

rheonics



inline process
density and viscosity
monitoring



MODBUS RTU Field Device Specification:
Rheonics, SME

Covers sensor Types: SR, SRV, SRD, DVP, DVM

GLOBAL SUPPORT

For Support, visit: <https://support.rheonics.com>

EUROPE HEADQUARTERS

Rheonics GmbH
Klosterstrasse. 19
8406 Winterthur Switzerland

US HEADQUARTERS

Rheonics, Inc.
3 Sugar Creek Center Blvd, Ste 100
Sugar Land, TX 77478
United States of America
Tel: +1 713 364 5427

Website: <https://rheonics.com>
Support Portal: <https://support.rheonics.com>
Sales E-Mail: info@rheonics.com
Support E-Mail: support@rheonics.com

© COPYRIGHT RHEONICS 2023

No part of this publication may be copied or distributed, transmitted, transcribed, stored in a retrieval system, or translated into any human or computer language, in any form or by any means, electronic, mechanical, manual, or otherwise, or disclosed to third parties without the express written permission of Rheonics. The information contained in this manual is subject to change without notice.

TRADEMARKS

Rheonics is a trademark of Rheonics, Inc. Other product and company names listed in this manual are trademarks or trade names of their respective manufacturers.

Contents

| | | |
|-----------|---|-----------|
| 1 | Before you begin | 4 |
| 2 | Product overview | 5 |
| 3 | Modbus RTU Wiring | 6 |
| 4 | Modbus RTU Configuration | 8 |
| 5 | Modbus Protocol | 10 |
| 6 | Linking tags with Modbus Poll | 16 |
| 7 | Modbus RTU Command and Response examples | 25 |
| 8 | Sensor status and parameter status | 27 |
| 9 | Units table | 30 |
| 10 | Troubleshooting | 31 |
| 11 | Reviews and approvals | 33 |
| 12 | Notes/Errata | 34 |

1 Before you begin

1.1 About the manual

This manual provides information on Modbus RTU support on Rheonics devices. This document specifies all the device-specific features and Modbus RTU Protocol implementation details.

Important Instructions

This manual assumes that the following conditions apply:

- The sensor has been installed correctly and completely according to the installation.
- The installation complies with all applicable safety requirements.
- The user is trained in relevant safety standards.

1.2 Purpose

This specification is designed to complement the SME Installation Manual by providing a complete, clear description of this Field Device from a Modbus RTU Communication perspective.

1.3 Who should use this document?

The specification is designed to be a technical reference for Modbus RTU end users. This document assumes the reader is familiar with Modbus Protocol requirements and terminology.

1.4 Warning

Before connecting the Modbus Communicator in an explosive atmosphere, make sure instruments are ordered and installed in accordance with intrinsically safe or EX classification-specific field wiring practices. Explosions can cause serious injury or death. Review Rheonics Ex Manual at <https://rheonics.com/resources/manuals/>.

1.5 Nomenclature

| Abbreviation (short form) | Full-term | Meaning |
|---------------------------|-------------------------------------|--|
| SRV | Symmetric Resonator Viscometer | Viscosity sensor |
| SRD | Symmetric Resonator Densitometer | Density and Viscosity sensor |
| DVP | Density Viscosity Probe | HPHT inline probe |
| DVM | Density Viscosity Module | HPHT inline module |
| RCP | Rheonics Control Panel | Software for data acquisition and configuration |
| SME | Smart Module Electronics | Sensor electronics |

Table 1. Defined Acronyms

1.6 Related Documentation

You can find all product documentation on the USB stick shipped with the SME and on our website at <https://rheonics.com/resources>.

2 Product overview

The Rheonics SME provides clients Modbus RTU interface to get digital data on all devices ordered. This document provides guidance for field connection by an end user.

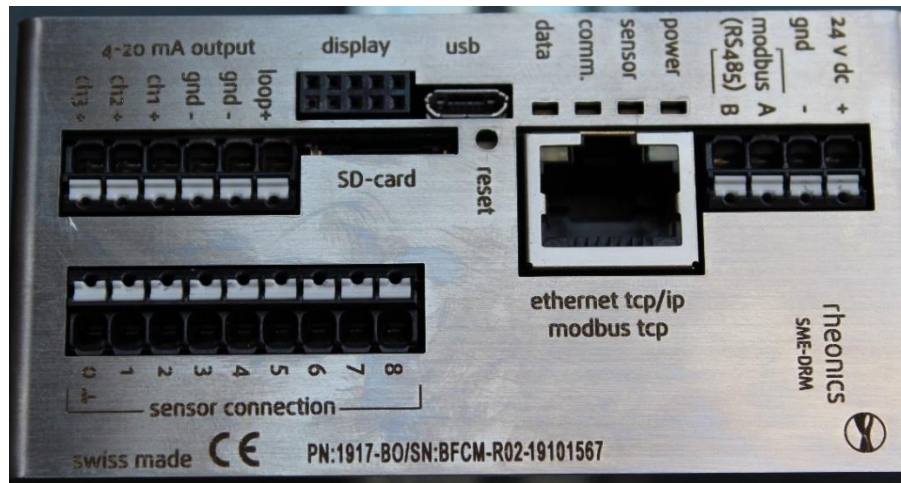


Figure 1. SME sensor electronics unit.

2.1 Process Interface

The SME is compatible with various Rheonics instruments. This includes Type: SR (SRV & SRD), Type: DV (DVP, DVM) and other instruments using the SME electronics from Rheonics.



Figure 2: Rheonics Sensor for Viscosity and density measurements.

2.2 Reference to other instruments.

Manuals and guides for digital instruments are modular. General instructions give information about the functioning and installation of instruments. Operational instructions explain the use of the digital instrument features and parameters. Fieldbus-specific information explains the installation and use of the instrument on that Fieldbus network. Related manuals are listed next and can be found in <https://rheonics.com/resources>:

- SRV USER MANUAL
- SRD USER MANUAL
- DVP USER MANUAL
- DVM USER MANUAL
- SENSOR INSTALLATION MANUAL
- RHEONICS CONTROL PANEL USER MANUAL

3 Modbus RTU Wiring

3.1 Instrument overview:

Rheonics SME Modbus RTU connection is by RS485-2W where the interface is not electrically isolated.

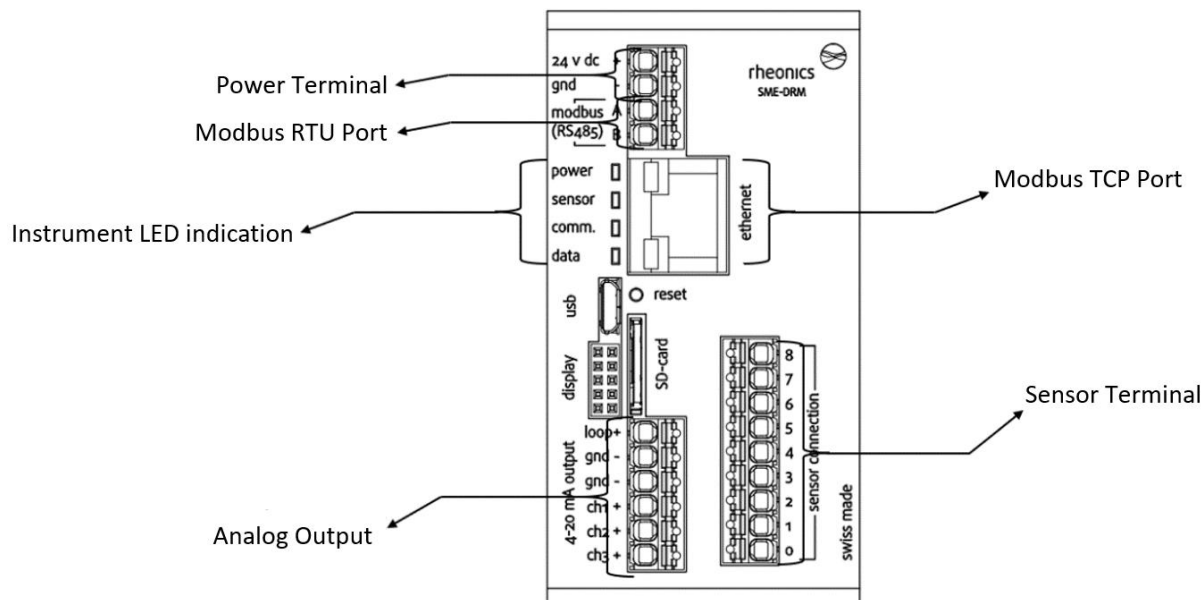


Figure 3. SME-DRM Diagram.

3.2 Rheonics Modbus RTU Parameters

All SME devices implement a MODBUS RTU compliant RTU slave.

Factory default settings of Modbus parameters are:

- Address = 0 or 1
- Baudrate = 38400
- Parity = odd

The system is capable of supporting:

- Parity settings of odd, even and none.
- Baudrates of 9600, 19200 and 38400

3.3 Rheonics Modbus RTU Wiring

Rheonics SME wiring for Modbus RTU is by RS485-2W. The device does not have internal biasing resistors and termination resistors and proper setup of the RS485 network have to be ensured. Find below the special consideration for the correct operation of Modbus RTU in Rheonics SME.

- **Termination:** Termination resistor should be only at the end and beginning of the bus to avoid reflections. Requirements on termination differ not only on the baud-rate used but also on the slew-rate of device internal drivers. As a general rule, if there are devices on the RS485 bus that allow fast transmission speeds, and cable lengths exceed a few meters, then termination is required.

- **Biasing:** Biasing of the RS485 bus is required. If biasing is omitted, communication may work unreliably or not at all. See figure below for Pull-Up/Pull-Down resistors.
- **Connections:** Connection stubs on the RS485 bus should be kept to the minimum.
- **Common:** RS485 does allow common mode voltages of +12V / -7V. It is therefore recommended to use a common wire to ensure common mode voltage is within safe limits for all receivers. In case of problems, it might be necessary to introduce RS485 isolators into the network. The common terminal for the SME device is the 0V terminal (GND) for the power supply.

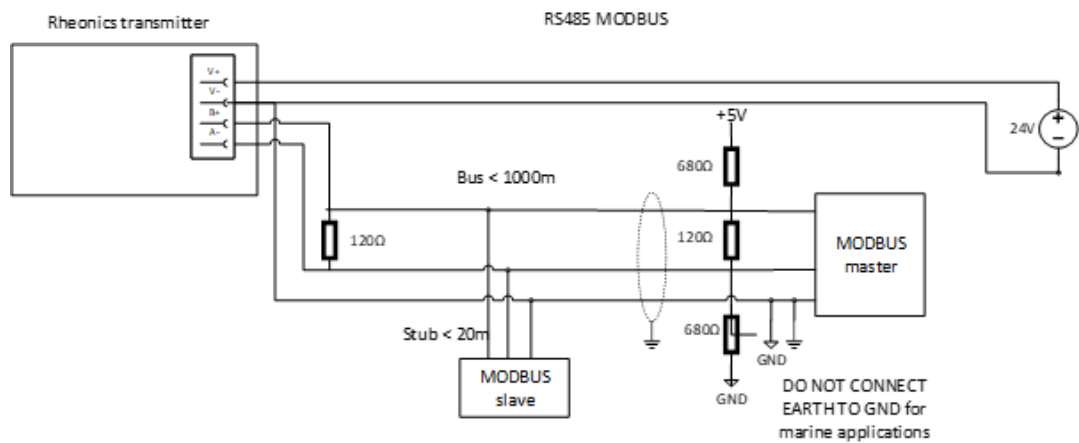


Figure 4: Modbus Network Diagram.

Figure 5 shows multiple Rheonics SME transmitters connected on the RS485 network as Slave 1 to “n”.

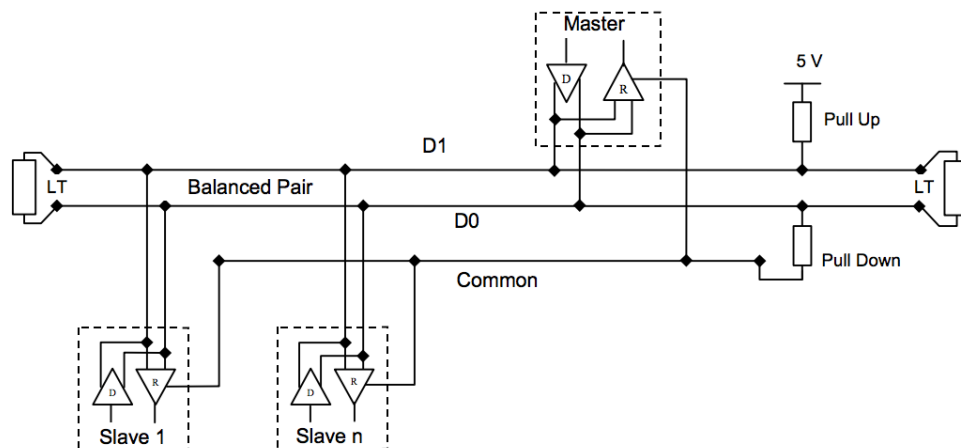


Figure 5: Multiple Device Network Diagram.

4 Modbus RTU Configuration

4.1 Components Used

- Rheonics SRV, SRD, DVP or DVM w/ Firmware V03.30/0 or higher
- Modbus Communication software (e.g. Modbus Poll)
- Software sensor: Rheonics Control Panel (RCP)
- Windows 10 - 64 bit
- USB to RS485 2-Wire Converter Terminal Block

4.2 System Connections

Connect the Rheonics sensor to a PC (with RCP and Modbus communication software installed) using the RS485 pins of the sensor electronics. This can be done with a USB-RS485 or RS232-RS485 Adapter.

4.3 Configure Rheonics SME

In this section, the SME ethernet configuration is set. We will use a PC with the RCP software and connect it to the SME using a USB. Verify the Address, Baud Rate and Polarity of the device. Figure 7 shows a standard configuration that can be used for the correct performance of the system. See the next steps as guidance.

- 4.3.1 Open the RCP software. On Settings Communication Tab, select USB, identify the COM Port of the desired SME to connect and click Apply. Wait for the Status Led to go green.

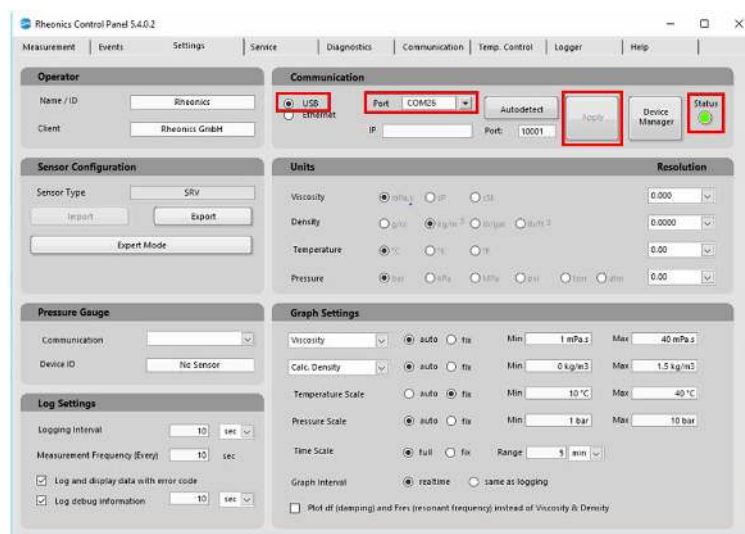


Figure 6: USB connection with SME.

4.3.2 For the RTU communication, verify the Modbus Address, Baud Rate and Parity.

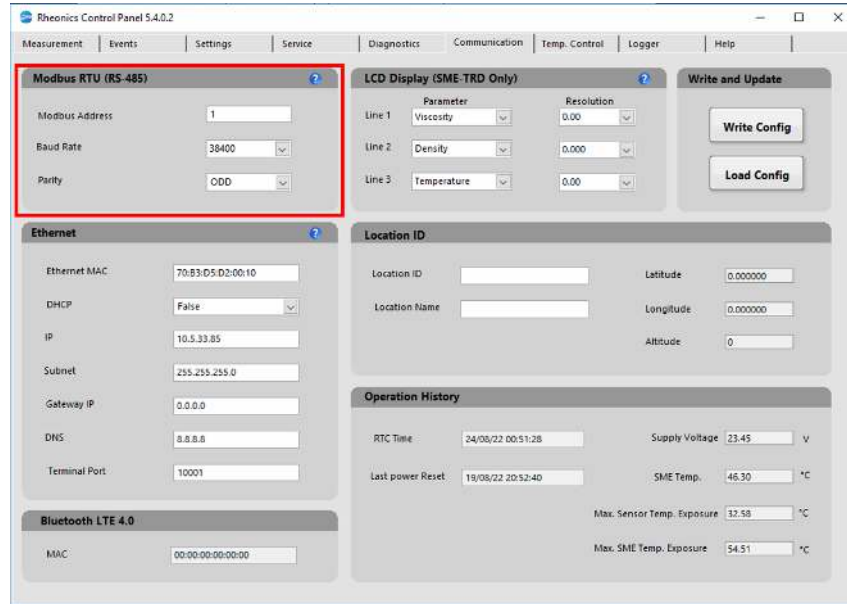


Figure 7: Review Modbus RTU Parameters in RCP.

4.3.3 Click “Write Config” to assure the configuration in the SME. Wait until the button turns green.

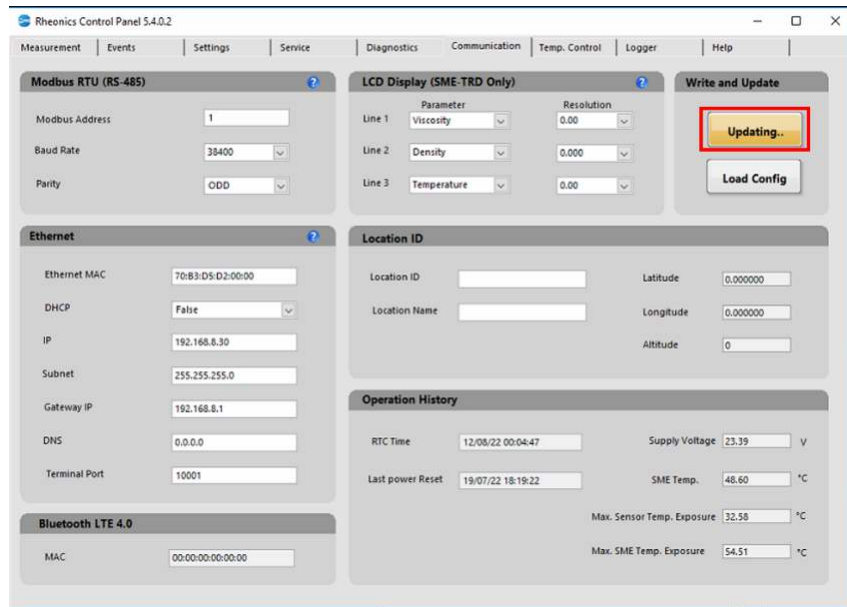


Figure 8. Upload configuration to the SME.

5 Modbus Protocol

The structure of the request and response body in Modbus RTU connections, from the function code to the end of the data portion, has the same layout and meaning as in the other MODBUS variants, such as Modbus TCP or Modbus ASCII. The only differences are the form of any ‘framing’ sequence, error check pattern, and address interpretation.

The request structure for Modbus RTU has the following bytes:

| | |
|----------------|--|
| byte 0: | Device Address |
| byte 1: | Functional Code |
| byte 2: | Address 1 st Register (High Byte) |
| byte 3: | Address 1 st Register (Low Byte) |
| byte 4: | Number of Registers (High Byte) |
| byte 5: | Number of Registers (Low Byte) |
| byte 6: | CRC (Low Byte) |
| byte 7: | CRC (High Byte) |

Table 2: Modbus RTU framing.

Unit Identifier (byte 0) for Rheonics SME through Modbus RTU by default is 1 (0x01), it can be verified in the RCP Software. Byte 1 is the Modbus functional code, mostly used are: 0x03 to read Holding Registers and 0x04 to read Input Registers. Byte 2 to 5 define the Data with its first Register address and the number of registers to be read. Byte 6 and 7 content the CRC or Error Check Code.

5.1 Measured Parameters List

The measured parameters are mapped into the Modbus input registers. There is a total of 23 Parameters available from the SME. Table 3 shows associated variable, Sensor and Error status.

| Parameter | Measurement | Description |
|---------------------|-------------------------------------|--|
| Parameter 0 | Viscosity median | Takes the median of the last 5 sensor measurements points |
| Parameter 1 | Density median | |
| Parameter 2 | Temperature median | |
| Parameter 3 | Kinematic Viscosity | Measured value divided by density squared |
| Parameter 4 | Density average | Calculates the average of the last 5 measurement points |
| Parameter 5 | Viscosity raw ‘process measurement’ | Reads the direct measurement taken by the sensor. |
| Parameter 6 | Density raw | |
| Parameter 7 | Temperature raw | |
| Parameter 8 | Resonant Frequency | Measured in Hertz |
| Parameter 9 | Compensated Resonant Frequency | |
| Parameter 10 | Damping | Damping from Measurement |
| Parameter 11 | Coil Temperature | |
| Parameter 12 | Viscosity Last Good | Keeps the last-good median measurement in memory when there is a measurement error |
| Parameter 13 | Density Last Good | |

| | | |
|----------------------|---|--|
| Parameter 14 | Displays of mapped value from Modbus register 512 | |
| Parameter 15 | Displays of mapped value from Modbus register 514 | |
| Parameter 16 | Displays of mapped value from Modbus register 516 | |
| Parameter 17 | Estimated Temperature | Calculated from internal temperature Algorithm |
| Parameter 18 | Temperature from PT1000 sensor | Read from internal sensor |
| Parameter 19 | Calculated parameter from viscosity model | Set from Advanced Calculations |
| Parameter 20 | Calculated parameter from density model | |
| Parameter 21 | Calculated parameter from concentration model | |
| Parameter 22 | Sensor Cleanliness Ratio | Used for estimating sensor cleanliness |
| Sensor Status | Sensor Error Status | Status of the sensor |
| Error | Error State | Error state of the sensor |

Table 3: Measured Parameters List.

5.2 Input Registers

For each parameter there are seven registers associated; the first two contain the value of the parameter in float data type, the third is the value scaled by 100 in Int16 format, fourth is the status of the parameter in UInt16. The fifth and sixth contain the uncalibrated value and the last one indicates the measurement units.

It's important to notice that Table 4 uses PDU addressing; addresses start at zero. Depending on the PLC used it might be required to add 1 to the addresses, to get the correct registers.

| Address (Dec) | Register (Dec) | Length (Reg.) | Type | Description |
|--------------------|--------------------|---------------|---------|---|
| Parameter 0 | Viscosity median | | | |
| 40 | 300041 | 2 | Float32 | Parameter 0 value as float |
| 42 | 300043 | 1 | SInt16 | Parameter 0 value scaled by 100 |
| 43 | 300044 | 1 | UInt16 | Parameter 0 status (See section 8.2) |
| 44 | 300045 | 2 | Float32 | Parameter 0 Uncalibrated value |
| 46 | 300047 | 1 | UInt16 | Parameter 0 Unit (See section 9) |
| Parameter 1 | Density median | | | |
| 48 | 300049 | 2 | Float32 | Parameter 1 value as float |
| 50 | 300051 | 1 | SInt16 | Parameter 1 value scaled by 100 |
| 51 | 300052 | 1 | UInt16 | Parameter 1 status |
| 52 | 300053 | 2 | Float32 | Parameter 1 Uncalibrated value |
| 54 | 300055 | 1 | UInt16 | Parameter 1 Unit |
| Parameter 2 | Temperature median | | | |
| 56 | 300057 | 2 | Float32 | Parameter 2 value as float |

| | | | | |
|--------------------|-------------------------------------|---|---------|---------------------------------|
| 58 | 300059 | 1 | SInt16 | Parameter 2 value scaled by 100 |
| 59 | 300060 | 1 | UInt16 | Parameter 2 status |
| 60 | 300061 | 2 | Float32 | Parameter 2 Uncalibrated value |
| 62 | 300063 | 1 | UInt16 | Parameter 2 Unit |
| Parameter 3 | Kinematic Viscosity | | | |
| 64 | 300065 | 2 | Float32 | Parameter 3 value as float |
| 66 | 300067 | 1 | SInt16 | Parameter 3 value scaled by 100 |
| 67 | 300068 | 1 | UInt16 | Parameter 3 status |
| 68 | 300069 | 2 | Float32 | Parameter 3 Uncalibrated value |
| 70 | 300071 | 1 | UInt16 | Parameter 3 Unit |
| Parameter 4 | Density Average | | | |
| 72 | 300073 | 2 | Float32 | Parameter 4 value as float |
| 74 | 300075 | 1 | SInt16 | Parameter 4 value scaled by 100 |
| 75 | 300076 | 1 | UInt16 | Parameter 4 status |
| 76 | 300077 | 2 | Float32 | Parameter 4 Uncalibrated value |
| 78 | 300079 | 1 | UInt16 | Parameter 4 Unit |
| Parameter 5 | Viscosity raw 'process measurement' | | | |
| 80 | 300081 | 2 | Float32 | Parameter 5 value as float |
| 82 | 300083 | 1 | SInt16 | Parameter 5 value scaled by 100 |
| 83 | 300084 | 1 | UInt16 | Parameter 5 status |
| 84 | 300085 | 2 | Float32 | Parameter 5 Uncalibrated value |
| 86 | 300087 | 1 | UInt16 | Parameter 5 Unit |
| Parameter 6 | Density raw | | | |
| 88 | 300089 | 2 | Float32 | Parameter 6 value as float |
| 90 | 300091 | 1 | SInt16 | Parameter 6 value scaled by 100 |
| 91 | 300092 | 1 | UInt16 | Parameter 6 status |
| 92 | 300093 | 2 | Float32 | Parameter 6 Uncalibrated value |
| 94 | 300095 | 1 | UInt16 | Parameter 6 Unit |
| Parameter 7 | Temperature raw | | | |
| 96 | 300097 | 2 | Float32 | Parameter 7 value as float |
| 98 | 300099 | 1 | SInt16 | Parameter 7 value scaled by 100 |
| 99 | 300100 | 1 | UInt16 | Parameter 7 status |
| 100 | 300101 | 2 | Float32 | Parameter 7 Uncalibrated value |
| 102 | 300103 | 1 | UInt16 | Parameter 7 Unit |
| Parameter 8 | Resonant Frequency | | | |
| 104 | 300105 | 2 | Float32 | Parameter 8 value as float |
| 106 | 300107 | 1 | SInt16 | Parameter 8 value scaled by 100 |
| 107 | 300108 | 1 | UInt16 | Parameter 8 status |
| 108 | 300109 | 2 | Float32 | Parameter 8 Uncalibrated value |
| 110 | 300111 | 1 | UInt16 | Parameter 8 Unit |
| Parameter 9 | Compensated Resonant Frequency | | | |
| 112 | 300113 | 2 | Float32 | Parameter 9 value as float |
| 114 | 300115 | 1 | SInt16 | Parameter 9 value scaled by 100 |
| 115 | 300116 | 1 | UInt16 | Parameter 9 status |

| | | | | |
|---------------------|---|---|---------|----------------------------------|
| 116 | 300117 | 2 | Float32 | Parameter 9 Uncalibrated value |
| 118 | 300119 | 1 | UInt16 | Parameter 9 Unit |
| Parameter 10 | Damping Frequency | | | |
| 120 | 300121 | 2 | Float32 | Parameter 10 value as float |
| 122 | 300123 | 1 | Slnt16 | Parameter 10 value scaled by 100 |
| 123 | 300124 | 1 | UInt16 | Parameter 10 status |
| 124 | 300125 | 2 | Float32 | Parameter 10 Uncalibrated value |
| 126 | 300127 | 1 | UInt16 | Parameter 10 Unit |
| Parameter 11 | Coil Temperature | | | |
| 128 | 300129 | 2 | Float32 | Parameter 11 value as float |
| 130 | 300131 | 1 | Slnt16 | Parameter 11 value scaled by 100 |
| 131 | 300132 | 1 | UInt16 | Parameter 11 status |
| 132 | 300133 | 2 | Float32 | Parameter 11 Uncalibrated value |
| 134 | 300135 | 1 | UInt16 | Parameter 11 Unit |
| Parameter 12 | Viscosity Last Good | | | |
| 136 | 300137 | 2 | Float32 | Parameter 12 value as float |
| 138 | 300139 | 1 | Slnt16 | Parameter 12 value scaled by 100 |
| 139 | 300140 | 1 | UInt16 | Parameter 12 status |
| 140 | 300041 | 2 | Float32 | Parameter 12 Uncalibrated value |
| 142 | 300043 | 1 | Slnt16 | Parameter 12 Unit |
| Parameter 13 | Density Last Good | | | |
| 144 | 300145 | 2 | Float32 | Parameter 13 value as float |
| 146 | 300147 | 1 | Slnt16 | Parameter 13 value scaled by 100 |
| 147 | 300148 | 1 | UInt16 | Parameter 13 status |
| 148 | 300149 | 2 | Float32 | Parameter 13 Uncalibrated value |
| 150 | 300151 | 1 | Slnt16 | Parameter 13 Unit |
| Parameter 14 | Displays of mapped value from Modbus register 512 | | | |
| 152 | 300153 | 2 | Float32 | Parameter 14 value as float |
| 154 | 300155 | 1 | Slnt16 | Parameter 14 value scaled by 100 |
| 155 | 300156 | 1 | UInt16 | Parameter 14 status |
| 156 | 300157 | 2 | Float32 | Parameter 14 Uncalibrated value |
| 158 | 300159 | 1 | Slnt16 | Parameter 14 Unit |
| Parameter 15 | Displays of mapped value from Modbus register 514 | | | |
| 160 | 300161 | 2 | Float32 | Parameter 15 value as float |
| 162 | 300163 | 1 | Slnt16 | Parameter 15 value scaled by 100 |
| 163 | 300164 | 1 | UInt16 | Parameter 15 status |
| 164 | 300165 | 2 | Float32 | Parameter 15 Uncalibrated value |
| 166 | 300167 | 1 | Slnt16 | Parameter 15 Unit |
| Parameter 16 | Displays of mapped value from Modbus register 516 | | | |
| 168 | 300169 | 2 | Float32 | Parameter 16 value as float |
| 170 | 300171 | 1 | Slnt16 | Parameter 16 value scaled by 100 |
| 171 | 300172 | 1 | UInt16 | Parameter 16 status |
| 172 | 300173 | 2 | Float32 | Parameter 16 Uncalibrated value |
| 174 | 300175 | 1 | Slnt16 | Parameter 16 Unit |

| | | | | |
|----------------------|--|---|---------|----------------------------------|
| Parameter 17 | Estimated Temperature | | | |
| 176 | 300177 | 2 | Float32 | Parameter 17 value as float |
| 178 | 300179 | 1 | SInt16 | Parameter 17 value scaled by 100 |
| 179 | 300180 | 1 | UInt16 | Parameter 17 status |
| 180 | 300181 | 2 | Float32 | Parameter 17 Uncalibrated value |
| 182 | 300183 | 1 | SInt16 | Parameter 17 Unit |
| Parameter 18 | Temperature from PT1000 sensor | | | |
| 184 | 300185 | 2 | Float32 | Parameter 18 value as float |
| 186 | 300187 | 1 | SInt16 | Parameter 18 value scaled by 100 |
| 187 | 300188 | 1 | UInt16 | Parameter 18 status |
| 188 | 300189 | 2 | Float32 | Parameter 18 Uncalibrated value |
| 190 | 300191 | 1 | SInt16 | Parameter 18 Unit |
| Parameter 19 | Calculated parameter from viscosity models | | | |
| 192 | 300193 | 2 | Float32 | Parameter 19 value as float |
| 194 | 300195 | 1 | SInt16 | Parameter 19 value scaled by 100 |
| 195 | 300196 | 1 | UInt16 | Parameter 19 status |
| 196 | 300197 | 2 | Float32 | Parameter 19 Uncalibrated value |
| 198 | 300199 | 1 | SInt16 | Parameter 19 Unit |
| Parameter 20 | Calculated parameter from density models | | | |
| 200 | 300201 | 2 | Float32 | Parameter 20 value as float |
| 202 | 300203 | 1 | SInt16 | Parameter 20 value scaled by 100 |
| 203 | 300204 | 1 | UInt16 | Parameter 20 status |
| 204 | 300205 | 2 | Float32 | Parameter 20 Uncalibrated value |
| 206 | 300207 | 1 | SInt16 | Parameter 20 Unit |
| Parameter 21 | Calculated parameter from concentration models | | | |
| 208 | 300209 | 2 | Float32 | Parameter 21 value as float |
| 210 | 300211 | 1 | SInt16 | Parameter 21 value scaled by 100 |
| 211 | 300212 | 1 | UInt16 | Parameter 21 status |
| 212 | 300213 | 2 | Float32 | Parameter 21 Uncalibrated value |
| 214 | 300215 | 1 | SInt16 | Parameter 21 Unit |
| Parameter 22 | Sensor Cleanliness Ratio | | | |
| 216 | 300217 | 2 | Float32 | Parameter 22 value as float |
| 218 | 300219 | 1 | SInt16 | Parameter 22 value scaled by 100 |
| 219 | 300220 | 1 | UInt16 | Parameter 22 status |
| 220 | 300221 | 2 | Float32 | Parameter 22 Uncalibrated value |
| 222 | 300223 | 1 | SInt16 | Parameter 22 Unit |
| Sensor Status | Sensor Error Status | | | |
| 39 | 300040 | 1 | | |
| Error | Error State | | | |
| 514 | 300515 | 1 | UInt16 | Error value |

Table 4: Input Registers.

5.3 Data Types

Float: IEEE754 floating point. This data type spans two registers which should be read together. Depending on the endianness of the host it might be necessary to swap high/low byte and/or the respective registers. To transform the value to decimal form, follow the IEEE 754 Standard for Floating Point Binary Arithmetic.

Int16: Signed 16 bit integer (register).

Uint16: Unsigned 16 bit integer.

6 Linking tags with Modbus Poll

The Rheonics devices offer many different modules and parameters through Modbus. To select parameters, the client's configuration tooling software is used, in this case, Modbus Poll.

6.1 Open the software and follow the steps:

6.1.1 Create a new Connection.

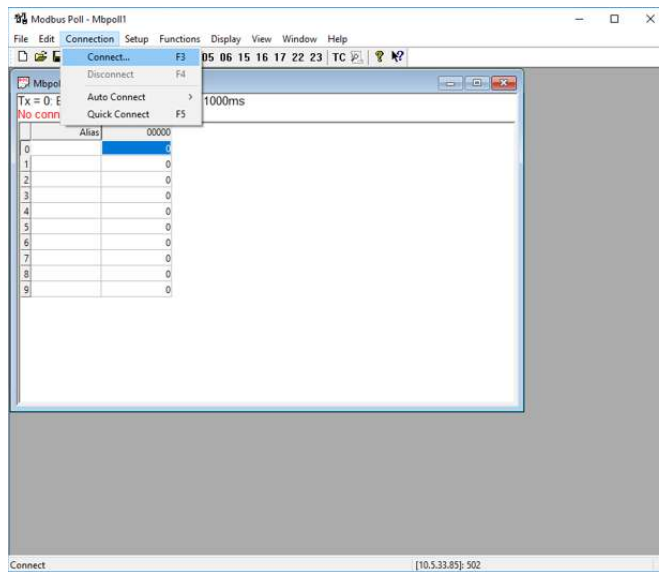


Figure 9: Modbus Poll new connection.

6.1.2 With the new window, set a Serial Port connection type (if you are connecting the PC to the RS485-2W terminals via USB) and select the correct COM Port from the SME. Set other Parameters as shown in the Figure and click OK.

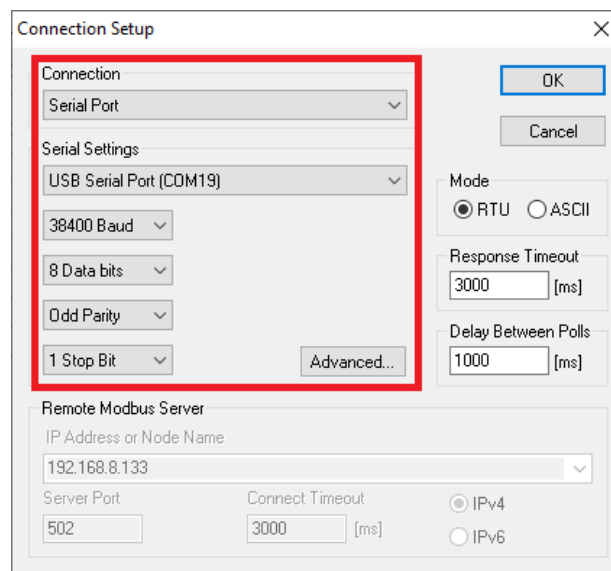


Figure 10: Connection Setup.

6.1.3 Go to “Set up” tab and select “Read/Write Definition”.

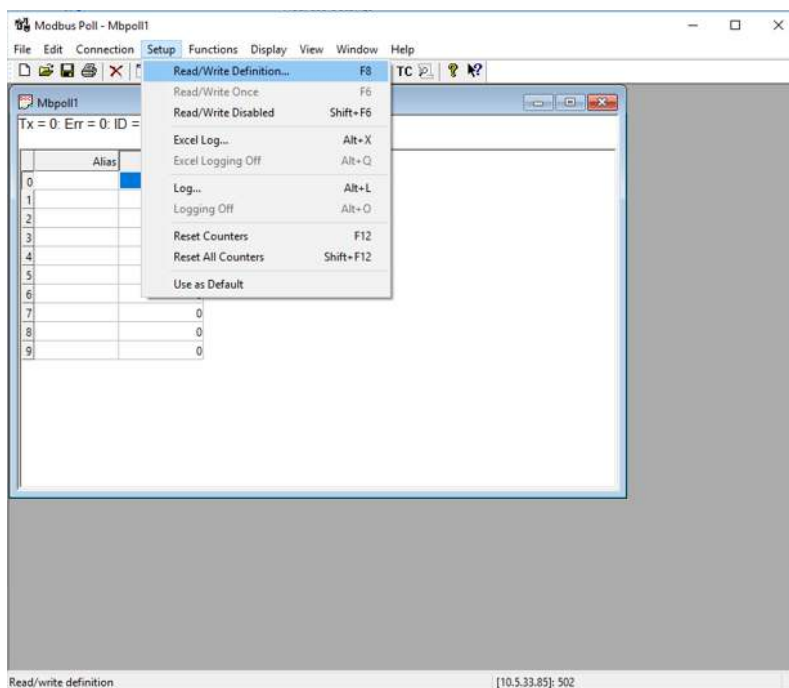


Figure 11: Reading Definition.

6.1.4 Once the new window is opened, set all the parameters as shown below (only quantity can vary).

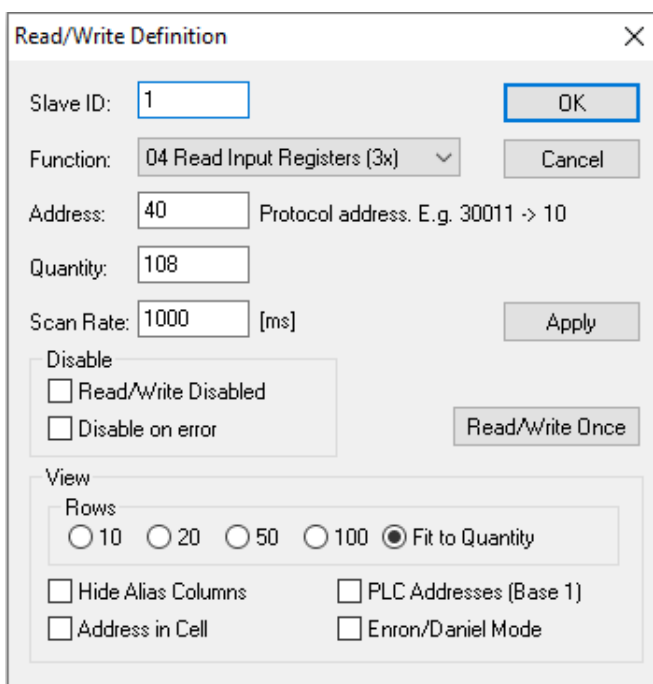


Figure 12: Configure Read/Write Definition.

- 6.1.5 The data from address 40 up to 49 is going to appear on the screen, this data corresponds to parameter 0. Check the following link for the mapped parameters.

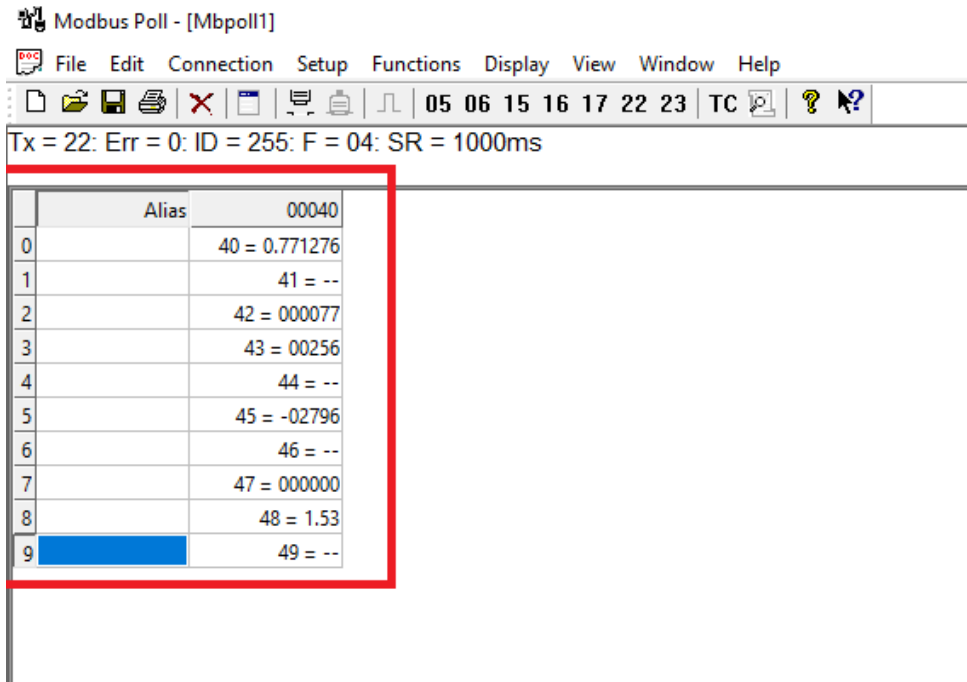


Figure 13: Example of Registers read.

- 6.1.6 By right-clicking any data, it can be seen that by default the data type is “signed”. This needs to be changed for every data according to Table 4: Input Registers.

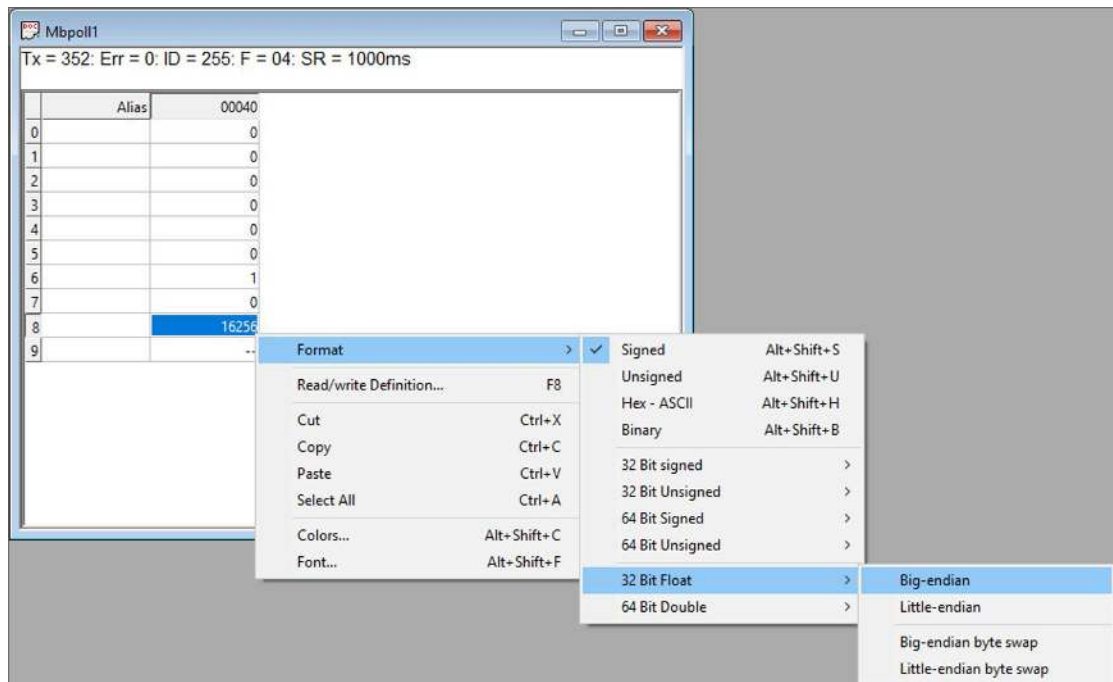


Figure 14: Configure Data format.

In this case, when Address 48 is configured as Float32 Big-endian, the value changes to 1, the current Density median. The data is better shown by configuring the correct data type and adding Alias to each address. The result would look like in the following Figure.

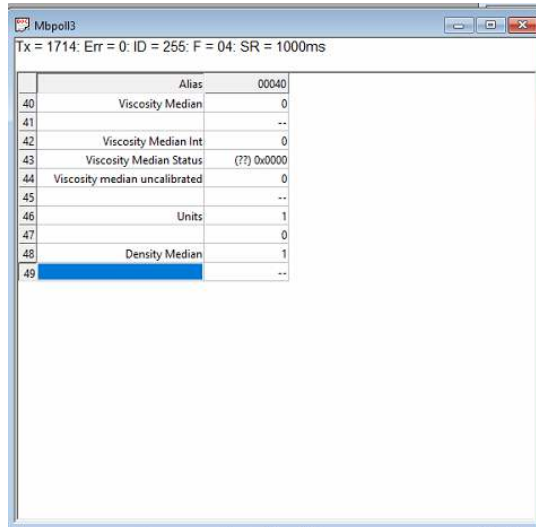


Figure 15: Data with correct format and alias.

6.2 Import Modbuspoll pre-configured files for Rheonics Modbus Devices

To read values through Modbus Poll easily, pre-configured files can be used. These already have the correct data types and alias. You can find them on Rheonics Support web page [here](#). To import them, follow the next steps.

6.2.1 Click Open on the File Tab.

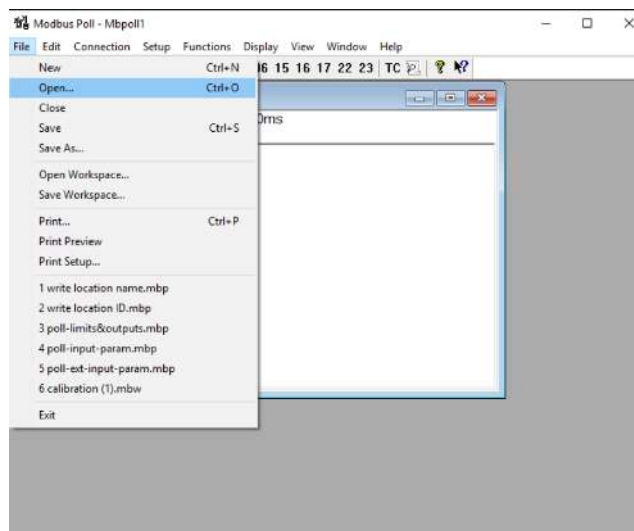


Figure 16: Import pre-configured file.

6.2.2 Choose the desired file between the files available on Rheonics web page.

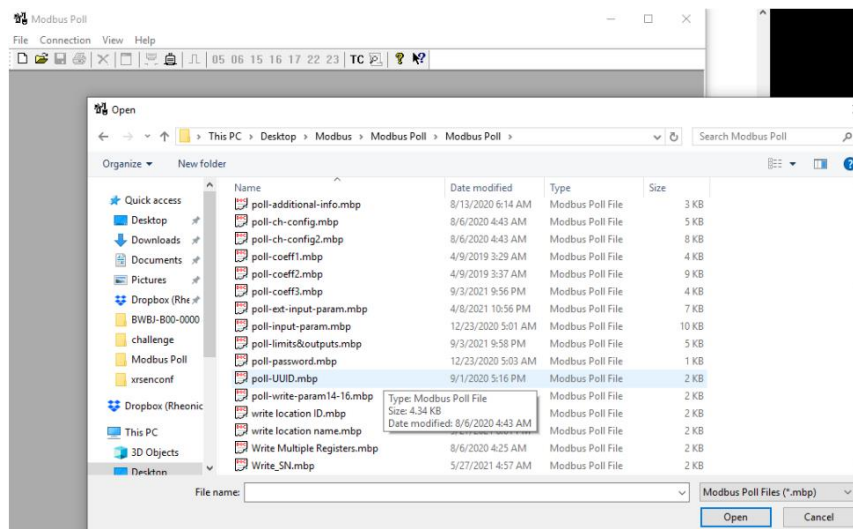


Figure 17: Select a pre-configured file.

6.2.3 The chosen File's Registers will appear with the correct data type and name.

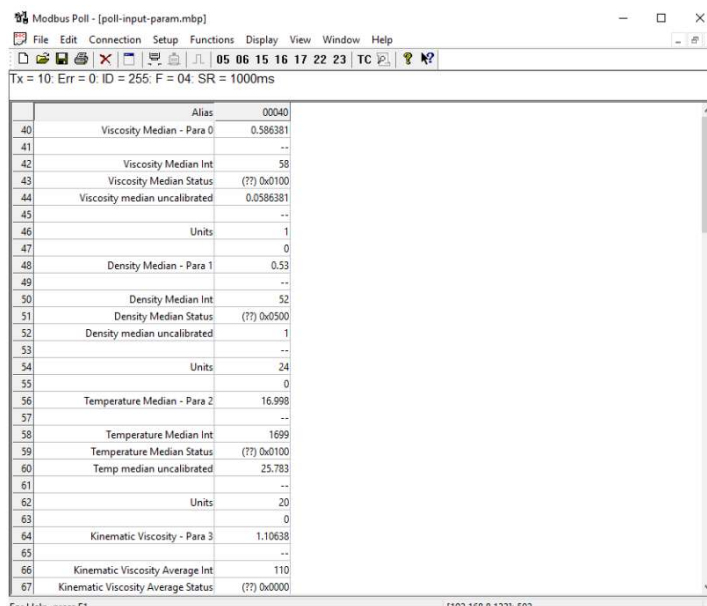


Figure 18: Loaded pre-configured file.

6.3 Density Input through Modbus Poll

In the case of the SRV sensor, the viscosity output (Parameter 0) is the product of dynamic viscosity and density. The default measurement output on the viscosity channel assumes a constant density of 1.0 g/cc. To get the actual dynamic or kinematic viscosity from the SRV, the sensor supports input of a constant or temperature-dependent density. For more information see the [Support Article](#) on Rheonics web page.

With density input, dynamic and kinematic viscosity can be directly output by the sensor over digital and analog channels without the need for any PC.

The steps below show how the user can set the density on the sensor using Modbus RTU to get dynamic or kinematic viscosity directly from the sensor electronics as well as on the RCP software. This configuration can also be done via the RCP software.

6.3.1 Introduction to density input for SRV

The Density SRV output can be described as a polynomial of the form:

$$D_{out} = D_{o5} \cdot T^5 + D_{o4} \cdot T^4 + D_{o3} \cdot T^3 + D_{o2} \cdot T^2 + D_{o1} \cdot T + D_{o0}$$

With this model, we can describe either a constant density or a temperature-dependent density.

a. Constant density input

To describe a constant density, the coefficient D_{o0} should be set equal to the constant density value and all the other coefficients should be set to zero.

b. Temperature-dependent density input

To describe a temperature-dependent density, assign values to the coefficients that reflect the density behavior of your fluid with temperature. If you have a fluid reference table (for example Cannon fluids, Nippon grease), you can fit the density to a polynomial model to get the coefficients.

The equation above expects temperature in °C. If you have data in °F or °K, then convert temperature to the corresponding °C before estimating the coefficients. Once you have the coefficients (D_{o0} , D_{o1} , ...), they can be set into the sensor electronics (SME).

6.3.2 Setting Density Input through Modbus RTU

Over Modbus (either RTU or TCP) there are a set of Registers associated with the density input for the SRV sensors. There are two holding registers associated with each of the D_{o} coefficients of the polynomial. Each pair of Holdings should form a float 32 data type in Big Endian format.

The registers are password protected. This means you should input the user password in the holding register 0 to be able to write the coefficients. The Modbus Function to write to multiple registers is 16 and 03 to read the holding registers.

| Register | Holding Register Address | Data Type |
|----------------------------|-------------------------------------|-----------|
| Password | Address = 0; User Password = 0x5312 | Uint16 |
| D_{o0} | 1212 | Float32 |
| D_{o1} | 1214 | Float32 |
| D_{o2} | 1216 | Float32 |
| D_{o3} | 1218 | Float32 |
| D_{o4} | 1220 | Float32 |
| D_{o5} | 1222 | Float32 |

Table 5: Density Coefficients Registers.

Follow the next steps to configure the Density Input, in this case, using Modbus Poll.

- i. Create a new Definition with the following data.

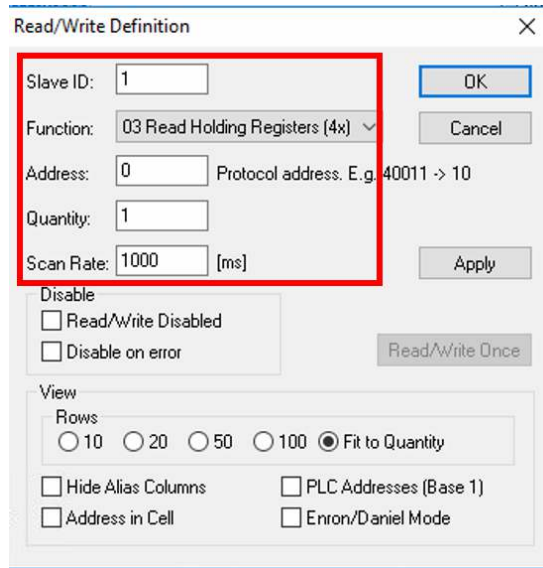


Figure 19: Reading Address 0 from Holding Register.

- ii. The register 0 from the Holding Register will appear, change the format to Hex and write the password as shown in Figure.

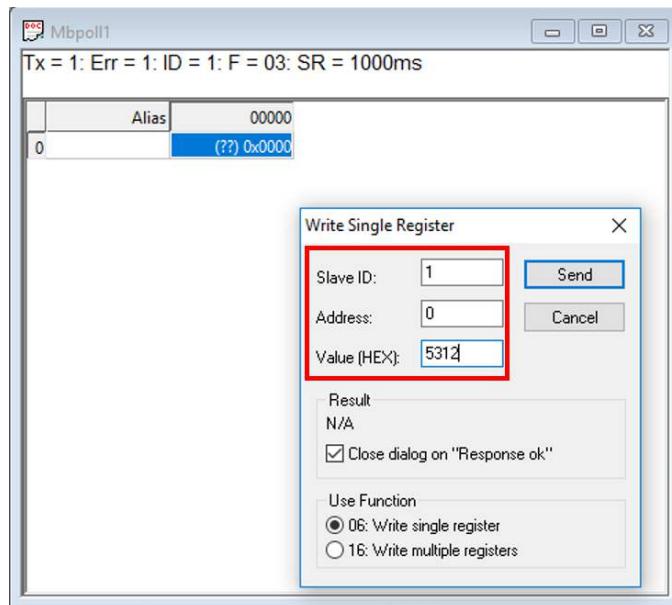


Figure 20: Writing Address 0 from Holding Register.

- iii. Create another Definition to read and write the D_o coefficients.

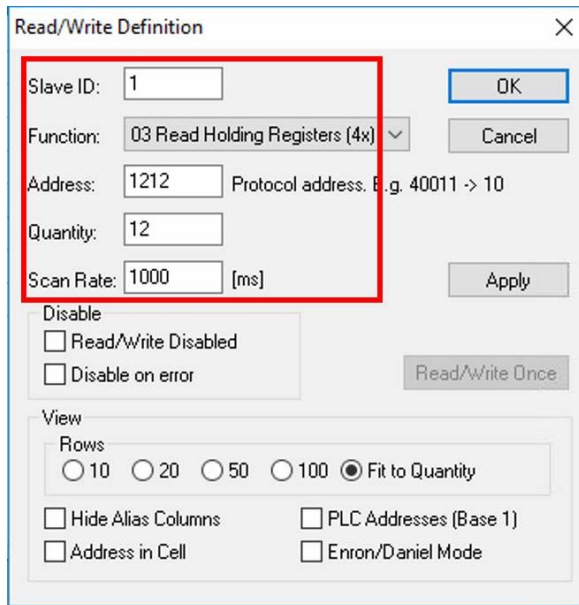


Figure 21: Reading Density Coefficients Registers.

- iv. Configure these registers with a float 32 format. For a constant density set D_{o0} to the desired value and all others to zero.

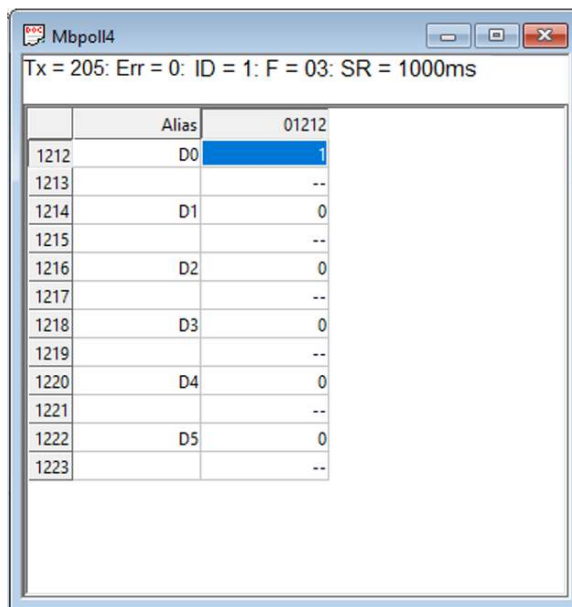
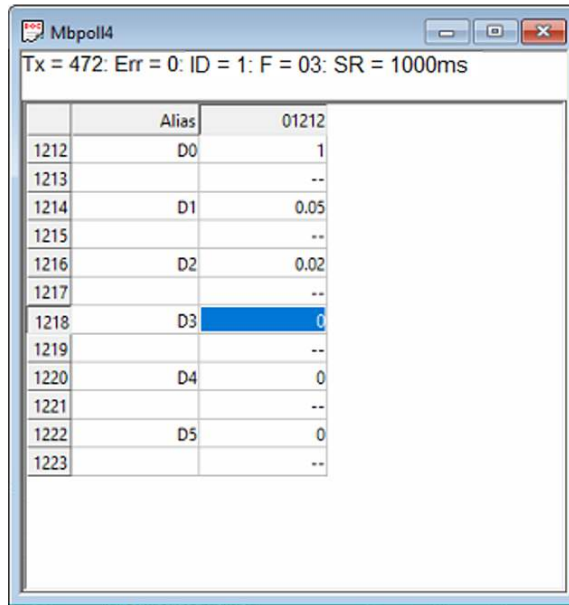


Figure 22: Example for constant density coefficients.

If a temperature-dependent density is needed, configure the other D_o parameters as needed.



| | Alias | 01212 |
|------|-------|-------|
| 1212 | D0 | 1 |
| 1213 | | -- |
| 1214 | D1 | 0.05 |
| 1215 | | -- |
| 1216 | D2 | 0.02 |
| 1217 | | -- |
| 1218 | D3 | 0 |
| 1219 | | -- |
| 1220 | D4 | 0 |
| 1221 | | -- |
| 1222 | D5 | 0 |
| 1223 | | -- |

Figure 23: Example for temperature dependent density coefficients.

7 Modbus RTU Command and Response examples

This section helps with the syntax for Modbus RTU Commands and Responses for the mostly used Rheonics parameters. Modbus RTU syntax was previously discussed in [Section 5](#).

7.1 Reading the parameter 2, Temperature median

This is a float data type and the syntax would be as follows:

- Device Address= 01 = 0x01 (byte).
- Address 40057-40001=56 = 0x0038 (word) (Address in Hex of the first register)
- Functional code= 04=0x0004 (word)-Read input registers
- Number of registers = 02=0x0002 (word)
- CRC = CRC16 (Word) (Little Endian)

Command Structure:

| Device Address | Functional Code | Address of the first register (Hi byte) | Address of the first register (Lo Byte) | Number of registers (Hi byte) | Number of registers (Lo byte) | CRC (Lo Byte) | CRC (Hi Byte) |
|----------------|-----------------|---|---|-------------------------------|-------------------------------|---------------|---------------|
| 01 | 04 | 00 | 38 | 00 | 02 | F0 | 06 |

Response Structure:

| Device Address | Functional Code | Number of Bytes | Value of the first register (Hi Byte) | Value of the first register (Lo Byte) | Value of the second register (Hi Byte) | Value of the second register (Lo Byte) | CRC (Lo Byte) | CRC (Hi Byte) |
|----------------|-----------------|-----------------|---------------------------------------|---------------------------------------|--|--|---------------|---------------|
| 01 | 04 | 04 | 41 | B1 | 5C | 29 | 47 | 41 |

Data = **0x41B15C29** = 22.17 (Float)

Temperature median = 22.17 °C (default unit)

7.2 Reading the parameter 0, Viscosity median Status

Status value is a UInt16 data type and the syntax would be as follows:

- Device Address= 01 = 0x01 (byte).
- Address 40044-40001=43 = 0x002B (word) (Address in Hex of the first register)
- Functional code= 04=0x0004 (word)-Read input registers
- Number of registers = 01=0x0001 (word)
- CRC = CRC16 (Word) (Little Endian)

Command Structure:

| Device Address | Functional Code | Address of the first register (Hi byte) | Address of the first register (Lo Byte) | Number of registers (Hi byte) | Number of registers (Lo byte) | CRC (Lo Byte) | CRC (Hi Byte) |
|----------------|-----------------|---|---|-------------------------------|-------------------------------|---------------|---------------|
| 01 | 04 | 00 | 2B | 00 | 01 | 41 | C2 |

Response Structure:

| Device Address | Functional Code | Number of Bytes | Value of the first register (Hi Byte) | Value of the first register (Lo Byte) | CRC (Lo Byte) | CRC (Hi Byte) |
|----------------|-----------------|-----------------|---------------------------------------|---------------------------------------|---------------|---------------|
| 01 | 04 | 02 | 00 | 01 | 78 | F0 |

Data = **0x0001** = 1 (UInt16) Referring to the parameter status, this value means a general error occurred.

7.3 Reading the Parameter 7, Temperature raw

This is a Int16 data type and the syntax would be as follows:

- Device Address= 01 = 0x01 (byte).
- Address 40099-40001=98 = 0x0062 (word) (Address in Hex of the first register)
- Functional code= 04=0x0004 (word)-Read input registers
- Number of registers = 01=0x0001 (word)
- CRC = CRC16 (Word) (Little Endian)

Command Structure:

| Device Address | Functional Code | Address of the first register (Hi byte) | Address of the first register (Lo Byte) | Number of registers (Hi byte) | Number of registers (Lo byte) | CRC (Lo Byte) | CRC (Hi Byte) |
|----------------|-----------------|---|---|-------------------------------|-------------------------------|---------------|---------------|
| 01 | 04 | 00 | 62 | 00 | 01 | 91 | 14 |

Response Structure:

| Device Address | Functional Code | Number of Bytes | Value of the first register (Hi Byte) | Value of the first register (Lo Byte) | CRC (Lo Byte) | CRC (Hi Byte) |
|----------------|-----------------|-----------------|---------------------------------------|---------------------------------------|---------------|---------------|
| 01 | 04 | 02 | 08 | 9E | 3F | 58 |

Data = **0x089E** = 2206 (Int16)

Since the register value is the temperature scaled by 100. The actual raw temperature value is 22.06 °C.

This register is useful for devices in which single-point conversion requires too much computation.

8 Sensor status and parameter status

All Rheonics sensors (SRV, SRD, DVP, DVM) for inline viscosity and density monitoring have inbuilt status. The status can be read through Modbus with Address 39 (Refer to Table 4: Input Registers.) and can be used over digital communication channels to understand when the sensor is operating correctly and when there is an issue.

8.1 Sensor Error Status

The sensor status can take any of the following values OR a combination of them:

| Bit | Hex | Name | Comment |
|---------------|--------|---------------------------|--|
| Bit 0 | 0x0001 | PLL frequency mismatch | The PLL frequency does not match the sensor frequency. Derived from the ASB string (E10) |
| Bit 1 | 0x0002 | PLL not locked | The PLL is not locked. Derived from the ASB string (E01) |
| Bit 2 | 0x0004 | PLL lock incorrect | The PLL has locked on a wrong frequency. Derived from the ASB string (E02) |
| Bit 3 | 0x0008 | ASB communication error | Issues with sensor electronics |
| Bit 4 | 0x0010 | Temperature sensor failed | The temperature sensor has failed. Derived from the ASB string if temperature is -273.0 |
| Bit 5 | 0x0020 | Sensor too hot | If temperature is above the hardcoded physical temperature limit. |
| Bit 6 | 0x0040 | ASB communication error | Communication issue between two electronics boards in the SME |
| Bit 7 | 0x0080 | Serial Changed | |
| Bit 8 | 0x0100 | Status not clean | Sensor is not clean (only SRV) |
| Bit 9 | 0x0200 | Status in Air | Determines if sensor is in air |
| Bit 10 -15 | Unused | | |

Table 6: Sensor error status bit code and description.

8.2 Parameter Status

The Parameter Status can take any of the following values, OR a combination of these states.

For example: If there is a config error, the status value will be 0x0003

If there is a config error and an internal error, the status will take a value of 0x0023

| Bit | Hex | Name | Comment |
|-------|--------|----------------|--|
| Bit 0 | 0x0001 | General error | This bit is always set in case there is an issue with the parameter. It can be used by the general user or application programmer to alert an issue with that parameter output. For details check the other bits |
| Bit 1 | 0x0002 | Config error | The parameter is not configured or there exists a configuration error. |
| Bit 2 | 0x0004 | Hardware error | The parameter cannot be calculated as the hardware failed. Example: Temperature sensor has failed. |

| | | | |
|--------|--------|----------------------|---|
| Bit 3 | 0x0008 | Dependent error | A parameter source for a dependent parameter is not available. Example: In case of a free formula a referenced parameter is NAN. |
| Bit 4 | 0x0010 | Not ready | No result is yet available. Example: No measurement has been taken yet. The algorithm requires a run-in time |
| Bit 5 | 0x0020 | Internal error | Internal error - Report to Rheonics |
| Bit 6 | 0x0040 | Calibration Error | Diagnostics |
| Bit 7 | 0x0080 | Further use | |
| Bit 8 | 0x0100 | Parameter Calibrated | Triggered when parameter has a calibration/scale factor/coefficient applied to it. |
| Bit 9 | 0x0200 | Model Loaded | Active when a model has been loaded in script parameters. Only valid for parameters 19,20,21 |
| Bit 10 | 0x0400 | Filtering Active | Active when there is a filter loaded for that parameter |
| Bit 11 | 0x0800 | Not stable | Parameter result not yet stable Example: Set for example on viscosity if sensor status is not okay. |
| Bit 12 | 0x1000 | Warning lower | Below lower warning limit (if configured for parameter) |
| Bit 13 | 0x2000 | Warning upper | Above upper warning limit (if configured for parameter) |
| Bit 14 | 0x4000 | Alarm lower | Below alarm limit (Hardcoded depending on parameter type) |
| Bit 15 | 0x8000 | Alarm upper | Above alarm limit (Hardcoded depending on parameter type) |

Table 7: Parameter Status bit code and description.

8.3 How to read sensor status?

Sensor status is a WORD data type, these status bits can be used over digital communication channels to understand when the sensor is operating correctly and when there is an issue.

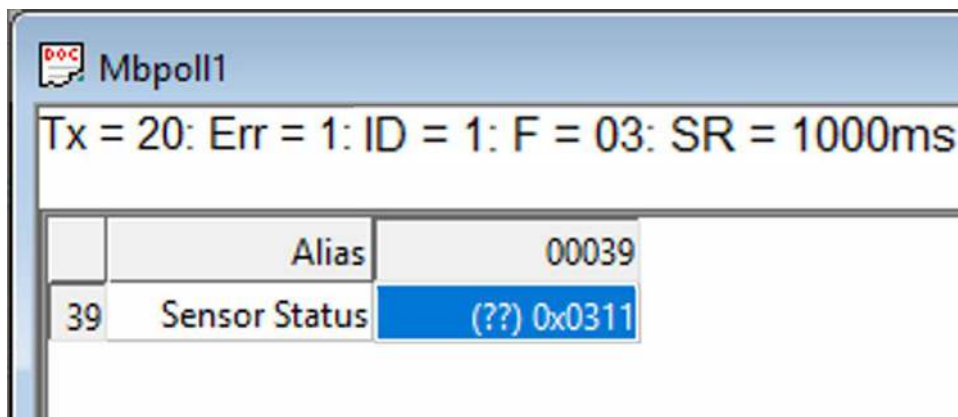


Figure 24: Sensor status example.

In the scenario described in Figure 24, **Sensor Status has a value 0x311**. Sensor error status is the combination of any bit and in this case is the combination (OR) of the following error bits:


| Bit | H | Comments |
|---------------|--------------|--|
| 0 | 0x001 | The PLL frequency does not match the sensor frequency. Derived from the ASB string (E10) |
| 4 | 0x010 | The temperature sensor has failed. Derived from the ASB string if temperature is -273.0 |
| 8 | 0x100 | Sensor is not clean (only SRV) |
| 9 | 0x200 | Status in Air |
| Result | 0x311 | |

Table 4. Sensor status bits for status 0117.

8.4 Which parameters should I read?

Each of the 23 parameters from Rheonics sensor are shown in [Section 5.2](#).

For up-to-date parameters information, please check the page:

| | |
|---|---|
| https://support.rheonics.com/support/solutions/articles/81000393235-parameter-list-access-for-field-devices |  |
|---|---|

| | |
|---|---|
| https://support.rheonics.com/support/solutions/articles/81000393237-units-translation-table-for-field-devices |  |
|---|---|

Customers normally choose Parameters 12 (Viscosity Median and last good), 13 (Density Median and last good), 2 (Temperature Median) and sensor status for their process as they provide good information of the fluid.

For custom parameters like concentration, compensated viscosity, and density; parameters 19, 20, and 21 can be used. **Contact Rheonics support for more information about mathematical models that the sensors support natively.**

9 Units table


| Unit Index | Unit Display | Unit Index | Unit Display |
|------------|--------------------------------|------------|--------------------|
| 0 | | 35 | °Baumé |
| 1 | mPa.s | 36 | °Brix |
| 2 | cP | 37 | %wt/v |
| 3 | Pa.s | 38 | %v/v |
| 4 | Poise | 39 | %vol |
| 5 | Reyn | 40 | Bar |
| 6 | mm ² /s | 41 | psi |
| 7 | cSt | 42 | m ³ /s |
| 8 | St | 43 | sccm |
| 9 | m ² /s | 44 | gpm |
| 10 | in ² /s | 45 | pH |
| 11 | SUS | 46 | m ³ |
| 12 | VI | 47 | gal |
| 13 | AV | 48 | STP |
| 14 | PV | 49 | Tref |
| 15 | YP | 50 | n _D |
| 16 | sec | 51 | %wt |
| 17 | μ | 52 | %Vol |
| 18 | η | 53 | mol/m ³ |
| 19 | v | 54 | alcohol |
| 20 | °C | 55 | ethanol |
| 21 | °F | 56 | Hz |
| 22 | °K | 57 | rhe |
| 23 | ref _{xx} ^γ | 58 | °P |
| 24 | g/cc | | |
| 25 | Kg/m ³ | | |
| 26 | lb/ft ³ | | |
| 27 | lbm/gal | | |
| 28 | lbs/gal | | |
| 29 | ppg | | |
| 30 | pptf | | |
| 31 | slug/ft ³ | | |
| 32 | SG | | |
| 33 | ρ | | |
| 34 | °API | | |

Table 7. Units translation table.


10 Troubleshooting

| Electronics Issues | |
|---|---|
| No LED lights, display, or output signals | <ol style="list-style-type: none"> 1. Check power supply and cabling |
| Viscosity is not stable | <ol style="list-style-type: none"> 1. SRV viscosity output is compared against only dynamic viscosity of the calibration fluids. 2. Sensor is not fully in fluid. |
| Probe Issues | |
| NaN values are displayed on the RCP software or PLC | <ol style="list-style-type: none"> 1. Check wiring from the probe to SME is correct. 2. Measure the internal resistance from the probe. |
| Communication Issues | |
| No data is visible | <ol style="list-style-type: none"> 1. Try to reset the SME and/or restart your master. 2. Verify all Modbus parameters (address, baud rate and parity). 3. Verify there is no duplicated address in the network. 4. Try to use the ping command to get a response from either the PLC or SME. 5. Consider Modbus RTU protocol should be enabled when ordering. Contact Rheonics Support if enabling is needed. |
| Data issues | |
| Measured value is different from my reference standards | <ol style="list-style-type: none"> 1. SRV viscosity output is compared against only dynamic viscosity of the calibration fluids |
| Wrong Unit from the data stream | <ol style="list-style-type: none"> 1. Unit is received as HEX representation, convert to decimal |
| Wrong readings for each parameter | <ol style="list-style-type: none"> 1. Verify each parameter is correctly mapped, some bytes are swapped 2. Make sure biasing and termination resistors are correctly installed. 3. Try swapping wire A and B from RS485 lines. |
| Sensor Status does not match any bit from the sensor status table | <ol style="list-style-type: none"> 1. Sensor status byte is swapped |
| CRC Errors | <ol style="list-style-type: none"> 1. Check baud rate and character format. 2. Check wiring. 3. Try swapping wire A and B from |

| | |
|--|--|
| | RS485 lines. 4. Check that Pre-Delay is at least 50 mS. |
|--|--|

| | |
|---|---|
| https://support.rheonics.com/support/solutions/articles/81000397717-modbus-rtu-troubleshooting |  |
|---|---|

- Contact RHEONICS support desk
 - support@rheonics.com
 - <https://support.rheonics.com>

| | |
|---|---|
| https://support.rheonics.com/support/home |  |
|---|---|

11 Reviews and approvals

| Version | Nature of changes | Approval | FW version | Date |
|---------|-------------------|-----------|------------|------------|
| 1 | Original | C. Arroyo | 3.30/15 | 16.02.2024 |

12 Notes/Errata

Contact Rheonics support for customization of system settings.

Notes

EUROPE HEADQUARTERS

Rheonics GmbH

Klosterstrasse. 19
8406 Winterthur Switzerland
Tel: +41 52 511 32 00

US HEADQUARTERS

Rheonics, Inc.

3 Sugar Creek Center Blvd, Ste 100
Sugar Land, TX 77478
United States of America
Tel: +1 713 364 5427

Website: <https://rheonics.com>
Support Portal: <https://support.rheonics.com>
Support E-Mail: support@rheonics.com
Sales E-Mail: info@rheonics.com

©2023 Rheonics, Inc. All rights reserved.
The Rheonics logo is a trademark and service mark of Rheonics Inc. Photographs, excerpts, trademarks, and logos are not endorsements and are used only for representation under fair use copyright laws. All rights stay with the respective owners.

