1900/65A General Purpose Equipment Monitor

Operation Manual

Bently Nevada Machinery Condition Monitoring

172250 Rev. F

Bently Nevada
a Baker Hughes business

1631 Bently Parkway South, Minden, Nevada USA 89423
Phone: 1.775.782.3611 (US) or Bently.com/support
Bently.com

Contents

1. General Safety

1.1 Receiving Inspection

Visually inspect the monitor for obvious shipping damage. If you detect shipping damage, file a claim with the carrier and submit a copy to Bently Nevada. Include all model numbers and serial numbers with the claim.

1.2 Handling and Storing Considerations

To prolong the service life of the system, handle components carefully, use best practices during installation, and practice diligent inspection procedures. Follow prescribed maintenance procedures and dispose of obsolete components in compliance with applicable electronic waste regulations.

Circuit Boards

Circuit boards contain devices that are susceptible to damage when exposed to electrostatic charges. Damage caused by obvious mishandling of the board voids any warranty. To avoid damage, observe the following precautions in the order given.

1. Do not discharge static electricity onto the circuit board. Avoid tools or procedures that would subject the circuit board to static damage. Some possible causes include ungrounded soldering irons, and non-conductive plastics and similar materials.
2. Personnel must be grounded with a suitable grounding strap before handling or maintaining a printed circuit board.
3. Transport and store circuit boards in electrically conductive bags or foil.
4. Use extra caution during dry weather. Relative humidity less than 30% tends to multiply the accumulation of static charges on any surface.

When performed properly, you may install this module into or remove this module from the rack while power is applied to the rack. Refer to the Rack Installation and Maintenance Manual (document 129766) for the proper procedure.

Potential Protection Loss

CAUTION

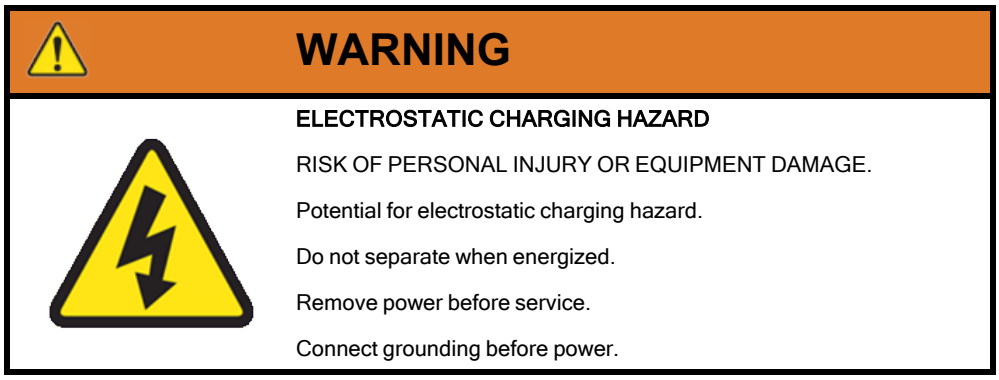
PROTECTION LOSS

MACHINERY PROTECTION MAY BE LOST when a processor or relay component is removed from the rack or base, or while a self-test is being performed.

1.3 Personal Safety Warnings

Labels and markings are provided on to guide the system integrator in the processes of choosing appropriate interface equipment, determining safe use conditions, and identifying recommended installation procedures. The format of these markings are governed by the standards that dictate safe use and environmental compliance in a variety of regions and regulated settings.

Potential Electrostatic Charging Hazard



Installations and maintenance tasks performed in potentially hazardous areas must be performed only after the area has been verified to be free of hazardous materials, atmospheres, and conditions.

- Do not discharge static electricity onto the circuit board. Avoid tools or procedures that would subject the circuit board to static damage. Some possible causes of static damage include ungrounded soldering irons, non-conductive plastics, and similar materials.
- Use a suitable grounding strap before handling or performing maintenance on a printed circuit board.
- Transport and store circuit boards in electrically conductive bags or foil.
- Use extra caution during dry weather. Relative humidity less than 30% tends to multiply the accumulation of static charges on any surface.

The following situations could cause a spark sufficient to cause an explosion:

- Potential of electrostatic discharge on plastic components, or
- Removal or placement of an energized connection.

1.4 Safe Disposal

Replacing Device and Failure Analysis

To return parts under warranty and request failure analysis, visit **Bently Nevada**.

Hazardous Materials

This device does not use hazardous materials outlined by RoHS or battery directive statutes. These regulations confirm that lead, mercury, cadmium, hexavalent chromium, polybrominated biphenyls, polybrominated diphenyl ether, and battery-related materials such as lithium are limited to no more than trace amounts within the system.

Recycling Facilities

Decommissioning of instrumentation should endeavor to minimize the impact of the waste created by disposal of system material. Refer to local or regional waste removal administration to collect information on proper material collection, reuse, and recycling.

Product Disposal Statement

Customers or third parties who are not member states of the European Union and who are in control of the product at the end of its life or at the end of its use, are solely responsible for diligent product disposal at the end of its useful life. No person, firm, corporation, association, or agency shall dispose of the product in a way that is in violation of any applicable international, federal, state, or local regulations. Visit www.weeerohsinfo.com for recycling information.

2. Introduction

This document covers the installation and use of the Modbus Gateway with the 1900/65A General Purpose Equipment Monitor. The Modbus Gateway is the interface this monitor uses to communicate with Modbus master devices over an Ethernet (Modbus/TCP) or a serial (Modbus/RTU) network using the Modbus protocol. Modbus is particularly useful in a client server architecture or for sharing monitor data simultaneously with multiple client devices as part of a Process Control System (PCS).

2.1 1900/65A Overview

The 1900/65A General Purpose Equipment Monitor is a machinery vibration monitor designed to continuously monitor and protect equipment used in a variety of applications and industries.

For additional information, refer to the Product Manual.

2.2 RS-232 and RS-485 Support

The 1900/65A Modbus Gateway supports serial Modbus, using the RTU mode, over RS-232 or RS-485. The serial connection cannot be used for configuration.

2.3 RS-422 Support

The 1900/65 and 1900/65A Modbus Gateways support serial Modbus over RS-422 with the following optional accessory:

Table 2 - 1: Firmware and software supporting the Modbus Gateway

Description	Part Number	Comment
Lantronix Xpress DR-IAP (XSDRIN-01)	172555	Modbus/TCP (Ethernet) to Mobus RTU (Serial) Converter

2.4 Response Time

Under most conditions the monitor will respond to a Modbus request within 500 ms. See Section 4.6 for a list of exceptions. This response time is measured from the time the monitor receives a request from a client, to the time it begins to transmit a response. The total response time seen by the client will be subject to network latencies so determining a worst case response time will require a detailed analysis of these latencies. In lieu of this analysis, consider assuming a total response time of 1 second for an Ethernet (Modbus/TCP) network. For a serial (Modbus/RTU) network, add in the baud rate multiplied by the maximum message length when determining the total response time.

3. Installing the Modbus Gateway

Modbus Gateway installation consists of upgrading your copy of the 1900 Configuration Software and your monitor firmware. If your monitor was originally ordered with the Modbus communications option or if you have a 1900/65A, you may skip these instructions.

Built-in support for RS-232 and RS-485 serial communications is available only with the 1900/65A monitor. The 1900/65 monitor must use an external converter. See Table 3 - 1: Firmware and software supporting the Modbus Gateway.

3.1 Check Firmware and Software Versions

The Modbus Gateway requires a 1900/65 monitor running the firmware and software versions listed in Table 3 - 1: Firmware and software supporting the Modbus Gateway. Major and minor revision numbers greater than the values shown in this table will also support the Modbus Gateway. 1900/65A monitors already meet the firmware requirement.

Table 3 - 1: Firmware and software supporting the Modbus Gateway

	Part Number	Part Revision	Major Revision	Minor Revision
Firmware	169065	E	2	17
Software	169253	B	3	0

Updates are available for download from the Bently Nevada Software Update Services page located at www.bently.com. Contact your Bently Nevada representative for licensing details.

Determine the Current Firmware Version

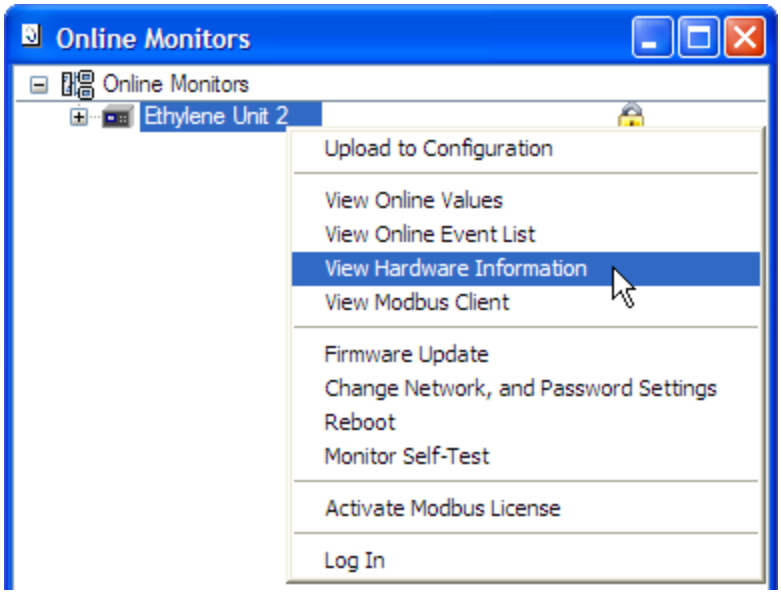


Figure 3 - 1: Opening the Hardware Information window.

1. Locate the monitor in the Online Monitors window.
2. Right click on the monitor to open the Context menu (Figure 3 - 1: Opening the Hardware Information window.).

3. Click on View Hardware Information to open the Hardware, Firmware Information window (Figure 3 - 2: Hardware, Firmware Information window.).

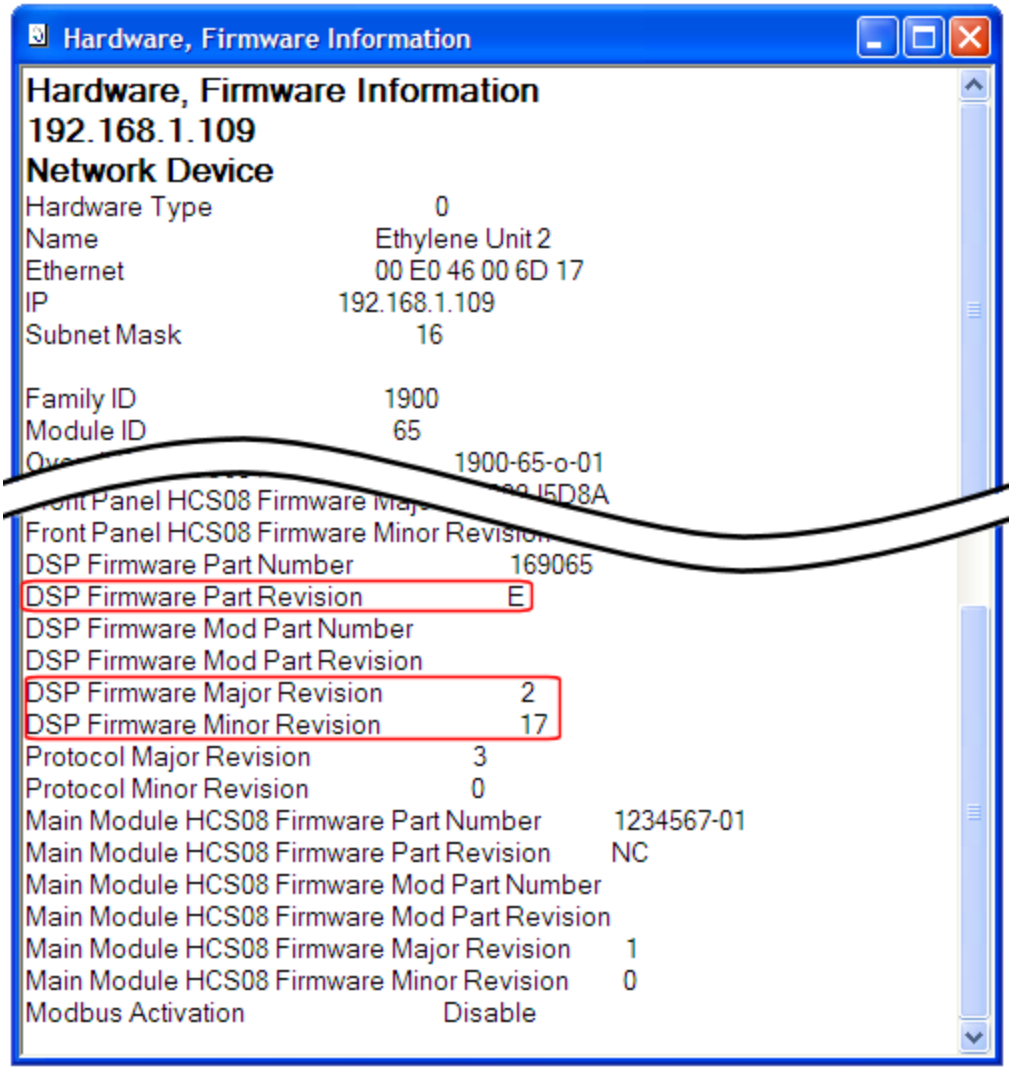


Figure 3 - 2: Hardware, Firmware Information window.

4. Note the revision information provided in this window. If the firmware version for your monitor is earlier than that listed in Table 3 - 1: Firmware and software supporting the Modbus Gateway, install new firmware.

Determine the Current Software Version

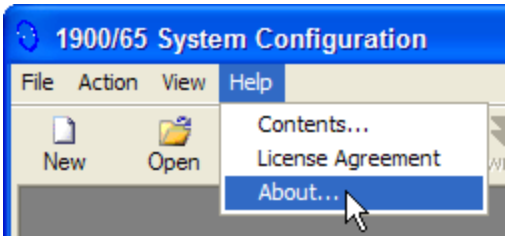


Figure 3 - 3: Opening the About window.

1. Click Help>About on the menu bar (Figure 3 - 3: Opening the About window.) to open the About window.
2. Note the revision information shown in this window. If the version of your software is earlier than that listed in Table 3 - 1: Firmware and software supporting the Modbus Gateway, install new software.

3.2 Installing New Software

System Requirements

Software may be installed on a notebook or desktop computer having these minimum requirements:

- Microsoft® Windows® 2000 or Windows XP Operating System
- 800 MHz Pentium® III class processor (or better)
- 256 MB RAM (minimum)
- 8 MB Video Card (or better)
- 8X CD-ROM drive (or faster)
- 100 MB available hard drive space
- 10 BaseT or 100BaseTX Ethernet interface (10 or 100 Mb/s)
- 1024 x 768 screen resolution (or higher)

Software Installation

You must install the 1900 Configuration Software from the 1900 Configuration Software CD onto your hard drive. You cannot run the program from the CD. To install the software:

1. Insert the 1900 Configuration Software CD in the CD-ROM drive of the computer.
2. The 1900 Configuration Software Setup will launch automatically. If it doesn't launch, browse to the CD-ROM drive and execute setup.exe.
3. Follow the setup Instructions (similar to Figure 3 - 4: Installation sequence..)

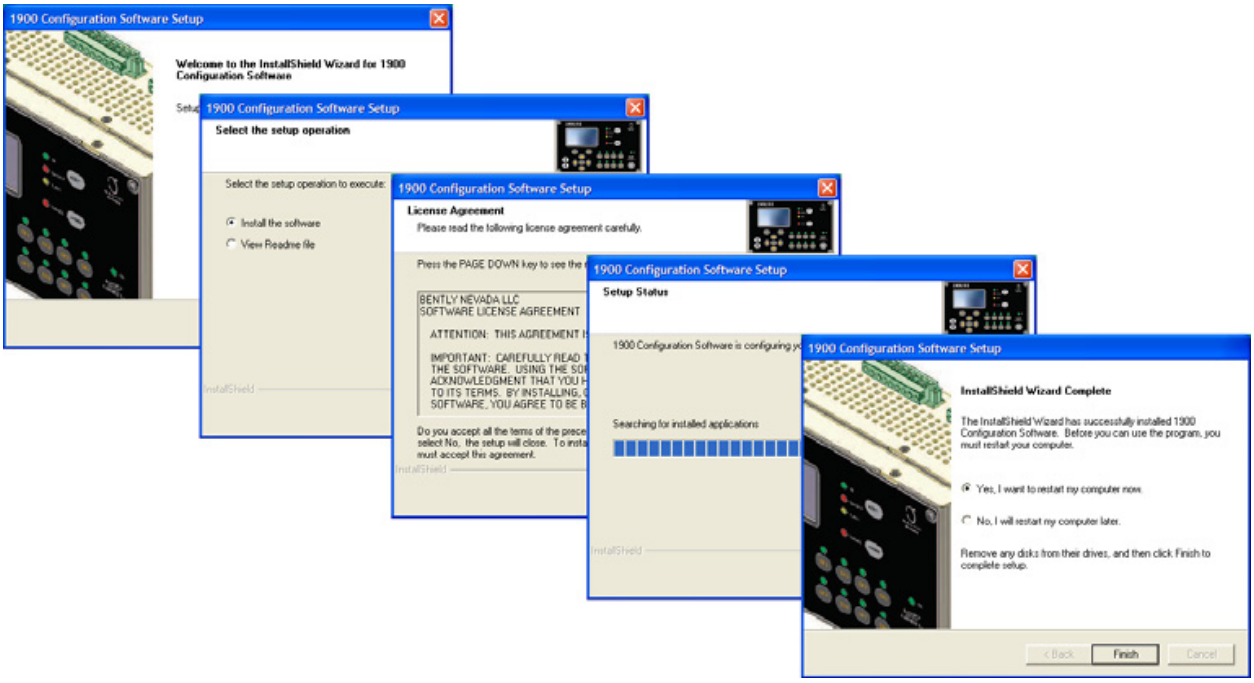


Figure 3 - 4: Installation sequence.

3.3 Installing New Firmware

If your installed version of 1900 Configuration Software supports Modbus, it will automatically handle upgrading the firmware. When you attempt to download a Modbus configuration to a monitor with older firmware, the software will prompt you to upgrade firmware. If you select yes, the software will upgrade the monitor's firmware using a version of firmware that supports Modbus. So performing the following steps which describe how to manually upgrade firmware should not be necessary.

1. Open the 1900 Configuration Software.
2. Click the Online icon in the Toolbar to display the 1900/65 monitors on your network.
3. Right-click on the monitor you want to update to open the context menu (Figure 3 - 5: Opening Firmware Update window.).

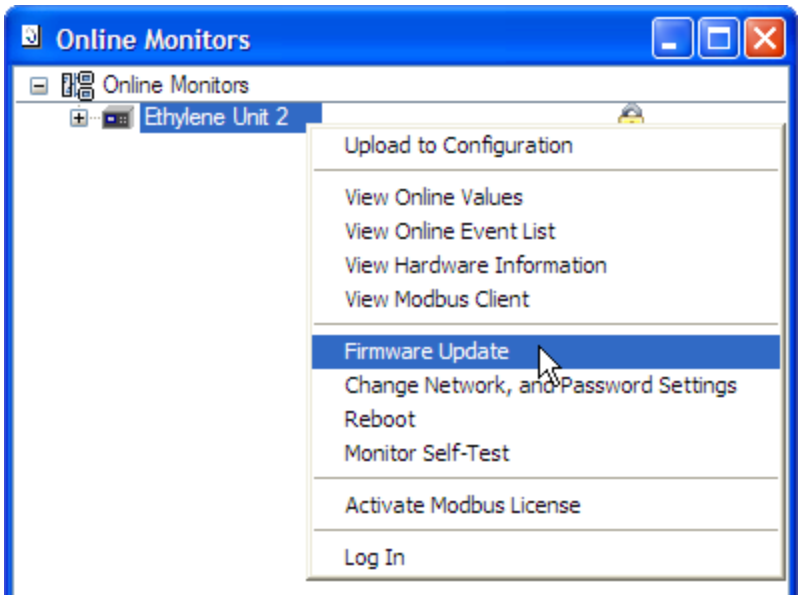


Figure 3 - 5: Opening Firmware Update window.

4. Click on Firmware Update to open the firmware update window (Figure 3 - 6: Firmware Update window.).

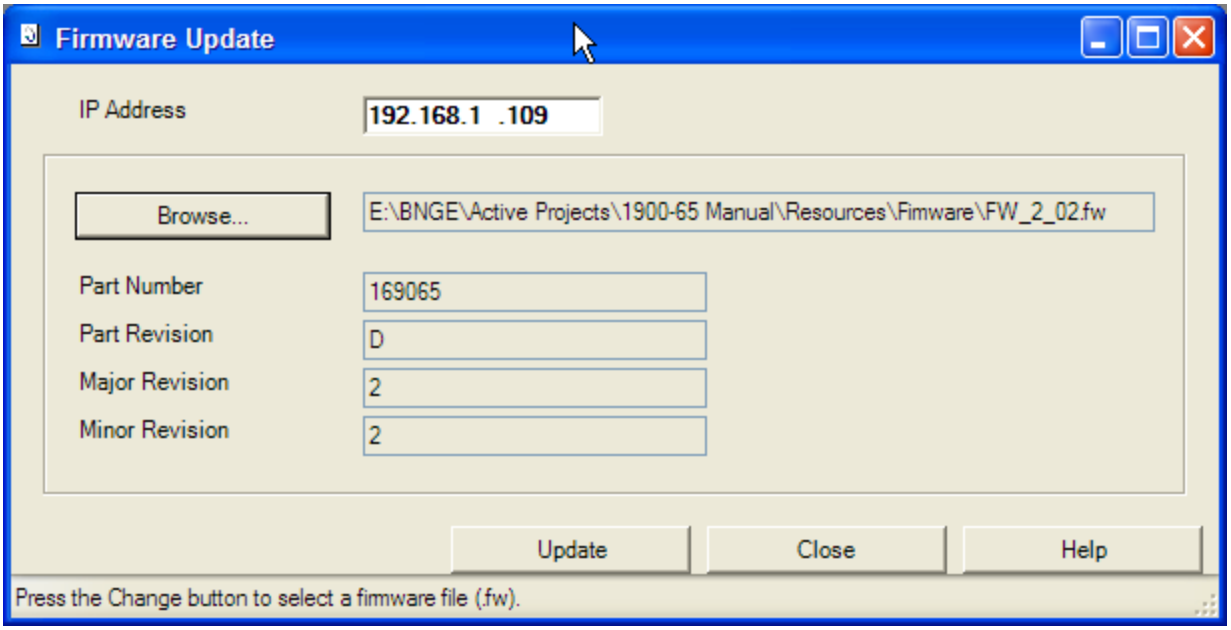


Figure 3 - 6: Firmware Update window.

5. Click the Browse button to locate the new firmware file.
6. Click the Update button to complete the upgrade.

4. Using the Modbus Gateway

This chapter describes how to use the Modbus gateway in the 1900/65A monitor. Modbus protocol and register definitions are covered in detail in Chapter 4 and Chapter 5.

4.1 Configuring Modbus Communications

The 1900/65A supports three types of Modbus communication:

Modbus Over Ethernet: The 1900/65A monitor acts as a server and is able to support simultaneous communication with up to 6 Modbus clients when connected over an Ethernet network. The hardware installation chapter of the 1900/65A Product Manual describes the physical connection to a plant network.

Modbus Over RS-232: The RS-232 port permits the direct connection of one 1900/65A slave to one Modbus master device. The hardware installation chapter of the 1900/65A Product Manual describes the physical connection details.

Modbus Over RS-485: The RS-485 port permits connecting multiple 1900/65A slaves to one Modbus master device. The hardware installation chapter of the 1900/65A Product Manual describes the physical connection details. The total number of devices on a single RS-485 link must be 32 or fewer.

To configure the communications, start with a valid monitor configuration and:

1. Generate a Modbus map. This maps monitor data to Modbus registers (Section Generate a Modbus Map).
2. Review the general Modbus Gateway properties. Change default settings as needed (Section Review Properties).
3. Review the automatically generated map. Add, remove or change register address assignments as needed (Section Review/Edit Modbus Map).
4. Save the configuration (Section Save Configuration).
5. Download the configuration to the 1900/65A monitor (Section Download Configuration).

The rest of this section describes this configuration process in detail.

Generate a Modbus Map

1. In the Configuration window (Figure 4 - 1: Expanded monitor node showing the Modbus Gateway Node, the Serial Modbus Node, and the Generate Map command.), expand the monitor you wish to map by clicking the plus (+) icon at the monitor node.
2. Right-click on the Modbus Gateway or the Serial Modbus Gateway node. The context menu will open.
3. Click on Generate Map at the top of the menu. A dialog will open to warn you that any existing configuration will be overwritten.

4. Click the OK button to generate a new Modbus Map.

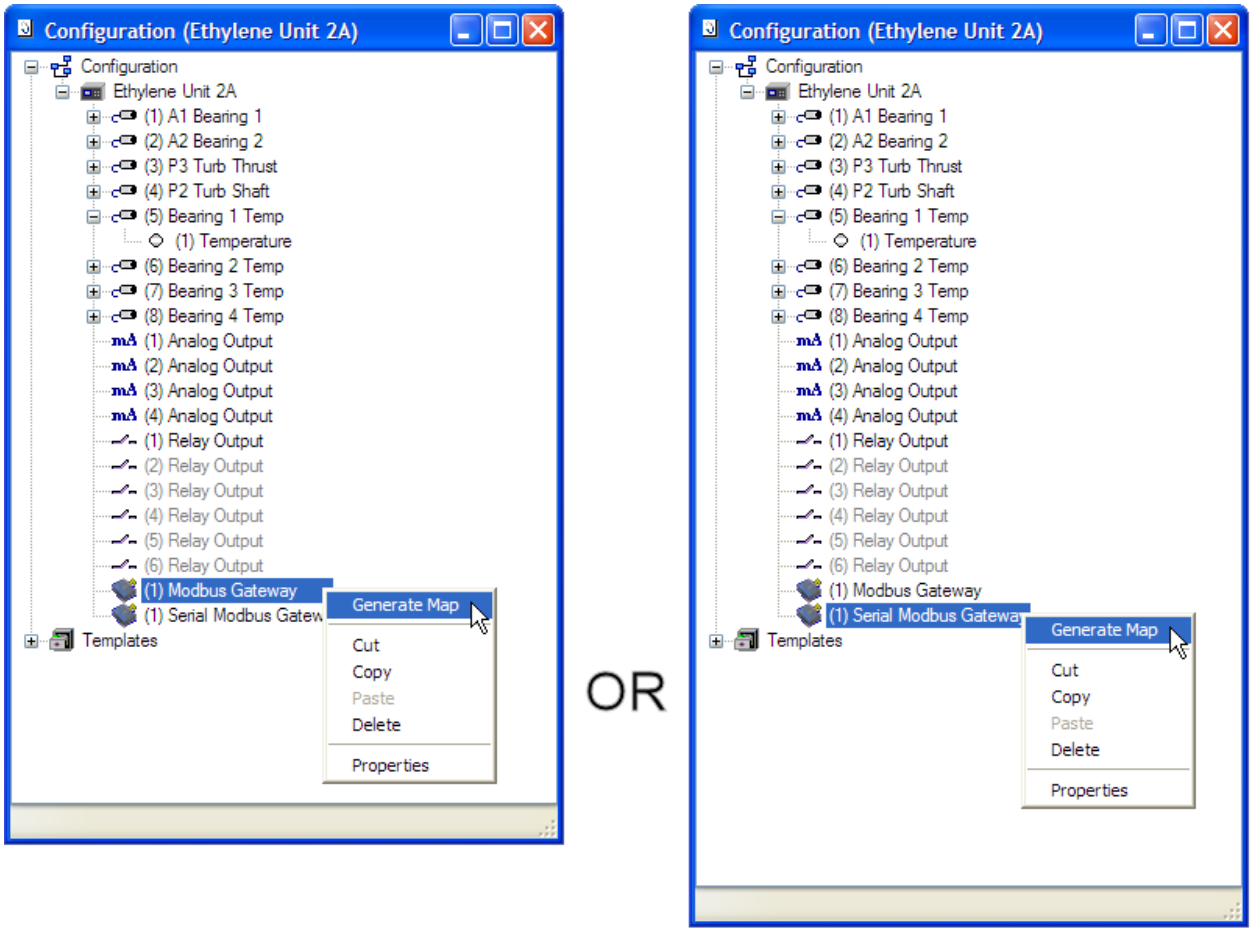


Figure 4 - 1: Expanded monitor node showing the Modbus Gateway Node, the Serial Modbus Node, and the Generate Map command.

Review Properties

General Tab

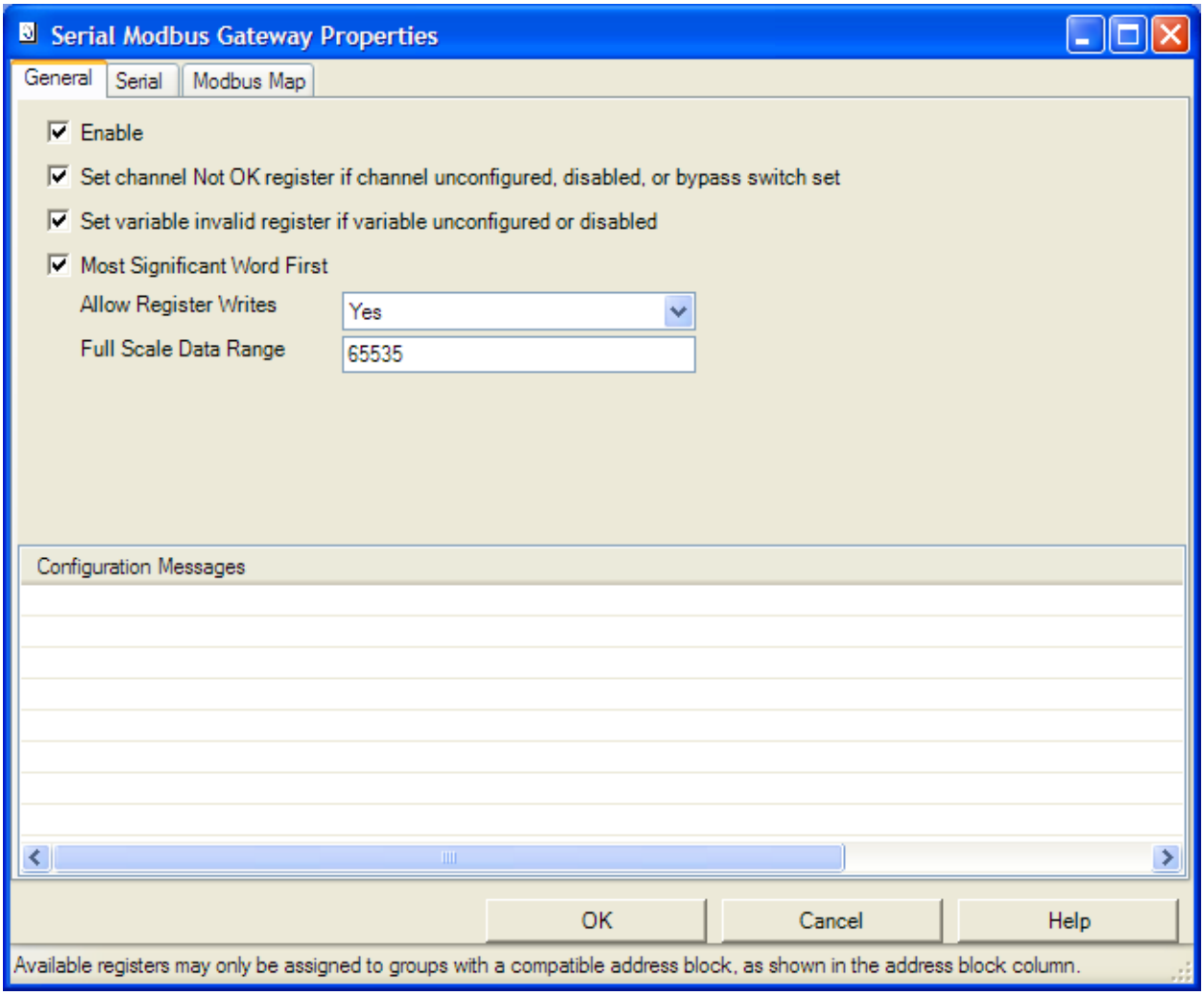


Figure 4 - 2: The Modbus Gateway properties General tab.

Enable: Click in the check box to enable (checked) or disable (unchecked) Modbus communications. The current property settings and Modbus map are preserved if a property is disabled.

For most applications you may use the default settings for the following properties.

Set channel Not OK register if channel unconfigured, disabled, or bypass switch set: Click the check box to force the monitor to set the Not OK register to 1 when the channel is unconfigured, disabled, or the channel bypass switch is set. Unclick the check box to prevent the Not OK register from responding to these conditions. The channel bypass switch is a software setting that will bypass a channel when active and can be set by a Modbus command, the 1900 Configuration Software, or the Display Module.

Set variable invalid register if variable unconfigured or disabled: Click the check box to force the monitor to set the variable invalid register to 1 when a variable is unconfigured or disabled. Unclick the check box so unconfigured or disabled variables do not affect the variable invalid status.

Most Significant Word First: This swaps the upper 16 bits with the lower 16 bits of a 32 bit value to accommodate different number formats. This applies only to Modbus data that uses two registers. A check sends the most significant word first. If the box is not checked, the least significant word is sent first. See Table 4 - 1: Most Significant Word First example showing a 32-bit value = 0x7B8C3F6E.

Table 4 - 1: Most Significant Word First example showing a 32-bit value = 0x7B8C3F6E

Map Address	Most Significant Word First Option	
	Enabled (checked)	Disabled (unchecked)
30020	0x7B8C	0x3F6E
30021	0x3F6E	0x7B8C

Allow Register Writes: Select the kind of configuration permission given to a Modbus client:

- No – Prohibits configuration by a Modbus client
- Yes – Permits a Modbus client to change the parameters listed below:
  - Write Time and Date
  - Reset Latched Statuses
  - Monitor Alarm Inhibit Switch
  - Monitor Bit Packed Status List
  - Channel Bypass Switch
  - Channel Danger Bypass Switch
  - Channel Bit Packed Status List
  - Setpoint Over Amplitude, Integer
  - Setpoint Under Amplitude, Integer
  - Setpoint Over Amplitude, Floating Point
  - Setpoint Under Amplitude, Floating Point

- With Sequence – Requires a write sequence to be used when the configuration parameters listed above are changed via Modbus. Immediately prior to writing to the configuration parameter to be changed, a value of 0x55AA (Hex) must be written to the Write Sequence Field register. See register definition in Section Write Sequence Number.

Full scale data range: Enter a value from 1 to 65535. Variable amplitudes are scaled to this value.

Configuration messages: Displays status information relating to recent configuration events. The log is not editable.

Three types of messages are listed:

 **Information:** This message type provides information about the configuration.

 **Warning:** This message type is an Alert indication that implies a possibly unintended result may require further attention.

 **Error:** This message type indicates that the configuration is invalid and must be corrected before proceeding further.

If the message is too long to be fully displayed in the window, double-click on the message to view the complete message in a separate window.

Serial Tab

The Serial tab is active only if *serial Modbus* is enabled.

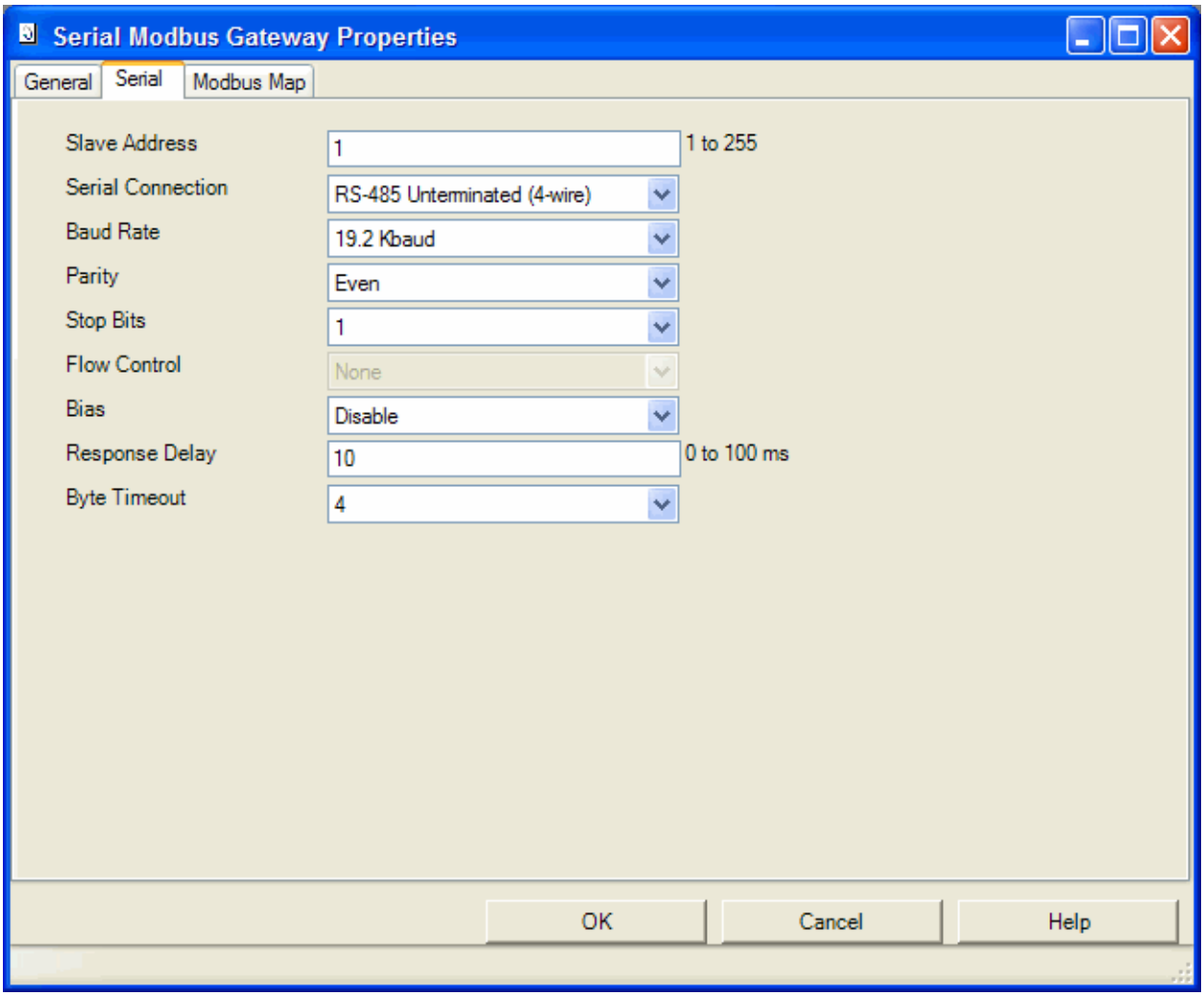


Figure 4 - 3: The Modbus Gateway properties Serial tab.

Slave Address: Enter a Slave Address for this monitor. Valid addresses are 1 to 255. The default Slave Address = 1.

Serial Connection: Select the type of serial connection that the monitor will use to communicate with other Modbus serial devices. The proper selection depends on the protocol used:

- *RS-232:* Select this option when connecting this 1900/65A to a Modbus master RS-232 interface. RS-232 permits only one monitor to communicate with a single Modbus master.
- *RS-485 Unterminated (2-wire):* Select this option when a 2-wire RS-485 bus is used and this 1900/65A is not located at either end of the bus.
- *RS-485 Terminated (2-wire):* Select this option when a 2-wire RS-485 bus is used and this 1900/65A is located at either end of the bus. In this case, the 1900/65A will provide electrical termination by placing an internal 120Ω resistor across the RX+ and RX- terminals that connect to the 2-wire bus.
- *RS-485 Unterminated (4-wire):* Select this option when a 4-wire RS-485 bus is used and this 1900/65A is not located at either end of the bus.
- *RS-485 Terminated (4-wire):* Select this option when a 4-wire RS-485 bus is used and this 1900/65A is located at either end of the bus. In this case, the 1900/65A will provide electrical termination by placing one internal 120Ω resistor across the RX+ and RX- terminals, and a second internal 120Ω resistor across the TX+ and TX- terminals.

Baud Rate: Select an appropriate baud rate. The default is 19,200 baud.

Parity: Select type of parity used: *Even*, *Odd*, or *None*. The default is *Even* parity.

Stop Bits: Select the number of stop bits used: 1 or 2. The default is 1 stop bit.

Flow Control: Select either *None* or *Hardware*. The default is *None*. You can edit the Flow control value only if the RS-232 serial connection is selected.

Bias: The default Bias selection is *Disable*. Typically, the Modbus master will bias the RS-485 bus. In this case (or if another device on the bus provides bias) select *Disable*. If the 1900/65A must provide bias, select *Enable*. This will connect RX- to common and RX+ to +5V through 1.3 kΩ resistors.

Response Delay: This is the minimum amount of time between the end of a master request (once the byte timeout has expired) and when the 1900/65A sends the response. This delay allows the DCS to have enough time to tristate the line and enter the listening mode before the 1900/65A begins transmitting a response.

Byte Timeout: Select a delay time in terms of transmitted bytes. This value defines how long the bus must be inactive (quiet) for the 1900/65A to detect the end of a request from the master. The byte timeout represents the time it takes to send a number of bytes transmitted at the baud rate used by the bus. For example, a byte timeout of 25 bytes transmitted at 19.2 kb/sec yields a delay of 14.3 ms. (25 bytes * 11 bits/byte / (19200 bits/sec) = 14.3 ms).

Review/Edit Modbus Map

Use the Modbus Map tab in the Modbus Gateway Properties window to review or edit existing register mapping, or to manually assign registers to a map. Registers can be dragged from the Available Registers pane to legal address blocks in the Modbus Map pane. Register groups can be added to or removed from the map to create a custom Modbus map. Expand individual register groups to show the registers assigned to that group.

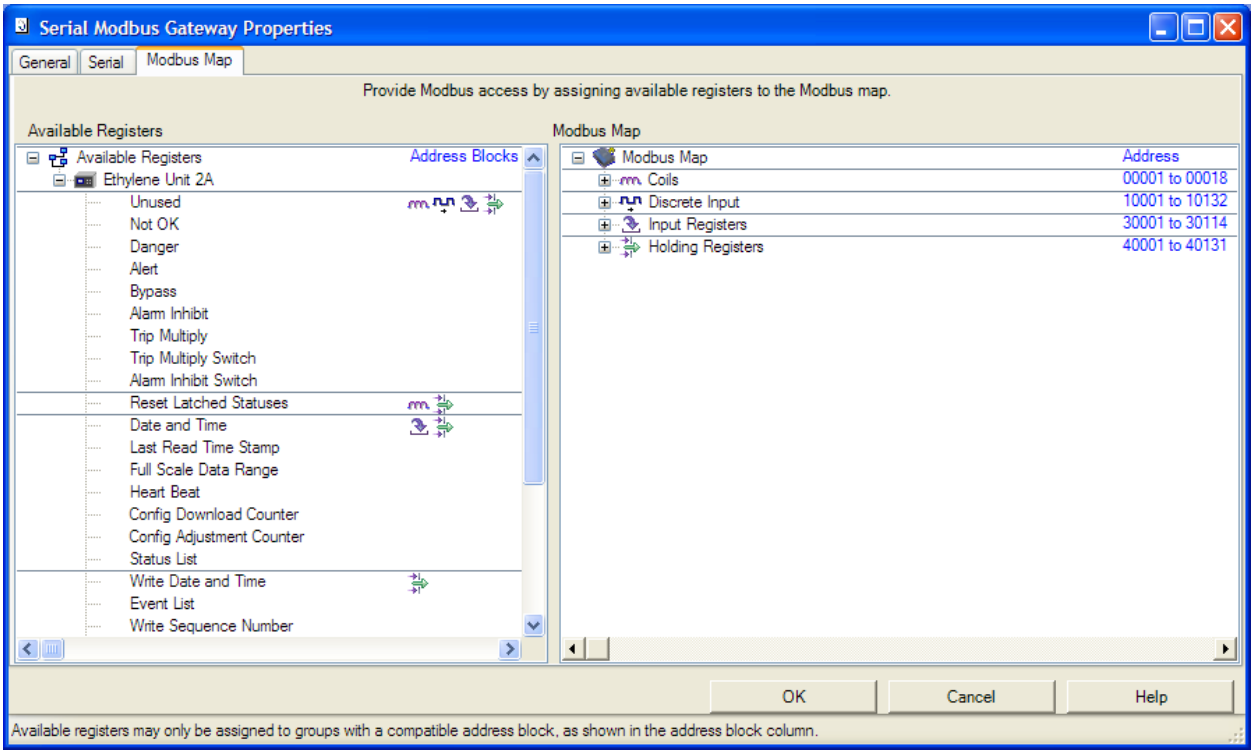


Figure 4 - 4: The Modbus Map tab

Available Registers Window

The Available Registers pane (left) groups 1900/65A registers with icons that represent the Modbus address blocks to which these registers can be mapped. For example, the 1900/65A register containing Reset Latched Statuses can be mapped into the Coils or Holding Registers address blocks.

Table 6-1 on page 49 provides a detailed summary of register to address block mapping.

Modbus Map Window

The Modbus Map right pane (Figure 4 - 4: The Modbus Map tab) lists the Modbus address groups, the registers mapped to these groups, and the range of mapped Modbus addresses. Expanding the group will list each data register that is configured in the group, organized by the type of data to be returned (Monitor, Channel, Variable or Relay). The Modbus Map window lists the register type, channel variable identification, register name, and block address for each mapped register.

Using the Modbus Map Features

Adding New Register Groups

Right click on the Modbus Map icon in the Modbus map pane. Click on the desired register group type to add the new register group.

Register Group Properties

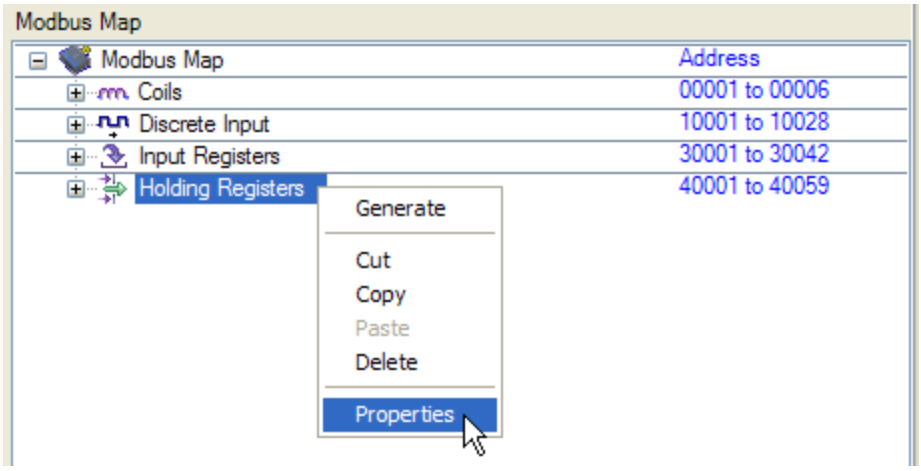


Figure 4 - 5: Opening a register group properties window.

Right click on the register group and select Properties (Figure 4 - 5: Opening a register group properties window.) to open the Register Group Properties window (Figure 4 - 6: Register Group Properties window.).

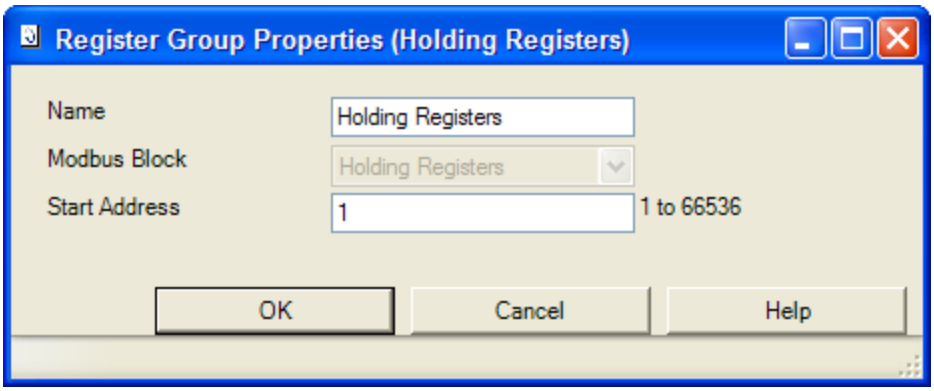


Figure 4 - 6: Register Group Properties window.

Use this window to edit the name of the group and adjust the start address of the register group. All registers added to this group will be assigned consecutive addresses beginning at the Start Address.

Generating a Register Group

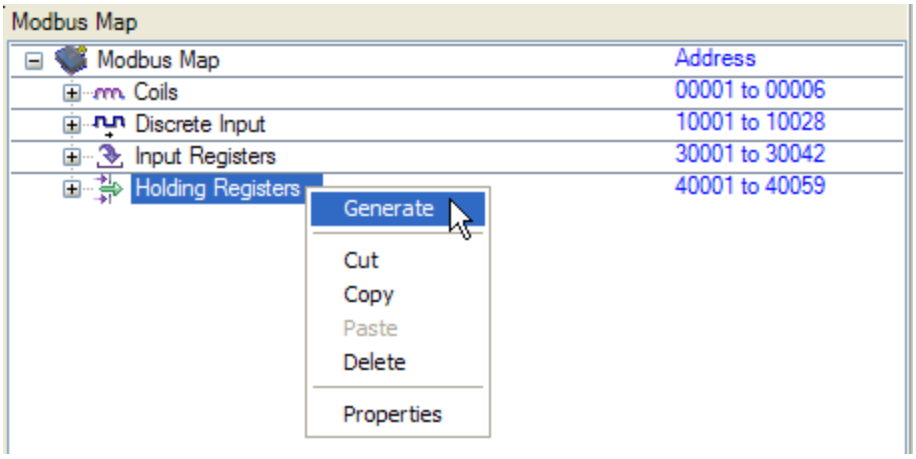


Figure 4 - 7: Opening the Generate Register Group window.

Right click on the address group and select Generate (Figure 4 - 7: Opening the Generate Register Group window.) to open the Generate Register Group window (Figure 4 - 8: Generate Register Group window.).

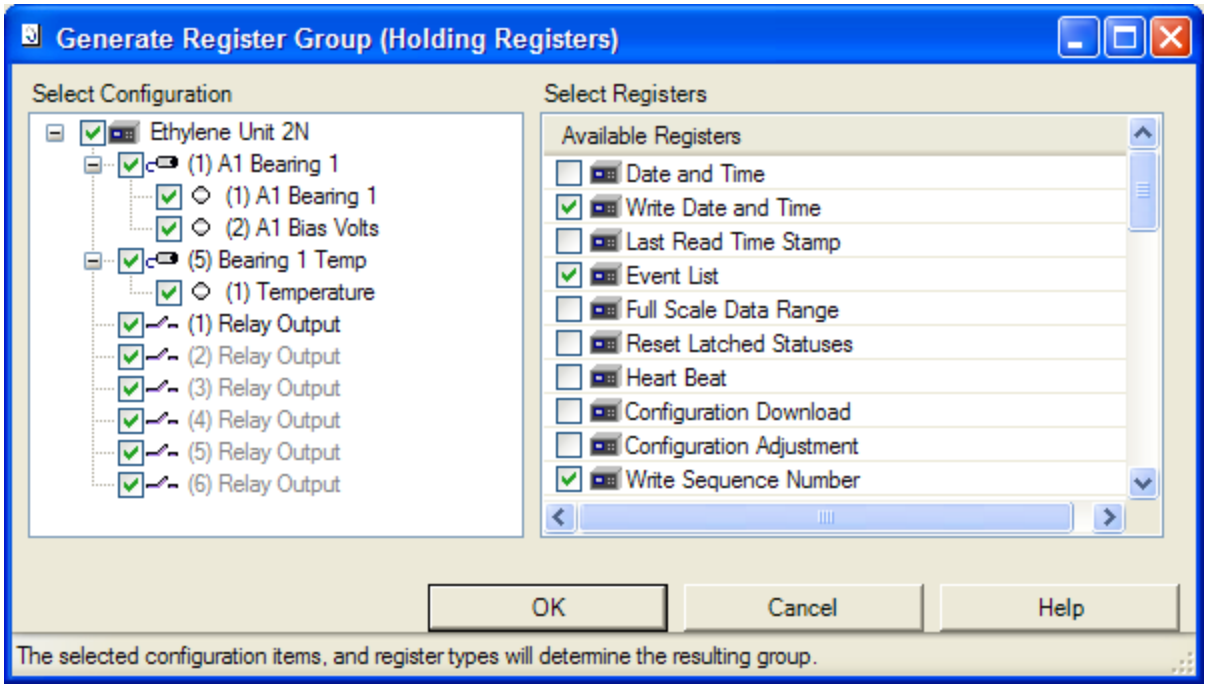


Figure 4 - 8: Generate Register Group window.

Select the desired configuration levels and register types to be mapped:

1. The *Select Configuration* pane (left) displays the available monitor, channel and variable level configurations. Check to select the desired configuration levels.
2. The *Select Registers* pane (right) shows the list of registers that can be mapped. Check the boxes to select the desired register types.
3. Click *OK*.

The software will automatically populate the address group with the data registers of configured points.

Example: Map an Input Register group starting at address 30500 that contains all configured Proportional Amplitude variables.

1. Right click on the *Modbus Map* and select *Add Input Registers Group* to add a new input registers group to the list.
2. Right click on the new *Input Registers* group and select *Properties*.
3. Rename the address group and change the starting address to 500. Click *OK*.
4. Right click on the new *Input Registers* group and select *Generate* to open the *Generate Register Group* dialog window.
5. Check the variables to be mapped in the *Select Configuration* window.
6. Check the *Amplitude (Proportional)* register type in the *Select Registers* window.
7. Click the *OK* button. All the proportional values for the configured points will be added to the address group.

Download Configuration

From the Configuration window (Figure 4 - 9: Starting Download from the context menu.):

1. Right-click on a monitor configuration to open the context menu.
2. Click *Download*. A message window will open.
3. Review the messages and click the *OK* button to complete the download.

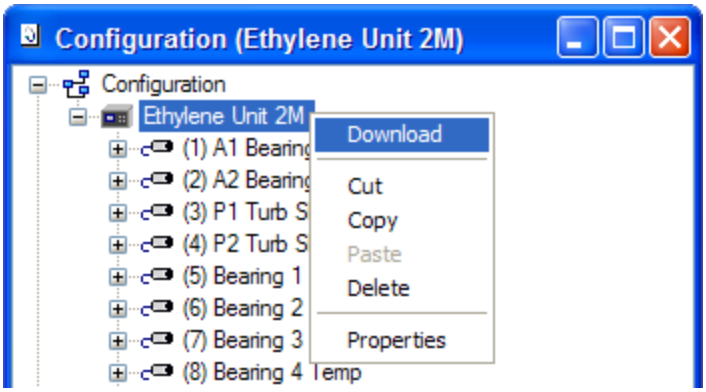


Figure 4 - 9: Starting Download from the context menu.

From the toolbar (Figure 4 - 10: Starting the Download from the toolbar.):

1. Click *Download* to open a window containing a list of online monitors.
2. Select a monitor on the list by clicking its checkbox. A message window will open.
3. Review the messages and click the *OK* button to complete the download.

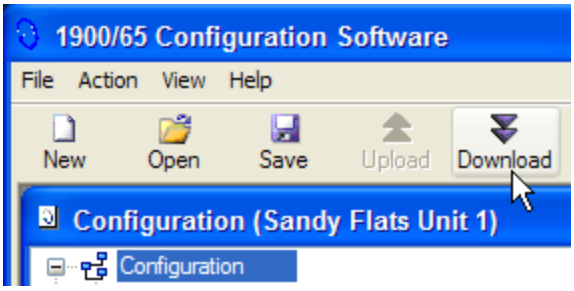


Figure 4 - 10: Starting the Download from the toolbar.

Save Configuration

Save the configuration by clicking the *Save* button on the toolbar or by selecting *File>Save* in the menu bar.

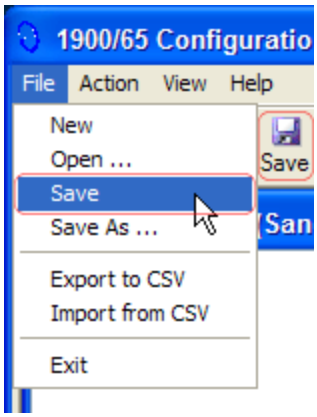


Figure 4 - 11: Saving a configuration.

Activating a Modbus License

If you receive the message shown in Figure 4 - 12: Activate license message., you will need to activate your license to enable the Modbus Gateway.

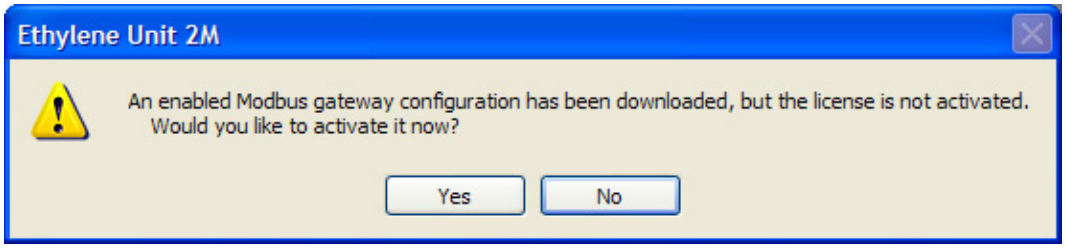


Figure 4 - 12: Activate license message.

1. Launch the 1900/65 (v4.2 and above) software and connect to the device.
2. Click on the **Online** button to locate online devices.

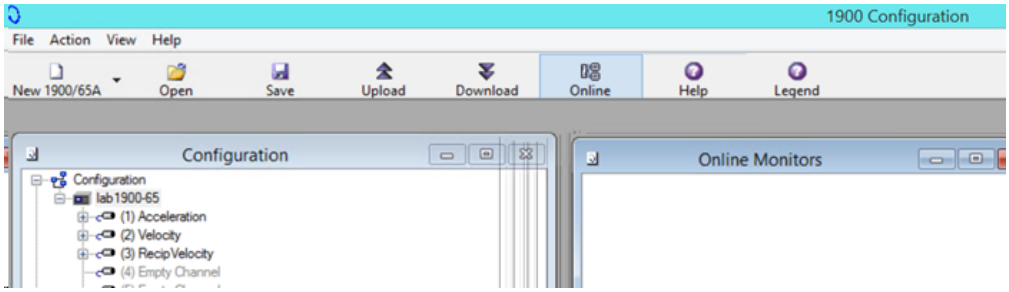


Figure 4 - 13: Online device search

3. Once located, right click on the device and click on option '**Activate Modbus License**'.

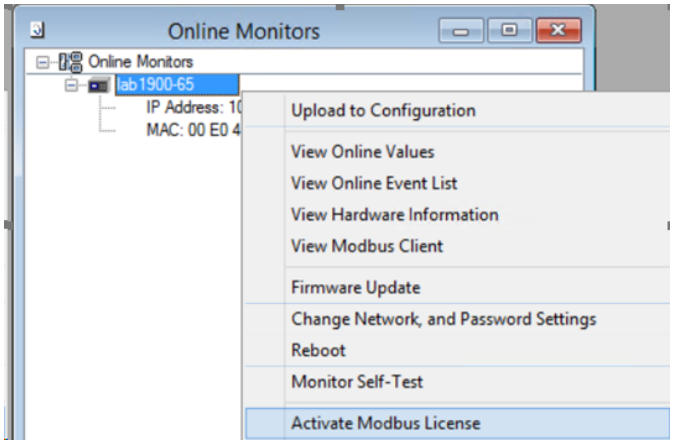


Figure 4 - 14: Activate Modbus License Menu

4. Upon successful activation, message *'Activate Modbus License operation was successfully completed'* will show up on the bottom screen.

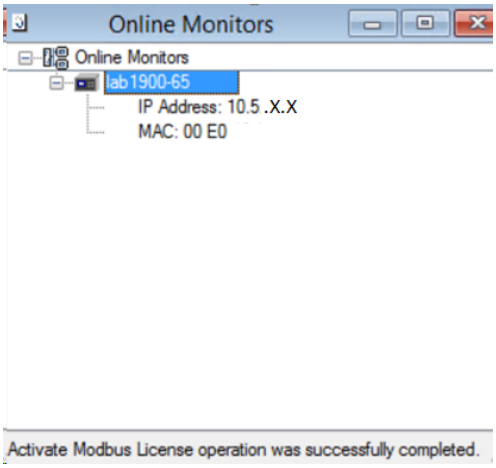


Figure 4 - 15: Activate Modbus License completed message

4.2 Checking Modbus Communications

The 1900 Configuration Software has an embedded Modbus client that allows you to verify the data returned in the Modbus map and compare the data with your control system. To view the Modbus Registers:

1. Open the *Online Monitors* window.
2. Right-click on a monitor. The context menu will open.
3. Click *View Modbus Client* to open the *Modbus Client* window (Figure 4 - 16: Modbus Client window.). This window shows the register group, Modbus address, and the value returned for each mapped register.

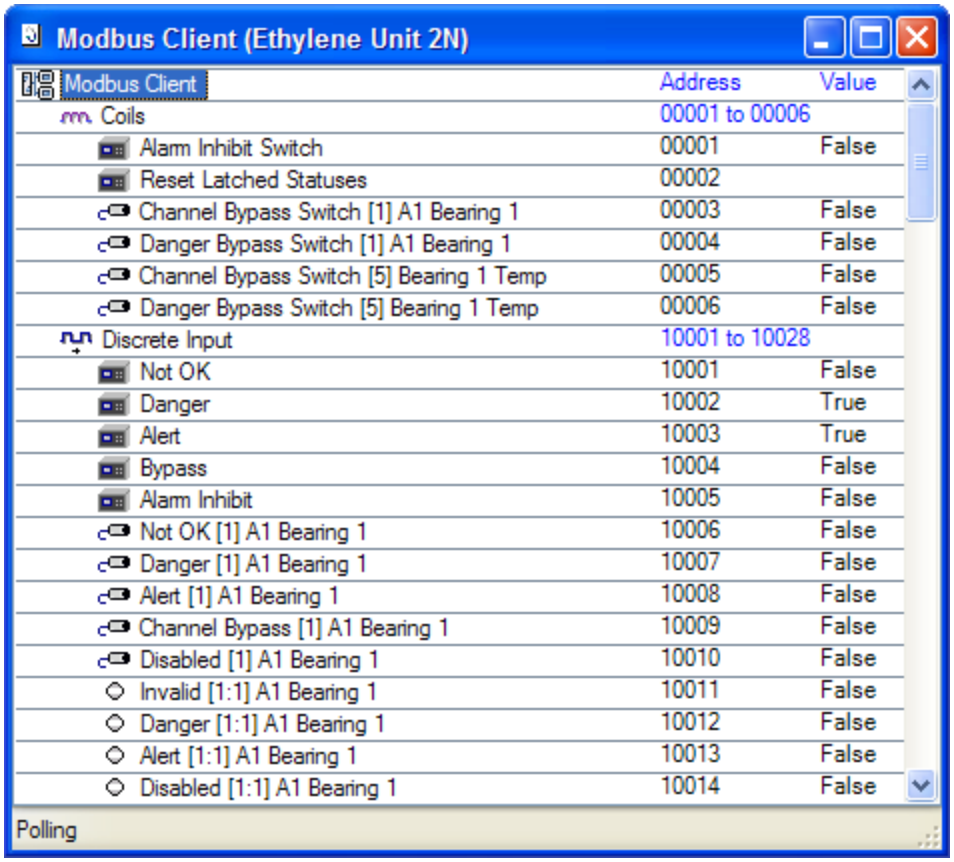


Figure 4 - 16: Modbus Client window.

4.3 Exporting a Modbus Map

The 1900 Configuration Software can export a Modbus map to a text file in CSV (Comma Separated Values) format. CSV files are compatible with spreadsheet programs like Microsoft Excel and text editors like Wordpad. Figure 4 - 17: Exported Modbus map file viewed in an ASCII text editor. and Figure 4 - 18: Exported Modbus map file viewed in a spreadsheet. show examples of a Modbus map file opened in a text editor and a spreadsheet, respectively.

Exporting from a Monitor Configuration

To export a Modbus map from a monitor configuration:

1. Upload the monitor to the configuration window.
2. Click on *File>Export to CSV* to open the *Export to CSV* dialog window.
3. Click the *Modbus Map* check box (to export the Modbus Map) and then click *OK*.
4. Enter the file name of the CSV file to create.
5. Click *OK*.

Modbus Map Examples

The CSV file organizes data into multiple columns with each row representing the data for one register. Commas delimit columns in the text file. The first row indicates the content of each column.

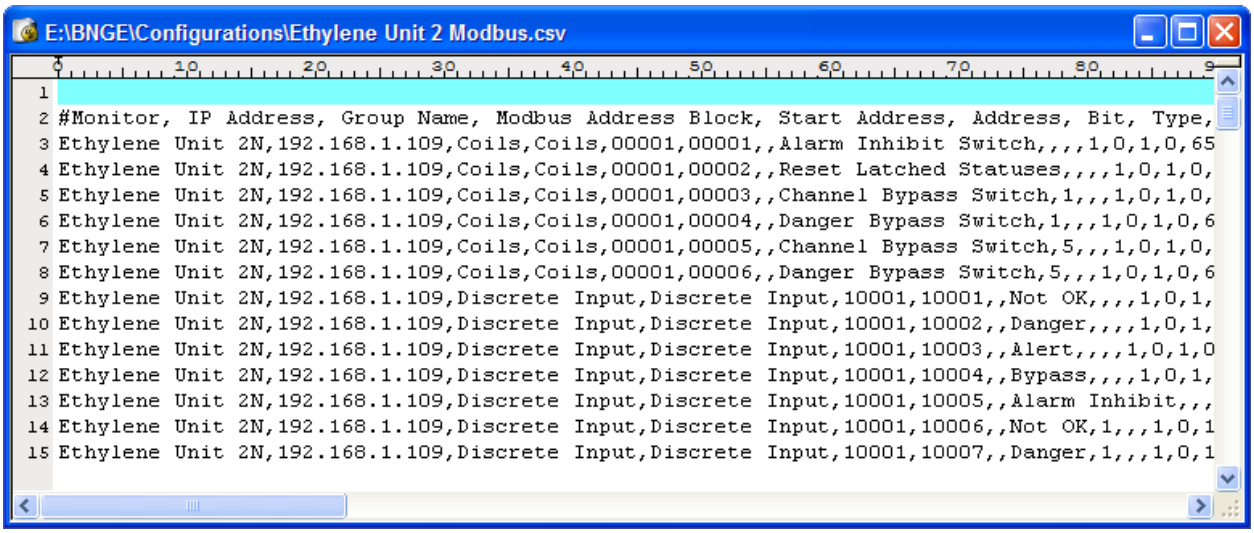
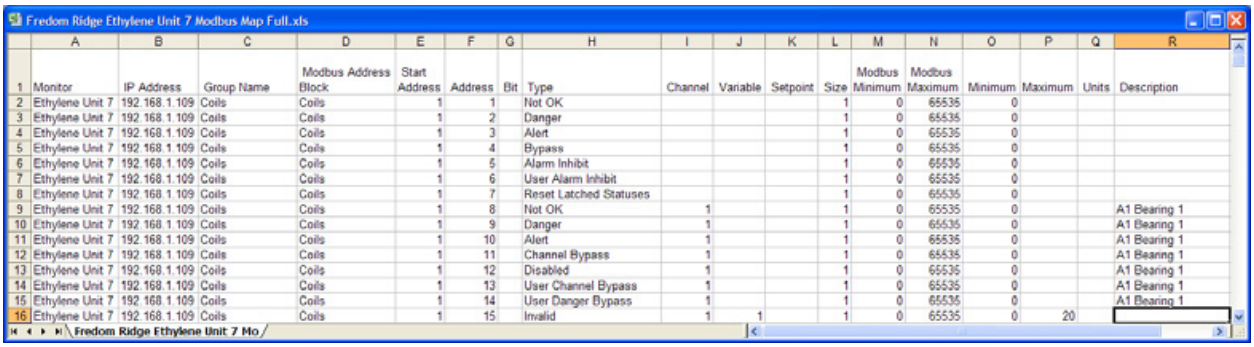


Figure 4 - 17: Exported Modbus map file viewed in an ASCII text editor.



Fredom Ridge Ethylene Unit 7 Modbus Map Full.xls

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
1	Monitor	IP Address	Group Name	Modbus Address Block	Start Address	Address	Bit	Type	Channel	Variable	Setpoint	Size	Modbus Minimum	Modbus Maximum	Minimum	Maximum	Units	Description
2	Ethylene Unit 7	192.168.1.109	Coils	Coils	1	1		Not OK				1	0	65535	0			
3	Ethylene Unit 7	192.168.1.109	Coils	Coils	1	2		Danger				1	0	65535	0			
4	Ethylene Unit 7	192.168.1.109	Coils	Coils	1	3		Alert				1	0	65535	0			
5	Ethylene Unit 7	192.168.1.109	Coils	Coils	1	4		Bypass				1	0	65535	0			
6	Ethylene Unit 7	192.168.1.109	Coils	Coils	1	5		Alarm Inhibit				1	0	65535	0			
7	Ethylene Unit 7	192.168.1.109	Coils	Coils	1	6		User Alarm Inhibit				1	0	65535	0			
8	Ethylene Unit 7	192.168.1.109	Coils	Coils	1	7		Reset Latched Statuses				1	0	65535	0			
9	Ethylene Unit 7	192.168.1.109	Coils	Coils	1	8		Not OK	1			1	0	65535	0			A1 Bearing 1
10	Ethylene Unit 7	192.168.1.109	Coils	Coils	1	9		Danger	1			1	0	65535	0			A1 Bearing 1
11	Ethylene Unit 7	192.168.1.109	Coils	Coils	1	10		Alert	1			1	0	65535	0			A1 Bearing 1
12	Ethylene Unit 7	192.168.1.109	Coils	Coils	1	11		Channel Bypass	1			1	0	65535	0			A1 Bearing 1
13	Ethylene Unit 7	192.168.1.109	Coils	Coils	1	12		Disabled	1			1	0	65535	0			A1 Bearing 1
14	Ethylene Unit 7	192.168.1.109	Coils	Coils	1	13		User Channel Bypass	1			1	0	65535	0			A1 Bearing 1
15	Ethylene Unit 7	192.168.1.109	Coils	Coils	1	14		User Danger Bypass	1			1	0	65535	0			A1 Bearing 1
16	Ethylene Unit 7	192.168.1.109	Coils	Coils	1	15		Invalid	1	1		1	0	65535	0	20		

Fredom Ridge Ethylene Unit 7 Mo

Figure 4 - 18: Exported Modbus map file viewed in a spreadsheet.

4.4 Creating a Configuration File for Import

If you prefer using a text editor to edit a configuration, consider exporting the configuration to a CSV file, editing the file, and importing the revised file.

To create a configuration file with a text editor:

1. Create a header row to define the data contained in each column. The header row must contain these terms separated with a comma:

   Monitor, IP Address, Group Name, Modbus Address Block, Start Address, Address, Bit, Type, Channel, Variable, Setpoint, Size, Modbus Minimum, Modbus Maximum, Minimum, Maximum, Units, Description, Gateway

2. List the appropriate data in each subsequent row below the header:

   Ethylene Unit 7,192.168.1.109,Coils,Coils,00001,

   0001,,Not OK,,,,1,0,65535,0,,,

The following rules apply for CSV validation:

- Data must be valid
- Registers are validated for input type, channel, variable type, and address
- Register types must be valid for their block type
- Empty lines are skipped
- A register must have the correct fields defined for a given register type

All errors are reported to the user after import is complete by line, field and description.

4.5 Importing a Modbus Map

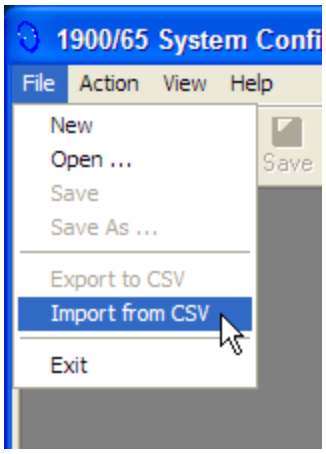


Figure 4 - 19: Selecting Import from CSV from menu.

Click *File>Import from CSV* on the menu bar (Figure 4 - 19: Selecting Import from CSV from menu.) to open the *Import Modbus Map* window (Figure 4 - 20: Import Modbus Map window.).

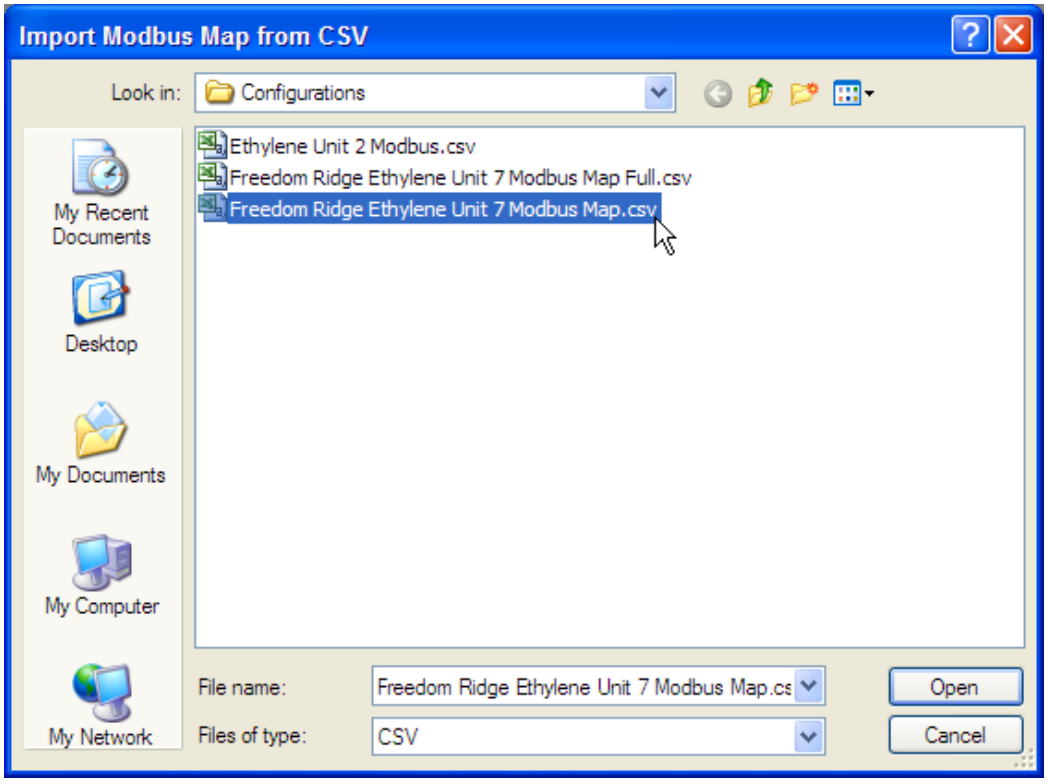


Figure 4 - 20: Import Modbus Map window.

Locate and select the desired CSV file to import. Click *Open* to load the file as a configuration.

5. Modbus / 1900/65A Protocol

This chapter discusses the Modbus protocol used by the 1900/65A monitor. When used in this context, the Modbus protocol provides client/server commu- nication over Ethernet, or master/slave communication over the RS-232/RS485 serial port.

The 1900/65A monitor supports simultaneous connection of up to 6 Modbus clients via Modbus Messaging on TCP/IP.

5.1 Modbus Reference Documents

This chapter covers the Modbus protocol at the PDU level as it applies to the1900/65A. Refer to the following documents for protocol information not covered here.

The Modbus specifications are covered in the following documents:

- Modbus Application Protocol version 1.1a
- Modbus Messaging on TCP/IP Implementation Guide version 1.0a
- Modbus Over Serial Line Specifications and Implementation Guide version 1.0

Documents are available free of charge at: www.modbus.org.

5.2 Terminology

Client/Server vs. Master/Slave

When the 1900/65A monitor communicates in a networked environment, the terms *client* and *server* are used instead of *master* and *slave* to describe the role of networked devices. For example, a Process Control System (PCS) acts as a client because it requests data, and a 1900/65A monitor acts as a server because it responds by providing the requested data.

Monitor Configuration Terms

The 1900/65A monitor uses a flexible configuration structure to define how the monitor processes and uses input information. Channels 1, 2, 3 and 4 support acceleration, velocity, speed and displacement transducers. Channels 5, 6, 7 and 8 support thermocouples and RTDs.

The configuration structure consists of the following items:

Configuration: This is information in a user-created file that represents one or more 1900/65 monitors in a facility or on a machine train.

Monitor: This contains the configuration for a specific 1900/65A monitor on the network.

Channel Type: This is one of seven transducer types (acceleration, recip acceleration, velocity, recip velocity, radial vibration, thrust position or speed) on channels 1 through 4, plus temperature on channels 5 through 8. The channel type must correspond to the type of transducer connected to the input.

Variables: Variables define the kind of measurements that the monitor obtains from the input signal source after the raw signal is processed and filtered.

Analog Output: This is one of four 4-20 mA outputs each of which may be assigned to represent a variable from any of the eight inputs.

Relay Output: This is one of six relay outputs that can be programmed to respond to logical combinations of monitor, channel, or variable status states.

Properties: Properties define how an object works. Objects in the hierarchy have one or more properties that can be edited.

5.3 Modbus Protocol Summary

Modbus Data Model

Modbus is a messaging protocol for client/server communication between devices that are connected on a bus or network. Client and server devices pass information by reading and writing to registers located in the server. Since the 1900/65A monitor acts as a server, these registers are located in the monitor.

The Modbus specification defines four register tables, each differentiated by register type and read-write access. Table 5 - 1: Modbus register tables describes these tables.

Table 5 - 1: Modbus register tables

Table Type	Object Type	Client Access	Comments
Discrete Inputs	1 bit	Read only	Client can read only. Only the server device can change its register values.
Coils	1 bit	Read/Write	Both client and server can read and write to these registers
Input Registers	16-bit word	Read only	Client can read only. Only the server device can change its register values.
Holding Registers	16-bit word	Read/Write	Both client and server can read and write to these registers.

Register Addresses

Table 5 - 2: Acceptable Modbus table address ranges. shows the acceptable address ranges that may be assigned the different table types.

Table 5 - 2: Acceptable Modbus table address ranges.

The PDU (Protocol Data Unit) must be offset by -1.

Table Type	Modbus Model Numbering	PDU Addressing
Coils	Coil 000001 - Coil 065536	000000 - 065535
Discrete Inputs	Discrete Input 100001 - Discrete Input 165536	100000 - 165535
Input Registers	Input Register 300001 - Input Register 365536	300000 - 365535
Holding Registers	Holding Register 400001 - Holding Register 465536	400000 - 465535

The Protocol Data Unit

The Modbus protocol is based on a transaction between a client (or master) and a server (or slave). A message between client and server is in the form of a request and reply defined by function codes. A function code is the basic element of the Modbus *Protocol Data Unit* (PDU).

A complete Modbus transaction consists of the transmission of a request from a client and a response from the addressed server. Request and response PDUs contain two fields as shown here:

Function Code	**Date Field (zero to many bytes)**
(1 byte)	Most significant byte sent first...Least significant byte sent last
	first - Order of Transmission - last

The length of each field is an integral multiple of 8 bit bytes. The function code field is sent first with the content of the data field following in the order shown.

Function Code

The function code is the first field of the PDU. It is a one-byte value from 0x00 to 0xFF, although values 0x80 to 0xFF are reserved for exception responses. The 1900/65A monitor supports the function codes listed in Table 5 - 3: Modbus function codes supported by the 1900/65A.

Table 5 - 3: Modbus function codes supported by the 1900/65A.

Code	Function
01	Read Coils
02	Read Discrete Inputs
03	Read Holding Registers
04	Read Input Registers
05	Write Single Coil
06	Write Single Register
08	Diagnostics
15	Write Multiple Coils
16	Write Multiple Registers
17	Report Slave ID

Data Field

The data field contains a register address and data associated with a request or response defined by the function code. The data field can vary in length from zero bytes to many bytes depending on the type of function.

Message Types

Static variable data, statuses, and parameter changes requested by a Master

(Client) will comprise the inputs to the Modbus gateway. The Modbus data field represents the type of data requested. The register address specifies the type of register valid for the request.

5.4 Function Code Definitions

The function codes described in this section are specific to the 1900/65A monitor. Refer to the Modbus Application Protocol (see Modbus Reference Documents on page 35.) for definitions of other function codes.

Diagnostics (08)

The diagnostic request provides a series of tests for checking the communication system between a client device (e.g. DCS) and the 1900/65A monitor. This request can also check certain internal error conditions within the monitor. A sub-function code sent with the request specifies specific diagnostic functions.

Request

Function Code (1 byte)	Diagnostic Code (2 bytes)		Data Byte 1 (1 byte)	Data Byte 2 (1 byte)
0x08	high byte	low byte	high byte	low byte

Diagnostic Code: This is a sub-function code that represents the diagnostic to execute. Table 5 - 4: Diagnostic codes. lists the diagnostics that the 1900/65A supports.

Table 5 - 4: Diagnostic codes.

Diagnostic Code (high byte)	Diagnostic Code (low byte)	Name of Diagnostic
0x00	0x00	Return Query Data (Section Return Query Data (Diagnostic Code 0x00))
0x00	0x05	Monitor Diagnostic Register (Section Monitor Diagnostic Register (Diagnostic Code 0x05))
0x00	0x06	Not Used
0x00	0x07	Not Used
0x00	0x08	Not Used
0x00	0x09	Not Used
0x00	0x0A	Clear Counters (Section Clear Counters (Diagnostic Code 0x0A))
0x00	0x0B	Message Count (Section Message Count (Diagnostic Code 0x0B))
0x00	0x0C	Communications Error Count (Section Communication Error Count (Diagnostic Code 0x0C))
0x00	0x0D	Exception Count (Section Exception Count (Diagnostic Code 0x0D))

Diagnostic Code (high byte)	Diagnostic Code (low byte)	Name of Diagnostic
0x00	0x0E	Return Slave Message Count (Section Slave Message Count (Diagnostic Code 0x0E))
0x00	0x0F	Return Slave No Response Count (Section Slave No Response Count (Diagnostic Code 0x0F))
0x00	0x11	Return Slave Busy Count (Section Slave Busy Count (Diagnostic Code 0x11))
0x00	0x12	Return Bus Character Overrun Count (Section Bus Character Overrun Count (Diagnostic Code 0x12))

Data bytes 1 and 2 and the corresponding response depend upon the requested diagnostic. See Diagnostic Codes.

Exception Response

Function Code	Valid Exception Code (1 byte)
0x08	0x01, 0x03, or 0x04

Valid Exception Codes: 01, 03, or 04. See Section 5.5 Exception Codes for definitions.

Diagnostic Codes

The 1900/65A supports all Serial Line Diagnostic Counters (0x0B, 0x0C, 0x0D, 0x0E, 0x0F, 0x11, and 0x12) except for “Return Slave NAK Count” (0x10). See page 35 of the Modbus over Serial Line Specification.

Return Query Data (Diagnostic Code 0x00)

This is a loopback test. The data in the request is returned in the response. The response message should be identical to the request.

Example: Return Query Data Request

Function Code (1 byte)	Diagnostic Code (2 bytes)		Data Byte 1 (1 byte)	Data Byte 2 (1 byte)
0x08	0x00	0x00	0xAA	0x55

Example: Return Query Data Response

Function Code (1 byte)	Diagnostic Code (2 bytes)		Data Byte 1 (1 byte)	Data Byte 2 (1 byte)
0x08	0x00	0x00	0xAA	0x55

Monitor Diagnostic Register (Diagnostic Code 0x05)

The data in the response message returns the contents of the 1900/65A monitor's diagnostic register.

Example: Monitor Diagnostic Register Response

Function Code (1 byte)	Diagnostic Code (2 bytes)		Data Byte 1[a]	Data Byte 2[a]
0x08	0x00	0x05	Expanded below	Expanded below

	Bit	Bit Value
Data Byte 1[a]	7	
	6	
	5	
	4	
	3	
	2	
	1	
	0	Display Source Fault[b]
Data Byte 2[a]	7	Channel 8 Hardware Error
	6	Channel 7 Hardware Error
	5	Channel 6 Hardware Error
	4	Channel 5 Hardware Error
	3	Channel 4 Hardware Error
	2	Channel 3 Hardware Error
	1	Channel 2 Hardware Error
	0	Channel 1 Hardware Error

a. A logic "1" in the bit pattern represents a FAILURE in the respective area.

b. Display Source Fault. The voltage source in the monitor used to power the Display Module is out of spec. A problem with the cable connecting the display to the monitor could result in this fault.

Clear Counters (Diagnostic Code 0x0A)

This diagnostic will clear all monitor diagnostic counters.

These counters are also cleared when the monitor is powered up.

Example: Clear Counters Request

Function Code (1 byte)	Diagnostic Code (2 bytes)		Data Byte 1 (1 byte)	Data Byte 2 (1 byte)
0x08	0x00	0x0A	0x00	0x00

Example: Clear Counters Response

Function Code (1 byte)	Diagnostic Code (2 bytes)		Data Byte 1 (1 byte)	Data Byte 2 (1 byte)
0x08	0x00	0x0A	0x00	0x00

Message Count (Diagnostic Code 0x0B)

The data in the response message returns the number of Modbus request messages that the monitor has received on the network since its last restart, power-up, or the last time its counters were cleared.

Example: Message Count Request

Function Code (1 byte)	Diagnostic Code (2 bytes)		Data Byte 1 (1 byte)	Data Byte 2 (1 byte)
0x08	0x00	0x0B	0x00	0x00

Example: Message Count Response

Function Code (1 byte)	Diagnostic Code (2 bytes)		Data Byte 1 (1 byte)	Data Byte 2 (1 byte)
0x08	0x00	0x0B	Message count high	Message count low

The value in Data Bytes 1 and 2 contains the number of Modbus request messages received by the monitor since the last reset, power-up or counter-clear request.

Communication Error Count (Diagnostic Code 0x0C)

The data in the response message returns the number of Modbus communication errors encountered by the monitor since its last restart, power-up, or the last time its counters were cleared. For example, the Modbus communication error count would be incremented if a request's message length were inconsistent with the function code's expected length.

Example: Communication Error Count Request

Function Code (1 byte)	Diagnostic Code (2 bytes)		Data Byte 1 (1 byte)	Data Byte 2 (1 byte)
0x08	0x00	0x0C	0x00	0x00

Example: Communication Error Count Response

Function Code (1 byte)	Diagnostic Code (2 bytes)		Data Byte 1 (1 byte)	Data Byte 2 (1 byte)
0x08	0x00	0x0C	Communication error count high	Communication error count low

The value in Data Bytes 1 and 2 contains the number of Modbus communication errors detected by the monitor since the last reset, power-up, or counter-clear request.

Exception Count (Diagnostic Code 0x0D)

The data in the response message returns the number of exception responses returned by the monitor since its last restart, power-up, or the last time its counters were cleared.

Example: Exception Count Request

Function Code (1 byte)	Diagnostic Code (2 bytes)		Data Byte 1 (1 byte)	Data Byte 2 (1 byte)
0x08	0x00	0x0D	0x00	0x00

Example: Exception Count Response

Function Code (1 byte)	Diagnostic Code (2 bytes)		Data Byte 1 (1 byte)	Data Byte 2 (1 byte)
0x08	0x00	0x0D	Exception count high	Exception count low

The value in Data Bytes 1 and 2 contains the number of Exception Responses detected by the monitor since the last reset, power-up, or counter-clear request.

Slave Message Count (Diagnostic Code 0x0E)

The data in the response message returns the quantity of messages addressed to the remote device or broadcast that the remote device has processed since its last restart, power-up, or the last time its counters were cleared.

Example: Slave Message Count Request

Function Code (1 byte)	Diagnostic Code (2 bytes)		Data Byte 1 (1 byte)	Data Byte 2 (1 byte)
0x08	0x00	0x0E	0x00	0x00

Example: Slave Message Count Response

Function Code (1 byte)	Diagnostic Code (2 bytes)		Data Byte 1 (1 byte)	Data Byte 2 (1 byte)
0x08	0x00	0x0E	Slave Message Count high	Slave Message Count low

The value in Data Bytes 1 and 2 contains the number of messages detected by the monitor since its last restart, power-up, or the last time its counters were cleared.

Slave No Response Count (Diagnostic Code 0x0F)

The data in the response message returns the quantity of messages addressed to the remote device for which it has returned no normal or exception response since its last restart, power-up, or the last time its counters were cleared.

Example: Slave No Response Count Request

Function Code (1 byte)	Diagnostic Code (2 bytes)		Data Byte 1 (1 byte)	Data Byte 2 (1 byte)
0x08	0x00	0x0F	0x00	0x00

Example: Slave No Response Count Response

Function Code (1 byte)	Diagnostic Code (2 bytes)		Data Byte 1 (1 byte)	Data Byte 2 (1 byte)
0x08	0x00	0x0F	Slave No Response Count high	Slave No Response Count low

The value in Data Bytes 1 and 2 contains the number of messages addressed to the remote device for which it has returned no normal or exception response since its last restart, power-up, or the last time its counters were cleared.

Slave Busy Count (Diagnostic Code 0x11)

The data in the response message returns the quantity of messages addressed to the remote device for which it returned a Slave Device Busy exception response since its last restart, power-up, or the last time its counters were cleared.

Example: Slave Busy Count Request

Function Code (1 byte)	Diagnostic Code (2 bytes)		Data Byte 1 (1 byte)	Data Byte 2 (1 byte)
0x08	0x00	0x11	0x00	0x00

Example: Slave Busy Count Response

Function Code (1 byte)	Diagnostic Code (2 bytes)		Data Byte 1 (1 byte)	Data Byte 2 (1 byte)
0x08	0x00	0x11	Slave Device Busy Count high	Slave Device Busy Count low

The value in Data Bytes 1 and 2 contains the number of messages addressed to the remote device for which it returned a Slave Device Busy exception response since its last restart, power-up, or the last time its counters were cleared.

Bus Character Overrun Count (Diagnostic Code 0x12)

The data in the response message returns the number of messages addressed to the remote device that it could not handle due to a character overrun condition since its last restart, power-up, or the last time its counters were cleared. A character overrun is caused by data characters arriving at the port faster than the receiving device can store them, or by the loss of a character due to a hardware malfunction.

Example: Bus Character Overrun Count Request

Function Code (1 byte)	Diagnostic Code (2 bytes)		Data Byte 1 (1 byte)	Data Byte 2 (1 byte)
0x08	0x00	0x12	0x00	0x00

Example: Bus Character Overrun Count Response

Function Code (1 byte)	Diagnostic Code (2 bytes)		Data Byte 1 (1 byte)	Data Byte 2 (1 byte)
0x08	0x00	0x12	Bus Character Overrun Count high	Bus Character Overrun Count low

The value in Data Bytes 1 and 2 contains the number of messages addressed to the remote device that it could not handle due to a character overrun condition since its last restart, power-up, or the last time its counters were cleared.

Slave Report ID (17, 0x11)

This reads status and configuration information specific to the 1900/65A monitor.

Request

Function Code
0x11

Function code: 17 (0x11).

No data is associated with this request.

Response

Function Code (1 byte)	Byte Count (1 byte)	Family ID 1	Family ID 2	Family ID 3	Family ID 4	Run indicator status	Major Revision Number	Minor Revision Number
0x11	0x07	high byte	byte	byte	low byte	bye	byte	byte

Family ID: 1900/65A Monitor Module part number 167706. Added space value in hex is:

- Family ID 1: 0x00
- Family ID 2: 0x02
- Family ID 3: 0x8F
- Family ID 4: 0x1A

Run indicator status: Always 0xFF. 0x00=off, 0xFF=on.

Major Rev Number: The current major firmware revision number.

Minor Rev Number: The current minor firmware revision number.

Exception Response

Function Code	Valid Exception Code (1 byte)
0x91	0x01, or 0x04

Valid Exception Codes: 0x01 or 0x04. See Section 5.5 Exception Codes for definitions.

5.5 Exception Codes

These are standard exception codes that both Ethernet and serial Modbus support.

A response indicates an error by setting the high order bit of the function code field to 1. The response also returns an exception code in the data field as a one-byte value. The error message and probable cause are given in Table 5 - 5: Exception codes.

Table 5 - 5: Exception codes.

Code	Error Message	Cause
01	ILLEGAL FUNCTION	Receipt of a request before the monitor receives a Modbus configuration. Receipt of a request having a function code not supported by the monitor.
02	ILLEGAL DATA ADDRESS	Receipt of a request to read or write a register outside valid address range.
03	ILLEGAL DATA VALUE	Value in the Data field is not an acceptable value. Receipt of a request to read or write more registers than the function supports. Receipt of a diagnostic request with an unsupported sub-function code.
04	SLAVE DEVICE FAILURE	Receipt of a request having a data length inconsistent with its function code. Receipt of a request to write an unacceptable value to a register. Receipt of a request to write to a register, which contains read-only information.
06	SLAVE DEVICE BUSY	Receipt of a request while the monitor is in initialization mode. A Modbus register request is attempted from the Modbus utility while a configuration is being downloaded to the monitor from the configuration software. The Modbus request will not be processed.

5.6 Increased Response Time

There are certain operations and scenarios that increase the 1900/65A monitor's response time. These are:

- Downloading firmware. Modbus requests will not be answered
- Downloading configuration
- Writing to multiple setpoints using a single Write Registers command

6. Mapping Data to Registers

This chapter defines how the 1900/65A monitor maps data to Modbus registers. See Register and Data table for the 1900/65A monitor. for a summary. Data/register associations are detailed in Section Data/Register Associations.

The data register mapping is the same for both Ethernet and Serial Modbus.

6.1 Terms

These terms apply when used in a register mapping context:

Register type: This term represents the specific 1900/65A monitor data to be returned.

Register Address: This is the one-based address of a Modbus register that contains specific 1900/65A monitor data or statuses. Data is assigned to registers in one of four register blocks: Discrete Inputs, Coils, Input Registers, or Holding Registers. Each register block implies an address range.

One-based: This implies that an address for a register is relative to a starting address of X00001, and the first register in a block has an address of X00001. By definition, Modbus addresses are provided as one-based values.

Zero-based: This implies that an address for a register is relative to a starting address of X00000, and the first register in a block has an address of X00000. By definition, addresses embedded in Modbus PDUs are provided as zero-based values.

Monitor: This implies that data is associated with the monitor in general, and not to a specific channel type or variable.

Channel: This implies that data is associated with a specific channel. There are up to 8 channels available per monitor.

Variable: This implies that data is associated with a specific variable. Variables define the kind of measurements obtained from the input signal source after the raw signal is processed and filtered.

6.2 Mapped Data Summary

Table 6 - 1: Register and Data table for the 1900/65A monitor.

	Client Access	Length	Register Block				Context		
			Coils	Discrete Inputs	Input Registers	Holding Registers	Monitor	Channel	Variable
Date and Time	R	7 Registers			X	X	X		
Write Date and Time	R/W	7 Registers				X	X		
Last Read Time Stamp	R	7 Registers			X	X	X		
Event List	R/W	18 Registers				X	X		
Full Scale Data Range	R	1 Register			X	X	X		
Reset Latched Statuses	W	1 Register	X			X	X		
Heart Beat	R	1 Register			X	X	X		
Config Download Counter	R	1 Register			X	X	X		
Config Adjustment Counter	R	1 Register			X	X	X		
Write Sequence Number	W	1 Register				X	X		
Monitor Not OK	R	1 Register	X	X	X	X	X		
Monitor Danger	R	1 Register	X	X	X	X	X		
Monitor Alert	R	1 Register	X	X	X	X	X		

	Client Access	Length	Register Block				Context		
			Coils	Discrete Inputs	Input Registers	Holding Registers	Monitor	Channel	Variable
Monitor Bypass	R	1 Register	X	X	X	X	X		
Alarm Inhibit	R	1 Register	X	X	X	X	X		
Alarm Inhibit Switch	R/W	1 Register	X	X	X	X	X		
Trip Multiply	R	1 Register	X	X	X	X	X		
Trip Multiply Switch	R/W	1 Register	X	X	X	X	X		
Monitor Status List	R/W	1 Register			X	X	X		
Channel Not OK	R	1 Register	X	X	X	X		X	
Channel Danger	R	1 Register	X	X	X	X		X	
Channel Alert	R	1 Register	X	X	X	X		X	
Channel Bypass	R	1 Register	X	X	X	X		X	
Channel Bypass Switch	R/W	1 Register	X	X	X	X		X	
Channel Danger Bypass Switch	R/W	1 Register	X	X	X	X		X	
Channel Disabled	R	1 Register	X	X	X	X		X	
Channel Status List	R/W	1 Register			X	X		X	

	Client Access	Length	Register Block				Context		
			Coils	Discrete Inputs	Input Registers	Holding Registers	Monitor	Channel	Variable
Variable Invalid	R	1 Register	X	X	X	X			X
Variable Danger	R	1 Register	X	X	X	X			X
Variable Alert	R	1 Register	X	X	X	X			X
Variable Disabled	R	1 Register	X	X	X	X			X
Variable Amplitude, Proportional	R	1 Register			X	X			X
Variable Amplitude, Floating Point	R	2 Registers			X	X			X
Variable Top Scale, Floating Point	R	2 Registers			X	X			X
Variable Bottom Scale, Floating Point	R	2 Registers			X	X			X
Variable Under Speed	R	1 Register	X	X	X	X			X
Variable Transducer 50% Error Period	R	1 Register	X	X	X	X			X
Variable Status List	R	1 Register			X	X			X

	Client Access	Length	Register Block				Context		
			Coils	Discrete Inputs	Input Registers	Holding Registers	Monitor	Channel	Variable
Variable Alert and Danger Over, Proportional	R/W	1 Register			X	X			X
Variable Alert and Danger Under, Proportional	R/W	1 Register			X	X			X
Variable Alert and Danger Over, Floating Point	R/W	2 Registers			X	X			X
Variable Alert and Danger Under, Floating Point	R/W	2 Registers			X	X			X
Relay Asserted	R	1 Register	X	X	X	X			X

6.3 Data/Register Associations

Date and Time

These registers contain the time used by the monitor to time stamp variable data and system events. The data will be based on the last time stamp sent from a Modbus client.

Access: Read Only

Length: 7 registers

Address Range: 300001 - 365530 (Input Registers)

400001 - 465530 (Holding Registers)

Context: Monitor

Format: Register 1: Year (00-99, 00 implies the year 2000)

Register 2: Month (1-12, 1 = January)

Register 3: Day (1-31)

Register 4: Hour (0-23, 24-hour clock; 0 = Midnight)

Register 5: Minute (0-59)

Register 6: Second (0-59)

Register 7: 1/1000 Second (0-999)

Exceptions: If the Modbus client hasn't sent a time stamp since it was last powered up, zeros will be returned in each of the 7 registers.

The time registered must be refreshed once a day to prevent significant time drift.

Write Date and Time

These registers are used to set the clock used by the monitor to time stamp Modbus variable data and system events. These registers are set by a Mod- bus client. They should not be used to read the actual time. When read, the registers will always return the last value written to them. When the 1/1000 Second register is written to, the Modbus time is updated with the contents of all 7 registers. The monitor will time stamp future system events and variable collections based on this new time. If this is the first instance that the time was set since the monitor was last powered up, the monitor will time stamp all previously generated system events based on this new time minus the amount of time that has elapsed since the event occurred. The same applies to time-stamped variable data.

If a client has not sent time and date data to the monitor since the monitor was last powered up, the monitor will return zeros in each of the seven time stamp registers when reading the Event List or the Last Read Time Stamp. System events still contain a sequence number indicating the order in which the events were collected.

Access: Read or write

Length: 7 registers

Address Range: 400001 - 465530 (Holding Registers)

Context: Monitor

Format: Register 1: Year (00-99, 00 implies the year 2000)

Register 2: Month (1-12, 1 = January)

Register 3: Day (1-31)

Register 4: Hour (0-23, 24-hour clock; 0 = Midnight)

Register 5: Minute (0-59)

Register 6: Second (0-59)

Register 7: 1/1000 Second (0-999)

Exceptions: If a Modbus client attempts to write an unacceptable value to one of the registers such as a 13 for the month, the monitor will return a SLAVE DEVICE FAILURE error.

If a Modbus client attempts to write the 1/1000 Second register and any of the other registers contain an unacceptable value such as 0 for the month or an invalid day for the month and year selected, the monitor will return a SLAVE DEVICE FAILURE error.

Last Read Time Stamp

These registers contain a time stamp value indicating when the last read vari- able amplitude (Direct, Vibration, Enveloped, Gap, Position, Bias Voltage, Speed or Temperature) was updated. When a client reads a variable amplitude through a Modbus register, the monitor puts the time stamp for that variable into the Last Read Time Stamp registers. To find the time that a variable value was obtained, first issue a read for a single variable amplitude. If a single read command requests multiple variables, the monitor will return time stamp data for the last requested register.

Reading variables that are not enabled will still update the Last Read Time Stamp registers with the current time.

Access: Read Only

Length: 7 registers

Address Range: 300001 - 365530 (Input Registers)

400001 - 465530 (Holding Registers)

Context: Monitor

Format: Register 1: Year (00-99, 00 implies the year 2000)

Register 2: Month (1-12, 1 = January)

Register 3: Day (1-31)

Register 4: Hour (0-23, 24-hour clock; 0 = Midnight)

Register 5: Minute (0-59)

Register 6: Second (0-59)

Register 7: 1/1000 Second (0-999)

Exceptions: If no variable amplitude registers have been read since the monitor was last powered up, the monitor will return zeros for each of the 7 registers.

If a Modbus client has not sent time to the monitor since the monitor was last powered up, the monitor will return zeros in each of the 7 registers.

Event List

The System Event List contains the latest 200 System Events. The monitor stores the list in non-volatile memory, so that if the monitor loses power, it will still retain the event list. The 1900/65A identifies each system event with a unique 32-bit sequence number. The Last Posted System Event register contains the sequence number of the latest system event. To request a system event, first write the desired sequence number to the Requested System Event registers using the Write Single Register or the Write Multiple Registers command. The specified event can then be read using the Read Holding Registers command. If you read the holding registers before the Requested System Event registers are updated, the monitor will return the last requested event.

Access: Registers 1 and 2 are read/write registers. All other registers are read only.

Length: 18 registers

Address Range: 400001 - 465519 (Holding Registers)

Context: Monitor

Format:

Register 1: Requested System Event Sequence Number (high word)

Register 2: Requested System Event Sequence Number (low word)

Register 3: Last Posted System Event Sequence Number (high word)

Register 4: Last Posted System Event Sequence Number (low word)

Register 5: Sequence Number (high word)

Register 6: Sequence Number (low word)

Register 7: Level (0 = Monitor, 1 = Channel, 2 = Variable, 3 = Relay)

Register 8: Channel Number (0 based) or Relay (0 based)

Register 9: Register 1: Variable (0 based)

Register 10: Register 1: Direction (0 = Non-Applicable, 1 = Entered, 2 = Exited)

Register 11: Event Code

1 1900 Has Rebooted

2 Not Ok

3 Alert

4 Danger

5 Relay Active

6 Config Changed

7 Config Change Failed

8 Reset Latching

9 Monitor Inhibit

10 Bypass

11 Danger Bypass

12 Flash Memory Failure

13 Config Lost

14 Inhibit/Bypass Lost

15 Events Lost

16 Self Test Passed

17 Self Test Failed

18 A/D Failure

19 CJC Failure

20 HCS08 Failure

21 +12 V Failure

22 -24 V Failure

23 +1.4 V Failure

24 +2.5 V Failure

25 Display Fault

26 +3.3 V Failure

27 +5 V Failure

28 Modbus Enabled

29 Port Failure

30 Trip Multiply

31 Transducer Signal Too Slow

32 Transducer 50% Error

33 Transducer Signal is Valid

34 Serial Port Reset

Register 12: Year (00-99, 00 implies the year 2000)

Register 13: Month (1-12, 1 = January)

Register 14: Day (1-31)

Register 15: Hour (0-23, 24-hour clock; 0 = Midnight)

Register 16: Minute (0-59)

Register 17: Second (0-59)

Register 18: 1/1000 Second (0-999)

Exceptions: If the requested event is more than 199 events older than the sequence number of the last posted system event, or the event requested has not been generated yet, registers 5-18 will contain zeros.

It is possible for the monitor to store an event without a time stamp. This results when an event occurs, but a Modbus client has not sent time to the monitor since the monitor was last powered up. In this case registers 12-18 will contain zeros.

If the event sequence number register has never been written to, registers 1-2 and 5-18 will contain zeros.

Full Scale Data Range

This register contains a copy of the Full Scale Data Range configuration setting. All variable amplitudes are scaled to this value. Reading this register will return a 16-bit integer value of the full-scale data range configured in the Modbus Gateway properties.

Access: Read Only

Length: 1 register

Address Range: 300001 - 365536 (Input Registers)

400001 - 465536 (Holding Registers)

Context: Monitor

Format: Register 1: Full-scale data range

Exceptions: None.

Reset Latched Statuses

Writing any non-zero value to this register will cause the monitor to reset any latched statuses or relays.

Access: Write Only

Length: 1 register

Address Range: 000001 - 065536 (Coils)

400001 - 465536 (Holding Registers)

Context: Monitor

Format: Register 1: Reset Latched Statuses

Exceptions: None.

Heart Beat

This register contains an integer number that will increment by one each time the register is read. This register can be configured as a heart beat signal in the DCS. The register will contain 0 when the monitor boots up and will roll over from 65535 to 0.

Access: Read Only

Length: 1 register

Address Range: 300001 - 365536 (Input Registers)

400001 - 465536 (Holding Registers)

Context: Monitor

Format: Register 1: Heart Beat

Exceptions: This register does not indicate the health of the monitor. It still increments when read even if errors (A/D errors, node voltage errors, channel Not OK conditions, etc.) exist, or the monitor is not updating variable ampli- tudes. The 1900/65A provides other registers that communicate these error conditions.

Config Download Counter

This register contains a value that will increment each time the 1900 Configuration Software downloads a configuration to the monitor. This register contains zero when shipped from the factory.

Access: Read Only

Length: 1 register

Address Range: 300001 - 365536 (Input Registers)

400001 - 465536 (Holding Registers)

Context: Monitor

Format: Register 1: Config Download Counter

Exceptions: None.

Config Adjustment Counter

This register contains a value that will increment each time the client or Display Module makes a runtime configuration change or software switch change. This register contains zero when shipped from the factory.

Access: Read Only

Length: 1 register

Address Range: 300001 - 365536 (Input Registers)

400001 - 465536 (Holding Registers)

Context: Monitor

Format: Register 1: Config Adjust Counter

Exceptions: None.

Write Sequence Number

This register must be used if the user configures the monitor to require a write sequence when configuration parameters are changed via Modbus. Writing the value 0x55AA to this register immediately before writing a change to a configuration parameter will allow the user to issue a single **WRITE SINGLE REGISTER** command (function code 6), **WRITE MULTIPLE REGISTERS** command (function code 16), **WRITE SINGLE COIL** (function code 5), or **WRITE MULTIPLE COILS** (function code 15). 0x55AA must first be written to the *Write Sequence Number* register each time that a configuration parameter is changed.

The following parameters require writing 0x55AA to this field if the user configures the monitor to use the write sequence:

- Write Date and Time
- Reset Latched Statuses
- Alarm Inhibit Switch
- Monitor Status List
- Channel Bypass Switch
- Channel Danger Bypass Switch
- Channel Status List
- Variable Alert Over, Proportional
- Variable Alert Over, Floating Point
- Variable Alert Under, Proportional
- Variable Alert Under, Floating Point
- Variable Danger Over, Proportional
- Variable Danger Over, Floating Point
- Variable Danger Under, Proportional
- Variable Danger Under, Floating Point

Access: Write Only

Length: 1 register

Address Range: 400001 - 465536 (Holding Registers)

Context: Monitor

Format: Register 1: Write Sequence Field

Exceptions: Using a single WRITE MULTIPLE REGISTERS command to write to the Write Sequence Number register and a configuration parameter register will not change the configuration parameter and will return a **SLAVE DEVICE FAILURE** error. The registers must be written using two separate consecutive write commands. If 0x55AA is written to the Write Sequence Number field and then a configuration is downloaded, 0x55AA must be re-written prior to a Mod- bus initiated run-time parameter change. Writing to a configuration parameter prior to writing to the Write Sequence Number register will cause the monitor to return a **SLAVE DEVICE FAILURE**.

Monitor Not OK

The monitor sets the value of this register to 1 if any enabled channel is Not OK. See the Channel Not OK register description for reasons why a channel may be Not OK.

Access: Read Only

Length: 1 register

Address Range: 000001 - 065536 (Coils)

100001 - 165536 (Discrete Inputs)

300001 - 365536 (Input Registers)

400001 - 465536 (Holding Registers)

Context: Monitor

Format: Register 1: 1 = Monitor Not OK, otherwise 0

Exceptions: Although the *"Set channel Not OK register if channel unconfigured, disabled, or bypass switch set"* option alters the Channel Not Ok register behavior, it has no affect on the Monitor Not Ok register. Channels that are unconfigured, disabled, or which have the Channel Bypass Switch enabled will not cause the monitor to set the value of the Monitor Not Ok register to 1.

Monitor Alert and Danger Alarm Status

Alert and Danger fields refer to the ALERT and DANGER setpoints that the user can activate and configure using the 1900 Configuration Software.

Monitor Danger

The monitor sets the value of this register to 1 if any Danger alarm is present on any variables on any channels in the monitor.

Access: Read Only

Length: 1 register

Address Range: 000001 - 065536 (Coils)

100001 - 165536 (Discrete Inputs)

300001 - 365536 (Input Registers)

400001 - 465536 (Holding Registers)

Context: Monitor

Format: Register 1: 1 = A Danger alarm is active, otherwise 0

Exceptions: None.

Monitor Alert

The monitor sets the value of this register to 1 if any Alert alarm is present on any variables on any channels in the monitor.

Access: Read Only

Length: 1 register

Address Range: 000001 - 065536 (Coils)

100001 - 165536 (Discrete Inputs)

300001 - 365536 (Input Registers)

400001 - 465536 (Holding Registers)

Context: Monitor

Format: Register 1: 1 = An Alert alarm is active, otherwise 0

Exceptions: None.

Monitor Bypass

The monitor sets the value of this register to 1 if either of the following conditions is true:

- Monitor Alarm Inhibit is active (see Alarm Inhibit register)
- Any enabled channel is in bypass (see Channel Bypass register)

Access: Read Only

Length: 1 register

Address Range: 000001 - 065536 (Coils)

100001 - 165536 (Discrete Inputs)

300001 - 365536 (Input Registers)

400001 - 465536 (Holding Registers)

Context: Monitor

Format: Register 1: 1 = Monitor is bypassed, otherwise 0

Exceptions: Channels that are unconfigured or disabled will not cause the monitor to set the value of this register to 1.

Alarm Inhibit

The monitor sets the value of this register to 1 if either of the following conditions is true:

- Monitor Alarm Inhibit terminals are shorted
- The Monitor Alarm Inhibit Switch is active

Access: Read Only

Length: 1 register

Address Range: 000001 - 065536 (Coils)

100001 - 165536 (Discrete Inputs)

300001 - 365536 (Input Registers)

400001 - 465536 (Holding Registers)

Context: Monitor

Format: Register 1: 1 = Monitor Alarm Inhibit is active, otherwise 0

Exceptions: None

Alarm Inhibit Switch

This register contains the current setting of the Monitor Alarm Inhibit Switch. A value of 0 indicates that the switch is off, while a 1 indicates that the switch is on. Reading this register will return the current switch setting. Writing 0 to this register will disable the switch, while writing any non-zero value will enable it. The user can also change the value of this switch through the 1900 Configuration Software or from the Display Module.

Access: Read or write

Length: 1 register

Address Range: 000001 - 065536 (Coils)

100001 - 165536 (Discrete Inputs)

300001 - 365536 (Input Registers)

400001 - 465536 (Holding Registers)

Context: Monitor

Format: Register 1: 1 = Monitor Alarm Inhibit Switch is active, otherwise 0

Exceptions: None

Trip Multiply

The monitor sets the value of this register to 1 if either of the following conditions is true:

- Monitor Trip Multiply terminals are shorted
- The Monitor Trip Multiply Switch is active

Access: Read Only

Length: 1 register

Address Range: 000001 - 065536 (Coils)

100001 - 165536 (Discrete Inputs)

300001 - 365536 (Input Registers)

400001 - 465536 (Holding Registers)

Context: Monitor

Format: Register 1: 1 = Monitor Trip Multiply is active, otherwise 0

Exceptions: None

Trip Multiply Switch

This register contains the current setting of the Trip Multiply Switch. A value of

0 indicates that the switch is off, while a 1 indicates that the switch is on. Reading this register will return the current switch setting. Writing 0 to this register will disable the switch, while writing any non-zero value will enable it. The user can also change the value of this switch through the 1900 Configuration Software or from the Display Module.

Access: Read or write

Length: 1 register

Address Range: 000001 - 065536 (Coils)

100001 - 165536 (Discrete Inputs)

300001 - 365536 (Input Registers)

400001 - 465536 (Holding Registers)

Context: Monitor

Format: Register 1: 1 = Monitor Trip Multiply Swtich is active, otherwise 0

Exceptions: None

Monitor Status List

The monitor sets the bits within this register according to the operation of the individual fields themselves. Writing to this register will not affect any bits except the Alarm Inhibit Switch bit and the Trip Multiply Switch bit.

Access: Read or write

Length: 1 register

Address Range: 300001 - 365536 (Input Registers)

400001 - 465536 (Holding Registers)

Context: Monitor

Format: Register 1:

Bit 0 (LSB): Monitor Not Ok

Bit 1: Monitor Danger

Bit 2: Monitor Alert

Bit 3: Monitor Bypass

Bit 4: Alarm Inhibit

Bit 5: Alarm Inhibit Switch

Bit 6: Trip Multiply

Bit 7: Trip Multiply Switch

Bit 8 -15: Unused (set to 0)

Exceptions: None

Channel Not OK

The monitor sets the value of this register to 1 if any of the following conditions are true:

- This channel's transducer is outside the OK limits
- There is a latched Not OK status
- The Timed OK Channel Defeat timer has not expired
- The *"Set channel Not OK register if channel unconfigured, disabled, or bypass switch set"* option is enabled and one of those conditions applies
- There is a node voltage failure that affects this channel
- There is an A/D failure on this channel
- There is an HCS08 failure
- There is a compensation circuit failure, and this is a thermocouple temperature channel

Access: Read only

Length: 1 register

Address Range: 000001 - 065536 (Coils)

100001 - 165536 (Discrete Inputs)

300001 - 365536 (Input Registers)

400001 - 465536 (Holding Registers)

Context: Channels 1 through 8

Format: 1 = The channel is Not OK, otherwise 0

Exceptions: The monitor sets the value of this register to 0 if the *"Set channel Not OK register if channel unconfigured, disabled, or bypass switch set"* option is disabled, and one of those conditions applies.

Channel Danger

The monitor sets the value of this register to 1 if the Danger alarm is present on any variables configured on the channel.

Access: Read only

Length: 1 register

Address Range: 000001 - 065536 (Coils)

100001 - 165536 (Discrete Inputs)

300001 - 365536 (Input Registers)

400001 - 465536 (Holding Registers)

Context: Channels 1 through 8

Format: 1 = Channel Danger alarm active, otherwise 0

Exceptions: If the channel is unconfigured or disabled, the monitor sets the value of this register to 0.

Channel Alert

The monitor sets the value of this register to 1 if the Alert alarm is present on any variables configured on the channel.

Access: Read only

Length: 1 register

Address Range: 000001 - 065536 (Coils)

100001 - 165536 (Discrete Inputs)

300001 - 365536 (Input Registers)

400001 - 465536 (Holding Registers)

Context: Channels 1 through 8

Format: 1 = Channel alert alarm active, otherwise 0

Exceptions: If the channel is unconfigured or disabled, the monitor sets the value of this register to 0.

Channel Bypass

The monitor sets the value of this register to 1 if any of the following conditions are true:

- Monitor Alarm Inhibit is active (see Alarm Inhibit register)
- The Channel Bypass Switch is active
- The Channel Danger Bypass Switch is active
- An enabled variable belonging to the channel is invalid and has enabled setpoints with non-latched alarms (see Variable Invalid register)
- The Timed OK Channel Defeat timer has not expired and a variable belonging to the channel has enabled setpoints with non-latched alarms

Access: Read only

Length: 1 register

Address Range: 000001 - 065536 (Coils)

100001 - 165536 (Discrete Inputs)

300001 - 365536 (Input Registers)

400001 - 465536 (Holding Registers)

Context: Channels 1 through 8

Format: 1 = Channel Bypassed, otherwise 0

Exceptions: If the channel is unconfigured or disabled, the monitor sets the value of this register to 0.

Channel Bypass Switch

This register contains the current setting of the Channel Bypass Switch. A value of 0 indicates that the switch is off, while a 1 indicates that the switch is on. Reading this register will return the current switch setting. Writing 0 to this register will disable the switch, while writing any non-zero value will enable it. The user can also change the value of this switch through the 1900 Configuration Software or from the Display Module.

Access: Read or write

Length: 1 register

Address Range: 000001 - 065536 (Coils)

100001 - 165536 (Discrete Inputs)

300001 - 365536 (Input Registers)

400001 - 465536 (Holding Registers)

Context: Channels 1 through 8

Format: 1 = Channel Bypass Switch is active, otherwise 0

Exceptions: If the channel is unconfigured or disabled, the monitor sets the value of this register to 0, and writing to this register will return a **SLAVE DEVICE FAILURE** error.

Channel Danger Bypass Switch

This register contains the current setting of the Channel Danger Bypass Switch. A value of 0 indicates that the switch is off, while a 1 indicates that the switch is on. Reading this register will return the current switch setting. Writing 0 to this register will disable the switch, while writing any non-zero value will enable it. The user can also change the value of this switch through the 1900 Configuration Software or from the Display Module.

Access: Read or write

Length: 1 register

Address Range: 000001 - 065536 (Coils)

100001 - 165536 (Discrete Inputs)

300001 - 365536 (Input Registers)

400001 - 465536 (Holding Registers)

Context: Channels 1 through 8

Format: 1 = Channel Danger Bypass Switch is active, otherwise 0

Exceptions: If the channel is unconfigured or disabled, the monitor sets the value of this register to 0, and writing to this register will return a **SLAVE DEVICE FAILURE** error.

Channel Disabled

The monitor sets the value of this register to 1 if the channel is unconfigured or disabled.

Access: Read only

Length: 1 register

Address Range: 000001 - 065536 (Coils)

100001 - 165536 (Discrete Inputs)

300001 - 365536 (Input Registers)

400001 - 465536 (Holding Registers)

Context: Channels 1 through 8

Format: 1 = Channel disabled, otherwise 0

Exceptions: None.

Channel Status List

The monitor sets the bits within this register according to the operation of the individual fields themselves. Writing to this register will not affect any bits except the Channel Bypass Switch bit and the Channel Danger Bypass Switch bit.

Access: Read or write

Length: 1 register

Address Range: 300001 - 365536 (Input Registers)

400001 - 465536 (Holding Registers)

Context: Channels 1 through 8

Format: Register 1:

Bit 0 (LSB): Channel Not Ok

Bit 1: Channel Danger

Bit 2: Channel Alert

Bit 3: Channel Bypass

Bit 4: Channel Bypass Switch

Bit 5: Channel Danger Bypass Switch

Bit 6: Channel Disabled

Bit 7 – 15: Unused (set to 0)

Exceptions: None.

Variable Invalid

The monitor sets the value of this register to 1 if any of the following conditions are true:

- The channel the variable belongs to is Not OK, the “Timed OK Channel Defeat” option is enabled, and this variable is not gap or bias voltage
- The Channel Bypass Switch is active on this variable’s channel
- The “Set variable invalid register if variable unconfigured or disabled” option is enabled, and one of those conditions applies
- There is a node voltage error that affects this variable’s channel
- There is an A/D error that affects this variable’s channel
- There is an HCS08 failure
- There is a compensation circuit failure, and this is a thermocouple temperature variable

Access: Read only

Length: 1 register

Address Range: 000001 - 065536 (Coils)

100001 - 165536 (Discrete Inputs)

300001 - 365536 (Input Registers)

400001 - 465536 (Holding Registers)

Context: Variable

Format: 1 = Variable invalid, otherwise 0

Exceptions: If the *“Set variable invalid register if variable unconfigured or disabled”* option is disabled, and one of those conditions applies, the monitor sets the value of this register to 0.

Variable Danger

The monitor sets the value of this register to 1 if the Danger alarm is present on the variable.

Access: Read only

Length: 1 register

Address Range: 000001 - 065536 (Coils)

100001 - 165536 (Discrete Inputs)

300001 - 365536 (Input Registers)

400001 - 465536 (Holding Registers)

Context: Variable

Format: 1 = Variable in Danger, otherwise 0

Exceptions: If the variable is unconfigured or disabled, the monitor will set the value of this register to 0.

Variable Alert

The monitor sets the value of this register to 1 if the Alert alarm is present on the variable.

Access: Read only

Length: 1 register

Address Range: 000001 - 065536 (Coils)

100001 - 165536 (Discrete Inputs)

300001 - 365536 (Input Registers)

400001 - 465536 (Holding Registers)

Context: Variable

Format: 1 = Variable in Alert, otherwise 0

Exceptions: If the variable is unconfigured or disabled, the monitor will set the value of this register to 0.

Variable Disabled

The monitor sets the value of this register to 1 if the variable is unconfigured or disabled. This register is not affected by the channel's Bypass status.

Access: Read only

Length: 1 register

Address Range: 000001 - 065536 (Coils)

100001 - 165536 (Discrete Inputs)

300001 - 365536 (Input Registers)

400001 - 465536 (Holding Registers)

Context: Variable

Format: 1 = Variable disabled, otherwise 0

Exceptions: None.

Variable Amplitude, Proportional

The value is the 16-bit integer amplitude value for the variable read.

The Full Scale Data Range variable referred to in the equations below is configured in the Modbus Gateway Properties window using the 1900 Configuration Software.

The monitor scales the value according to the top and bottom display scales which are set in the 1900 Configuration Software. The returned counts will be calculated as follows:

Returned counts = (reading - bottom scale) / (top scale – bottom scale) * Full Scale Data Range

If the reading is smaller than the bottom scale, the returned counts will be equal to 0.

If the reading is larger than the top scale, the returned counts will be equal to the Full Scale Data Range.

Access: Read only

Length: 1 register

Address Range: 300001 - 365536 (Input Registers)

400001 - 465536 (Holding Registers)

Context: Variable

Format: Integer variable amplitude in counts

Exceptions: If the variable is unconfigured or disabled, the monitor sets the value of this register to 0.

If the variable does exist but is invalid (see Variable Invalid register), then the monitor will calculate the returned counts as follows:

Returned counts = (clamp value – bottom scale) / (top scale – bottom scale) * Full Scale Data Range

Variable Amplitude, Floating Point

The value returned is the IEEE single precision floating point value that represents the variable amplitude.

Access: Read only

Length: 2 registers

Address Range: 300001 - 365535 (Input Registers)

400001 - 465535 (Holding Registers)

Context: Variable

Format:

Register 1: Most significant word of the IEEE single precision floating point value if the *"Most Significant Word First"* option is enabled.

Register 2: Least significant word of the IEEE single precision floating point value if the *"Most Significant Word First"* option is enabled.

Exceptions: If the variable is unconfigured or disabled, the monitor sets the value of these registers to 0.

If the variable does exist but is invalid, the monitor will return the Clamp Value that was configured for the variable in the 1900 Configuration Software.

Variable Top Scale, Floating Point

The value returned is the IEEE single precision floating point value that represents the top scale, which is set in the 1900 Configuration Software.

Access: Read only

Length: 2 registers

Address Range: 300001 - 365535 (Input Registers)

400001 - 465535 (Holding Registers)

Context: Variable

Format:

Register 1: Most significant word of the IEEE single precision floating point value if the *"Most Significant Word First"* option is enabled.

Register 2: Least significant word of the IEEE single precision floating point value if the *"Most Significant Word First"* option is enabled.

Exceptions: If the variable is unconfigured or disabled, the monitor sets the value of these registers to 0.

Variable Bottom Scale, Floating Point

The value returned is the IEEE single precision floating point value that represents the bottom scale, which is set in the 1900 Configuration Software.

The integer bottom scale value is 0.

Access: Read only

Length: 2 registers

Address Range: 300001 - 365535 (Input Registers)

400001 - 465535 (Holding Registers)

Context: Variable

Format:

Register 1: Most significant word of the IEEE single precision floating point value if the *"Most Significant Word First"* option is enabled.

Register 2: Least significant word of the IEEE single precision floating point value if the *"Most Significant Word First"* option is enabled.

Exceptions: If the variable is unconfigured or disabled, the monitor sets the value of these registers to 0.

Variable Underspeed

The monitor sets the value of this register to 1 if the variable type is Speed and is less than 1 RPM or if the speed input signal is less than 1 event per minute.

Access: Read Only

Length: 1 register

Address Range: 000001 - 065536 (Coils)

100001 - 165536 (Discrete Inputs)

300001 - 365536 (Input Registers)

400001 - 465536 (Holding Registers)

Context: Variable (specific to the Speed variable type)

Exceptions: If the variable is unconfigured or disabled or the variable type is not Speed, the monitor sets the value of this register to 0.

Variable Transducer 50% Error Period

The monitor sets the value of this register to 1 if the variable type is Speed, and it is invalid due to a speed input signal that had a period change of more than 50% between consecutive periods. This detection is only performed if the speed is greater than 100 RPMs.

Access: Read Only

Length: 1 register

Address Range: 000001 - 065536 (Coils)

100001 - 165536 (Discrete Inputs)

300001 - 365536 (Input Registers)

400001 - 465536 (Holding Registers)

Context: Variable (specific to the Speed variable type)

Exceptions: If the variable is unconfigured or disabled or the variable type is not Speed, the monitor sets the value of this register to 0.

Variable Status List

The monitor sets the bits within this register according to the operation of the individual fields themselves.

Access: Read only

Length: 1 register

Address Range: 300001 - 365536 (Input Registers)

400001 - 465536 (Holding Registers)

Context: Variable

Format: Register 1:

Bit 0 (LSB): Variable Invalid

Bit 1: Variable Danger

Bit 2: Variable Alert

Bit 3: Variable Disabled

Bit 4: Variable Underspeed

Bit 5: Variable 50% Error

Bit 6 – 15: Unused (set to 0)

Exceptions: None.

Variable Alert and Danger Over, Proportional

This register contains an over setpoint amplitude scaled to an integer using the configured *Full Scale Data Range*. Writing to this register will change the setpoint amplitude, but will not disrupt monitoring.

The *Full Scale Data Range* variable referred to in the equations below is configured in the Modbus Gateway Properties window using the 1900 Configuration Software.

The monitor scales the value according to the top and bottom display scales which are set in 1900 Configuration Software. Scaling is according to the equation:

Returned counts = (setpoint amplitude - bottom scale) / (top scale – bottom scale) * Full Scale Data Range

For Example:

Given: Setpoint Amplitude = 1.2 in/s, Top Scale = 2 in/s, Bottom Scale = 0, Full Scale Data Range = 65535

Returned Counts = (1.2 - 0) / (2 - 0) * 65535 = 39321

Access: Read or write

Length: 1 register

Address Range: 300001 - 365536 (Input Registers)

400001 - 465536 (Holding Registers)

Context: Variable

Format: Integer setpoint amplitude

Exceptions: A value read from this register may differ from the value written to it due to the resolution of the register. The monitor rounds setpoints to the nearest digit of precision based on the variable's full-scale range. The resolution is defined in Table 6 - 2: Resolution definition table.:

Table 6 - 2: Resolution definition table.

Full Scale Range	Digits After the Decimal Place
0 to 0.999	3
1 to 99.99	2
100 to 999.9	1
1000 or greater	0

Calculate the full-scale range by taking the absolute value of (Top Scale - Bottom Scale). For example, if top scale is 20 g pk and bottom scale is 0 g pk, the full scale range would be 20 which means that the monitor will round all setpoints to the nearest one hundredth. So if an integer value that represents 13.736 g pk is written to the setpoint register, the setpoint register will return an integer value that represents 13.74 g pk.

If the setpoint is unconfigured or disabled, this register will contain 0.

The following conditions will result in the monitor returning a **SLAVE DEVICE FAILURE**:

- Writing a value to this register if the setpoint is unconfigured or disabled.
- Writing a value to this register that is larger than *Full Scale Data Range*.
- Writing a value to this register that exceeds the linear range configured for the transducer.
- Writing a value to this register that violates the rule that the Over setpoint be greater than the Under setpoint.

Variable Alert and Danger Under, Proportional

This register contains an under setpoint amplitude scaled to an integer using the configured Full Scale Data Range. Writing to this register will change the setpoint amplitude, but will not disrupt monitoring.

The Full Scale Data Range variable referred to in the equations below is configured in the Modbus Gateway Properties window using the 1900 Configuration Software.

The monitor scales the value according to the top and bottom display scales which are set in 1900 Configuration Software. Scaling is according to the equation:

Returned counts = (setpoint amplitude - bottom scale) / (top scale – bottom scale) * Full Scale Data Range

For Example:

Given: Setpoint Amplitude = 1.2 in/s, Top Scale = 2 in/s, Bottom Scale = 0, Full Scale Data Range = 65535

Returned Counts = (1.2 - 0) / (2 - 0) * 65535 = 39321

Access: Read or write

Length: 1 register

Address Range: 300001 - 365536 (Input Registers)

400001 - 465536 (Holding Registers)

Context: Variable

Format: Integer setpoint amplitude

Exceptions: A value read from this register may differ from the value written to it due to the resolution of the register. The monitor rounds setpoints to the nearest digit of precision based on the variable's full-scale range. The resolution is defined in Table 6 - 3: Resolution definition table.:

Table 6 - 3: Resolution definition table.

Full Scale Range	Digits After the Decimal Place
0 to 0.999	3
1 to 99.99	2
100 to 999.9	1
1000 or greater	0

Calculate the full-scale range by taking the absolute value of (Top Scale - Bottom Scale). For example, if top scale is 20 g pk and bottom scale is 0 g pk, the full scale range would be 20 which means that the monitor will round all setpoints to the nearest one hundredth. So if an integer value that represents 13.736 g pk is written to the setpoint register, the setpoint register will return an integer value that represents 13.74 g pk.

If the setpoint is unconfigured or disabled, this register will contain 0.

The following conditions will result in the monitor returning a **SLAVE DEVICE FAILURE**:

- Writing a value to this register if the setpoint is unconfigured or disabled.
- Writing a value to this register that is larger than *Full Scale Data Range*.
- Writing a value to this register that exceeds the linear range configured for the transducer.
- Writing a value to this register that violates the rule that the Over setpoint be greater than the Under setpoint.

Variable Alert and Danger Over, Floating Point

These registers contain the Over setpoint amplitude. When read, they will always return the current value that the monitor is using. Writing to register 1 will not change the setpoint at that time. The monitor does not update the setpoint until register 2 is written to. Prior to writing to register 2, reads will return the old setpoint value used for alarming.

This operation will not disrupt monitoring.

Access: Read or write

Length: 2 registers

Address Range: 300001 - 365535 (Input Registers)

400001 - 465535 (Holding Registers)

Context: Variable

Format:

Register 1: Most significant word of the IEEE single precision floating point value if the *"Most Significant Word First"* option is enabled.

Register 2: Least significant word of the IEEE single precision floating point value if the *"Most Significant Word First"* option is enabled.

Exceptions: The following conditions will result in the monitor returning a SLAVE DEVICE FAILURE:

- Writing a value to this register if the setpoint is unconfigured or disabled.
- Writing a value to this register that is greater than the configured top scale or less than the configured bottom scale.
- Writing a value to this register that exceeds the linear range configured for the transducer.
- Writing a value to this register that violates the rule that the Over setpoint be greater than the Under setpoint.
- Writing an invalid IEEE single precision floating point value to these registers.

A value read from this register may differ from the value written to it due to the resolution of the register. The monitor rounds setpoints to the nearest digit of precision based on the variable's full-scale range. The resolution is defined in Table 6 - 4: Resolution definition table.:

Table 6 - 4: Resolution definition table.

Full Scale Range	Digits After the Decimal Place
0 to 0.999	3
1 to 99.99	2
100 to 999.9	1
1000 or greater	0

Calculate the full-scale range by taking the absolute value of (Top Scale - Bottom Scale). For example, if top scale is 20 g pk and bottom scale is 0 g pk, the full scale range would be 20 which means that the monitor will round all setpoints to the nearest one hundredth. So if a value of 13.736 g pk is written to the setpoint register, the setpoint register will return a value of 13.74 g pk.

Variable Alert and Danger Under, Floating Point

These registers contain the Under setpoint amplitude. When read, they will always return the current value that the monitor is using. Writing to register 1 will not change the setpoint at that time. The monitor does not update the setpoint until register 2 is written. Prior to writing to register 2, reads will return the old setpoint value used for alarming.

This operation will not disrupt monitoring.

Access: Read or write

Length: 2 registers

Address Range: 300001 - 365535 (Input Registers)

400001 - 465535 (Holding Registers)

Context: Variable

Format:

Register 1: Most significant word of the IEEE single precision floating point value if the *"Most Significant Word First"* option is enabled.

Register 2: Least significant word of the IEEE single precision floating point value if the *"Most Significant Word First"* option is enabled.

Exceptions: The following conditions will result in the monitor returning a **SLAVE DEVICE FAILURE**:

- Writing a value to this register if the setpoint is unconfigured or disabled.
- Writing a value to this register that is greater than the configured top scale or less than the configured bottom scale.
- Writing a value to this register that exceeds the linear range configured for the transducer.
- Writing a value to this register that violates the rule that the Over setpoint be greater than the Under setpoint.
- Writing an invalid IEEE single precision floating point value to these registers.

A value read from this register may differ from the value written to it due to the resolution of the register. The monitor rounds setpoints to the nearest digit of precision based on the variable's full-scale range. The resolution is defined in Resolution definition table.-:

Table 6 - 5: Resolution definition table.

Full Scale Range	Digits After the Decimal Place
0 to 0.999	3
1 to 99.99	2
100 to 999.9	1
1000 or greater	0

Calculate the full-scale range by taking the absolute value of (Top Scale - Bottom Scale). For example, if top scale is 20 g pk and bottom scale is 0 g pk, the full scale range would be 20 which means that the monitor will round all setpoints to the nearest one hundredth. So if a value of 13.736 g pk is written to the setpoint register, the setpoint register will return a value of 13.74 g pk.

Relay Asserted

The monitor sets the value of this register to 1 if the relay logic configured to drive the relay is true.

Access: Read only

Length: 1 register

Address Range: 000001 - 065536 (Coils)

100001 - 165536 (Discrete Inputs)

300001 - 365536 (Input Registers)

400001 - 465536 (Holding Registers)

Format: Register 1: 1 = Relay assserted, otherwise 0

Exceptions: If the relay is unconfigured or disabled, the monitor sets the value of this register to 0.