Est. USA 1981

# Modbus RTU & TCP on the securityProbe

## Table of contents



## Section #1 - Introduction to Modbus

MODBUS RTU is a non-proprietary serial communications protocol that is widely used in the process control industry actuation. The hardware over which MODBUS RTU communications are performed is not defined by the protocol.

MODBUS RTU is supported on RS232, RS422, RS485, Ethernet and other electrical standards. The securityProbe supports, RS485 and Ethernet. The Modbus/RTU protocol defines how a “master” device polls one or more “slave” devices to read and write data in real time over the RS485 serial data communication connection.

Modbus/TCP, an extension of Modbus/RTU, defines how Modbus/RTU and Modbus/ASCII messages are encoded within and transported over TCP/IP-based networks. Modbus/TCP is just as simple to implement and flexible to apply as the original Modbus/RTU. The securityProbe can represent both “master” and “slave” devices and supports both Modbus RTU and Modbus TCP protocols.

### Transactions on Modbus Networks

Controllers communicate using a master–slave technique, in which only one device (the master) can initiate transactions (called ‘queries’). The other devices (the slaves) respond by supplying the requested data to the master, or by taking the action requested in the query. Typical master devices include host processors and programming panels. Typical slaves include programmable controllers.

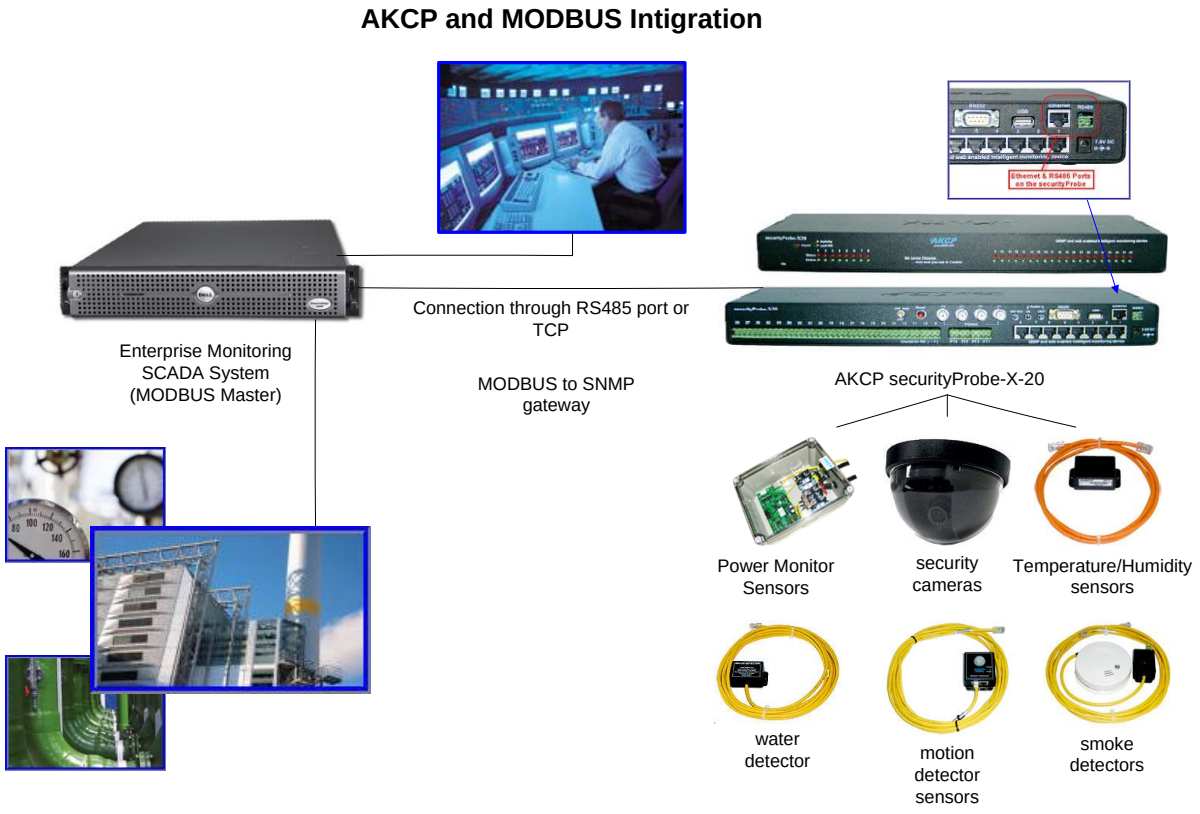
The master can address individual slaves, or can initiate a broadcast message to all slaves. Slaves return a message (called a ‘response’) to queries that are addressed to them individually. Responses are not returned to broadcast queries from the master.

The Modbus protocol establishes the format for the master’s query by placing into it the device (or broadcast) address, a function code defining the requested action, any data to be sent, and an error–checking field.

The slave’s response message is also constructed using Modbus protocol. It contains fields confirming the action taken, any data to be returned, and an error–checking field. If an error occurred in receipt of the message, or if the slave is unable to perform the requested action, the slave will construct an error message and send it as its response.

Important Note:

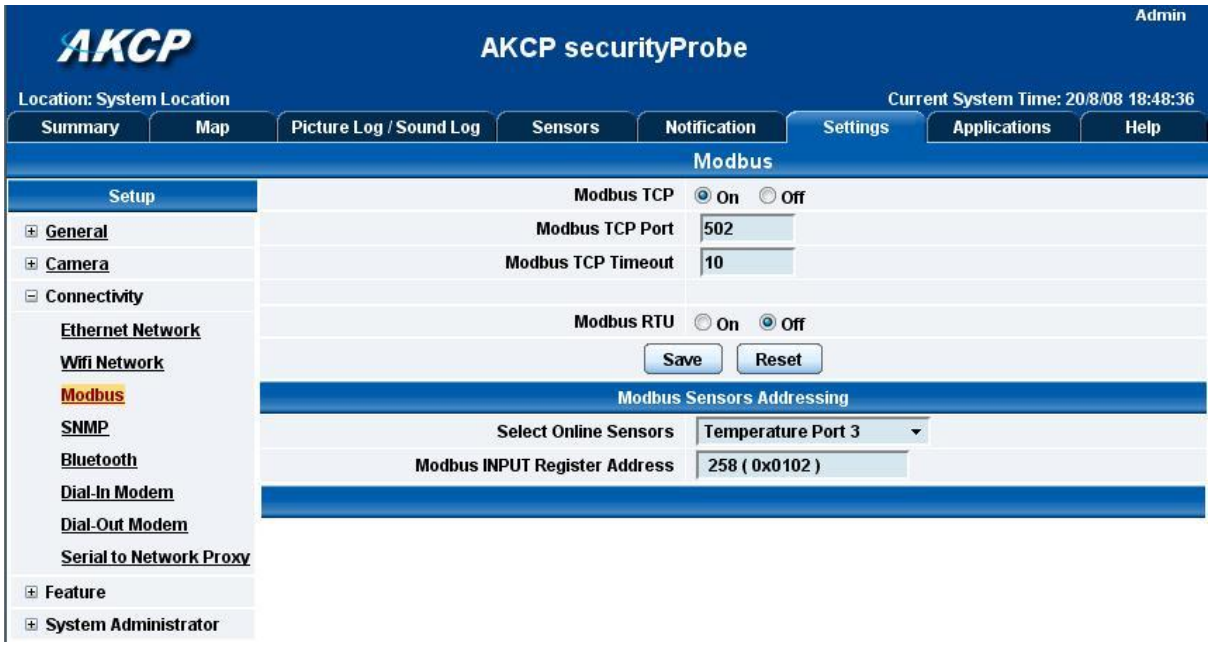
## Section #2 - AKCP and Modbus Integration diagram




## Section # 3 - Setting up MODBUS on the securityProbe


1. From the “Settings” tab in the securityProbes web interface, expand “Connectivity, then click on “Modbus” in the left hand column. Set the Modbus TCP and Modbus RTU to off for the Master device (this is the Reading unit). See screen shot above.

2. Now set the Modbus TCP or Modbus RTU (this depends on your connection type) setting to “on” for your Client device



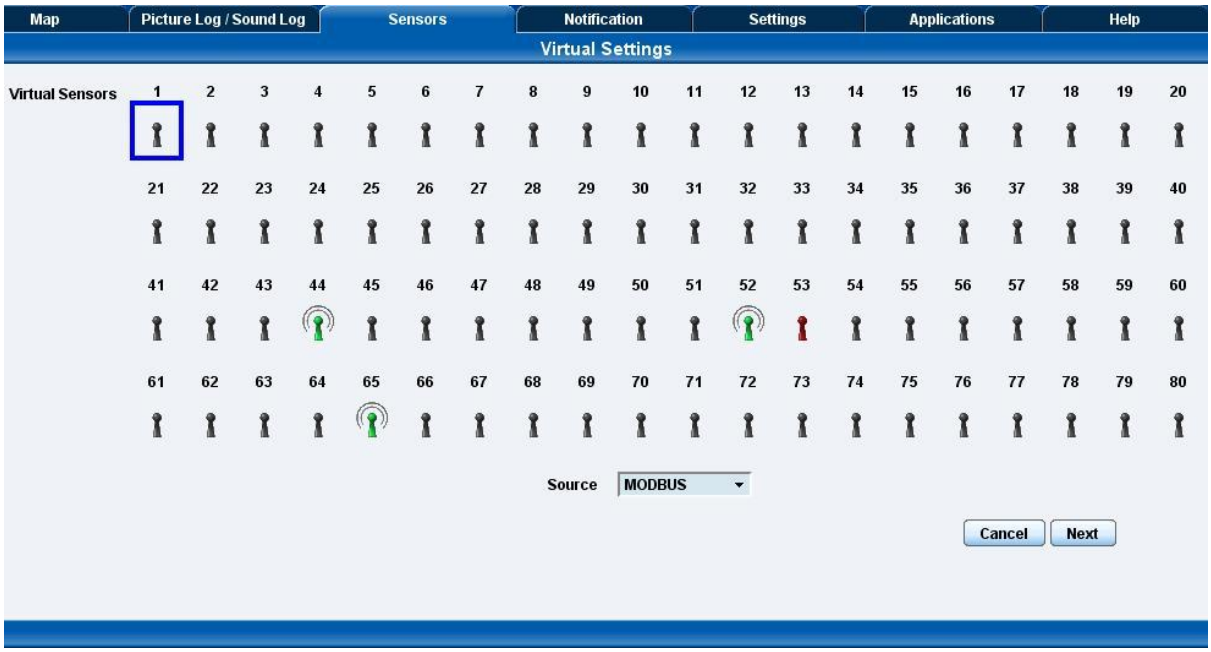

2.1 For setting up the Modbus TCP, first set the TCP port and TCP timeout (default TCP port is 502 and TCP timeout is 10 seconds). See screen shot above.


2.2 For setting up the Modbus RTU

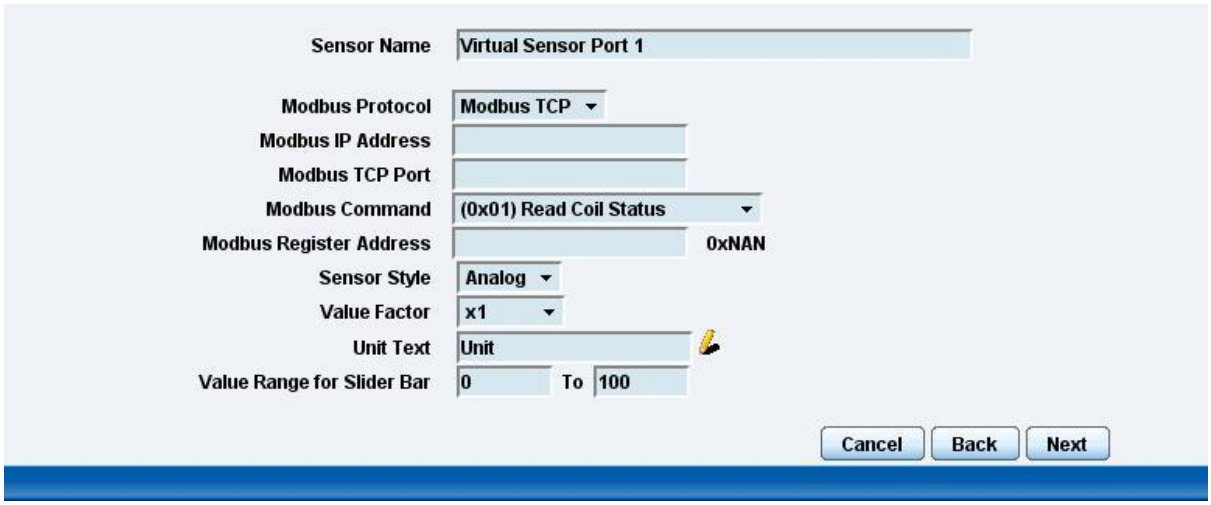
- First set the Modbus Address,
- Serial Port, Serial Port Speed,
- Serial Port Parity
- Serial Port Stop bits

(The default Modbus Address is 1, Serial Port is RS485, Serial Port Speed is 9660, Serial Port Parity is none and Serial Port Stop bit is 1)

3. The Modbus Sensors Addressing is now showing the Modbus INPUT Register Address of each sensor that is online on client device. Example; The Modbus INPUT Register Address of the Temperature on port 3 is 258 (0x0102)

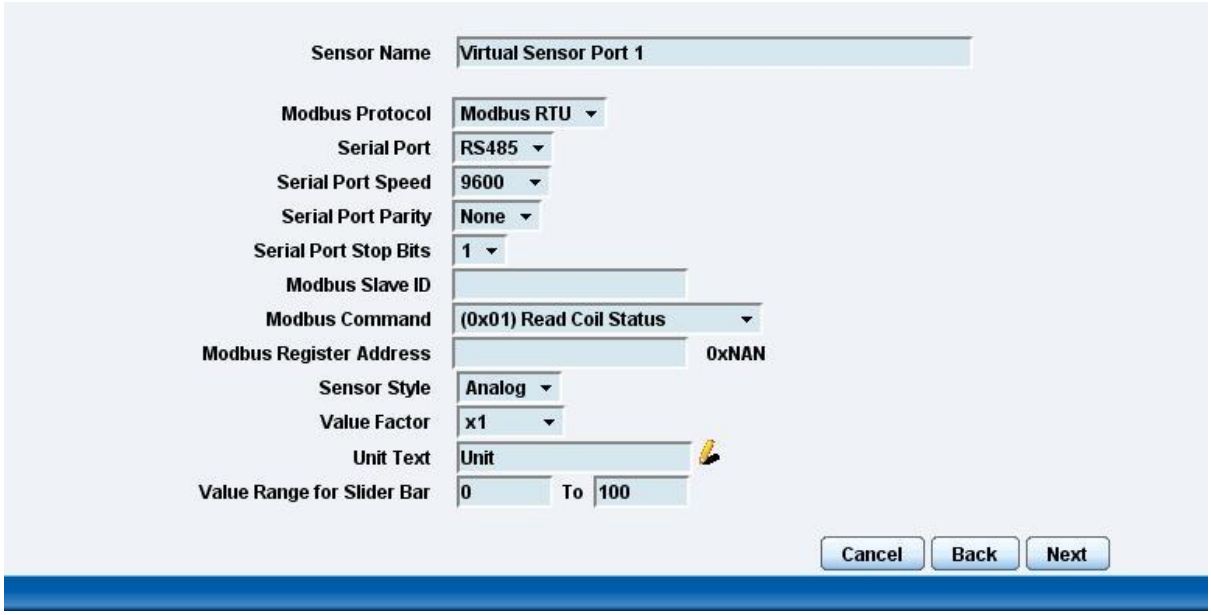



4. From the “Sensors” tab (on Master device) select virtual sensors, then set the “Source” to Modbus and then click “Next”. See screen shot above




4.1 For the Modbus TCP

- Set the “Modbus Protocol” to Modbus TCP
- Enter the Client device IP address in “Modbus IP Address”
- Enter the TCP port in “Modbus TCP Port”
- Set the “Modbus command: to get the sensor value or get the sensor status, (0x03 Read Holding Register for getting the sensor status and 0x04 Read Input Register for getting the sensor value).
- Now set the Modbus INPUT Register Address value to Modbus Register Address (see #3 above for the Modbus INPUT Register Address Value) Example; Set 258 for the read value of the Temperature on port 3 is 258). See screen shot above

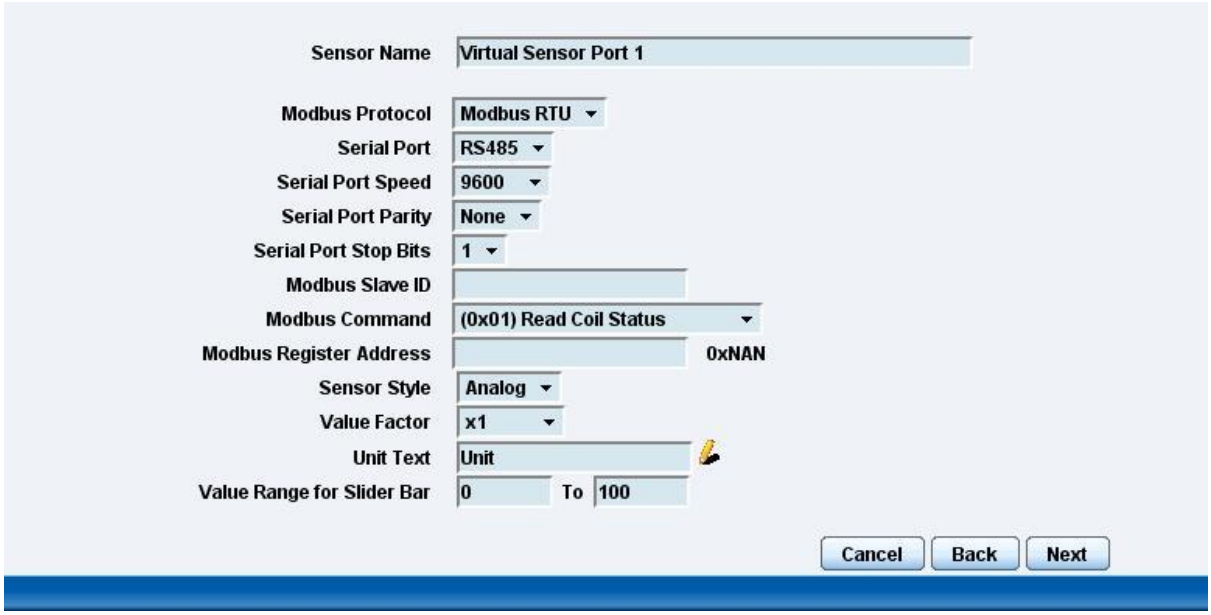



4.2 For Modbus RTU

- Set “Modbus Protocol” to Modbus RTU
- Set “Serial Port” to RS485 or other (this setting depends on the connection you are using)
- Set “Serial Port Speed” (this setting should be the same as shown in number 2.2 above)
- Set “Serial Port Parity” (this setting should be the same as shown in number 2.2 above)
- Set “Serial Port Stop Bits” (this setting should be the same as shown in number 2.2 above)
- Input the same “Modbus Address” value from step number 2.2 into “Modbus Slave ID”
- Select the Modbus command for getting the sensor value or the sensor status (0x03 Read Holding Register for getting the sensor status and 0x04 Read Input Register for getting the sensor value).
- Input the Modbus INPUT Register Address value into “Modbus Register Address”. An example of the Modbus INPUT Register Address Value can be found in step number 3 (Example input of 258 for the read value of the Temperature on port 3 is 258)

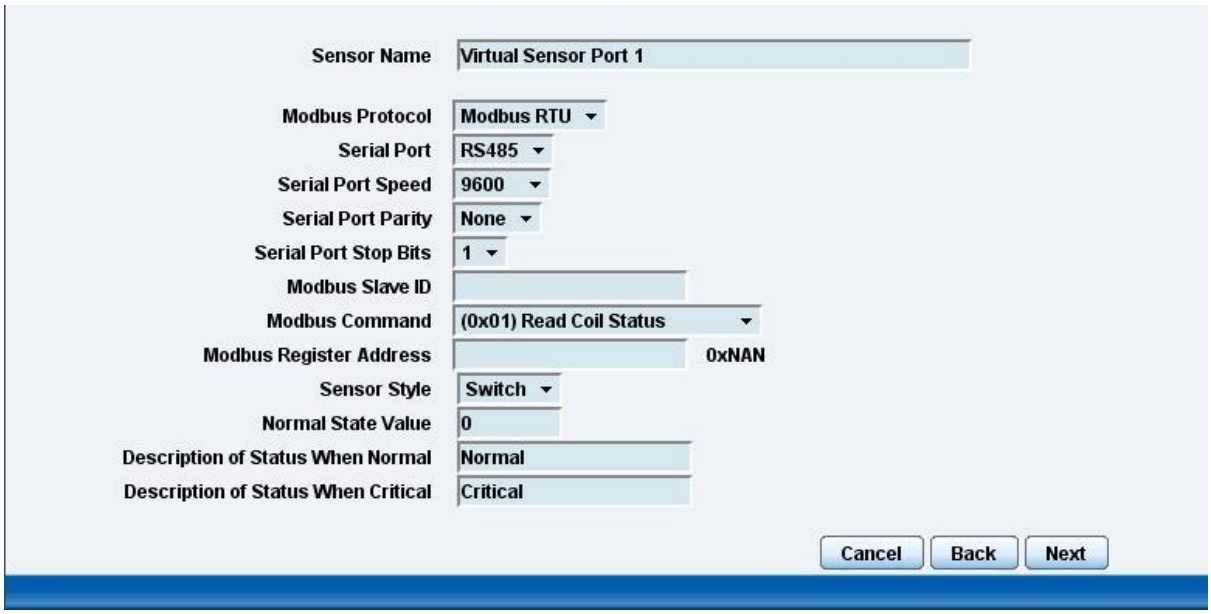
4.3 Sensor Style. Set the Sensor Style to Analog

4.3.1 Analog Style “Value Factor”, Set “Value Factor” for multiple with normal value (Example; if raw value is 1234 and needs to show a value to 12.34, then this should be set to x0.01. Default is x1). Input the Unit of sensor value into the “Unit Text” field, or click to input a special character





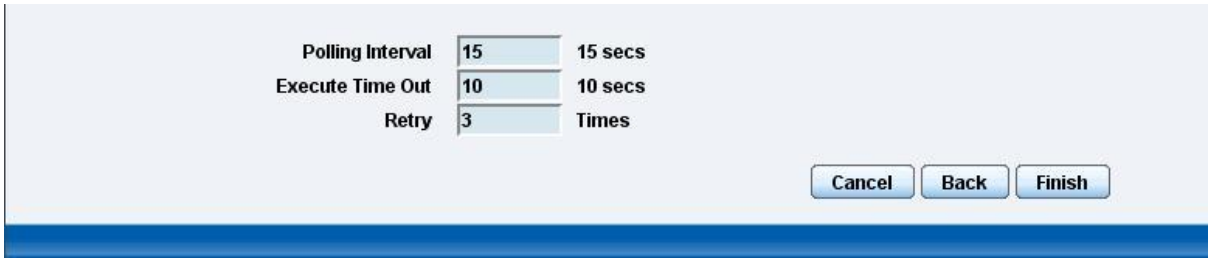
Input the Minimum and Maximum of the sensor values into the “Value Range for Slider Bar” field then click “Next”
Set the Low Critical, Low Warning, High Warning and High Critical thresholds as shown in the screen shot above, then click “Next” again



4.3.2 Switch Style

- Set the “Sensor Style” to Switch
- Input the Normal value of the sensor into “Normal State Value”.
- Input the Description of Normal and Critical status into “Description of Status When Normal” and “Description of Status When Critical”

(See screen shot on previous page)




5. Set Time to Polling

- Input the time to polling into “Polling Interval” (Default is 15 secs)
- Input “Execute Time Out” (Default is 10 secs) and “Retry” (Default is 3 times)
- Click the “Finish” button

## Section # 4 – Return value from Holding Register

The following is a list of what each of the return values from the Holding Register represent

1 No Status (Sensor has not been initialized)
2 Normal
3 High Warning
4 High Critical
5 Low Warning
6 Log Critical
7 Sensor Error
8 Switch Low (Output)
9 Switch High (Output)

For the INPUT Register

If the sensor is OFFLINE, the INPUT register will return 32767
If the sensor is No Status or Sensor Error, the INPUT register is -32768

## Section #5 - Troubleshooting & Example of configuration of a 3rd party power meter

### Some troubleshooting steps to try when having problems with Modbus RTU:-

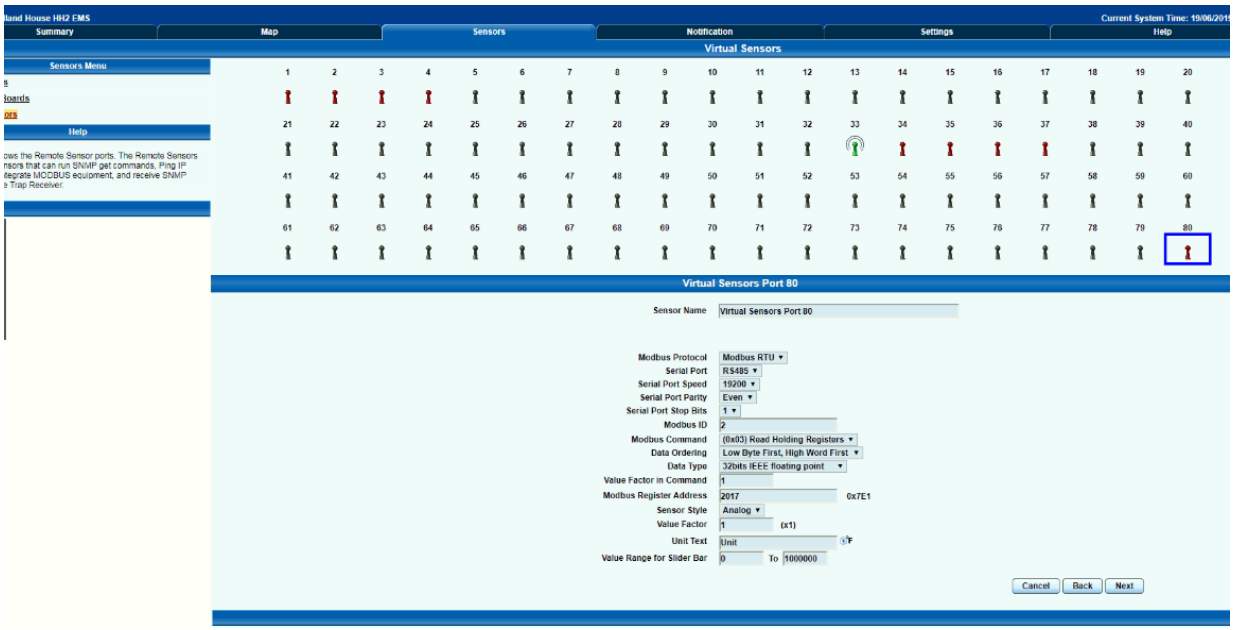
1. Make sure that the securityProbe is not configured as a Modbus **slave** device in the Modbus settings in the units web UI if you are monitoring FROM the securityProbe with the Modbus equipment connected to the RS485 port, or you will not see any readings. The Modbus setting must be set to Master in order to receive the Modbus data from the equipment connected to the RS485 port.

2. Verify the cabling and wire polarity to the Modbus slave device are correct as the manual explains above.
3. Verify the Modbus slave device is powered.
4. Verify that the Modbus address is correct and register parameters are correct in the Modbus virtual sensor settings.
For example : often the address is actually the register – 1 or the data ordering is often "Low byte first, High word first."

## How to configure a 3rd party power meter on the securityProbe

In this example the customer is trying to connect a Schneider PM5110 power meter.

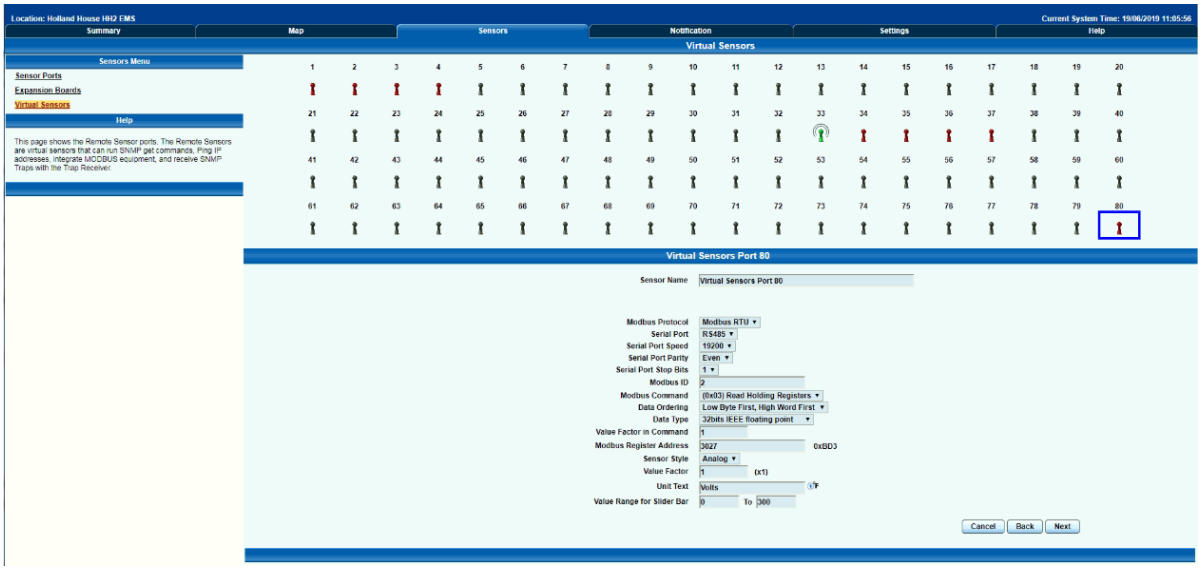
After our engineer changed the settings he was able to receive values for the register "Nominal voltage" (address = 2020, value = 230 volts). See the screenshot below in the securityProbe's Modbus virtual sensor settings.




These settings were changed to the following:
Even parity
Register address = address - 1
Low byte first, High word first
The last one is float
The first one is unsigned integer

Then he went ahead and checked the voltage settings & readings as follows:
Voltage A-N, Register 3028
Voltage = 242.0 V

As shown in the screenshots below.


Location: Holland House HH2 EMS
Current System Time: 19/06/2019 11:05:56
Summary
Map
Sensors
Notification
Settings
Help
Virtual Sensors
Sensors Menu
Sensor Ports
Expansion Boards
Virtual Sensors
Help
This page shows the Remote Sensor ports. The Remote Sensors are virtual sensors that can run SNMP get commands, Ping IP addresses, integrate MODBUS equipment, and receive SNMP Traps with the Trap Receiver.
Virtual Sensors Port 80
Sensor Name
Virtual Sensors Port 80
Modbus Protocol
Modbus RTU
Serial Port
RS485
Serial Port Speed
19200
Serial Port Parity
Even
Serial Port Stop Bits
Modbus ID
Modbus Command
(0x03) Read Holding Registers
Data Ordering
Low Byte First, High Word First
Data Type
32bits IEEE floating point
Value Factor in Command
Modbus Register Address
3027
0xBD3
Sensor Style
Analog
Value Factor
(x1)
Unit Text
Volts
°F
Value Range for Slider Bar
To
Cancel
Back
Next


Location: Holland House HH2 EMS
Current System Time: 19/06/2019 11:06:36
Summary
Map
Sensors
Notification
Settings
Help
Virtual Sensors
Sensors Menu
Sensor Ports
Expansion Boards
Virtual Sensors
Help
This page shows the Remote Sensor ports. The Remote Sensors are virtual sensors that can run SNMP get commands, Ping IP addresses, integrate MODBUS equipment, and receive SNMP Traps with the Trap Receiver.
Virtual Sensors Port 80
Normal Settings
Advanced Settings
Continuous Time Settings
Minimum Time Settings
Sensor Name
Virtual Sensors Port 80
Source
MODBUS
Change Configurations
Current Reading
242.0 Volts
Status
High Critical
Sensor Currently
Online
242 Volts
Low Critical
60
120
180
240
High Critical
Low Warning
High Warning
Low Critical
Low Warning
High Warning
High Critical
60
120
180
240
Save
Reset

The customer then inquired if the engineer was getting these registers from the Schneider documentation. Our engineer confirmed he was getting these from the PM5100_PM5300_ModbusRegisterList.xls document. This document was obtained from the meter manufacturer by the customer and previously sent to us.

The customer then asked where it was dictated about the data type needing to change.
Which our engineer replied, the voltage is 3028, but they had used 3027 and got a value, however if in the excel doc the Data Type was declared as "FLOAT32" then this data type needed to be set to "32 bits IEEE floating point"
If in the excel doc the Data Type was declared as "INT16U" then this data type would need to be set to "..."

Moreover, the address should be decreased by 1 and the data ordering always should be "Low byte first, High word first".

So, there were 2 issues: with address and the byte ordering. The address (no matter what it is) needs to be decreased by 1. And to add you need wait a little bit before the readings will be displayed from the meter in the securityProbe's web UI.

The customer then enquired about the "Active Power" reading. They tested with active power 3054 and entered in 3053 and got 0

This would be expected as the power meter is not monitoring anything, just plugged in to mains/wall power socket. Because if there is no current the active power is going to be 0.

In this environment only the voltage and product are the only readings that can really be tested, however it does prove that the meter is communicating with the securityProbe.

For the frequency the register 3110 (need to enter 3109) which was now reading 50, or 50.1hz.

Regarding the question about the address (no matter what it is) needs to be decreased by 1.
For this PMS, yes the addressing of the registers in the manuals can vary from manufacturer to manufacturer. There is no specific reason for this it just depends on the manufacturer's implementation of the Modbus protocol and how they enter the information into their manuals.

Modbus operates with the term "register", which starts from 1, but most all manufacturers use "address" which starts from 0.

**Please contact support@akcp.com if you have any further technical questions or problems.**

# Thanks for Choosing AKCP!