MODBUS USER INSTRUCTION

For Sensorex Smart Sensors

JUNE 16, 2021

SENSOREX CORPORATION
11751 Markon Drive, Garden Grove, CA, 92841
www.sensorex.com

Thank you for choosing Sensorex Smart Sensor products. This instruction manual is provided to guide the user in the use of the smart sensor products. It covers all aspects of programming and communication for the sensors. If any question not covered in this document arise, please contact your Sensorex supplier or support@sensorex.com

QUICK START INSTRUCTIONS:

Sensor Wiring:

Wire Color	Function
Red	V+ (7-40VDC in Modbus mode)
Black	V-
White	Modbus A
Green	Modbus B

Smart Sensor Remote Electronics Wiring:
See Smart Sensor remote electronics manual.

Sensor Mdbus Defaults:

Baud Rate: 19200
Framing: 8N1
Slave ID: 240

Voltage & Power Requirements:

Sensorex Smart sensors with Modbus 485 output can be powered with 7-40V DC
In Modbus mode, the minimum power requirements are:

1. pH <90mw
2. ORP <90mW
3. EC(Conductivity) <50mW
4. DO <90mW
5. Toroidal <75mW
6. FCL <90mW

Start waiting time is >10 seconds**(see pg 3)

ABOUT THIS DOCUMENT:

Rev Level	Date	Notes
1.0	6/14/2021	AZ/SE- 1st release
1.1	6/24/2021	SE- Added startup wait note to 1. DEVICE ADDRESS. Page 3

Table of Contents

Sensorex Smart Sensor Modbus User Instruction

Modbus is a simple and robust standard communication protocol. Sensorex Smart Sensor is designed to function as a Modbus Slave on a multi-drop RS485 serial connection using the Modbus RTU (remote terminal unit) protocol. Please refer to *Modbus Application Protocol Specification V1.1b3 (April 26, 2012)* for more detailed about each protocol specification. The Remote Terminal Unit serial protocol is described in *MODBUS over Serial Line, Specification and Implementation Guide, V1.02.*

Each sensor has four wires: two for RS485 (white is A, green is B) and two for power (red is V+, Black is V-). When in the Modbus mode, a Modbus master devise can read and configure the sensor. Since RS-485 uses half-duplex communication, master switches to transmitting state to send a command then switches to receiver state and waits for a response. The waiting time for master to receive a response should be at least 200 milliseconds for the slave to process the command and send out the response.

It is important to remember that all setup changes requires a power (on then off) cycle or soft reset to accept any changes.

Sensorex Smart Sensors are capable of reading and writing in the 4-20mA loop current mode. Any changes to the 4-20mA scale require the Modbus communication interface.

1. Device Address

The device address in register 0 is also called slave ID. Valid device addresses are 1-247 decimal. Factory default is 240. Slave ID 0 is reserved for broadcast mode and will not supported at this time. ****Note: All smart sensors have a default setup at the startup window time. The user must wait at least 10 seconds at the power up time before starting Modbus communication for slave ID's other than 240, or multiple drops connections.***

2. Baud Rate

The Baud Rate in register 1 is selectable via the Modbus Communications Interface. The selectable baud rates are 9600, 19.2k or 38.4K bits per second. Value 9 is 9600, 19 is 19.2K and 38 is 38.4K. The default baud rate is 19.2K.

Baud Rate	Value	Access
9600	9	Read/Write
19,200	19	Read/Write
38,400	38	Read/Write

3. Data Format

The Data Format in register 2 is selectable via the Modbus Communications Interface. The selectable data formats are as follows. The default value is 0 that means 8-N-1 format.

Data Bits	Parity	Stop Bit	Format	Register Value
8	None	1	8-N-1	0

8	Even	1	8-E-1	1
8	Odd	1	8-O-1	2
8	None	2	8-N-2	3

Important note: The default communication setup Slave ID=240, Baud rate 19.2K bit/second, 8 data bits, no parity and one stop bit. In case the configuration set by user is forgotten and communication is not working, your smart sensor has a startup window to rebuild the communication with the default setup upon power up.

4. Read Register (Function Code 03/0x03)

4.1 Modbus Read Query Message

Byte	Modbus	Range	Referenced to Sensor
1st	Slave Address	1-247* (Decimal)	Sensor ID (Address)
2nd	Function Code	03	Function Code
3rd	Starting Register Hi*	00-FF (Hex)	Currently Not Used (00)
4th	Starting Register Lo*	00-FF (Hex)	Commands
5th	No. of Registers Hi	00	Currently Not Used (00)
6th	No. of Registers Lo	01	No. of 16-Bit Registers
7th	CRC Lo	00-FF (Hex)	CRC Lo Byte
8th	CRC Hi	00-FF (Hex)	CRC Hi Byte

* Note: Start register can be a maximum of 9999 Address Locations (0000-270E)

Example 1- read six registers (3 float point data) from address 03 for a pH sensor (Slave ID 240).

Command:	**Read probe value, Temperature, probe alternate value**	
byte index	**byte description**	**Example (Hex)**
0	Slave ID	F0
1	Function Code	3
2	Reg_Addr_H	0
3	Reg_Addr_L	3
4	Number of regester_H	0
5	Number of regester_L	6
6	CRC16_L	20
7	CRC16_H	E9

4.2 Modbus Read Response Message

Byte	Modbus	Range	Referenced to Sensor
1st	Slave Address	1-247* (Decimal)	Sensor ID (Address)
2nd	Function Code	03	Read Holding Registers
3rd	Byte Count	02	No. of Data Bytes
4th	Data Hi	00-FF (Hex)	Hi Byte Status Data

5th	Data Lo	00-FF (Hex)	Lo Byte Status Data
6th	CRC Lo	00-FF (Hex)	CRC Lo Byte
7th	CRC Hi	00-FF (Hex)	CRC Hi Byte

Example 2 –Return to request of example 1.

Response:	pH=10.37, Temp.=24.67, mV=-235.65		
byte index	byte description	Example(Hex)	
0	Slave ID	F0	
1	Function Code	3	
2	byte count	C	
3	pH_HH	41	
4	pH_HL	25	
5	pH_LH	FF	
6	pH_LL	55	
7	Temp_HH	41	
8	Temp_HL	C5	
9	Temp_LH	57	
10	Temp_LL	60	
11	mV_HH	C3	
12	mV_HL	6B	
13	mV_LH	A7	
14	mV_LL	72	
15	CRC16_L	78	
16	CRC16_H	F6	

5. Write Single Register (Function Code 06 / 0x06)

5.1 Unlock Write Protection

Sensorex Smart sensor has write protection feature. It is required to use a password to access the register value change. An unlock command must be carried out before writing to single register. The command is as below. ***Password is "SX" that in hex is 0x5358.***

Command:	**sub command 87 with value 0x5358**	**Unlock write protection**
byte index	**byte description**	**Example(Hex)**
0	Slave ID	F0
1	Function Code	6
2	Sub Command H	0
3	Sub Command L	57
4	password _H	53
5	password _L	58

6	CRC16_L	10
7	CRC16_H	31

Unlock Command Response Message

The slave sensor will respond with the same message as master sent.

5.2 Modbus Write Query Message

After the slave response for the unlock command has been received by the master, the master will send the write command as below in order to write a single 16 bit register in the sensor.

Byte	Modbus	Range	Referenced to Sensor
1st	Slave Address	1-247* (Decimal)	Sensor ID
2nd	Function Code	06	Preset Single Register
3rd	Start Register Hi	00-FF (Hex)	Sensor register Hi
4th	Start Register Lo	00-FF (Hex)	Sensor register Lo
5th	Preset Data Hi	00-FF (Hex)	Value Hi Byte
6th	Preset Data Lo	00-FF (Hex)	Value Lo Byte
7th	CRC Lo	00-FF (Hex)	CRC Lo Byte
8th	CRC Hi	00-FF (Hex)	CRC Hi Byte

Example 3 Change Slave ID to 1

Command	Change Slave ID to 1	
byte index	byte description	Example (Hex)
0	Slave ID	F0
1	Function Code	6
2	Register H	0
3	Register L	0
4	Value number _H	0
5	Value number _L	1
6	CRC16_L	5D
7	CRC16_H	2B

Modbus Write Response Message is as the same as master sent as Example 3.

Byte	Modbus	Range	Referenced to Sensor
1st	Slave Address	1-247* (Decimal)	Sensor ID
2nd	Function Code	06	Preset Single Register
3rd	Start Register Hi	00	Sensor register Hi
4th	Start Register Lo	00-FF (Hex)	Sensor register Lo
5th	Preset Data Hi	00-FF (Hex)	Value Hi Byte
6th	Preset Data Lo	00-FF (Hex)	Value Lo Byte
7th	CRC Lo	00-FF (Hex)	CRC Lo Byte
8th	CRC Hi	00-FF (Hex)	CRC Hi Byte

6. Write Multiple Registers (Function Code 16 / 0x10)

6.1 Unlock Write Protection

As shown in section 5, an unlock command must be carried out before you can write to single register. The command is as below. *Password is "SX" that in hex is 0x5358*.

Command:	sub command 87 with value 0x5358	Unlock write protection
byte index	**byte description**	**Example(Hex)**
0	Slave ID	F0
1	Function Code	6
2	Sub Command H	0
3	Sub Command L	57
4	password _H	53
5	password _L	58
6	CRC16_L	10
7	CRC16_H	31

Unlock Command Response Message

The slave sensor will respond with the same message as master sent.

6.2 Modbus Write Query Message

After the slave response has been received for the unlock command, send the write command as shown below. (Example to write four registers)

Byte	Modbus	Range	Referenced to Sensor
1st	Slave Address	1-247* (Decimal)	Sensor ID
2nd	Function Code	16	Preset Single Register
3rd	Start Register Hi	00-FF (Hex)	Sensor register Hi
4th	Start Register Lo	00-FF (Hex)	Sensor register Lo
5th	Number of Register Hi	00	number Hi Byte
6th	Number of Register Lo	0x04	number Lo Byte
7th	Byte count	0x08	number of register x2
8th	Byte 0	00-FF (Hex)	data to write
9th	Byte 1	00-FF (Hex)	data to write
10th	Byte 2	00-FF (Hex)	data to write
11th	Byte 3	00-FF (Hex)	data to write
12th	Byte 4	00-FF (Hex)	data to write
13th	Byte 5	00-FF (Hex)	data to write
14th	Byte 6	00-FF (Hex)	data to write
15th	Byte 7	00-FF (Hex)	data to write
16th	CRC Lo	00-FF (Hex)	CRC Lo Byte
17th	CRC Hi	00-FF (Hex)	CRC Hi Byte

Example 4. Write a float point number 10.0 into register 90

Command:	**write 10.0 into address 90**	
byte index	**byte description**	**Example(Hex)**
0	Slave ID	F0
1	Function Code	10
2	Reg_Addr_H	0
3	Reg_Addr_L	5A
4	Number of regester_H	0
5	Number of regester_L	2
6	Byte Count	4
7	Value_HH	41
8	Value_HL	20
9	Value_LH	0
10	Value_LL	0
11	CRC16_L	64
12	CRC16_H	E5

6.3 Modbus Write Response Message

The response returned from slave is 8 bytes.

Byte	**Modbus**	**Range**	**Referenced to Sensor**
1st	Slave Address	1-247* (Decimal)	Sensor ID
2nd	Function Code	16	command code
3rd	Start Register Hi	00-FF (Hex)	Sensor register Hi
4th	Start Register Lo	00-FF (Hex)	Sensor register Lo
5th	Number of Register Hi	00	number Hi Byte
6th	Number of Register Lo	0x02	number Lo Byte
7th	CRC Lo	00-FF (Hex)	CRC Lo Byte
8th	CRC Hi	00-FF (Hex)	CRC Hi Byte

Example 5. Write Response Message for example 4.

byte index	byte description	Example(Hex)
0	Slave ID	F0
1	Function Code	10
2	Reg_Addr_H	0
3	Reg_Addr_L	5A
4	Number of regester_H	0
5	Number of regester_L	2

6	CRC16_L	74
7	CRC16_H	FA

7. Sensor Measurement Data Register

7.1 Probe Value, Temperature and Alternate value

Registers 3 to 7 are float point data for probe value, temperature in degree C and alternate probe value. The table below is the list for the Sensorex smart sensor. There is no data available for ORP temperature value. Register 80-81 is for DO sensor ppm value. Conductivity sensor and toroidal conductivity sensors use the same units. The alternate value is salinity in ppt (parts per thousands)

Sensor Type	pH	ORP	DO	FCL	Conductivity
Register 3-4	pH	mV	%	ppm	uS
Register 5-6	°C	xxx	°C	°C	°C
Register 7-8	mV	mV_raw	mV	nA	ppt

The float point is 32 bit IEEE751 standard that reads out with high bytes first. See example 2.

7.2 Probe Raw Value

Register 86-87 is the probe manufacture calibrated value before user calibration of Register 3-4.

It is a float point data. This data should be used for user calibration.

Example 6. Read probe raw value

Command:	**Read probe raw value**	
byte index	**byte description**	**Example(Hex)**
0	Slave ID	F0
1	Function Code	3
2	Reg_Addr_H	0
3	Reg_Addr_L	56
4	Number of regester_H	0
5	Number of regester_L	2
6	CRC16_L	31
7	CRC16_H	3A

The Sensor response message.

Example : pH raw value =11.16		
byte index	**byte description**	
0	Slave ID	F0
1	Function Code	3
2	byte count	4
3	pH_raw_HH	41
4	pH_raw_HL	32
5	pH_raw_LH	91
6	pH_raw_LL	97

7	CRC16_L	83
8	CRC16_H	31

8. Sensor Calibration Data Register

8.1 Calibration Data Storage

Registers 90 to 133 are for Calibration data storage. There are two data pairs for calibration. Data pair A is Calibration Reference A and measurement reading A. Data pair B is Calibration Reference B and measurement reading B. All are float point data. Calibration data should be stored into sensor memory. See table below.

When register 90 data is overwritten, the previous data is copied into register 104 Cal-Point-A1, and register 104 data is copied into register 118 Cal-Point-A2. The same processing is carried by sensor for register 92, 94, 96 and 98. Register 98-103 is 12 ASCII bytes for a time stamp such as YYYYMMDDHHmm (year, month, day, hour, and minutes). Writing the time stamp will also make an increment for the calibration number of register 132.

Register Addr.	Register name	Register count	Bytes	Read write access
90	Cal_point_A	2	4	Read/write
92	Meas_point_A	2	4	Read/write
94	Cal_point_B	2	4	Read/write
96	Meas_point_B	2	4	Read/write
98	Cal_Time	6	12	Read/write
104	Cal_point_A1	2	4	Read
106	Meas_point_A1	2	4	Read
108	Cal_point_B1	2	4	Read
110	Meas_point_B1	2	4	Read
112	Cal_Time1	6	12	Read
118	Cal_point_A2	2	4	Read
120	Meas_point_A2	2	4	Read
122	Cal_point_B2	2	4	Read
124	Meas_point_B2	2	4	Read
126	Cal_Time2	6	12	Read
132	Cal_Number	1	2	Read

8.2. User Calibration

In most cases, a two-point calibration is recommended. In case of use one point calibration, point A and B should be different value for sensor to calculate slope and offset.

Slope =(Cal_point_B-Cal_point_A)/(Meas_point_B-Meas_point_A)

Offset =Cal_point_A-slope*Meas_point_A

Here is the calibration step:

1. Put probe into buffer A. read the probe raw measurement of register 86-87 see Example 6. When the reading value is stable, write down the reference value and measurement value. Such as reference buffer is 4.0pH, measured value is 3.86pH.
2. Rinse the probe and put probe into buffer B. Read the probe raw measurement of register 86-87. When the reading value is stable, write down the reference value and measurement value, such as reference buffer is 10.0pH, measured value is 9.56pH.
3. Write value 4.0 to register 90-91. Write 3.86 to register 92-93. Write 10.0 to register 94-95. Write value 9.56 to register 96-97. Write 12 bytes ASCII string into register 98-103. **Remember to unlock write protection for every write command**. There is no combined write for all calibration data, in another word, one-write command for one float point. Because the sensor will do the copy processing for every calibration data storage. See unlock command and example 4.

Example 7. Write 12 bytes of Calibration Time Digits into register 98-103.

Command:	write '201903221130' into register 98	
byte index	byte description	Example(Hex)
0	Slave ID	F0
1	Function Code	10
2	Reg_Addr_H	0
3	Reg_Addr_L	62
4	Number of regester_H	0
5	Number of regester_L	6
6	Byte Count	C
7	byte0	32
8	byte1	31
9	byte2	31
10	byte3	39
11	byte4	30
12	byte5	33
13	byte6	32
14	byte7	32
15	byte8	31
16	byte9	31
17	byte10	33
18	byte11	30
19	CRC16_L	B2
20	CRC16_H	8D

8.3 Temperature Calibration

Register 66-67 is the offset for temperature calibration. To do the calibration, first read out the offset value in register 66-67. Then read the measured temperature in the register 5-6.

The new offset value is

Offset_new =T_ref-T_meas+Offset.

Here ***T_ref*** is reference temperature. ***T_meas*** is measured temperature.

Write the Offset_new into register 66-67, the updated temperature reading should be close to reference temperature.

8.4 ***Free Chlorine sensor Calibration will be added in a later revision of this manual.***

9. Operation Mode and 4-20mA Scale Setup

9.1 Operation Mode

Sensorex smart sensor can communication in one of two ways: Modbus or 4-20mA current loop signaling over the power line. The operation mode can be switched from Modbus to 4-20mA loop current mode by changing the operating mode to 2 in register 52 (default is Modbus). Please follow the **example 3** of writing a single register for the operating mode change. Recycle the power for the new setup to take effect.

9.2 Current Loop 4-20mA Scale Setup

The Modbus is function in the 4-20mA loop current mode. The user can configure the 4-20mA scale by changing the value in register 11-12 (4mA probe value) and register 13-14 (20mA probe value). Recycle the power after the new 4-20mA scale setup to write these changes to the sensor. Please follow **Example 4** of write multiple register for the 4-20mA scale change.

10. Temperature Coefficient Data Register

Temperature coefficient in register 46-47 is used for the temperature compensation of conductivity measurement. The default value is 0.02. That is 2% per degree C. The user can change the value as the application requires. Follow the **example 4** for the float point data write.

11. Firmware Version

Register 34-39, Sensor firmware version is a 12 bytes ASCII string. There are four fields separated by dashes. The first field is sensor type such as “ph”, “orp” or “cl”. A number behind represents a new design update. The second field number represents the firmware owner (example 3 is for Sensorex). The third number is a major revision and the fourth field is a minor revision.

Example 8. Firmware Revision

“cp2-3-0-4” – Contact conductivity sensor with second design change, firmware for Sensorex sensors. Major reversion 0 and minor reversion 4.
“ph-3-0-4” – pH sensor.
“cl-3-0-4” – Free Chlorine sensor.
“do-3-0-4” –- Dissolved oxygen sensor.
“to2-3-0-4” –- Toroidal conductivity sensor with second design change.
“orp-3-0-4” – ORP sensor.

12. Model Number

Register 16-21. Model number is a 12 byte ASCII string indicating product model number such as EM802-EC-MB2.

13. Serial Number

Register 22-27. It is a 12 byte ASCII string indicating manufacturing serial number such as 2021012811.

14. User label

Register 28-33. It is a space of 12 bytes ASCII reserved for user labeling the sensor.

15. Manufacture Date

Register 28-33. It is a 12 bytes ASCII string indicating manufacturing date such as 2019-02-2714.

16. Exception Responses and Exception Codes

16.1 Exception Response

In a normal communications query and response, the master device sends a query to the Sensor and the Sensor receives the query without a communications error and handles the query normally within the master device's allowable timeout. The Sensor then returns a normal response to the master. An abnormal communications produces one of four possible events.

1) If the Sensor does not receive the query due to a communications error, then no response is returned from the Sensor and the master device will eventually process a timeout condition for the query.
2) If the Sensor receives the query, but detects a communication error (CRC, etc.), then no response is returned from the Sensor and the master device will eventually process a timeout condition for the query.
3) If the Sensor receives the query without a communications error, but cannot process the response to the master within the master's timeout setting, then no response is returned from the Sensor and the master device will eventually process a timeout condition for the query. **In order to prevent this condition from occurring the maximum response time for the** Sensor **is 200 milliseconds. Therefore the Master's Timeout Setting should be set to 200 milliseconds or greater.**
4) If the Sensor receives the query without a communications error, but cannot process it due to reading or writing to a non-existent Sensor command register, then the Sensor will return an exception response message informing the master of the error.

The exception response message (ref. No. 4 above) has two fields that differentiate it from a normal response:

Byte	Modbus	Range	Referenced to Sensor
1st	Slave Address	1-247* (Decimal)	Sensor ID (Address)
2nd	Function Code	83, 86 or 90 (Hex)	Function Code + 0x80(Hex)

3rd	Exception Code	01 - 06 (Hex)	Appropriate Exception Code (See Below)
4th	CRC Lo	00-FF (Hex)	CRC Lo Byte
5th	CRC Hi	00-FF (Hex)	CRC Hi Byte

16.2 Exception Code

Exception Code Field: In a normal response, the Sensor returns data and status in the data field, which was requested in the query from the master. In an exception response, the Sensor returns an exception code in the data field, which describes the Sensor condition that caused the exception. Below is a list of exception codes that are supported by the Sensorex Modbus:

Code	Name	Description
01	Illegal Function	The function code received in the query is not an allowable action for the Sensor.
02	Illegal Data Address	The data address received in the query is not allowable address for the Sensor.
03	Illegal Data Value	A value contained in the query data field is not an allowable value for the Sensor.
04	Slave Device Failure	An unrecoverable error occurred while the Sensor was attempting to perform the requested action.
05	Acknowledge	Sensor has accepted the request and is processing it, but a long duration of time will be required to do so. This response is returned to prevent a timeout error from occurring in the master.
06	Device Busy	The Sensor is engaged in processing a long-duration program command. The master should retransmit the message later when the slave is free.

17. Soft Reset Sensor

Sensorex Modbus has a soft reset command that resets (reboots) the sensor without cycling the power. That is convenient in configuration for the multi-drop application. The following is the command detail. The sub-command is 89(0x59) and the password is ASCII “RX” that is 0x5258 in hex.

Command	Soft Reset sensor	
byte index	byte description	Example (Hex)
0	Slave ID	F0
1	Function Code	6
2	Register H	0
3	Register L	59
4	Password _H	52
5	Password _L	58
6	CRC16_L	70
7	CRC16_H	62

The return response is the same as the message the master sent. Then slave sensor will take a few seconds for reboot.

18. Sensorex Modbus Register List

This register List is also a memory map for sensor to access the manufacturing calibration parameters when processing the measurement data. Some of them are for manufacture calibration mode. Many important factor data is using for calculation on the run time. Please take precaution when writing into register. Any improper write will cause sensor malfunction.

Address	Register Name	Format	Register count	Bytes	ACCESS
0	modbus_address	uint8	1	2	Read/write
1	baud_rate	uint8	1	2	Read/write
2	serial_format	uint8	1	2	Read/write
3	probe_value	float	2	4	Read only
5	probe_temp_c	float	2	4	Read only
7	probe_alternate_value	float	2	4	Read only
9	loop_current_ma	float	2	4	Read only
11	probe_value_min	float	2	4	Read/write
13	probe_value_max	float	2	4	Read/write
15	probe range	uint16	1	2	Read only
16	model_number	char12	6	12	Read only
22	serial_number	char12	6	12	Read only
28	user_label	char12	6	12	Read/write
34	firmware_version	char12	6	12	Read only
40	manufacture_date	char12	6	12	Read only
46	temperature_coefficient	float	2	4	Read/Write
48	pressure_torr	uint16	1	2	Read/write
50	salinity_ppm	uint16	1	2	Read
51	reed_switch_active	uint16	1	2	Read
52	operating_mode	uint16	1	2	Read/write
53	override_mode	uint16	1	2	Read
54	override_value	float	2	4	Read
56	probe_duty_cycle_c2	float	2	4	Read
58	probe_duty_cycle_c1	float	2	4	Read
60	probe_duty_cycle_c0	float	2	4	Read
62	probe_temp_c2	float	2	4	Read
64	probe_temp_c1	float	2	4	Read
66	probe_temp_c0	float	2	4	Read/write
68	probe_value_c2	float	2	4	Read
70	probe_value_c1	float	2	4	Read
72	probe_value_c0	float	2	4	Read

74	calib_mode_c2	float	2	4	Read
76	calib_mode_c1	float	2	4	Read
78	calib_mode_c0	float	2	4	Read
80	DO/ppm	float	2	4	Read
82	probe_value_raw	float	2	4	Read
84	probe_temp_raw	float	2	4	Read
86	instrument_temp_raw	float	2	4	Read
88	instrument_temp_c	float	2	4	Read
90	Cal_point_A	float	2	4	Read/write
92	Meas_point_A	float	2	4	Read/write
94	Cal_point_B	float	2	4	Read/write
96	Meas_point_B	float	2	4	Read/write
98	Cal_Time	char12	6	12	Read/write
104	Cal_point_A1	float	2	4	Read
106	Meas_point_A1	float	2	4	Read
108	Cal_point_B1	float	2	4	Read
110	Meas_point_B1	float	2	4	Read
112	Cal_Time1	char12	6	12	Read
118	Cal_point_A2	float	2	4	Read
120	Meas_point_A2	float	2	4	Read
122	Cal_point_B2	float	2	4	Read
124	Meas_point_B2	float	2	4	Read
126	Cal_Time2	char12	6	12	Read
132	Cal_number	uint16	1	2	Read
134	Cal_ppm0	float	2	4	Read/write
136	Meas_nA0	float	2	4	Read/write
138	Cal_ppm1	float	2	4	Read/write
140	Meas_nA1	float	2	4	Read/write
142	Cal_Temp	float	2	4	Read/write
144	Cal_pH	float	2	4	Read/write
146	FCL_Cal_Time	char12	6	12	Read/write
152	FactorA1	float	2	4	Read
154	FactorA2	float	2	4	Read
156	FactorA3	float	2	4	Read
158	FactorA4	float	2	4	Read
160	FactorB1	float	2	4	Read
162	FactorB2	float	2	4	Read
164	FactorB3	float	2	4	Read
166	FactorB4	float	2	4	Read
168	FactorC1	float	2	4	Read
170	FactorC2	float	2	4	Read

172	FactorC3	float	2	4	Read
174	FactorC4	float	2	4	Read
176	Cell_Const1	float	2	4	Read
178	MiddlePoint	float	2	4	Read
180	Zero_Cutoff	float	2	4	Read
182	Parameter0	float	2	4	Read
184	Parameter1	float	2	4	Read
186	Parameter2	float	2	4	Read
188	Parameter3	float	2	4	Read
190	Parameter4	float	2	4	Read
192	Parameter5	float	2	4	Read
194	Parameter6	float	2	4	Read
196	Pwm_Counts_4mA	uint16	1	2	Read/write
197	Pwm_Counts_20mA	uint16	1	2	Read/write
198	Pwm_Counts	uint16	1	2	Read