input *or* output. Also note the size of the Fieldbus Interface is variable, depending on the type of fieldbus.

XYR300X Configuration Software provides user configurable *Fieldbus Mappings* to link the required Fieldbus I/O to the Radio Interface. *Write* mappings write I/O values from the Radio Interface *to* the Fieldbus IN Area. *Read* mappings read I/O values *from* the Fieldbus OUT Area to the Radio Interface.

If you want to send a value from the XYR 300G to the host device, use a Fieldbus *Write* Mapping. The input data from the Radio Interface (i.e. input data that has either come in from the radio or from local I/O) will be transferred to the IN Area via the fieldbus write mapping. The host device can then read this input data from the IN Area.

If you want to send a value from the host device to the XYR 300G, use a Fieldbus *Read* Mapping. The host device can write output data to the OUT Area. The output data from the OUT

Area will then be transferred to the Radio Interface via the fieldbus read mapping. The radio driver can then either send this output over the radio or to a local I/O.

Several different configurable *transfer modes* are also available for fieldbus mappings to ensure the I/O is formatted according to the requirements of the particular fieldbus protocol or host device. The six possible types of Fieldbus Mapping are outlined in the table below.

### Fieldbus Mapping Types

<i>Transfer Mode</i>	<i>Read Mapping</i>	<i>Write Mapping</i>
<i>Single Bit</i>	The XYR 300G reads a block of consecutive bits from Fieldbus OUT Area and stores each bit in consecutive I/O Registers, as hex FFFF or 0000.	The XYR 300G takes the MSB (most significant bit) of a block of consecutive I/O Registers, converting the 16 bit I/O register values into 0 or 1, and writes to consecutive bits of Fieldbus IN Area.
<i>Byte (8-bit)</i>	The XYR 300G reads consecutive bytes (8-bit values) from Fieldbus OUT Area and stores each byte in the most significant 8-bits of a consecutive I/O register.	The XYR 300G takes the most significant 8-bits of consecutive I/O registers and writes them to consecutive bytes (8-bit values) of the Fieldbus IN area.
<i>Word (16-bit)</i>	The XYR 300G reads consecutive words (2x8-bit values) form Fieldbus OUT Area and stores each word in a consecutive I/O Register.	The XYR 300G takes consecutive I/O registers and writes them to consecutive words (2x8-bit values) of Fieldbus IN Area.

### 4.8.2 Transfer Mode

Radio Interface registers are all 16-bit general-purpose input or output registers. That is, analog inputs or outputs are stored as a 16-bit value. Digital inputs or outputs occupy a whole 16-bit register and are stored as either 0000(hex) or FFFF(hex) for compatibility with the Honeywell Radio Protocol. However, the Fieldbus Interface may contain (depending on the protocol) significantly less registers than the Radio Interface (see diagram above). Also, certain protocols may require a different I/O structure than that used by the Radio Interface registers.

Consequently, depending on the fieldbus mapping transfer mode (see above table), Radio Interface registers may or may not be compressed.

“Word” transfer mode offers no compression, but rather a direct transfer of 16-bit registers between Radio Interface and Fieldbus Interface. This mode would suit the transfer of registers containing pulse counts or analog values with no loss of resolution.

“Byte” transfer mode operates on only the most significant BYTE (the first 8 bits) of Radio Interface registers, but allows these bytes to be consecutively packed in the Fieldbus Interface. This mode would suit the transfer of analog values in low-resolution, in cases where I/O space is at a premium. *Byte Address Mode* is recommended when using byte transfer mode (see Address Mode section below).

Bit transfer mode operates on only the most significant BIT of Radio Interface registers, but allows these bits to be consecutively packed in the Fieldbus Interface. This mode would suit the

transfer of digital I/O in cases where it is not desirable (or possible) to use a whole 16-bit register just to store a 0 or 1 value.

### 4.8.3 Address Mode

Configuration software allows the Fieldbus Interface IN and OUT areas to be addressed as an array of 8-bit bytes (*Byte Address Mode*) or an array of 16-bit words (*Word Address Mode*). The address mode may be required to change depending on the transfer mode, the protocol, or the particular host device. The Address Mode option is included so that the configuration software can be setup to use the same I/O addressing method used by the host device. The actual structure of I/O in this database can only be physically altered via the transfer mode.

The Fieldbus Interface IN and OUT areas are simply a block of I/O memory, exchanged according to the configured protocol. For example, with a Modbus TCP slave that supports 2048 bytes of inputs, the fieldbus interface IN area could be addressed either as byte locations 1 to 2048 or as word locations 1 to 1024. Note that in either case, the underlying database structure is unchanged, the difference is limited to the Fieldbus IN/OUT Area address that is displayed by configuration software.

Certain protocols have an inherent or preferred byte or word structure – for example, Modbus is a protocol that usually operates on 16-bit (word) registers. Consequently, configuration software will default to the most common address mode for that protocol. Configuration software may also apply an offset and/or scaling to the IN/OUT Area addressing to suit the particular protocol. For example Modbus/TCP areas start from location 1, but other fieldbuses may start at location 0.

Note:

- The Fieldbus Interface IN *and* OUT Area both number from 0 - that is, there is an input 0 as well as an output 0 (an offset may apply for some protocols).
- All IN/OUT Area locations accessed by the fieldbus must be part of a fieldbus mapping in the XYR 300G - that is, if a host device is writing to bytes 0 – 100 in the OUT Area, there must be at least one fieldbus read mapping that uses these locations - if not, the Fieldbus Interface will generate an error response message.
- Fieldbus mappings to/from the IN/OUT areas should always start at location 0 if possible (or the lowest available unused location). Configuration Software will always automatically choose the next lowest available location – it is strongly recommended that this topology be used so as not to place unnecessary processing overhead on the module.

### 4.8.4 Fieldbus Mapping Configuration

The example below shows the Fieldbus Mapping configuration screen when adding new or editing existing Fieldbus Mappings. Starting from the left of the screen, the I/O Register selection specifies the starting I/O Register from the Radio Interface (press the “...” button to make a selection graphically). The I/O Count parameter specifies how many consecutive I/O Registers are to be transferred or *linked*. Command Type and Transfer Mode specify the type of

Fieldbus Mapping (see Fieldbus Mappings table above). Finally, I/O Location specifies the IN or OUT Area location in the Fieldbus Interface (see earlier diagram).

**Ethernet Configuration**

Ethernet Configuration for: XYR 300G#5

**Radio Interface**

I/O Register (16 bit)

I/O Count

**Command Type**

Transfer Mode:

☐ Single Bit Mode

☐ Byte Mode (8 bits)

☒ Word Mode (16 bits)

**Fieldbus Interface**

☒ MS Byte First

I/O Location (16 bit)

I/O Instance

1

Disable ☐

Apply

Cancel

**Wireless IO and Gateway Configuration Utility**

File View Utilities Unit Options Help

Build 154 Test

- Units
  - XYR 300G#1
  - XYR 300G#2
  - XYR 300G#3
  - XYR 300G#4
  - XYR 300G#5
- Mappings
- Block Mappings
- Fieldbus Config
- Sensitivities
- Ethernet Settings
- UpdateTimes
- Comm's Fail Times
- Unit Details
- Serial Units

**Fieldbus Options:**

Status Location: 4500

☒ Disable

☐ Clear Reg's when Offline

Address Mode:

☐ Byte (8 bits)

☒ Word (16 bits)

☒ Modbus TCP

New Fieldbus Mapping

Edit Fieldbus Mapping

Delete Fieldbus Mapping

I/O Reg	Fieldbus Lcn	I/O Count	Cmd Type	BIT Offset
I/O Reg 10	1	5	Write WORD	N/A
I/O Reg 30	16400	12	Read BIT	0
I/O Reg 4320	16412	8	Read BIT	4

Comm Port 4 Selected

Version: 1.41.0 Build: 302

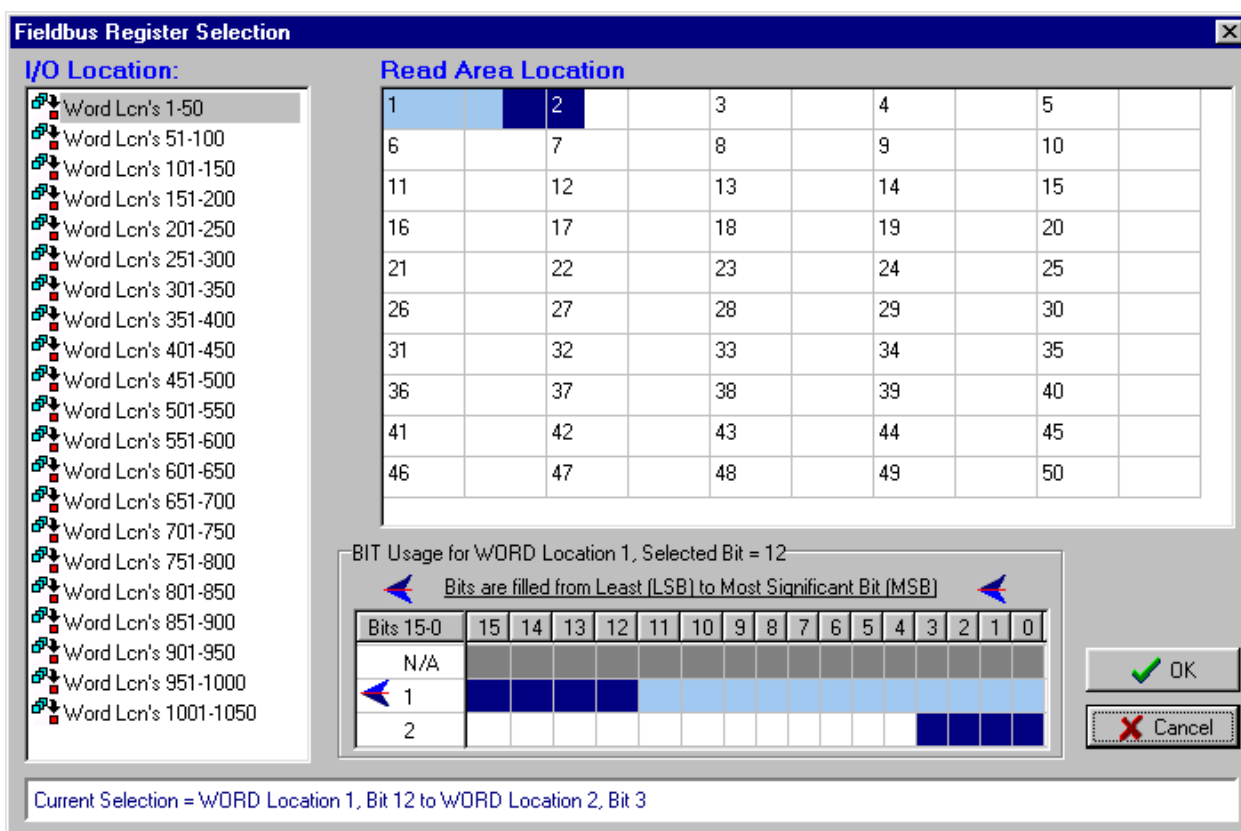
Three Fieldbus Mappings are illustrated in the example above. Note that “Word Address Mode” is selected, meaning that the Fieldbus Interface IN and OUT Areas will be treated as word-addressed arrays by configuration software. The parameters for each fieldbus mapping were setup using the mapping configuration screen as described above.



The first Fieldbus Mapping is a “Write WORD” mapping, writing I/O Registers 10 – 15 from the Radio Interface to word-locations 1 – 5 in the Fieldbus IN Area. Because the *transfer mode* is “word” complete 16-bit registers are transferred.

The second mapping is a “Read BIT” mapping, reading 12 *bits* from Fieldbus OUT Area *word-location* 1(word address mode is selected) to I/O Registers 30 – 41. Remember that for such a BIT transfer, that each individual bit in the Fieldbus Interface is transferred to an entire 16-bit I/O Register. Note also that there is a word-location 1 for both the Fieldbus OUT *and* IN areas.

The third mapping is another “Read BIT” mapping, reading 8 bits from Fieldbus OUT Area word-location 1 to I/O Registers 4320 – 4327 (i.e. local DOT 1 – 8). Note here that we are again reading from Fieldbus OUT Area word-location 1 (as with the previous mapping). However, since each word-location contains 16-bits and the last mapping used only 12 of those, we have been able to follow on from the previous mapping (see below).



The Fieldbus Register Selection screen above was shown when selecting the Fieldbus OUT Area location for the third mapping in the above example. This screen shows the currently used portion of the Fieldbus OUT Area, and allows the user to graphically select the location for the current mapping. NOTE – by default configuration software will always choose the next available Fieldbus Interface register for fieldbus mappings. Allowing configuration software to automatically make the selection is strongly recommended wherever possible.

Clicking on the required location in the top panel will alter the currently selected word-location. Further, clicking individual bits in the “Bit Usage” panel at the bottom of the screen, allows the current BIT mapping to be specified at the bit-level of the currently selected word.

The lighter blue areas indicate the extent of already existing fieldbus mappings. It can be seen that bits 0 – 11 of word location 1 have already been used (by the second mapping in the example). The dark blue area in the register selection screen above shows the extent and location of the current fieldbus mapping graphically. The status panel at the bottom of the window always displays the extent of the current selection, which can be seen to be word 1, bit 12 to word 2, bit 3.

A status location (4500) may be used to give the host device status information about the Fieldbus Interface. This register will be value 0x0000 if the Fieldbus Interface is “on-line” and communicating with the fieldbus, or value 0xFFFF if it is “off-line”. If you wish to use a status register, select the “Enable Status Location” box. This register could be mapped to a remote module or local output as an alarm.

## 4.9 Fieldbus Configuration - Ethernet

---

The XYR 300G provides the following Ethernet functionality:

1. Modbus/TCP. The module supports the Modbus/TCP protocol and conforms to the Modbus/TCP specification 1.0 (full information on this protocol can be obtained from <http://www.modicon.com/openmbus/index.html>). Refer to section 4.12.2 below for configuration details.
2. IT-Functionality. The Ethernet XYR 300G has several IT features, including Internet functionality.
  - Filesystem. The module features a flexible file system with two security levels. The size available for user files is approximately 1.4 Mbyte of non-volatile memory.
  - FTP Server. The FTP Server provides easy file management using standard FTP clients.
  - Telnet Server. The Telnet server features a command line interface similar to the MS-DOS™ environment.
  - HTTP Server. The module features a flexible HTTP server with SSI functionality. This enables the user to configure a web interface (or web page) accessing I/O values in the XYR 300G.
  - Email Client (SMTP). Predefined messages stored within the file system can be sent, triggered by a specified I/O value in the XYR 300G. It is also possible to include I/O values in emails, using SSI functionality.
  - IP Access Control. It is possible to configure which IP addresses and what protocols that are allowed to connect to the module.

For further details, refer to Appendix 2.

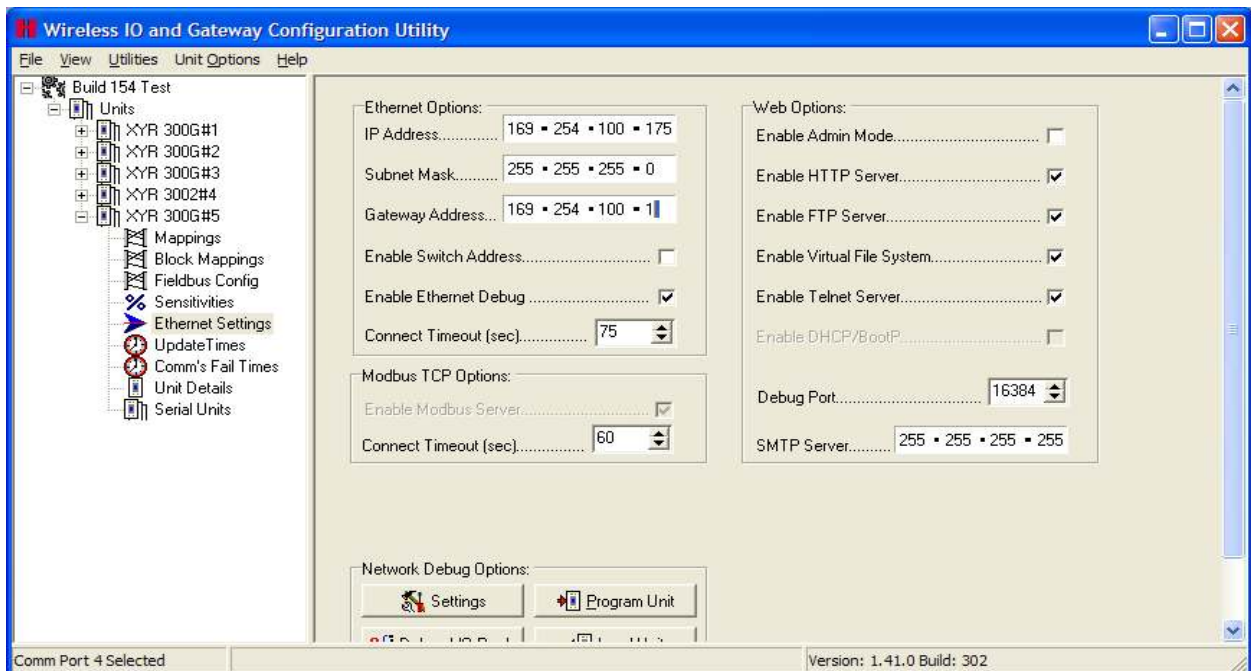
### 4.9.1 Setting IP Address

The Ethernet IP address can be set from the configuration software or via the Ethernet port or via the selector switches in the top end-plate of the module. If the “Enable Switch Address” box is not selected, then the address entered in the program will be used and the switch value ignored. The IP address can be overwritten from the Ethernet port. If the “Enable Switch Address” box is selected, then the address entered in the configuration program will be ignored and the rotary switch read on start-up of the XYR 300G.

The IP address is used to identify each node on the Ethernet network. Therefore, each node on the network must have a unique IP address. IP addresses are written as four decimal integers (0-255) separated by periods, where each integer represents the binary value of one byte in the IP address. This is called dotted-decimal notation. Example: 169.254.100.175

#### Subnet Mask

An IP Address is divided into two main parts *subnet ID* and *host ID*. All devices on the same local network must have the same subnet ID, but a unique host ID. To separate these two parts a *subnet mask* is used. In its simplest form, the subnet mask is a four byte pattern where a value of 255 allocates the corresponding byte of the IP Address to the subnet ID, and a value of 0 allocates the corresponding byte of the IP Address to the host ID.



For example, a common subnet mask is shown in the example below. Looking at the IP Address located directly above the Subnet Mask in this example, it can be seen that the IP Address values directly above a subnet mask value of 255 correspond to the subnet ID. Conversely, the IP Address values directly above a subnet mask value of 0 correspond to the host ID. So, in this example, the subnet ID is 169.254.100 and the host ID is 175.

## Special case IP addresses

Devices on an Ethernet network are not allowed to be configured to the following IP addresses; therefore do not configure the module to use any of them.

0.x.x.x - IP address where the first byte is zero  
127

127.x.x.x - IP address where the first byte is

x.x.x.0 - IP address where the last byte is zero

x.x.x.255 - IP address where the last byte is 255

## Gateway

The Gateway IP is the IP address of the LAN server or the host device.

## Connect Timeout

The Connect Timeout parameter in the IP addressing section of the display refers to the IP functionality of the module. If an IP connection to the module has not been active for this amount of time, the XYR 300G will timeout and disconnect that connection. Note that there can be several active connections at the same time - only the inactive connection will be disconnected.

## Enable Ethernet Debug

Select this box if you wish to enable Ethernet Diagnostics on the XYR 300G via configuration software (see section 6.3 for details). For security reasons, disabling this option will disallow all Ethernet diagnostics functions accessible to configuration software, and can only be reactivated via serial port configuration.

### 4.9.2 Modbus TCP

To use Modbus TCP, select the Enable Modbus Server box and deselect the Enable Ethernet/IP box. This will automatically remove the “I/O Instance” selection for all fieldbus mappings. It is possible for both Modbus TCP and Ethernet/IP to be selected - in this case, select “Disable I/O Instance” individually for each Modbus TCP fieldbus mapping.

**Supported Commands:**

Function Code	Function Name	Class	Affects Area	Address Method
1	Read coils	1	IN/OUT	Bit
2	Read Input discretes	1	IN/OUT	Bit
3	Read multiple registers	0	IN/OUT	Word
4	Read input registers	1	IN/OUT	Word
5	Write coil	1	OUT	Bit
6	Write single register	1	OUT	Word
7	Read exception status	1	-	-
15	Force multiple coils	2	OUT	Bit
16	Force multiple registers	0	OUT	Word
22	Mask write register	2	OUT	Word
23	Read/Write registers	2	IN/OUT	Word

**Supported Exception Codes:**

Exception Code	Name	Description
01	Illegal function	The module does not support the function code in the query
02	Illegal data address	The data address received in the query is outside the initialized memory area
03	Illegal data value	The data in the request is illegal

**Modbus/TCP Addressing**

The IN and OUT areas of the Ethernet interface are addressed under Modbus/TCP according to the tables below. Since Modbus uses a 16-bit format, “Word (16-bit) Address Mode” will be automatically applied whenever the “Modbus/TCP” checkbox is checked. If Ethernet/IP is also enabled, the “Disable I/O Instance” option must be selected for each fieldbus mapping to which Modbus/TCP Addressing is to apply.

**IN Area Modbus TCP Addresses (XYR 300G Write Locations 0 – 1023\*)**

<b>IN Area Location</b>	<b>Modbus Word Address</b>	<b>Modbus Bit Address</b>						
		<b>Bit 15</b>	<b>Bit 14</b>	<b>Bit 13</b>	<b>---</b>	<b>Bit 2</b>	<b>Bit 1</b>	<b>Bit 0</b>
<b>0</b>	1	1	2	3	---	14	15	16
<b>1</b>	2	17	18	19	---	30	31	32
<b>---</b>	<b>---</b>	<b>---</b>	<b>---</b>	<b>---</b>	<b>---</b>	<b>---</b>	<b>---</b>	<b>---</b>
<b>1022</b>	1023	16353	16354	16355	---	16382	16383	16384
<b>1023</b>	1024	16369	16370	16371	---	16382	16383	16384

**OUT Area Modbus TCP Addresses (Fieldbus READ Locations 0 – 1023\*)**

<b>OUT Area Location</b>	<b>Modbus Word Address</b>	<b>Modbus Bit Address</b>						
		<b>Bit 15</b>	<b>Bit 14</b>	<b>Bit 13</b>	<b>---</b>	<b>Bit 2</b>	<b>Bit 1</b>	<b>Bit 0</b>
<b>0</b>	1025	16385	16386	16387	---	16398	16399	16400
<b>1</b>	1026	16401	16402	16403	---	16414	16415	16416
<b>---</b>	<b>---</b>	<b>---</b>	<b>---</b>	<b>---</b>	<b>---</b>	<b>---</b>	<b>---</b>	<b>---</b>
<b>1022</b>	2047	32737	32738	32739	---	32750	32751	32752
<b>1023</b>	2048	32753	32754	32755	---	32766	32767	32768


\* Assumes Word Mode Addressing is selected in Configuration Software

The Fieldbus IN and OUT areas can be configured to a maximum size of 1024 words (2048 bytes) each, depending on the configured fieldbus mappings. The highest mapped location will correspond to the highest available Modbus register (or coil) available to a Modbus/TCP client. A Modbus/TCP client must use the appropriate Modbus Coil or Modbus Word addresses corresponding to configuration software, as well as the correct function code (see 4.12.2 Supported Commands).

Ethernet Configuration

**Ethernet Configuration for: XYR 300G#1**

**Radio Interface**



**I/O Register** (16 bit)  
4320

**I/O Count**  
8

**Command Type**

← Read

Transfer Mode:


☒ Single Bit Mode

☐ Byte Mode (8 bits)

☐ Word Mode (16 bits)

**Fieldbus Interface**

☒ MS Bit First



**I/O Location** (16 bit)  
16385

✓ Apply

✗ Cancel

Fieldbus Register Selection

**I/O Location:**

- Word Lcn's 16385-17177
- Word Lcn's 17185-17977
- Word Lcn's 17985-18777
- Word Lcn's 18785-19577
- Word Lcn's 19585-20377
- Word Lcn's 20385-21177
- Word Lcn's 21185-21977
- Word Lcn's 21985-22777
- Word Lcn's 22785-23577
- Word Lcn's 23585-24377
- Word Lcn's 24385-25177
- Word Lcn's 25185-25977
- Word Lcn's 25985-26777
- Word Lcn's 26785-27577
- Word Lcn's 27585-28377
- Word Lcn's 28385-29177
- Word Lcn's 29185-29977
- Word Lcn's 29985-30777
- Word Lcn's 30785-31577
- Word Lcn's 31585-32377
- Word Lcn's 32385-33177

**Read Area Location**

16385		16401		16417		16433		16449	
16465		16481		16497		16513		16529	
16545		16561		16577		16593		16609	
16625		16641		16657		16673		16689	
16705		16721		16737		16753		16769	
16785		16801		16817		16833		16849	
16865		16881		16897		16913		16929	
16945		16961		16977		16993		17009	
17025		17041		17057		17073		17089	
17105		17121		17137		17153		17169	

Usage for BIT Locations 16385 - 16400.

BIT Locations 16385 - 16392 are Selected.

Location+	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
N/A																
▶ 16385																
16401																

✓ OK

✗ Cancel

Current Selection = BIT Location 16385 to BIT Location 16392

Appropriate Modbus prefixes may need to be added to the Modbus Address depending on the host device. For example, a “word write” fieldbus mapping in the XYR 300G to Modbus location 10, can be read by a host device as 30010 (30000 for an input register + 10 as the address). Alternatively, a “word read” fieldbus mapping in the XYR 300G from Modbus Location 1025, can be written to by a host device as 41025 (40000 for an output register + 1025 as the address).

Conversely, for Modbus bit/binary commands the appropriate 0x or 1x prefix may need to be added depending on the host device. The example below shows 8 bits being read from Modbus locations 16385 – 16392 into I/O registers 4300 – 4307 (DOT 1-8). The Modbus/TCP host device would write to these as Modbus addresses 016385 – 016392 (using the 0x prefix to denote output coils).

### **Connect Timeout**

The Connect Timeout parameter in the Modbus TCP section of the display refers to the Modbus TCP functionality of the module. If a TCP connection to the module has not been active for this amount of time, the XYR 300G will timeout and disconnect that connection. Note that there can be several active connections at the same time - only the inactive connection will be disconnected.

IO (i.e. IO data configured using fieldbus *read* commands). The same IO are also available in the vendor specific objects I/O data input mapping object (class A0h) and IO data output mapping object (class A1h) respectively as instance attributes 1 – 6. (See Object Specifications below)



## 4.10 Connecting Serial I/O

NOTE – Serial I/O Expansion is only possible for XYR 300G Firmware versions 1.50 onwards.

Up to 31 x Serial addresses can be connected to each XYR 300G depending on, I/O count and module type being used.

To enable the XYR 300G serial port for Serial expansion, select the “Enable Serial Expansion” box in the configuration software.

Note that enabling Serial expansion also disables on-line diagnostics via the serial port.

Mapping to or from the Serial I/O is the same as if the Serial modules are connected to a

XYR 300X

I/O module.

Each Serial

module has

an address

between 96

and 127.

The XYR

300G acts as

a repeater

address in

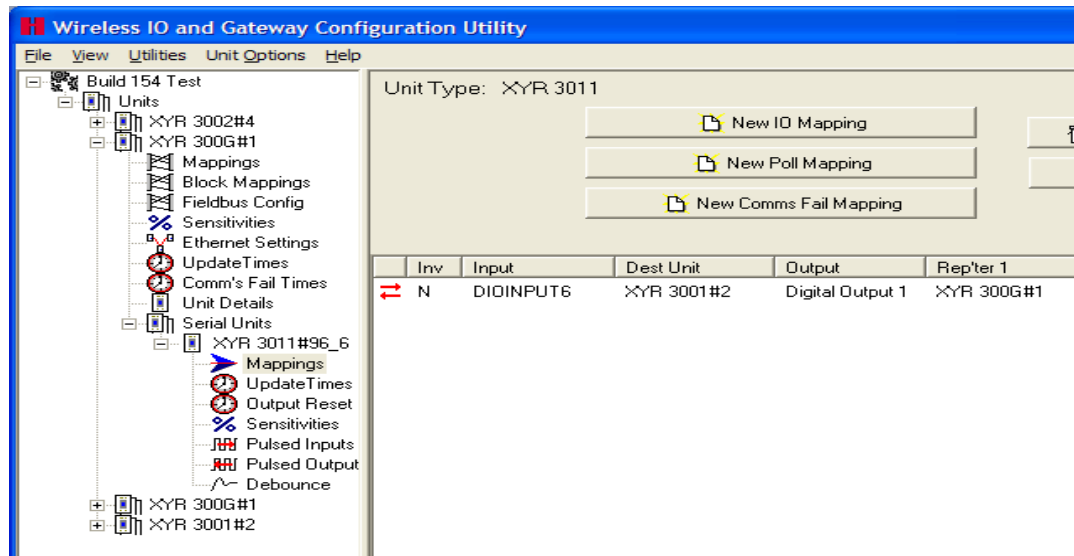
the mapping.

The XYR

300G I/O

registers can

also be mapped to/from the Serial Modules.



## 4.11 Access to Message Buffer Count

---

The number of messages in buffers is stored in I/O registers for access from the data bus. This provides a powerful diagnostics feature for troubleshooting busy systems. The number of “free” messages is also provided - this is the amount of space available in the message buffers.

<b>I/O Reg</b>	<b>Description</b>
----------------	--------------------

4350	– Number of Free COS (change-of-state) messages (max. is 1500)
------	--

4351	– Number of Free Block Messages (for queuing block mappings and repeated messages – max. is 200)
------	--

4352	– Number of Free Ack Messages (max. is 10)
------	--

4353	– Number of Free “Rx Messages for Ethernet Monitor Comms only” (Max is 20)
------	--

4354	– Repeater messages queue (number of queued messages to be Repeated waiting to be sent)
------	---

4355	– Block Message queue (number of block mappings queued waiting to be sent)
------	--

4356	– COS Message queue (number of COS messages queued waiting to be sent)
------	--

4357	– Update Message queue (number of update messages queued waiting to be sent)
------	--

4358	– ACK queue (number of ACK messages queued waiting to be sent)
------	--

4359	– Radio Data Change queue (number of COS received on radio waiting to be sent through to fieldbus)
------	--

The following four are buffer empty alarms (i.e. hex 0000 for OK, hex FFFF for buffer empty)

4360	– Free COS message buffer empty alarm (i.e. triggered when reg 4350 is 0)
------	---

4361	– This register counts the number of times the above alarm has been triggered
------	---

4362	– Free Block message buffer empty alarm (i.e. triggered when reg 4351 is 0)
------	---

4362	- This register counts the number of times the above alarm has been triggered
------	---

4364	- Free COS buffers empty latch.
------	---------------------------------

4365	– Free Block Message empty latch
------	----------------------------------

# Chapter 5

# SPECIFICATIONS

<b>General</b>		
Housing	130 x 185 x 60mm DIN rail mount Refer section 5.1 for dimensioned drawing	Powder-coated, extruded aluminum
Terminal blocks	Removable	Suitable for 12 gauge / 2.5 mm <sup>2</sup> conductors
LED indication	Power supply/OK, Active operation, digital I/O, Radio RX and TX, Serial RX and TX	
Operating Temperature	XYR 300G-M XYR 300G-E	-40 to 140 degF, -40 to 60 degC 30 to 140 degF, 0 to 60 degC
Humidity	0 – 99% RH non-condensing	
<b>Power Supply</b>		
Battery supply	11.3 - 15.0 VDC	
AC supply	12 - 24 VAC, 50/60 Hz	Overvoltage protected
DC supply	9 - 30 VDC	Overvoltage and reverse voltage protected > 17VDC required for charging battery
Battery Charging circuit	Included, suitable for 12Vsealed lead acid batteries	Regulated to max 1.5 amp charging current
Normal Current Drain at 12VDC	XYR 300G-M XYR 300G -other	150 mA 270 mA 270 mA add 5mA per active I/O
Normal Current Drain at 24VDC	XYR 300G-M XYR 300G -other	90 mA 170 mA add 3mA per active I/O

Radio transmitter inrush	XYR 300X 900MHz License-free	350mA @ 13.8VDC; 250mA @ 24VDC
	Fixed frequency licensed	450mA @ 13.8VDC (0.5W) 600mA @ 13.8VDC (1W) 800mA @ 13.8VDC (2W) 1.25A @ 13.8VDC (5W)
Power fail status	Monitored	Can be transmitted to remote modules
Battery voltage	Monitored	Analog value can be transmitted Low voltage status can be transmitted
<b>Radio Transceiver (XYR 300X)</b>		
<b>900MHz License-free</b>		
XYR 300X Radio standards	FCC Part 15A, Part 15.247	902 – 928 MHz, 1W
Frequency	USA/Canada Australia New Zealand	902 – 928 MHz 915 – 928 MHz 922 – 928 MHz
Transmission Power	1W	
Signal detect / RSSI	-120 to -40 dBm	
Expected line-of-sight range (subject to local conditions)	20 miles + @ 4W ERP 15 km + @ 1W ERP depending on local conditions	USA / Canada Australia / New Zealand Range may be extended by up to 5 intermediate modules as repeaters
Antenna Connector	Female SMA coaxial	
Data transmission rate	19200 baud	
<b>Fixed Frequency Licensed</b>		
Fixed Frequency	Channel spacing 12.5 / 25 KHz	150 – 230MHz 360 – 512MHz
Radio standards	FCC Part 90, Part 15, RSS-119	150 – 230 MHz, 360 – 512 MHz, 12.5 / 25KHz, 0.5 – 5W
Transmission Power	Configurable	0.5 – 5W
Signal detect / RSSI	-120 to -50 dBm	
Expected line-of-sight range (subject to local conditions)	70 miles @ 10W ERP 25 miles @ 2W ERP depending on local conditions	ERP allowed depends on license conditions Range may be extended by up to 5 intermediate modules as repeaters
Antenna Connector	Female SMA coaxial	
Data transmission rate		4800 b/s

<b>Serial Ports</b>		
RS232 Port	DB9 male DCE	RTS/CTS hardware signals provided
RS485 Port	2 pin terminal block	Typical distance 1 - 2 km
Data rate (bit/sec) -	9600	
Byte format	8 data bits	1 Stop, 1 Start, No parity bits
<b>Ethernet Port</b>		
RJ45	Transformer isolated	10/100 Mbit/sec
<b>Digital I/O</b>	Eight on-board I/O	3000V surge protection input, voltage free contact output, FET 30VDC 500mA

# Chapter 6

# DIAGNOSTICS

Before installing a new system, it is always best to set up the system on a bench to test the system configuration. It is always easier to detect problems when the modules are together.

After installation, test the radio paths, using the radio strength testing function described later in this section. Record the radio strength and background noise measurements for later reference (refer section 6.2.2 for this feature). If a later test shows that the radio path has changed, this may be the cause of a new problem.

## 6.1 Diagnostics Chart

The LED indicators on the XYR 300G have the following meanings: -

INDICATOR	CONDITION	MEANING
OK	OFF continuously ON continuously	Module power off, or module failure Normal Operation
RADIO TX	Flashes yellow	Radio transmitting
RADIO RX	Flashes green Flashes red	Radio receiving good radio signal Radio receiving weak radio signal
SERIAL TX	Flashes yellow Brief flash each second	Sending serial data Configuration Mode
SERIAL RX	Flashes green Flashes red	Receiving serial data Serial RX buffer full
ACTIVE	OFF continuously ON continuously Flashes Yellow	Start-up initializing sequence Diagnostic or configuration menu Module in active operation Re-configuration required

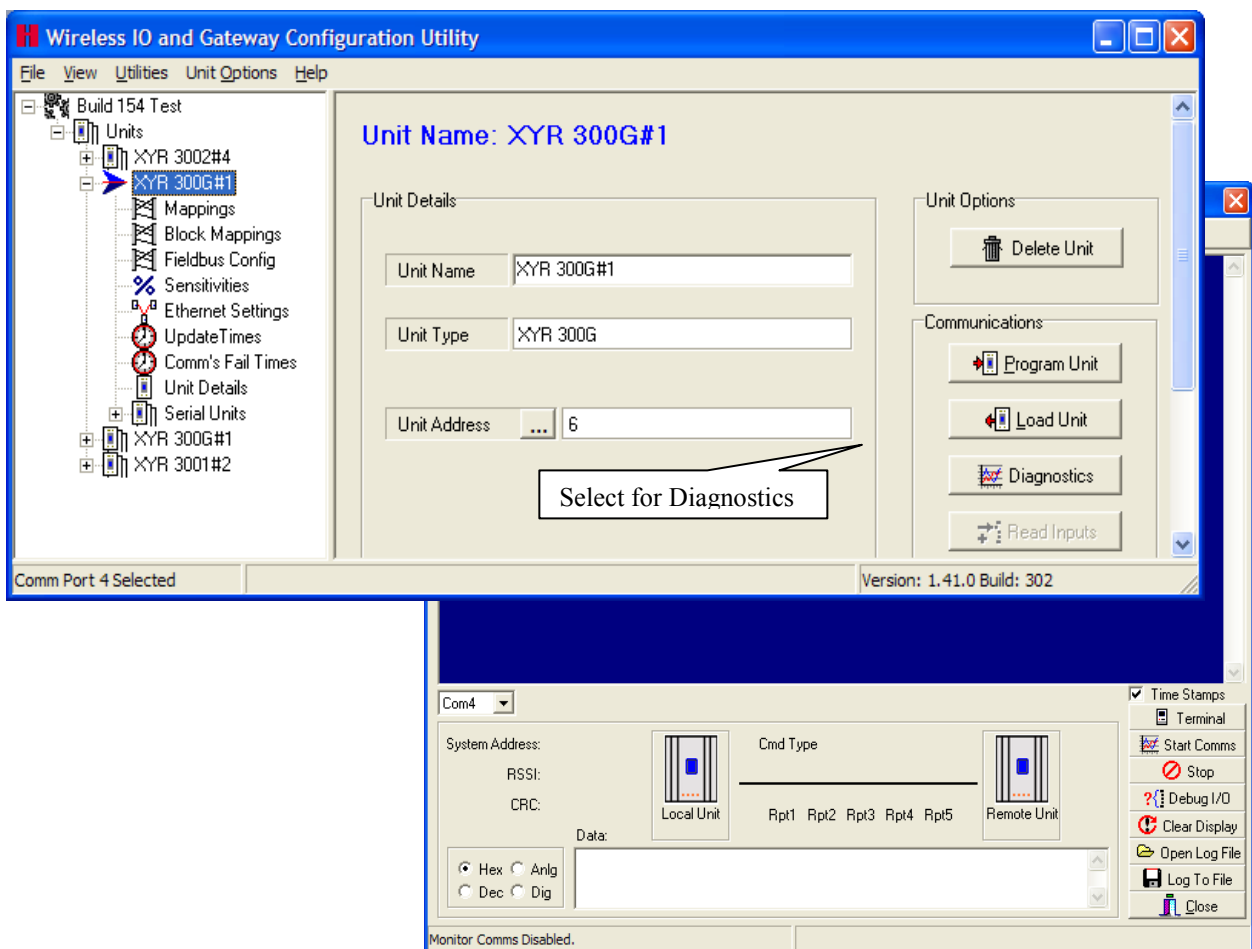
The Ethernet module also has four diagnostic LED's on the end-plate - refer section 6.4.

## 6.2 Diagnostics Menu

The XYR 300G provides both offline and online diagnostic features to assist with troubleshooting. The offline diagnostics disable both the radio and fieldbus interface drivers, and are only used for simple radio tests such as “RSSI Measurement” or “Tone Reversals”. The online diagnostics provide more powerful debugging features such as access to the internal I/O Registers and Radio Communications Monitoring while the module is running (i.e. online).

When the configured protocol driver uses the shared RS232/485 port (i.e. Modbus, or Honeywell Serial Driver), the online diagnostics must disable the serial protocol driver since the same serial port must be made available for diagnostics. However, the diagnostics still has full access to the radio network. For all other protocol drivers (Ethernet), the serial port is already free and therefore online diagnostics can be used while the module is fully operational.

The module diagnostics can be accessed via any ‘terminal’ package (i.e. HyperTerminal), or via configuration software using the terminal available in the “Diagnostics” section. First, ensure that the XYR 300G is connected to the PC using the RS232 configuration cable, and that the corresponding com port is selected. To access the terminal, select the XYR 300G and press “Diagnostics”. Press the “terminal” button in the diagnostics window to open the terminal.



### 6.2.1 Offline Diagnostics:

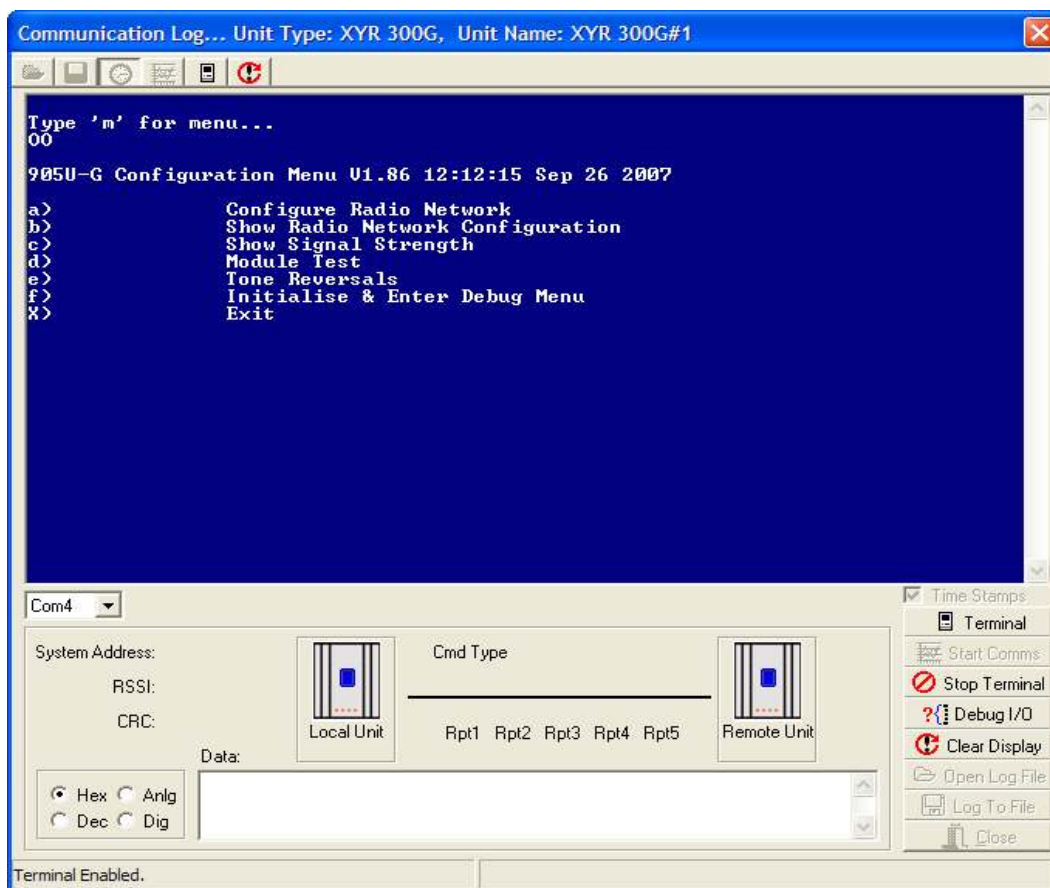
The offline diagnostics menu disables the radio protocol driver *and* the fieldbus protocol driver. Before displaying the offline debug menu open and start the “terminal” window in configuration software (see above), or use any third party terminal package.

To display the offline diagnostics menu:

- Put the XYR 300G into configuration mode by pressing the small pushbutton switch in the end plate of the module for 5 seconds (as per section 4.9) until the ACT led flashes - then release (then the ACT is off and the Serial TX LED flashes once every second);
- Type ‘m’ in the terminal window to get the off-line diagnostics menu.

The module will stop normal operations and a menu like the following will appear on the PC screen for all XYR 300G versions.

**Note:** Options a), b) and d) are used in factory test and should not be selected.



#### (c) Show Signal Strength

This option allows measurement of radio path between two locations. This is done by the display of the received radio signal strength at the connected XYR 300G. With no transmitted signal from the other site, the display will show the strength of the background noise, which is normally between -100 and -130 dBm. At the other site, the transmitter may be turned on (select “e” at



the other XYR 300G, or “Tone Reversals” if the other module is a XYR 300X). The display will now show the received radio signal from the other transmitter.

The display will initially show the background noise of the radio band. Determine the approximate average of the noise level. The remote unit may then be set up for tone reversals (refer below). Determine the approximate average of the received signal strength. It is normal for the measured values to continually change - the radios are continually changing frequency. Calculate the best average for both the noise and signal.

For reliable operation, the average signal strength should be better than -98dBm (that is, -90dBm, not -100dBm) provided the average background noise is less than -108dBm (between -108 and -130 dBm). If the average noise is greater than -108, the difference between the noise level and the transmitter signal should be at least 10dB for reliable operation. For example, if the average noise level is -101dBm, then a transmitter signal of better than -91dBm is required for reliable operation.

*Note the RSSI (received signal strength indication) of a received message is also stored in the database registers when the module is online - refer to section 2.5.1*

#### **e) Tone Reversals**

If you select this option, the module will continuously transmit - you can use this feature for radio tests. Note that if you are powering the module from a battery only, the battery will be discharged quickly.

#### **f) Initialize and Enter Debug Menu**

This option will put the XYR 300G in *online* debug mode. In online mode, the module will initialize the radio driver and go *online* to the radio network. Where possible, the fieldbus driver will also be initialized (i.e. for Ethernet) – for Modbus, and Honeywell Serial Driver the fieldbus driver will be disabled so that the serial port can be used for diagnostics. Note: before going online, the XYR 300G must complete any “startup polls” that are configured – this may take some time depending on how many polls are configured.

#### **x) Exit**

The module will restart via its normal power-up and initialization sequence, and resume its normal operation mode. Select “Stop Terminal” to shut down the terminal and close the com port.

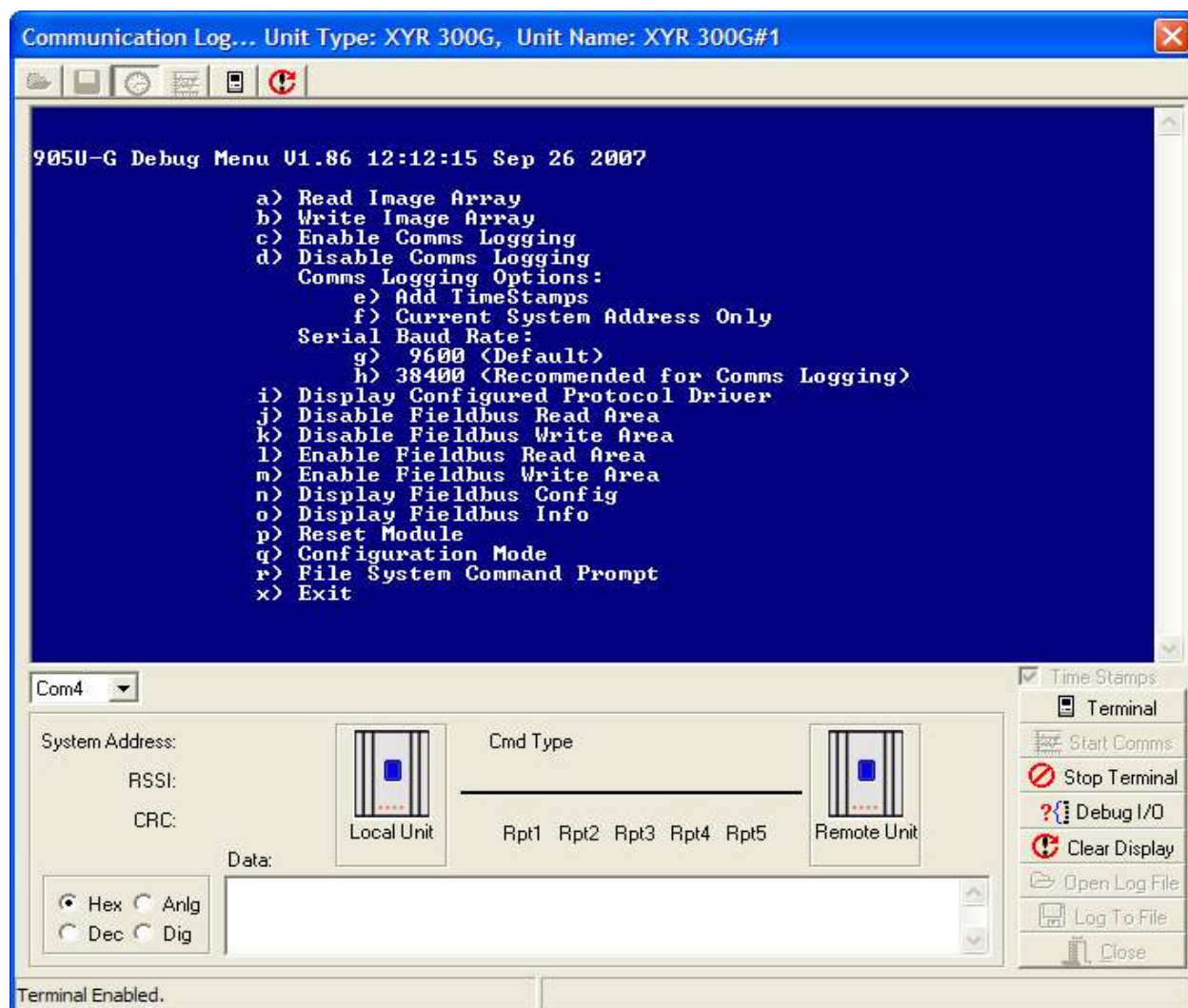
### **6.2.2 Online Diagnostics**

The online diagnostics menu enables the radio protocol driver *and* the fieldbus protocol driver (where possible) to provide online diagnostic information while the module is running. When the configured protocol driver uses the shared RS232/485 port (i.e. Modbus, or Honeywell Serial Driver), the online diagnostics must disable the serial protocol driver since the same serial port must be made available for diagnostics.

To access the online diagnostics menu, first connect to the “terminal” in configuration software (see above) or use any third party terminal package. Once the terminal is connected, display the menu using the following procedure:

- If the configured protocol driver uses the shared RS232/485 port (i.e. Modbus, or Honeywell Serial Driver), first enter the *offline* diagnostics menu (see 6.2.1 “Offline Diagnostics” above). From the offline menu, select option “f) Initialize and Enter debug Menu”. Once initialized the online menu will be displayed.
- For Ethernet protocol drivers, simply press "Enter" to display the menu. If the module was previously in configuration mode or the offline menu, then first reset power to the module.

A menu like the following will appear on the PC screen for the all models, however Modbus model will not have options h) through k).



The online diagnostics menu is also referred to as the “Debug” menu. The Debug Menu allows the Radio Interface (I/O Registers) to be viewed and modified to confirm the operation of the radio network. These options may be used to check operation of outputs at remote sites, and to check the values of inputs reported from remote sites. When the protocol driver does *not* use the

shared RS232/485 port (Ethernet version only) data is also exchanged with the fieldbus and the I/O Registers according to the configured fieldbus mappings.

### **Option a) Read Image Array**

Displays the I/O registers of the Radio Interface - the register values for a block of 50 registers are updated every 1 second. For example, to display the I/O Database value at locations 0 to 49.

Select a), then enter Location: 0

```
0 0x0000 0x0000 0x0000 0x0000 0x0000 0x0000 0x0000 0x0000 0x0000
10 0x0000 0x0000 0x0000 0x0000 0x0000 0x0000 0x0000 0x0000 0x0000
20 0x0000 0x0000 0x0000 0x0000 0x0000 0x0000 0x0000 0x0000 0x0000
30 0x0000 0x0000 0x0000 0x0000 0x0000 0x0000 0x0000 0x0000 0x0000
40 0x0000 0x0000 0x0000 0x0000 0x0000 0x0000 0x0000 0x0000 0x0000
```

Note that I/O Image locations are specified in decimal, whereas register values are displayed and specified in hexadecimal. If you want the XYR 300G to stop the host device writing values to the I/O database at the same time, then select option i) Disable Fieldbus Write Area.

Press “Enter” to go back to the menu.

### **Option b) Write Image Array**

This option allows you to change the value of an I/O register in the Radio Interface.

To change the value of a register, select option b) write image array.

Enter the location, then the value to be written to the register – for example

**b**

Location: **12**

New Value: 0x**FFFF**

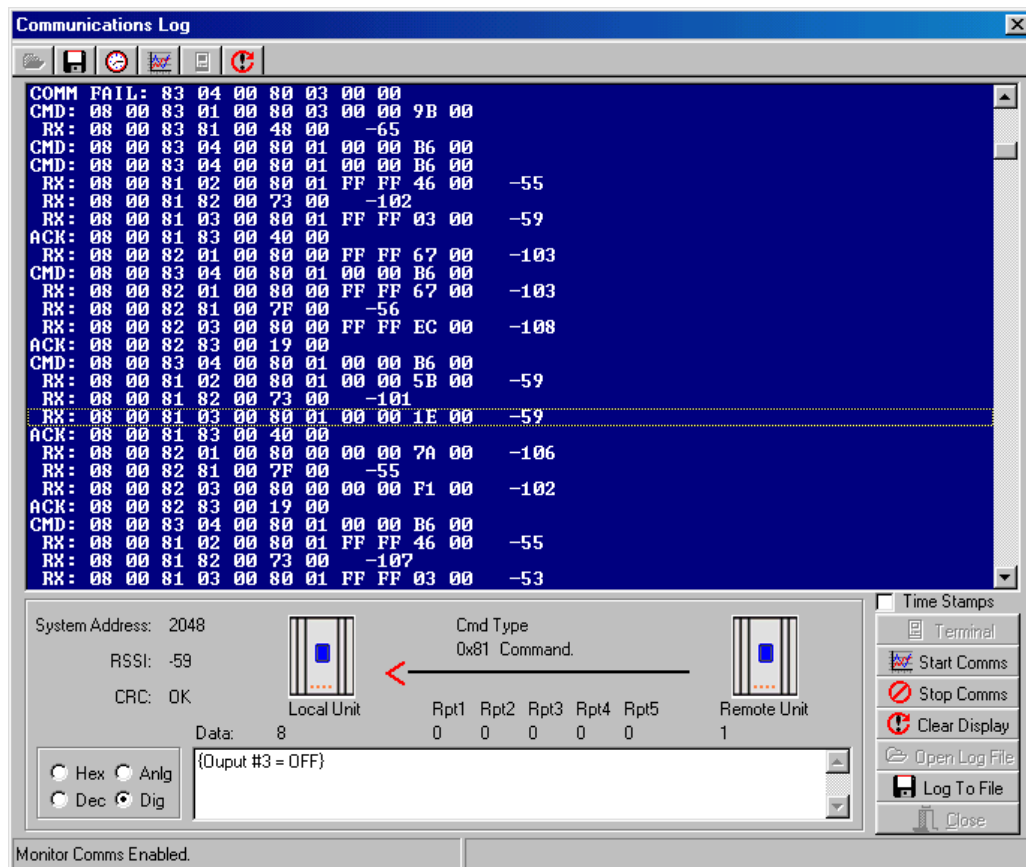
Register values should always be written in hexadecimal format. If you want the XYR 300G to stop the host device reading or writing values to the I/O database at the same time, then select option h) or i).

### **Options c), d) Enable/Disable Comms logging**

These options allow logging and display of radio communications. Once enabled, the radio communications that are displayed is the radio traffic in raw format (i.e. the raw data frame for each received packet is displayed in hexadecimal format). To decode the meaning of each radio data packet, configuration software can be used to decode the data frames.

- To decode the radio traffic using configuration software, first start communications logging by selecting option “c) Enable Comms Logging” from the debug menu. Next, in the diagnostics screen select ‘Stop Terminal’ and then ‘Start Comms’. Configuration software now expects the XYR 300G to be in monitor comms mode, and will decode all radio communications.

The display will show radio messages transmitted and received. Messages starting with RX are received messages, CMD are transmitted messages and ACK are acknowledgment messages. At the end of each received message is the RSSI (radio signal strength indication) in dBm.



If you select any message line with the mouse, information about the message will be displayed at the bottom of the screen - the system address, RSSI and CRC (error-check) status. The “text box” at the bottom middle of the screen decodes the message - that is, it decodes the message to display I/O channel and value. Note – Configuration software can only decode the message completely if the same configuration project corresponding to the system being monitored is open.

You can display the register values in Decimal by selecting “Dec” at the bottom of the screen. If you select “Dig”, the values will be displayed as a 0 or 1 digital value (1 if the 16-bit value is greater than 50% - that is, the most significant bit is 1). If you select “Anlg”, the value will be displayed as a 4-20mA range.

To stop the decoding of “comms logging”, select the “Stop Comms” button. You should then also stop the XYR 300G from outputting radio comms by pulling up the terminal menu (i.e. press “terminal” and then hit enter in the terminal screen) and selecting “d) Disable Comms Logging”.

Further Information in for Comms logging using different XYR 300G module types can be found in Appendix 3

### **Option e) Add Time Stamps**

This option in the debug menu will add a timestamp to each displayed radio message. The timestamp is based on the XYR 300G internal real time clock. This option is normally used only if monitoring is done from a terminal package only, and configuration software is *not* being used to decode the communications.

When configuration software is being used to decode the radio comms (see above) time stamps can be added by selecting the “Time Stamps” checkbox. This will display the current time and date (according to the PC Clock) alongside each message. The “Comms log” can be saved to a file for future reference by selecting “Log to File”.

### **Option f) Current System Address only**

This option will ensure that only radio messages that have the same system address as the connected XYR 300G are displayed. If you have another system with a different system address these messages will not be displayed if you choose this option. This option is useful where there is more than one system in the same area so that only the radio messages relevant to the desired system will be displayed.

### **Option g) Display Configured Protocol Driver**

This option displays the configured Protocol Driver for this unit e.g.

Configured Protocol is: Ethernet TCP-IT

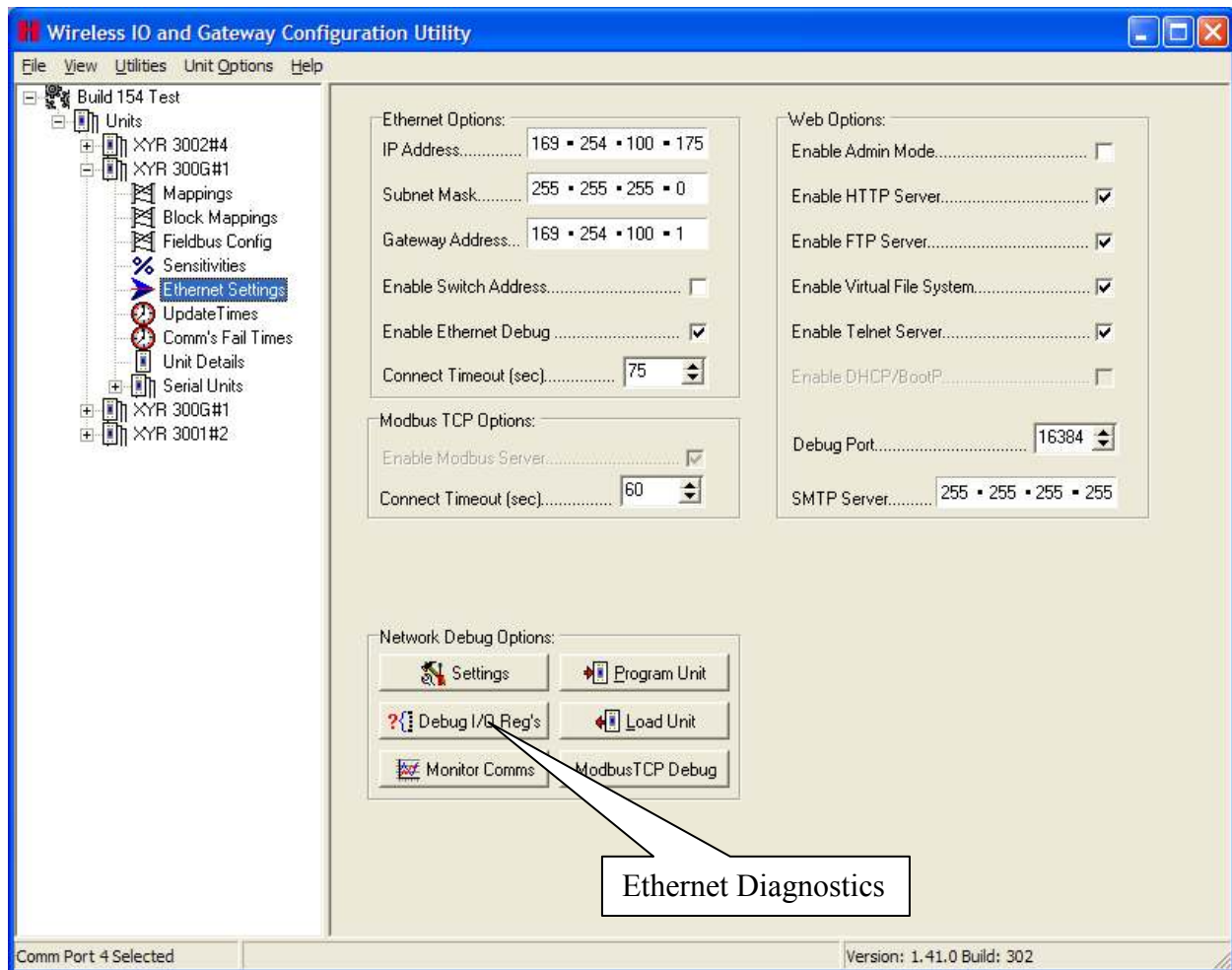
### **Option h, i, j, k) Enable/Disable Fieldbus Read/Write Area**

(These options not available on the XYR300G--M version)

This option is used to halt data exchange between the Fieldbus Interface and the Radio Interface (I/O Registers). This is mainly used when trying to read or write image arrays. If the Fieldbus read area “h” is not disabled when trying to read or write to the I/O registers then the value in the Fieldbus database will overwrite the I/O register and you may get an incorrect value.

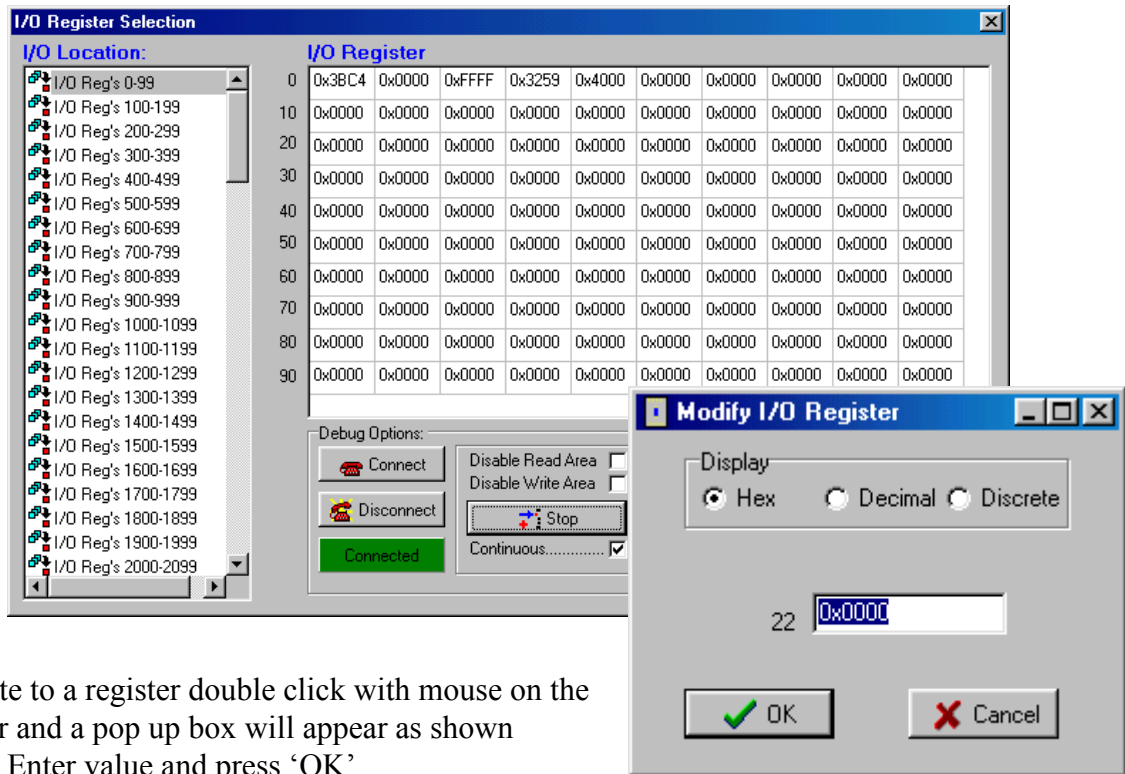
*When doing read/write image array and the module has been configured with Fieldbus mappings, you may need to disable the Fieldbus read area option h. This stops the Fieldbus database overwriting the radio database.*

## 6.3 Ethernet Diagnostics



Read and Write image array can also be done via the Ethernet port by selecting 'Debug I/O Registers' from within the Ethernet Settings window in the configuration software. The IP address of the module must have previously been configured in the module - refer to section 4.8.2 for setting IP address.

To debug the registers you will need to select 'Connect' under Debug Options. The Green / Red box will indicate the Connected / Disconnected State. Once connected select "Read" and check "Continuous". The display option allows you to view the registers in different formats, and you can select which I/O register you want to view from the left-hand side of the screen.



To write to a register double click with mouse on the register and a pop up box will appear as shown below. Enter value and press ‘OK’.

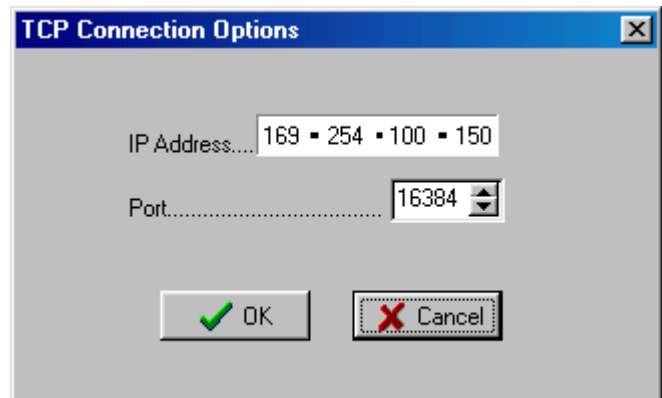
You can disable the links between the I/O registers and the Ethernet interface by selecting “Disable Read Area” and “Disable Write Area” - if you do this, remember to re-enable before you leave the diagnostics screen.

### Settings

You can change your TCP Connection setting i.e. IP address and port.

### Program / Load Unit

These options allow you to program and upload the configuration from the module via the Ethernet port. Must ensure the IP address has been set on the module before uploading the Configuration



### Monitor Comms

Configuration software also provides the option to monitor the radio network communications via the Ethernet port. This allows radio traffic to be monitored from any location where an Ethernet connection to the XYR 300G can be established. Simply select ‘Monitor Comms’ from the Network Debug Options section of the Ethernet Settings page. Functionality is as per section 6.2.2c above.

## 6.4 Fieldbus Indicating LEDs

All XYR 300G modules (*except* MD1) are equipped with four fieldbus indication LED's located in the module end plate, used for diagnostics purposes. The meaning of the LED's for each fieldbus is described below.

### 6.4.1 Ethernet Indicating LED's

The XYR 300G-E module can communicate Modbus TCP or EtherNet IP based protocols. The LED sequence will vary depending on Protocol being used and Ethernet Board firmware.

Changes to Ethernet Board firmware effect LED 2 indications when flashing. When flashing this only indicates that an EtherNet IP connection has not been made to the XYR 300G-E. Example would be RSLogix has not had a Generic Ethernet Device added with XYR 300G-E IP address and connection data.

If using Modbus TCP then LED's 2 & 3 have no function in regards to the Modbus TCP communications of the module



Led No	Colour	State	Description
1	Green	-	The Link led indicates that the module is connected to an Ethernet network.
2	Green	Off	No power applied to module.
2	Green	Steady	Device operating correctly.
2	Green	Flashing	Module configured, Scanner in Idle State ( <b>EtherNet IP only</b> )
2	Red	Flashing	Minor recoverable fault has been detected.
2	Red	Steady	Major internal error has been detected.
2	Green/Red	Flashing	Power on self-test.
3	Green	Off	No power applied or no IP address has been assigned.
3	Green	Steady	Module has at least one Ethernet/IP connection established.
3	Green	Flashing	No Ethernet/IP connections to the module.
3	Red	Flashing	Connection timeout
3	Red	Steady	Duplicate IP address
3	Green/Red	Flashing	Power on self-test.
4	Green	Flashing	Flashes each time a packet is received or transmitted.



## 6.5 Radio Path Testing

---

To carry out a radio path test, you will need two XYR 300X modules. One module will be “fixed” and the other “mobile”. Both units will need power supplies and antennas. The power supply for the mobile unit is normally a 12V battery, but make sure that the battery is fully charged - batteries with low voltage will lead to low radio power which will affect the test result.

The object of the test is to determine whether radio paths are reliable, marginal or unreliable. A reliable path will have a margin of at least 10dB above the background noise level in good weather - this margin is enough to ensure that the radio path remains reliable in poor conditions. A marginal path will work reliably in good conditions, however will fail during poor conditions. If the test is carried out during rainy or foggy weather, then a margin of only 5dB is required.

Procedure:

- ❑ Configure the modules to the same system address, and on each module, configure DI1 to DO2 on the other module. At the fixed module, wire DO2 to DI1 such that DI1 will turn ON when DO2 turns ON. Connect a switch to DI1 on the mobile unit.
- ❑ When the modules are close to each other, test the system - close the switch, forcing the mobile unit to transmit. The mobile unit will transmit to the fixed unit, and the fixed unit will transmit back to the mobile unit, activating DO2. Turning off the switch will result in two radio transmissions, turning off DO2. Each time the switch is changed, there should be two radio messages (two sets of TX/RX flashes) at the mobile unit. Note that when the modules are within a couple of metres, they may not work well with antennas connected - in this case, test without antennas.
- ❑ Set up the fixed module in one of the test positions - this is normally at a control centre or repeater site. Fix the antenna in a temporary fashion. You will need to make an initial assessment on how high the antenna should be mounted.
- ❑ Take the mobile module to the other end of the radio path. The antenna at this end can be either held by the tester, or fixed in a temporary fashion. Note that a person’s body will affect the radiation pattern of an antenna, so if the antenna is hand-held and the test is not successful, try again with the antenna fixed to a 1 metre length of plastic pipe or timber. The tester holds the length of pipe or timber with the antenna above head height.
- ❑ Test the radio path by operating the switch. If the radio path is short, and there is a high level of confidence that the radio path will be reliable, the result can be checked by simply looking at the TX/RX leds on the mobile unit. If each TX flash is followed immediately by a RX flash (that is, the TX flash does not flash twice or more times before the RX flashes), then the radio path is likely to be reliable. Operate the switch several times - do not rely on one test. If the test is being done outside, the leds will need to be shaded to view the flashes.
- ❑ If the radio path is uncertain, then the result should be measured by connecting a laptop computer, following the procedure outlined in this manual for measuring the radio signal strength. Before the switch is operated, the background noise level should be measured and recorded. This measurement is likely to “jump around” or oscillate, to determine an

average measurement. Now operate the switch several times - take the average measurement of the signal transmitted from the fixed unit.

- ❑ The radio path is reliable if the transmitted signal is 10dB above the noise level, or better than -98dBm. For example, if the noise level is -115dBm, then the minimum level for reliability is -98dBm. If the noise level is -100dBm, then you need -90dBm for a reliable path. If the laptop displays a scale measurement instead of a numerical measurement, then the transmitted signal should be at least 3 divisions, and at least 2 divisions above the noise level.
- ❑ If the weather is poor during the test, then the transmitted signal needs to be 5dB above noise, or 1 division. It is best not to do radio tests during poor weather.
- ❑ Record these measurements for comparison later during commissioning or if the system has problems later.

If the radio path test is not successful:

1. Increasing the height of the antenna at either module, or at both modules can significantly improve the result. Sometimes moving the antenna to the side helps, if there is an obvious obstruction in the radio path.
2. Change one or both antennas to a higher gain if regulations allow.
3. Use a shorter coaxial cable between the antenna and the XYR 300X.(this may involve moving XYR 300X nearer to antenna mounting), or use a different coaxial cable with lower loss.
4. If a reliable radio path is not possible because of distance or path obstructions, you will need to consider using a repeater module. The ideal repeater is another module in the system, in a good location to act as a repeater. If this is not the case, you need to consider installing a module to act specifically as a repeater.

# Chapter 7

# WARRANTY

We are pleased that you have purchased this product.

Honeywell products are warranted to be free from manufacturing defects for the “serviceable lifetime” of the product. The “serviceable lifetime” is limited to the availability of electronic components. If the serviceable life is reached in less than 18 months following the original purchase from Honeywell, Honeywell will replace the product with an equivalent product if an equivalent product is available.

This warranty does not extend to:

- Failures caused by the operation of the equipment outside the particular product's specification, or
- Use of the module not in accordance with this User Manual, or
- Abuse, misuse, neglect or damage by external causes, or
- Repairs, alterations, or modifications undertaken other than by an authorized Service Agent.

Honeywell’s liability under this warranty is limited to the replacement or repair of the product. This warranty is in lieu of and exclusive of all other warranties. This warranty does not indemnify the purchaser of products for any consequential claim for damages or loss of operations or profits and Honeywell is not liable for any consequential damages or loss of operations or profits resulting from the use of these products. Honeywell is not liable for damages, losses, costs, injury or harm incurred as a consequence of any representations, warranties or conditions made by Honeywell or its representatives or by any other party, except as expressed solely in this document..

Full product specifications and maintenance instructions are available from your Service Agent, your source of purchase, or from the master distributor in your country upon request and should be noted if you are in any doubt about the operating environment for your equipment purchase

In the unlikely event of your purchase being faulty, your warranty extends to free repair or replacement of the faulty unit, after its receipt at the master distributor in your country. Our warranty does not include transport or insurance charges relating to a warranty claim.

Should you wish to make a warranty claim, or obtain service, please forward the module to the nearest Honeywell Sales office or sales distributor along with proof of purchase.

# Appendix 1

# Status Registers

## I/O Status Registers 5000 - 9499

Bit	Information	Meaning										
15	Communications failure	For inputs, this bit is set (“on”) if no message has been received from the remote address within the timeout period configured for this input. The bit is reset ("off") when a message is received. For outputs, this bit is set (“on”) if transmission to the remote was unsuccessful after five attempts. The bit is reset ("off") when a message is transmitted successfully. This bit may also be set if the Disable Output Transmissions on Comms Fail option is selected - see the Radio Comms Failure section.										
14	Start-up status	For inputs, this bit remains set ("on") following start-up until a message has been received for this input to give an initial input value. For outputs, this bit remains set ("on") following start-up until the XYR 300G sends the first radio message for this output to the remote address.										
13	Input / Output status	This bit is set ("on") if this I/O point has been configured as an input.										
12	Active status	This bit is set ("on") if the register has been configured as an I/O point.										
11-10	Timer Units	This field determines whether the timer counts down every 10 seconds, every minute, or every hour.  <table><tr><td>Timer Units</td><td>Timer timebase</td></tr><tr><td>00</td><td>Every 10 seconds</td></tr><tr><td>01</td><td>Every minute</td></tr><tr><td>10</td><td>Every Hour</td></tr><tr><td>11</td><td>Every Hour</td></tr></table>	Timer Units	Timer timebase	00	Every 10 seconds	01	Every minute	10	Every Hour	11	Every Hour
Timer Units	Timer timebase											
00	Every 10 seconds											
01	Every minute											
10	Every Hour											
11	Every Hour											
9 - 0	Timer	For inputs, the timer value is set to the configured comms fail time for the input whenever a message has been received for this input. The timer value will decrease until another message is received. When the timer value reaches zero, the comms fail status is set. If the configured comms fail time is zero, then the comms fail status for this input is never set.  For outputs, the timer value is set to the configured update time for the output whenever a message is transmitted by the XYR 300G to the remote address. The timer value decreases. When the timer value reaches zero, another update message is transmitted to the remote address. If the configured update time is zero, no update messages are transmitted for this output.										

## Block Message Status Registers 9500 - 9999

Bit	Information	Meaning
15	Communications failure	For read commands – Read Bits and Read Words – This bit is set if no response is received to the read command after a timeout, or if a communication fail response is received to a read.  For Write Commands this bit is set if a communication failure response is received to the write command.  For a Poll command, this bit should not be set.
14	Startup	This bit is set initially, and remains set until the first time the command executes.
13	Force	*To force the command to happen immediately regardless of the current timer value, write a '1' to this bit.
12	Waiting	This bit is set when the command is active. For Write commands, the command delays before sending to see if any more changes occur. For Read commands, the command delays while waiting for a response from the remote device.
11-0	Timer	When the Waiting bit is clear, this field is either zero, or contains the time (in seconds) until the command next becomes active. If this field is zero, the field will be loaded with the configured delay value at the next update time.  When the Waiting bit is set, and the command is a read command, this field contains the time in seconds, within which a reply is expected. If no reply is received within this time, the Communications failure bit is set.  When the Waiting bit is set, and the command is a write command, the field contains the time, in seconds before the write command is transmitted.

### Using the Force Bit:

Only Bit 13 of registers 9500 – 9999 may be altered by a host device (i.e. via the fieldbus interface). For XYR 300G modules with firmware versions later than 1.50, the setting of registers 9500 – 9999 must follow the new change-of-state algorithm. The *Force* bit will only be activated on a transition from 0 – 1. For example to force the corresponding block mapping, first set the Force bit to '0', then set the value of the Force bit to '1' (i.e. by always first writing the value 0 this ensures that the change-of-state from 0 will be detected). Values must be held (i.e. not change) for approx. 200msec to be detected.

## XYR 300G-E Ethernet module only

### Filesystem

The filesystem is a fixed-size storage area with a hierarchical directory structure. Any user- or application data can be stored in files within the filesystem. Files can be grouped in directories for increased readability.

The filesystem features two security levels. Depending on security level, different users can have access to different files and directories. The filesystem can be accessed via FTP, Telnet, and HTTP.

- **Case Sensitivity**

The file system is case sensitive. This means that the file 'Honeywell.txt' is not identical to the file 'HONEYWELL.TXT'.

- **Filename / Pathname length**

Filenames can be a maximum of 48 characters long. Pathnames can be 256 characters in total, filename included.

- **File size**

File size is not restricted. However, the size cannot exceed the space available in the file system.

- **Free space**

Approximately 1.4MB non-volatile (FLASH).

### Security

The file system features two security levels; Admin and Normal. Security level is set at a per user basis, or globally via setting Admin Mode in configuration software Ethernet Settings.

- **Normal Mode**

This mode is recommended for normal operation, so that web pages and other settings are protected from FTP and Telnet access. In this mode, the FTP and Telnet servers are enabled *only* if there is a subdirectory called “\user”. When a normal user connects via FTP or Telnet, this directory will be their root directory. The user will not be able to access files outside this directory and its subdirectories.

If user/password protection for FTP and Telnet is required in normal mode, a file called “sys\_pswd.cfg” must be placed in the directory “\user\pswd\”. Files in this directory cannot be accessed from a web browser. If Admin Mode has not been enabled by configuration software and a valid admin password file (See “System Files”) is found, the module will operate in this mode (i.e. an admin password file with at least one entry *must* exist, *and* the “\user” directory *must* exist to enable this mode).

- **Admin Mode**

Admin users have full access to the filesystem through FTP and Telnet. This enables the user to access areas of the filesystem that are restricted or inaccessible in Normal mode. The Admin user accounts are defined in the file 'ad\_pswd.cfg'.

If no admin password file (See "System Files") is found or Admin Mode is set by configuration software, the module will run in Admin Mode; i.e. all users will have Admin access rights. No login is needed for Telnet, and the FTP server accepts any username/password combination. Admin Mode is primarily intended for product configuration and testing.

Files within the file system can be protected from web (i.e. HTTP) access through username/password authorization, see sections below on "System Files" and "web\_accs.cfg". It is also possible to configure which IP addresses and what protocols are allowed to connect to the module, see "ip\_accs.cfg".

## **System Files**

The module uses system files for configuration purposes (see file system "Structure" below). In most cases these files have the file extension '.cfg' and must be created or edited by the user to achieve the desired configuration. The system files are ASCII (text) files and can be edited with any text editor, or copied/moved to/from the file system using FTP or Telnet. Depending on security settings, the files may be inaccessible for normal users. Generally, the module has to be restarted in order for any changes in these files to have effect.

**Note:** It is very important to follow the exact syntax specifications for each configuration file, otherwise the module might have problems interpreting it, which can result in a faulty or non-expected behaviour.

### **ad\_pswd.cfg & sys\_pswd.cfg**

User/password information for FTP and Telnet is stored in the files 'sys\_pswd.cfg' (Normal users) and 'ad\_pswd.cfg' (Admin users) – see "Security" above. These files must be placed in 'user\pswd' and 'pswd\' respectively. These directories are protected from web browser access.

The file format is the following:

User1:password1

User2:password2

...

User3:password3

*Example:*

Honeywell: XYR 300G

In this example, the username is 'Honeywell', and the password is 'XYR 300G'. If no ':' is present, the password will be equal to the username.

## **web\_accs.cfg**

To protect a directory from web access, a file called 'web\_accs.cfg' must be placed in the directory to protect. This file shall contain a list of users that are allowed to browse the protected directory and its subdirectories. Multiple of these password files may be present in the system, allowing different users to access different files and directories.

The file format is the same as for the 'ad\_pswd.cfg' and 'sys\_pswd.cfg' files, except that the optional parameter 'AuthName' can be added. The value of this parameter will be presented in the login window. If it is not given, the requested file/pathname will be presented instead.

*File format:*

```
User:Password  
[AuthName]  
(Message goes here)
```

The contents of this file can be redirected by placing the line '[File path]' on the first row, followed by a list of password files.

*Example:*

```
[File path]  
\\user\pswd\my_passwords\web_pswd.cfg
```

If any errors in the format of these files are detected the user/password protection will be ignored

## **ip\_accs.cfg**

It is possible to configure which IP addresses and what protocols that are allowed to connect to the module. This information is stored in the file 'ip\_accs.cfg'. The file contains one or several of the headers below.

```
[Web]  
[FTP]  
[Telnet]  
[Modbus/TCP]  
[Ethernet/IP]  
[All]
```

Under each header the allowed IP addresses are written. The wildcard '\*' can be used to allow series of IP addresses. If a protocol header is not given, the system will use the configuration set below the header 'All'. If the 'All' header is not given, the protocol will not accept any connections.

*Example:*

```
[Web]  
10.10.12.*  
10.10.13.*  
[FTP]  
10.10.12.*  
[Telnet]  
10.10.12.*  
[All]  
*.*.*.*
```



The above example will allow all IP addresses beginning with 10.10.12 to access all protocols in the module. Addresses beginning with 10.10.13 will be able to access the web server, but not the FTP and Telnet servers. The Modbus/TCP and Ethernet/IP servers will accept connections from any IP address.

The contents of this file can be redirected by placing the line '[File path]' on the first row, and a file path on the second. This procedure is exactly the same as with the system file "web\_accs.cfg" (see above).

### **telwel.cfg**

The default Telnet welcome message can be changed by creating this file. It shall contain the new welcome message in ASCII form. The contents of this file can be redirected by placing the line '[File path]' on the first row, and a file path on the second.

*Example:*

```
[File path]
\\my_settings\telnet_welcome_message.txt
```

### **ethcfg.cfg**

This file contains the network configuration and is read by the module at start up. The settings in this file may be affected by configuration software and SSI commands. The format of the file is the following:

```
[IP address]
192.168.0.150
[Subnet mask]
255.255.255.0
[Gateway address]
192.168.0.1
[DHCP/BOOTP]
OFF                                (allowable values are "ON" and "OFF")
[Speed]
Auto                              (allowable values are "Auto", "100", or "10")
[Duplex]
Auto                              (allowable values are "Auto", "Full", or "Half")
[SMTP address]
0.0.0.0
[SMTP username]
username
[SMTP password]
password
[DNS1 address]      (Primary DNS)
0.0.0.0
[DNS2 address]      (Secondary DNS)
0.0.0.0
[Domain name]
Honeywell.com
[Host name]
honeywell
```

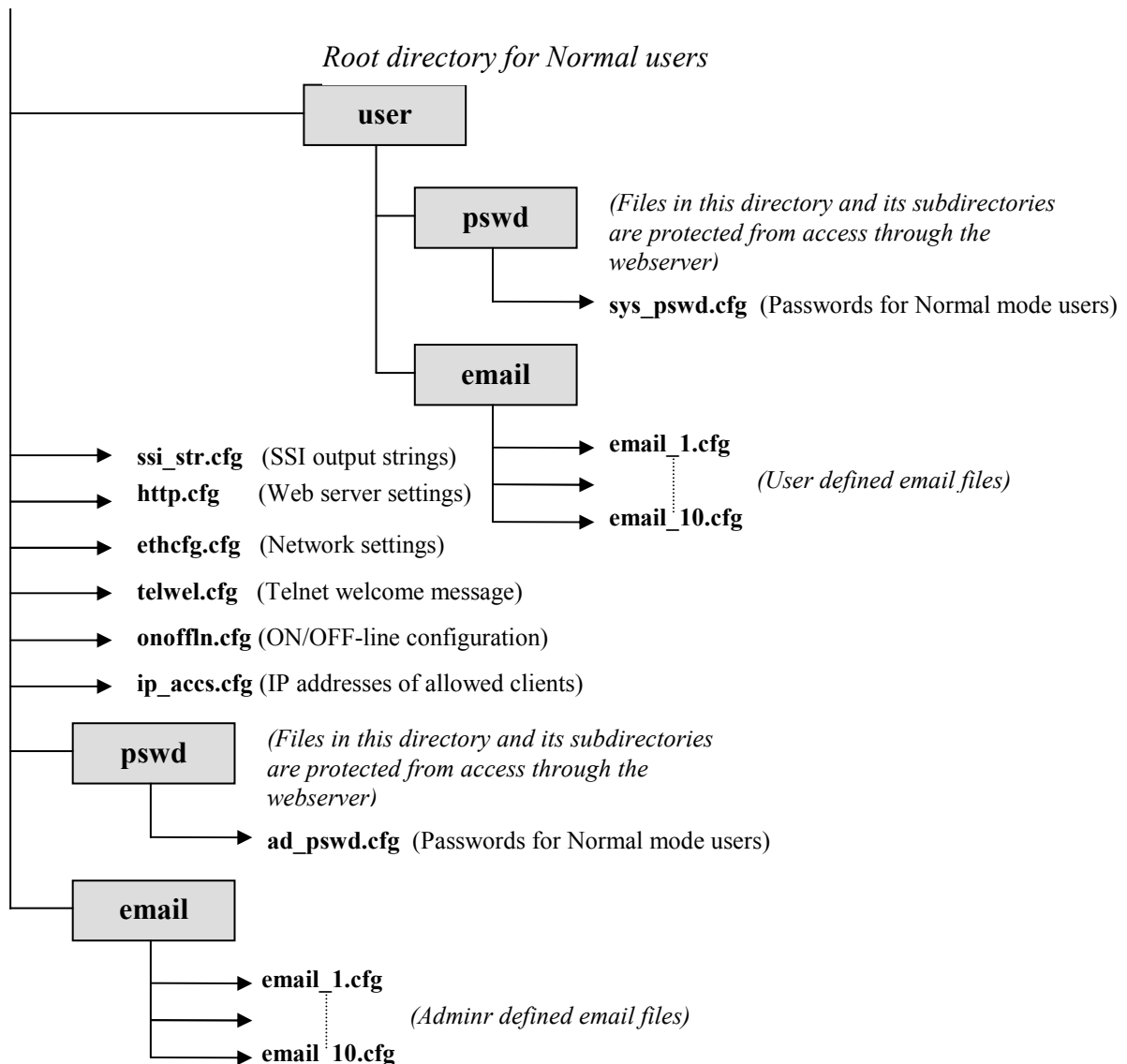
NOTE: In the current firmware implementation "IP Address", "Subnet Mask", "Gateway Address", and "SMTP Address" will *always* be overridden by the values used in configuration software (i.e. those values cannot be set by writing to this file).

The contents of this file can be redirected by placing the line '[File path]' on the first row, and a file path on the second. This procedure is exactly the same as with the system file "ip\_accs.cfg" (see above). For example, redirecting the contents of this file to the "user" directory would allow "Normal Mode" users to have access to this file.

## Structure

The figure below illustrates the structure of the file system, where the system files are located, and which areas Normal/Admin users can access. The files and directory structure must be created by the user using FTP or Telnet. The required .cfg file structures are outlined in the 'System Files' section below.

*Root directory for Admin users*



## Virtual File System

The module also contains a virtual file system containing a set of files used to build the default configuration webpage. The virtual file system can be overwritten or disabled, but not erased; A file with the same name in the file system replaces the file in the virtual file system

until it is removed. The entire virtual file system can be disabled using configuration software on the Ethernet Settings page.

Replacing the virtual files makes it possible to for example replace the default logo by uploading a new logo named '\logo.jpg'. It is also possible to make links from a web page to the virtual configuration page. In that case the link shall point to '\config.htm'.

The virtual file system contains the following files:

\index.htm	- Points to the contents of config.htm
\config.htm	- Configuration frame page
\configform.htm	- Configuration form page
\configform2.htm	- Configuration form page
\store.htm	- Configuration store page
\logo.jpg	- HMS logo
\configuration.gif	- Configuration picture
\boarder.bg.gif	- picture
\boarder_m_bg.gif	- picture

## FTP Server

It is possible to upload/download files to/from the file system using a standard FTP client. Depending on security settings, different parts of the filesystem can be accessed by the user (see Security above). Internet Explorer within the Windows Operating System, for example, may also operate as an FTP Client simply by preceding the address in the address bar with "ftp:" instead of "http:"

The FTP Server can be disabled via configuration software on the Ethernet Settings page.

## Server Side Include (SSI) Functionality

The SSI functionality makes it possible to display or alter I/O data and configuration settings on a web page. It is also possible to use SSI functions in email messages (see "SSI in Email Messages"). Since this functionality allows reading/writing of I/O values in the Fieldbus Interface, some of the functions described below will use an "offset" parameter to specify the I/O Location within the Fieldbus Interface. It should be noted that the "offset" parameter will *always* refer to a byte-addressed offset from the start of the Fieldbus Interface (i.e. the "Address Mode" in configuration software should be set to "Byte" and the "Modbus TCP Address Mode" option should be disabled – see 4.8 Fieldbus Configuration).

## Functions

### DisplayIP

*Syntax:* <?--#exec cmd\_argument='DisplayIP'-->

This function returns the currently used IP address.

### DisplaySubnet

*Syntax:* <?--#exec cmd\_argument='DisplaySubnet'-->

This function returns the currently used Subnet mask

#### DisplayGateway

*Syntax:* <?--#exec cmd\_argument='DisplayGateway'-->

This function returns the currently used Gateway address

#### DisplayDNS1

*Syntax:* <?--#exec cmd\_argument='DisplayDNS1'-->

This function returns the address of the primary DNS server.

#### DisplayDNS2

*Syntax:* <?--#exec cmd\_argument='DisplayDNS2'-->

This function returns the address of the secondary DNS server

#### DisplayHostName

*Syntax:* <?--#exec cmd\_argument='DisplayHostName'-->

This function returns the hostname.

#### DisplayDomainName

*Syntax:* <?--#exec cmd\_argument='DisplayDomainName'-->

This function returns the default domain name.

#### DisplayDchpState

*Syntax:*

<?--#exec cmd\_argument='DisplayDhcpState( "Output when ON", "Output when OFF")'-->

This function returns whether DHCP/BootP is enabled or disabled.

#### DisplayDhcpSupport

*Syntax:* <?--#exec cmd\_argument='DisplayDhcpSupport( "Arg1", "Arg2" )'-->

DHCP support can be disabled using configuration software. This function returns 'Arg1' if it's enabled and 'Arg2' if it's disabled.

#### DisplayEmailServer

*Syntax:* <?--#exec cmd:argument='DisplayEmailServer'-->

This function returns the currently used SMTP server address.

#### DipslaySMTPUser

*Syntax:* <?--#exec cmd:argument='DisplaySMTPUser'-->

This function returns the username used for SMTP authentication.

#### DipslaySMTPPswd

*Syntax:* <?--#exec cmd:argument='DisplaySMTPPswd'-->

This function returns the password used for SMTP authentication.

GetText (Note - This function cannot be used within email messages)

*Syntax:*

```
<?--#exec cmd argument='GetText( "ObjName", OutWriteString ( offset ), n )'-->
```

This SSI function gets the text from an object and stores it in the OUT area.

ObjName - Name of object.

offset - Specifies the offset from the beginning of the OUT area (i.e. Fieldbus Location).

n - Specifies maximum number of characters to read (Optional)

### **printf**

*Syntax:* <?--#exec cmd\_argument='printf("String to write", Arg1, Arg2, ..., ArgN)'-->

This SSI function includes a formatted string, which may contain data from the Fieldbus IN/OUT area, on a web page. The formatting of the string is equal to the standard C function printf().

Like the standard C function printf() the "String to write" for this SSI function contains two types of objects: Ordinary characters, which are copied to the output stream, and conversion specifications, each of which causes conversion and printing of the next successive argument to printf. Each conversion specification begins with the character % and ends with a conversion character. Between the % and the conversion character there may be, in order:

- Flags (in any order), which modify the specification:
  - which specifies left adjustment of the converted argument in its field.
  - + which specifies that the number will always be printed with a sign (space) if the first character is not a sign, a space will be prefixed.
  - 0 for numeric conversions, specifies padding to the field with leading zeroes.
  - # which specifies an alternate output form. For o, the first digit will be zero. For x or X, 0x or 0X will be prefixed to a non-zero result. For e, E, f, g and G, the output will always have a decimal point; for g and G, trailing zeros will not be removed.
- A number specifying a minimum field width. The converted argument will be printed in a field at least this wide, and wider if necessary. If the converted argument has fewer characters than the field width it will be padded on the left (or right, if left adjustment has been requested) to make up the field width. The padding character is normally space, but can be 0 if the zero padding flag is present.
- A period, which separates the field width from the precision.
- A number, the precision, that specifies the maximum number of characters to be printed from a string, or the number of digits to be printed after the decimal point for e, E, or F conversions, or the number of significant digits for g or G conversion, or the minimum number of digits to be printed for an integer (leading 0s will be added to make up the necessary width)

- A length modifier h, l, or L. "h" Indicates that the corresponding argument is to be printed as a short or unsigned short; "l" or "L" indicates a long or unsigned long.

The conversion characters and their meanings are shown below. If the character after the % is not a conversion character, the behavior is undefined.

| Character | Argument type | Converted to   |
|-----------|---------------|--|
| d, i      | byte, word    | decimal notation (For signed representation. Use signed argument)  |
| o         | byte, word    | octal notation (without a leading zero).   |
| x, X      | byte, word    | hexadecimal notation (without a leading 0x or 0X), using abcdef for 0x or ABCDEF for 0X.   |
| u         | byte, word    | decimal notation.  |
| c         | byte, word    | single character, after conversion to unsigned char.   |
| s         | char*         | characters from the string are printed until a '\0' (i.e. NULL) is reached or until the number of characters indicated by the precision have been printed  |
| f         | float         | decimal notation of the form [-]mmm.ddd, where the number of d's is specified by the precision. The default precision is 6; a precision of 0 suppresses the decimal point.                         |
| e, E      | float         | decimal notation of the form [-]m.dddddd e+-xx or [-]m.ddddddE+-xx, where the number of d's specified by the precision. The default precision is 6; a precision of 0 suppresses the decimal point. |
| g, G      |               | %e or %E is used if the exponent is less than -4 or greater than or equal to the precision; otherwise %f is used. Trailing zeros and trailing decimal point are not printed.                       |
| %         | -             | print a %  |

The arguments that can be passed to the SSI function *printf* are:

| Argument                       | Description  |
|--------------------------------|--|
| InReadSByte( <i>offset</i> )   | Reads a signed byte from position <i>offset</i> in the IN area                     |
| InReadUByte( <i>offset</i> )   | Reads an unsigned byte from position <i>offset</i> in the IN area                  |
| InReadSWord( <i>offset</i> )   | Reads a signed word (short) from position <i>offset</i> in the IN area             |
| InReadUWord( <i>offset</i> )   | Reads an unsigned word (short) from position <i>offset</i> in the IN area          |
| InReadSLong( <i>offset</i> )   | Reads a signed longword (long) from position <i>offset</i> in the IN area          |
| InReadULong( <i>offset</i> )   | Reads an unsigned longword (long) from position <i>offset</i> in the IN area       |
| InReadString( <i>offset</i> )  | Reads a string (char*) from position <i>offset</i> in the IN area                  |
| InReadFloat( <i>offset</i> )   | Reads a floating point (float) value from position <i>offset</i> in the IN area    |
| OutReadSByte( <i>offset</i> )  | Reads a signed byte from position <i>offset</i> in the OUT area                    |
| OutReadUByte( <i>offset</i> )  | Reads an unsigned byte from position <i>offset</i> in the OUT area                 |
| OutReadSWord( <i>offset</i> )  | Reads a signed word (short) from position <i>offset</i> in the OUT area            |
| OutReadUWord( <i>offset</i> )  | Reads an unsigned word (short) from position <i>offset</i> in the OUT area         |
| OutReadSLong( <i>offset</i> )  | Reads a signed longword (long) from position <i>offset</i> in the OUT area         |
| OutReadULong( <i>offset</i> )  | Reads an unsigned longword (long) from position <i>offset</i> in the OUT area      |
| OutReadString( <i>offset</i> ) | Reads a NULL terminated string (char*) from position <i>offset</i> in the OUT area |
| OutReadFloat( <i>offset</i> )  | Reads a floating point (float) value from position <i>offset</i> in the OUT area   |

## scanf

*Syntax:*

<?--#exec cmd\_argument='scanf( "ObjName", "format", Arg1, ..., ArgN), ErrVal1, ..., ErrvalN'-->

This SSI function reads a string passed from an object in a HTML form, interprets the string according to the specification in format, and stores the result in the OUT area according to the passed arguments. The formatting of the string is equal to the standard C function call scanf()

ObjName                      - The name of the object with the passed data string

format                      - Specifies how the passed string shall be formatted

Arg1 - ArgN                      - Specifies where to write the data

ErrVal1 -ErrValN                      - Optional; specifies the value/string to write in case of an error.

| Character | Input data, Argument Type   |
|-----------|---|
| d         | Decimal number; byte, short   |
| i         | Number, byte, short. The number may be in octal (leading 0(zero)) or hexadecimal (leading 0x or 0X)   |
| o         | Octal number (with or without leading zero); byte, short  |
| u         | Unsigned decimal number; unsigned byte, unsigned short  |
| x         | Hexadecimal number (with or without leading 0x or 0X); byte, short  |
| c         | Characters; char*. The next input characters (default 1) are placed at the indicated spot. The normal skip over white space is suppressed; to read the next non-white space character, use %1s. |
| s         | Character string (not quoted); char*, pointing to an array of characters large enough for the string and a terminating "\0" that will be added.   |
| e, f, g   | Floating-point number with optional sign, optional decimal point and optional exponent; float*  |
| %         | Literal %; no assignment is made.   |

The conversion characters d, i, o, u and x may be preceded by l (small case L) to indicate that a pointer to 'long' appears in the argument list rather than a 'byte' or a 'short'

The arguments that can be passed to the SSI function scanf are:

| Argument                        | Description   |
|---------------------------------|---|
| OutWriteByte( <i>offset</i> )   | Writes a byte to position <i>offset</i> in the OUT area                         |
| OutWriteWord( <i>offset</i> )   | Writes a word (short) to position <i>offset</i> in the OUT area                 |
| OutWriteLong( <i>offset</i> )   | Writes a long to position <i>offset</i> in the OUT area                         |
| OutWriteString( <i>offset</i> ) | Writes a string to position <i>offset</i> in the OUT area                       |
| OutWriteFlost( <i>offset</i> )  | Writes a floating point (float) value to position <i>offset</i> in the OUT area |

## IncludeFile

*Syntax:* <?--#exec cmd\_argument='IncludeFile( "File name" )'-->

This SSI function includes the contents of a file on a web page.

*Default output:*

- Success           - <File content>
- Failure           - Failed to open <filename>

## SaveToFile

*Syntax:*

<?--#exec cmd\_argument='SaveToFile( "File name", "Separator", [Append|Overwrite] )'-->



This SSI function saves the contents of a passed form to a file. The passed name/value pair will be written to the file "File name" separated by the "Separator" string. The contents can either be Appended to the file or overwrite the current content of the file.

*Default output:*

|         |                       |
|---------|-----------------------|
| Success | - Form saved to file  |
| Failure | - Failed to save form |

## **Web Server**

The module features a complete web server with SSI functionality. It is possible to upload web pages to the module, giving access to parameters in the Fieldbus Interface using a customizable interface.

By default the HTTP server is enabled, but it can be enabled/disabled by configuration software on the Ethernet settings page.

## **Email Client**

It is possible to send emails from the module. To send an email, the SMTP server address must be configured. Without a valid SMTP address the module will not be able to send any email messages.

### **Sending a predefined email on data event**

It is possible to send predefined email messages, triggered by an event in the Fieldbus Interface. The Fieldbus Interface is scanned once every 0.5 second. This means that an event must be present longer than 0.5 seconds to ensure that it is detected by the module. It is possible to have up to 10 user defined, and 10 admin defined emails, triggered on different events. These shall be placed in the directories “\user\email” for user configurable emails and “\email” for non-user configurable emails. The files must be named ‘email\_1.cfg’, ‘email\_2.cfg’ ... ‘email\_10.cfg’.

The files shall have the following format:

```
[Register]
Area, Offset, Type
[Register match]
Match Value, Mask, Match operand
[To]
Recipient(s)
[From]
Sender
[Subject]
Subject line
[Headers]
Extra Headers
[Message]
Message body
```

| Parameter     | Description   |
|---------------|---|
| Area          | Source Fieldbus Interface Area. Possible values are 'IN' or 'OUT'   |
| Offset        | Source offset in Fieldbus Area, shall be written in decimal or hexadecimal.   |
| Type          | Source data type. Possible values are 'byte', 'word', and 'long'  |
| Match Value   | Value to compare with the source data. Shall be written in decimal or hexadecimal.  |
| Mask          | The module performs a logical 'and' on the source data and this Mask before the value is compared with the Match Value. The value shall be written in decimal or hexadecimal. |
| Match Operand | Specifies how the data shall be compared with the Match Value. Possible values: '<', '=', '>'   |
| Recipient(s)  | Destination email addresses, semicolon separated  |
| Sender        | Sender email address  |
| Subject line  | Email subject (One line only)   |
| Extra Headers | Optional. May be useful for advanced users when for example sending HTML emails etc.  |
| Message Body  | The actual email message.   |

The data is read in the Fieldbus Interface from the area and offset specified by the parameters Area, and Offset. The data size to read is specified by the Type parameter. The module performs a logical 'AND' between the read data and the parameter Mask. The result is compared with the parameter Match Value. How the data shall be compared is specified by the Match Operand.

*Example:*

```
[Register]
IN, 0x0003, byte
[Register match]
0x20, 0x7F, >
[To]
support@honeywell.com
[From]
user@companyname.com
[Subject]
Status
[Message]
All data correct.
```

In the above example:

- A byte is read from the Fieldbus IN area, at byte address 0003h
- The module performs a logical <data> AND 7Fh.

If the result is larger than 20h, the email message is sent to support@honeywell.com

**Note:** If the [Register] or [Register match] information is changed, a reset is required for changes to take effect. Other changes will take effect directly without a reset.

**Note:** Hexadecimal values must be written in the format 0xN where 'N' is the hexadecimal value.

## **SSI in Email Messages**

For predefined emails it is possible to include data in the mails. This is done in a similar way as data is added to web pages with SSI includes. Due to natural reasons, some SSI functions cannot be used in email messages.

The supported SSI commands for emails are:

- DisplayIP
- DisplaySubnet
- DisplayGateway
- DisplayDNS1
- DisplayDNS2
- DisplayHostName
- DisplayDomainName
- DisplayEmailServer
- DisplaySMTPUser
- DisplaySMTPPwd
- DisplayDhcpState
- DisplayDhcpSupport
- printf
- IncludeFile
- SsiOutput

## **Telnet Server**

Through a Telnet client, the user can access the filesystem using a command line interface similar to MS-DOS™. Depending on security settings, different parts of the filesystem can be accessed by the user (see Security above).

The telnet server can be disabled via configuration software on the Ethernet Settings page.

## **General Commands**

**help**

*Syntax:* help [general|diagnostic|filesystem]

**version**

This command will display version information, serial number and MAC ID

**exit**

This command closes the Telnet session.

**Diagnostic Commands**

The following commands can be viewed by the command 'help diagnostic'

**arps**

Display ARP stats and table

**iface**

Display net interface stats

**sockets**

Display socket list

**routes**

Display IP route table

**File System Operations**

For commands where filenames, directory names or paths shall be given as an argument the names can be written directly or within quotes. For names including spaces the filenames must be surrounded by quotes. It is also possible to use relative pathnames using '.', '\ ' and '..'

**dir**

*Syntax:* dir [path]

Lists the contents of a directory. If no path is given, the contents of the current directory are listed.

**md**

*Syntax:* md [[path][directory name]]

Creates a directory. If no path is given, the directory is created in the current directory.

**rd**

*Syntax:* rd [[path][directory name]]

Removes a directory. The directory can only be removed if it is empty.

**cd**

*Syntax:* cd [path]

Changes current directory.

**format**

Formats the filesystem. This is a privileged command and can only be called in administration mode.

**del**

*Syntax:* del [[path]][filename]]

Deletes a file.

**ren**

*Syntax:* ren [[path]][old name]] [[path]][new name]]

Renames a file or directory.

**move**

*Syntax:* move [[source path]][source file]] [[destination path]]

This command moves a file or directory from the source location to a specified destination.

**copy**

*Syntax:* copy [[source path]][source file]] [[destination path]][destination file]]

This command creates a copy of the source file at a specified location.

**type**

*Syntax:* type [[path]][filename]]

Types (displays) the contents of a file.

**mkfile**

*Syntax:* mkfile [[path]][filename]]

Creates an empty file.

**append**

*Syntax:* append [[path]][filename]] ["The line to append"]

Appends a line to a file.

**df**

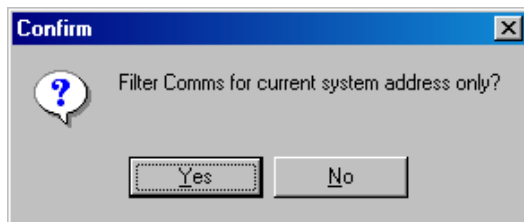
Displays filesystem info.

## Appendix 3

## Comms Logging

1. Start E-Series Configuration Utility and open up the appropriate project.
2. Select from the site list the module you wish to monitor comms at.
3. Press the “Diagnostic” button on the right hand side of the configuration screen and a new window will open headed “Communications Log”.
4. If the model is an “XYR300G-E” press “Start Comms” and then select whether you wish to filter current system address.

“Yes” for current system address only or “No” for all System addresses.

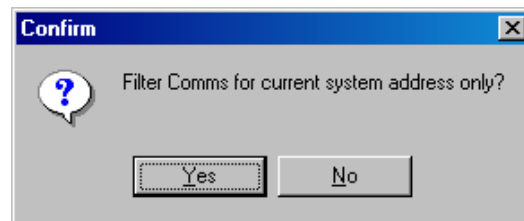


Go to step 7

5. If model is an “XYR300G-M” press the “Terminal” button on the right hand side of this window.
6. To put the module into configuration mode you will need to hold down the indented black button (on the end of the module) until the ACT LED flashes (about 5 seconds), release button and you should see “Type ‘m’ for menu.....” press “m” and a Configuration Menu will appear.

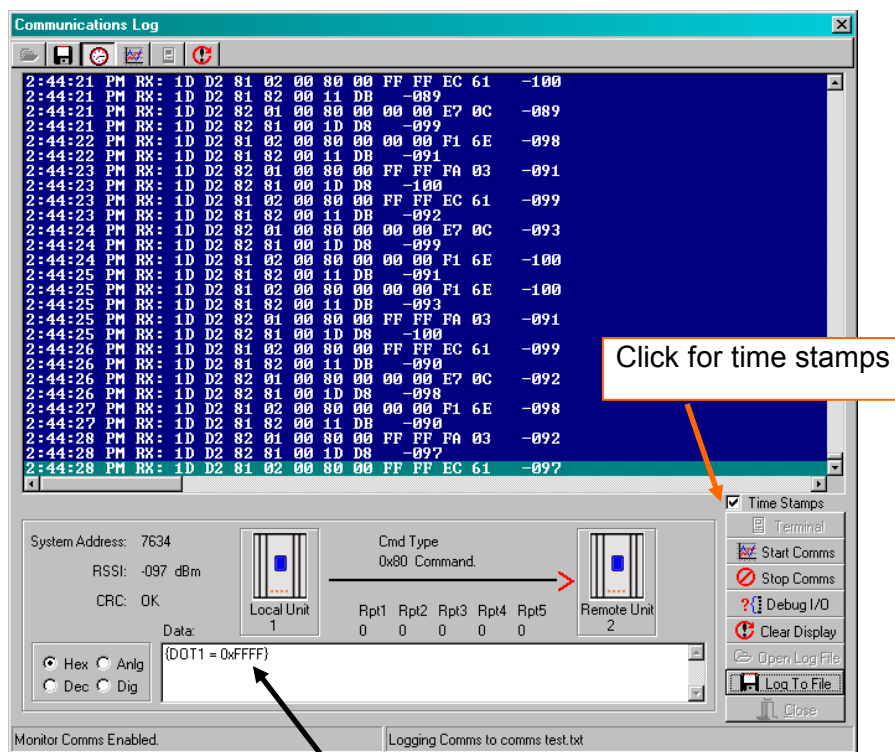
*Note: If “Enable Comms Logging” is selected from the Debug menu, Monitor Comms mode will stop functioning and serial port will revert back to Modbus communications.*

Click on “Stop Terminal” and then “Start Comms” and select whether you wish to filter current system address. “Yes” for current system address only or “No” for all System addresses.



7. Tick the “Time stamps” option and then select “Log to File” and type in the name of the log file.
8. From now on everything that is received will be logged to this file with a time stamp.
9. The data frames can be decoded by selecting a message and then viewing the panel below the Terminal windows. (See Below). You can see the System address, Message type, from and to

addresses as well as the data value, which can be viewed in a number of different formats. I.e Hex, Dec, Analog and Digital.



Decoded Data