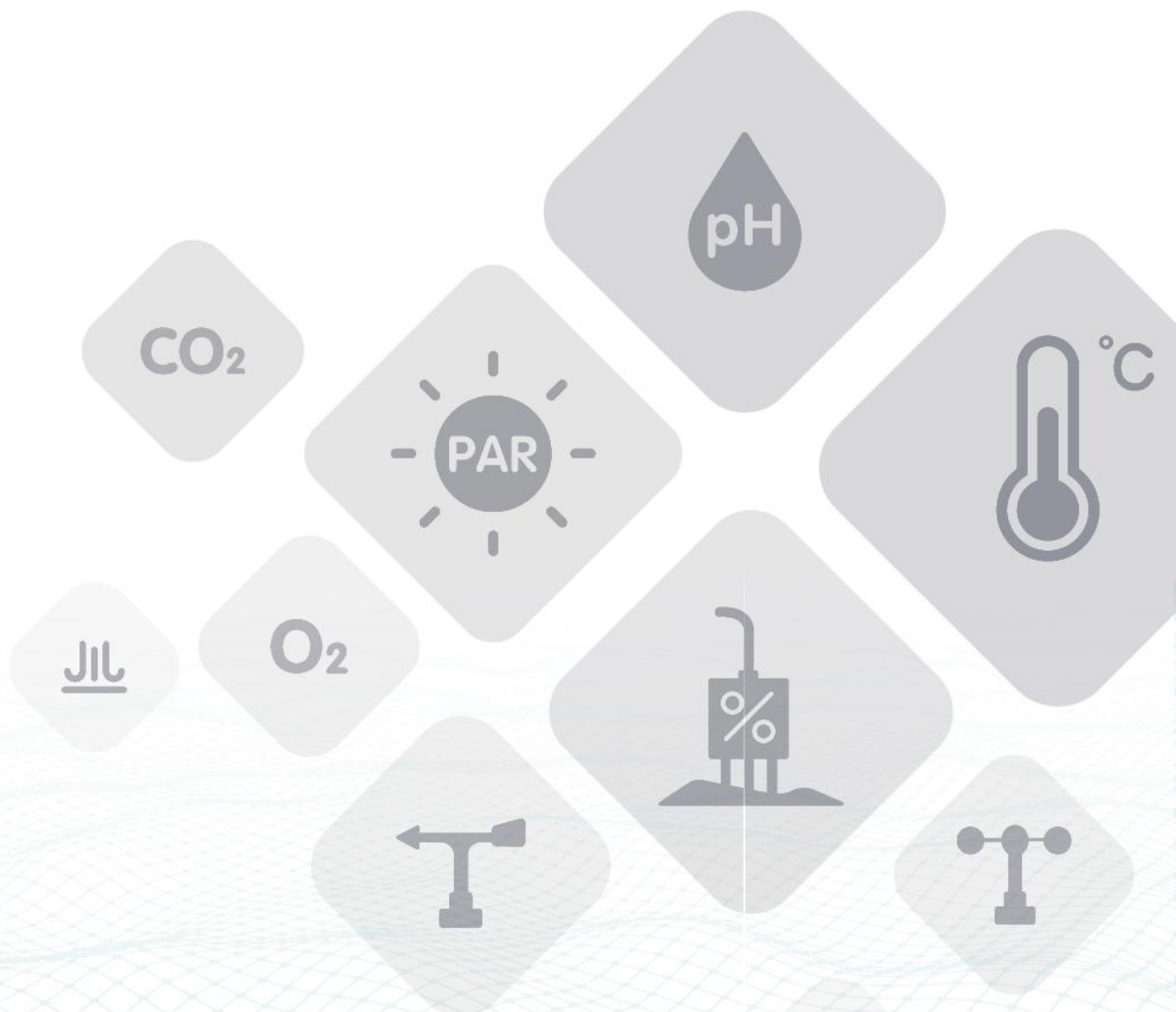




SENSECAP

# Leaf Wetness and Temperature Sensor User Guide

Version: V1.0



## Content

1. Introduction .....	3
2. Wiring diagrams .....	4
3. Dimension .....	5
4. Safety ,Care and Installation .....	6
5. Output Signal Conversion .....	7
6. RS485 Modbus Protocol .....	8
6.1 Modbus Protocol .....	8
6.2 Modbus Register .....	9
6.3 Modbus Register Detail Description .....	11
6.4 Modbus Function Code .....	14
6.4.1 Function Code 3 Protocol Example .....	14
6.4.2 Function Code 4 Protocol Example .....	15
6.4.3 Function Code 6 Protocol Example .....	16
6.4.4 Function Code 16 Protocol Example .....	17
7. Software Configuration Utility .....	19

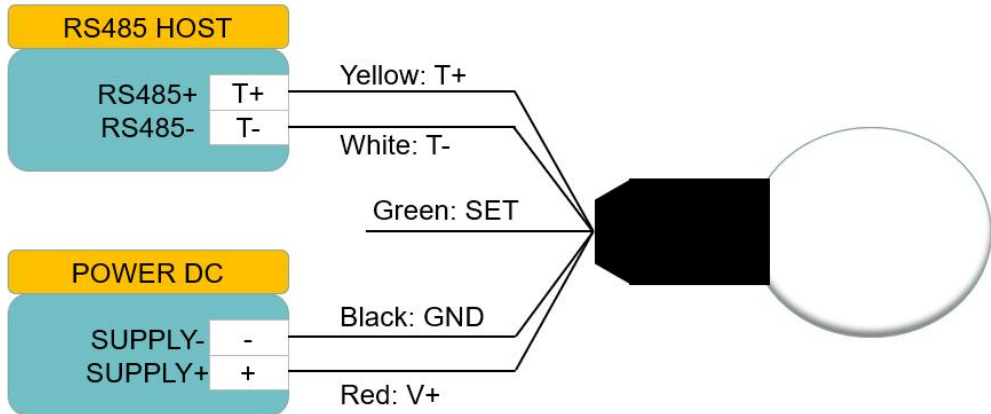
# 1. Introduction

S-YM-01 measures leaf wetness and temperature. It sealed with resin packaged plastic body can be used for long time monitor. Wetness measurement value is proportional to the percentage of sensing area covered with water. The output signal can be RS485, Analog Voltage or Analog Current. The sensor is applicable for science research, bacteria prevention, foliage dressing spraying, greenhouse, smart agriculture etc.

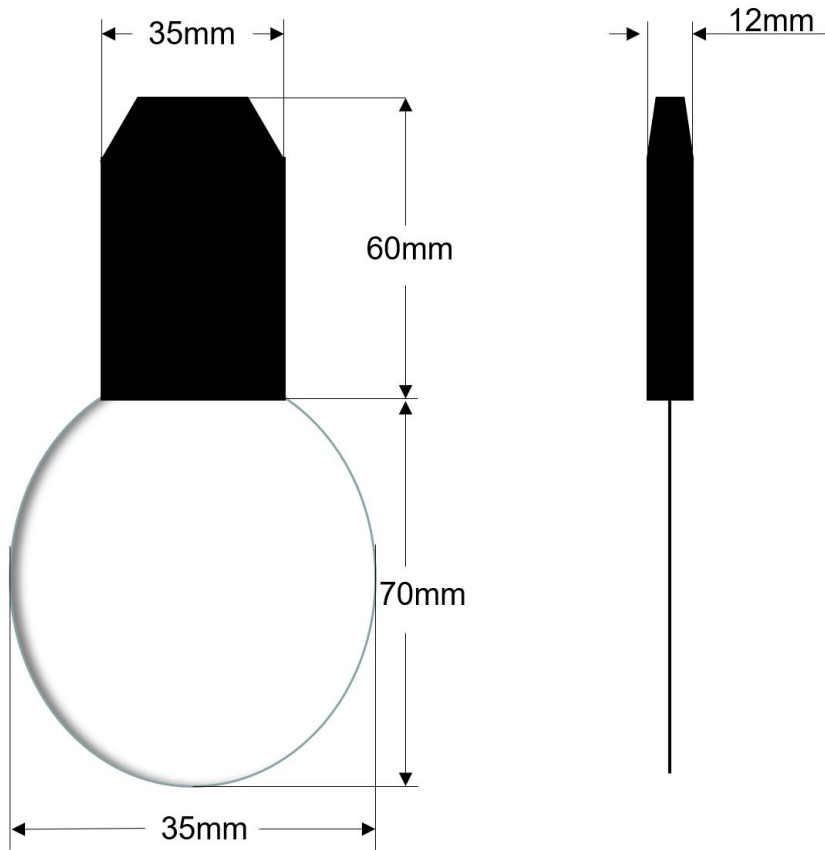
- Integrated with leaf wetness and temperature measurement
- Output Interface with RS485, Voltage, Current
- Mimics the real leaf thermal resistance by sensing shape and thickness
- Water, ice, foggy detection
- Water proof to IP68 ratings
- High accuracy with excellent stability
- Reverse power protection and Built-in TVS/ESD protection

<b>Specifications</b>	
<b>Output Interface</b>	RS485 Modbus-RTU
<b>Power Supply</b>	3.6-30V/DC
<b>Power Consumption</b>	6mA@24V DC
<b>Leaf wetness Measurement</b>	Range: 0-100%, Resolution: 0.1%, Accuracy:5%
<b>Temperature Compensation</b>	Range: -40~80°C, Resolution:0.1°C, Accuracy: ±0.5°C
<b>Measurement Technique</b>	FDR
<b>IP Ratings</b>	IP68
<b>Operating Temperature</b>	-40~85°C
<b>Sensor Sealed</b>	Epoxy resin
<b>Installation</b>	Mounting kit
<b>Cable Length</b>	2 meters
<b>Dimension</b>	65*13*145mm

## 2. Wiring diagrams

Type	Wiring diagram
<b>RS485 Modbus</b>	<p data-bbox="347 412 1423 813">                     Red (V+): Power Supply +                      Black (G): Power Supply -                      Yellow (T+): RS485+/A/T+                      White (T-): RS485-/B/T-                      Green (SET): SETTING mode. When sensor power-up with the SET wire connected to Power Supply +, then sensor using setting mode communication parameters for RS485. When sensor power-up with the SET wire connected to Power Supply - or unconnected, the sensor using communication parameters in register for RS485. Please refer to the usage below.                 </p>  <p data-bbox="347 1323 1406 1473">                     ALL RS485 communication parameters (Modbus Slave Address, baud rate, parity, data bits, stop bits ) are set in internal register and can be saved when power down, the factory setting is ADDRESS=1, BAUDRATE=9600bps, PARITY=NONE, DATABITS=8bits, STOPBITS=1bit;                 </p> <p data-bbox="347 1489 1382 1809">                     Sometimes you may FORGET the communication settings, In this case, you can connect the GREEN &amp; RED wire together to Power Supply+, black wire to Power Supply-, then re-power up the sensor, then the sensor start-up with a fixed communication settings (we call it setting mode) ADDRESS=0, BAUDRATE=9600bps, PARITY=NONE, DATABITS=8bits, STOPBITS=1bit; Communicate with the sensor using this parameters and then set your desired communication parameters, then disconnect the green wire from Power Supply+, then re-power up the sensor, and the sensor will communicate with your settings.                 </p>

### 3. Dimension



## 4. Safety ,Care and Installation

Sensor can be mounted on the branch of plants or on the weather station. Please keep the sensing area on the upside and form an angle with ground.

For foliage dressing spraying, use multiple sensor at different level of plant to guarantee the spraying effect.

Seeed Technology Co., Ltd. Authorised

## 5. Output Signal Conversion

Output Interface	Parameters Range	Conversion Formula
<b>RS485 Modbus-RTU</b>	Temperature: -40-80°C	TEMP=(REGISTER VALUE)/100. When REGISTER VALUE=2013, then TEMP= 2013/100=20.13°C.
	Wetness: 0-100%	WETNESS =(REGISTER VALUE)/100. When REGISTER VALUE=2013, then WETNESS = 2013/100=20.13%.

Seeed Technology Co., Ltd. Authorised

## 6. RS485 Modbus Protocol

### 6.1 Modbus Protocol

Modbus Protocol is widely used to establish master-slave communication between intelligent devices or sensors. A MODBUS message sent from a master to a slave contains the address of the slave, the function code (e.g. 'read register' or 'write register'), the data, and a check sum (LRC or CRC).

The sensor is RS485 interface with Modbus protocol. The default serial communication settings is slave address 1, Modbus RTU, 9600bps, 8 data bits and 1 stop bit. All communication settings can be changed with Modbus command, and take effective after re-power up the sensor.

Following Modbus function code are supported by sensor.

Modbus Function Code 0x03 : used for reading holding register.

Modbus Function Code 0x04 : used for reading input register.

Modbus Function Code 0x06 : used for writing single holding register.

Modbus Function Code 0x10: used for writing multiple holding register.



## 6.2 Modbus Register

Parameters	Register Addr. (HEX/DEC)	Data Type	Modbus Function Code(DEC)	Range and Comments	Default Value
TEMPERATURE	0x0000 /0	INT16 RO	3/4	-4000-8000 for -40.00~80.00°C.	N/A
WETNESS	0x0001 /1	UINT16 RO	3/4	0-10000 for 0-100%	N/A
TEMPUNIT	0x0020 /32	UINT16 R/W	3/6/16	0: °C 1: °F	0
TEMPCALIB	0x0021 /33	INT16 R/W	3/6/16	-999-999 for -9.99~9.99°C	0
SLAVEADDRESS	0x0200 /512	UINT16 R/W	3/6/16	0-255	1
BAUDRATE	0x0201 /513	UINT16 R/W	3/6/16	0-6 0:1200bps 1:2400bps 2:4800bps 3:9600bps 4:19200bps 5:38400bps	3:9600bps
PROTOCOL	0x0202 /514	UINT16 R/W	3/6/16	0-1 0: Modbus RTU 1: Modbus ASCII	0: Modbus RTU
PARITY	0x0203 /515	UINT16 R/W	3/6/16	0-2 0:None 1:Even 2:Odd	0:None Parity
DATABITS	0x0204 /516	UINT16 R/W	3/6/16	1 1:8 databits	1:8 databits
STOPBITS	0x0205 /517	UINT16 R/W	3/6/16	0-1 0:1 stopbit 1:2 stopbits	0:1 stopbit
RESPONSEDELAY	0x0206 /518	UINT16 R/W	3/6/16	0-255 for 0-2550 milliseconds	0
ACTIVEOUTPUTINTERVAL	0x0207 /519	UINT16 R/W	3/6/16	0-255 for 0-255 seconds.	0

NOTE: UINT16:16 bit unsigned integer, INT16:16bit signed integer

NOTE: RO: Register is Read Only, R/W: Register is Read/Write

NOTE: HEX is Hexadecimal (data with 0x/0X prefix), DEC is Decimal

Seeed Technology Co., Ltd. Authorised

## 6.3 Modbus Register Detail Description

<b>TEMPERATURE</b>		
Data Range	-4000-8000 For -40.00~80.00°C	Default: N/A
Power Down	N/A	
Save		

Note: Temperature value (Binary complement).

Example: When REGISTER = 0x0702 (HEX format), then  $VALUE = (0x07 * 256 + 0x02) / 100 = 17.94^{\circ}C$ . When REGISTER = FF05H (HEX format), then  $VALUE = ((0xFF * 256 + 0x05) - 0xFFFF - 0x01) / 100 = (0xFF05 - 0xFFFF - 0x01) / 100 = -2.51^{\circ}C$ .

<b>WETNESS</b>		
Data Range	0-10000 For 0-100%	Default: N/A
Power Down	N/A	
Save		

Note: Leaf wetness measurement value, is proportional to the percentage of sensing area covered with water.

Example: When REGISTER = 0x0702 (HEX format), then  $VALUE = (0x07 * 256 + 0x02) / 100 = 17.94\%$

<b>TEMPUNIT</b>		
Data Range	0:°C 1:°F	Default: 0
Power Down	YES	
Save		

Note: Temperature Unit

<b>TEMPCALIB</b>		
Data Range	-999-999 for -9.99~9.99°C	Default: 0
Power Down	YES	
Save		

Note: Temperature Calibration Value

<b>SLAVEADDRESS --- Modbus Slave Address</b>		
Data Range	0-255	Default: 1
Power Down	YES	
Save		

Note: Please re-power on the sensor to take effective after set.

**BAUDRATE --- Serial Comm Baudrate**

Data Range	<b>0-5</b> <b>0:</b> 1200bps <b>1:</b> 2400bps <b>2:</b> 4800bps <b>3:</b> 9600bps <b>4:</b> 19200bps <b>5:</b> 38400bps	Default: 3
Power Down Save	YES	

Note: Please re-power on the sensor to take effective after set.

**PROTOCOL --- Serial Comm Protocol**

Data Range	0-1 0:Modbus RTU 1:Modbus ASCii	Default: 0
Power Down Save	YES	

Note: Please re-power on the sensor to take effective after set.

**PARITY --- Serial Comm Parity**

Data Range	0-2 0:NONE 1:EVEN 2:ODD	Default: 0
Power Down Save	YES	

Note: Please re-power on the sensor to take effective after set.

**DATABITS --- Serial Comm Databits**

Data Range	1 1:8 databits	Default: 1
Power Down Save	YES	

Note: Please re-power on the sensor to take effective after set.

<b>STOPBITS --- Serial Comm Stopbits</b>		
Data Range	0-1 0:1 stopbit 1:2 stopbits	Default: 0
Power Down Save	YES	

Note: Please re-power on the sensor to take effective after set.

<b>RESPONSEDELAY --- Serial Comm Response Delay</b>		
Data Range	0-255 for 0-2550 milliseconds, 0 for disabled	Default: 0
Power Down Save	YES	

Note: Please re-power on the sensor to take effective after set.

Note: Sensor will delay a period before response to master request command.

Example: When set to 5 and receive a request from master device, then sensor will delay  $5 \times 10\text{ms} = 50\text{ms}$ , then response to master.

<b>ACTIVEOUTPUTINTERVAL --- Serial Comm Active Output Interval time</b>		
Data Range	0-255 for 0-255 seconds, 0 for disabled	Default: 0
Power Down Save	YES	

Note: Please re-power on the sensor to take effective after set.

Note: Sensor will output the data actively without any master request command.

Note: Only ONE sensor should be on RS485 network, or there will be data collision and corrupt the data on line.

Note: Refer to SETTING mode to exit the Active Output Mode.

Example: When set to 5 then sensor will output the data every 5 seconds without any master request command.

## 6.4 Modbus Function Code

For description below, data started with 0X/0x means that it's in HEX format.

### 6.4.1 Function Code 3 Protocol Example

#### Master Request: AA 03 RRRR NNNN CCCC

AA	1 byte	Slave Address,0-255
0x03	1 byte	Function Code 3
RRRR	2 byte	Starting Register Addr
NNNN	2 byte	Quantity of Register to read
CCCC	2 byte	CRC CHECKSUM

#### Slave Response: AA 03 MM VV0 VV1 VV2 VV3... CCCC

AA	1 byte	Slave Address,0-255
0x03	1 byte	Function Code 3
MM	1 byte	Register Data Byte Count
VV0,VV1	2 byte	Register Value (High8bits first)
VV2,VV3	2 byte	Register Value (High8bits first)
...	...	Register Value (High8bits first)
CCCC	2 byte	CRC CHECKSUM

**Example: Read register 0x0200-0x0201,that is slave address and baudrate.**

#### Master Request:01 03 0200 0002 C5B3

Slave Addr.	1 byte	0x01
Function Code	1 byte	0x03
Starting Register Addr.	2 byte	0x0200
Quantity of Register to read	2 byte	0x0002
Checksum	2 byte	0xC5B3

**Slave Response:01 03 04 00 01 00 03 EB F2**

Slave Addr.	1 byte	0x01
Function Code	1 byte	0x03
Register Data Byte Count	1 byte	0x04
Register Value:	2 byte	0x00(HIGH 8 Bits)
Address		0x01(LOW8 Bits)
Register Value:	2 byte	0x00(HIGH 8 Bits)
Baud rate		0x03(LOW8 Bits)
Checksum	2 byte	0xEBF2

### 6.4.2 Function Code 4 Protocol Example

**Master Request: AA 04 RRRR NNNN CCCC**

AA	1 byte	Slave Address,0-255
0x04	1 byte	Function Code 4
RRRR	2 byte	Starting Register Addr
NNNN	2 byte	Quantity of Register to read
CCCC	2 byte	CRC CHECKSUM

**Slave Response: AA 04 MM VV0 VV1 VV2 VV3... CCCC**

AA	1 byte	Slave Address,0-255
0x04	1 byte	Function Code 4
MM	1 byte	Register Data Byte Count
VV0,VV1	2 byte	Register Value (High8bits first)
VV2,VV3	2 byte	Register Value (High8bits first)
...	...	Register Value (High8bits first)
<b>CCCC</b>	2 byte	CRC CHECKSUM

**Example: Read register 0x0000-0x0001, that is temperature, wetness.**

**Master Request: 01 04 0000 0002 71CB**

Slave Addr.	1 byte	0x01
Function Code	1 byte	0x04
Starting Register Addr.	2 byte	0x0000
Quantity of Register to read	2 byte	0x0002
Checksum	2 byte	0x71CB

**Slave Response: 01 04 04 07 E0 0F 01 09 23 F5 AF**

Slave Addr.	1 byte	0x01
Function Code	1 byte	0x04
Register Data Byte Count	1 byte	0x04
Register Value: Temperature	2 byte	0x07(HIGH 8 Bits) 0xE0(LOW8 Bits)
Register Value: Soil Moisture	2 byte	0x0F(HIGH 8 Bits) 0x01(LOW8 Bits)
Checksum	2 byte	0x3F36

Temperature =  $(0x07 * 256 + 0x E0) / 100 = 2016 / 100 = 20.16 \text{ } ^\circ\text{C}$

Wetness =  $(0x0F * 256 + 0x01) / 100 = 3841 / 100 = 38.41 \%$

### 6.4.3 Function Code 6 Protocol Example

**Master Request: AA 06 RRRR VVV CCCC**

AA	1 byte	Slave Address, 0-255
0x06	1 byte	Function Code 6
RRRR	2 byte	Register Addr (High8bits first)
VVV	2 byte	Register Value (High8bits first)
CCCC	2 byte	CRC CHECKSUM



**Slave Response: AA 06 RRRR VVV CCCC**

AA	1 byte	Slave Address,0-255
0x06	1 byte	Function Code 6
RRRR	2 byte	Register Addr (High8bits first)
VVV	2 byte	Register Value (High8bits first)
CCCC	2 byte	CRC CHECKSUM

**Example: Write Register 0x0200,that is change Modbus slave address to 2.**

**Master Request: 01 06 0200 0002 09B3**

Slave Addr.	1 byte	0x01
Function Code	1 byte	0x06
Register Addr.	2 byte	0x0200
Register Value	2 byte	0x0002
Checksum	2 byte	0x09B3

**Slave Response: 01 06 0200 0002 09B3**

Slave Addr.	1 byte	0x01
Function Code	1 byte	0x06
Register Addr.	2 byte	0x0200
Register Value	2 byte	0x0002
Checksum	2 byte	0x09B3

### 6.4.4 Function Code 16 Protocol Example

**Master Request: AA 10 RRRR NNNN MM VVV1 VVV2 ...CCCC**

AA	1 byte	Slave Address,0-255
0x10	1 byte	Function Code 0x10
RRRR	2 byte	Starting Register Addr
NNNN	2 byte	Quantity of Register to write
MM	1 byte	Register Data Byte Count
VVV1	2 byte	Register Value(High8bits first)
VVV2	2 byte	Register Value(High8bits first)
...	...	Register Value(High8bits first)
CCCC	2 byte	CRC CHECKSUM

**Slave Response: AA 10 RRRR NNNN CCCC**

AA	1 byte	Slave Address,0-255
0x10	1 byte	Function Code 0x10
RRRR	2 byte	Starting Register Addr
NNNN	2 byte	Quantity of Register to write
CCCC	2 byte	CRC CHECKSUM

**Example: Write Register 0x0200-0x0201,that is set slave address to 1,and baudrate to 19200bp.**

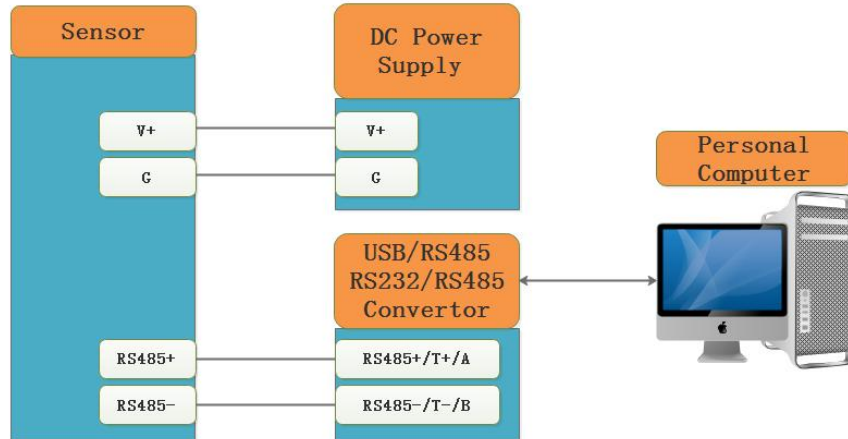
**Master Request:01 10 0200 0002 04 0001 0004 BACC**

0x01	1 byte	Slave Addr.
0x10(HEX)	1 byte	Function Code 0x10
0x0200	2 byte	Starting Register Addr
0x0002	2 byte	Quantity of Register to write
0x04	1 byte	Register Data Byte Count
0x0001	2 byte	Register Value: Slave Address 1
0x0004	2 byte	Register Value: Baud rate 19200bps
0xBACC	2 byte	CRC CHECKSUM

**Salve Response:01 10 0200 0002 4070**

0x01	1 byte	Slave Addr.
0x10(HEX)	1 byte	Function Code 0x10
0x0200	2 byte	Starting Register Addr(High8bits first)
0x0002	2 byte	Quantity of Register to write(High8bits first)
0x4070	2 byte	CRC CHECKSUM

## 7. Software Configuration Utility



You can use software listed below to try reading/writing the register of sensor,  
<https://github.com/ed-chemnitz/qmodbus/releases>