Modbus on Senseair Sunrise

Table of contents:

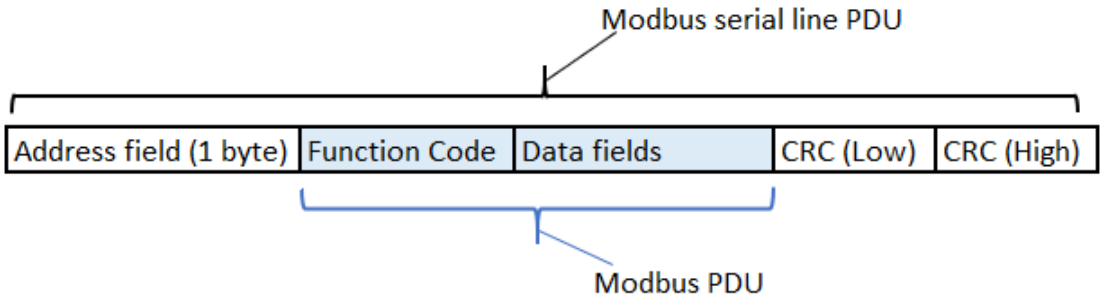
1. Modbus protocol

Modbus is a simple, open protocol for both PLC and sensors [1][2]. Details on Modbus can be found on the website www.modbus.org.

1.1. Serial line frame and addressing

Serial line frame

Modbus over serial line specification [2] distinguishes Modbus Protocol PDU and Modbus serial line PDU in the following way:



Addressing rules

The addressing rules are summarised in the table:

Address	Modbus over serial line V1.0	Senseair Sunrise
0 (0x00)	Broadcast address	No broadcast commands currently implemented
1 to 247 (0x01 to 0xF7)	Slave individual address	Slave individual address
248 to 253 (0xF8 to 0xFD)	Reserved	Nothing[1)]
254 (0xFE)	Reserved	“Any sensor” [2)]
255 (0xFF)	Reserved	Nothing[1)]

Notes:

1) “Nothing” means that the sensor doesn’t recognise Modbus serial line PDUs with this address as addressed to the sensor. The sensor does not respond.

2) “Any sensor” means that any sensor with any slave individual address will recognise serial line PDUs with address 254 as addressed to them. They will respond. This address is for production / test purposes only and must not be used in the installed network. This is a violation against the Modbus specification [1].

1.2. Bus timing

Parameter	Min	Typ	Max	Units
Response time-out			180	ms

“Response time-out” is defined to prevent the master (host system) from staying in “Waiting for reply” state indefinitely. Refer to page 9 of MODBUS over serial line specification [2].

For slave device “Response time-out” represents maximum time allowed to take by “processing of required action”, “formatting normal reply” and “normal reply sent” alternatively by “formatting error reply” and “error reply sent”, refer to the slave state diagram on page 10 of the document mentioned above.

1.3. Function code descriptions (PUBLIC)

Description of exception responses

If the PDU of the received command has wrong format:

No Response PDU, sensor doesn’t respond

If Function Code isn’t equal to any implemented function code:

Exception Response PDU

Function code	1 byte	Function Code + 0x80
Exception code = *Illegal Function*	1 byte	0x01

If one or more of addressed Registers is not assigned (register is reserved or Quantity of registers is larger than maximum number of supported registers):

Exception Response PDU

Function code	1 byte	Function Code + 0x80
Exception code = *Illegal Data Address*	1 byte	0x02

01 (0x01) Read Coils

One bit read/write registers.

Not implemented.

02 (0x02) Read Discrete Input

One bit read only registers.

Not implemented.

03 (0x03) Read Holding Registers

16 bits read/write registers.

Refer to Modbus specification [1].

Request PDU

Function code	**1 byte**	**0x03**
Starting Address Hi	1 byte	Address Hi
Starting Address Lo	1 byte	Address Lo
Quantity of Registers Hi	1 byte	Quantity Hi
Quantity of Registers Lo	1 byte	Quantity Lo

Response PDU

Function code	**1 byte**	**0x03**
Byte Count	1 byte	2 x N*
Register Value	N* x 2 bytes	

* N = Quantity of Registers

If Address is out of range:

Exception Response PDU

Function code	**1 byte**	**0x83**
Exception code = *Illegal Data Address*	1 byte	0x02

If Quantity=0 or Quantity>Number of Registers:

Exception Response PDU

Function code	**1 byte**	**0x83**
Exception code = *Illegal Data Value*	1 byte	0x03

04 (0x04) Read Input Registers

16 bits read only registers.

Refer to Modbus specification [1].

Quantity of Registers is limited to 32.

Request PDU

Function code	**1 byte**	**0x04**
Starting Address Hi	1 byte	Address Hi
Starting Address Lo	1 byte	Address Lo
Quantity of Registers Hi	1 byte	Quantity Hi
Quantity of Registers Lo	1 byte	Quantity Lo

Response PDU

Function code	**1 byte**	**0x04**
Byte Count	1 byte	2 x N*
Register Value	N* x 2 bytes	

* N = Quantity of Registers

If Address is out of range:

Exception Response PDU

Function code	**1 byte**	**0x84**
Exception code = *Illegal Data Address*	1 byte	0x02

If Quantity=0 or Quantity>Number of registers:

Exception Response PDU

Function code	**1 byte**	**0x84**
Exception code = *Illegal Data Value*	1 byte	0x03

05 (0x05) Write Single Coil

One bit read/write register.

Not implemented.

06 (0x06) Write Single Register

16 bits read / write register.

Not implemented.

15 (0x0F) Write Multiple Coils

One bit read / write registers.

Not implemented.

16 (0x10) Write Multiple Registers

16 bits read/write register.

Refer to Modbus specification [1].

Address of Modbus Holding Registers for 1-command reading/writing is limited in range 0x0000..0x002F.

Request PDU

Function code	**1 byte**	**0x10**
Starting Address Hi	1 byte	Address Hi
Starting Address Lo	1 byte	Address Lo
Number of Register Hi	1 byte	Value Hi
Number of Register Lo	1 byte	Value Lo
The Number of Data Bytes	1 byte	2 x N*
Register Value to Write	2 x N* bytes	Value to write

* N = Quantity of Registers

Response PDU (is an echo of the Request)

Function code	**1 byte**	**0x10**
Starting Address Hi	1 byte	Address Hi
Starting Address Lo	1 byte	Address Lo
Number of Register written Hi	1 byte	Value Hi
Number of Register written Lo	1 byte	Value Lo

If Address is out of range:

Exception Response PDU

Function code	**1 byte**	**0x90**
Exception code = *Illegal Data Address*	1 byte	0x02

20 (0x14) Read File record

Not implemented.

21 (0x15) Write File record

Not implemented.

22 (0x16) Mask Write Register

16 bits read/write register.

Not implemented.

23 (0x17) Read / Write Multiple Registers

16 bits read/write register.

Not implemented.

43 / 14 (0x2B / 0x0E) Read Device Identification

Refer to Modbus specification …

The sensor only supports Read Device ID code 4, individual access.

Objects 0x00 ..0x02 (basic identification) are available (see table)

Object ID	Object Name / Description	Type	Modbus status	Category	Implementation status
0x00	Vendor Name	ASCII string*	Mandatory	Basic	Implemented
0x01	ProductCode	ASCII string*	Mandatory	Basic	Implemented
0x02	MajorMinorRevision	ASCII string*	Mandatory	Basic	Implemented
0x03	VendorUrl	ASCII string*	Optional	Regular	Not Implemented
0x04	ProductName	ASCII string*	Optional	Regular	Not Implemented
0x05	ModelName	ASCII string*	Optional	Regular	Not Implemented
0x06	UserApplicationName	ASCII string*	Optional	Regular	Not Implemented
0x07.. 0x7F	Reserved				
0x80	Memory map version	1 byte unsigned	Optional	Extended	Not Implemented
0x81	Firmware revision, consists of: Firmware type, Revision Main, Revision Sub	3 bytes unsigned	Optional	Extended	Not Implemented
0x82	Sensor serial number (sensor ID)	4 bytes unsigned	Optional	Extended	Not Implemented
0x83	Sensor type	3 bytes unsigned	Optional	Extended	Not Implemented

*The ASCII strings are different for different models. As an example:

Vendor Name = “Senseair” (length 8 bytes)

Product Code = “Sunrise” (length 7 bytes)

MajorMinorRevision = “1.00” (length 4 bytes)

Example: Read objects of category “Basic”

Request PDU, Object ID 0x00 to 0x02

Function code	**1 byte**	**0x2B**
MEI Type	1 byte	0x0E
Read Device ID code	1 byte	0x04 (individual access only)
Object ID	1 byte	0x00..0x02

Response PDU, Object ID 0x00 to 0x02

Function code	**1 byte**	**0x2B**
MEI Type	1 byte	0x0E
Read Device ID code	1 byte	0x04, same as in request
Conformity level	1 byte	0x81, basic identification for individual or stream access
More Follows	1 byte	0x00
Next Object ID	1 byte	0x00
Number of objects	1 byte	0x01
Object ID	1 byte	0x00..0x02
Object length	1 byte	0x0B or 0x07 or 0x04 (see definition of ASCII strings)
Object value	n byte	Object Data

If wrong MEI Type:

Exception Response PDU

Function code	**1 byte**	**0xAB**
Exception code = *Illegal Function Code*	1 byte	0x01

If Object ID is not in range 0x00..0x03:

Exception Response PDU

Function code	**1 byte**	**0xAB**
Exception code = *Illegal Data Address*	1 byte	0x02

If wrong Device ID:

Exception Response PDU

Function code	**1 byte**	**0xAB**
Exception code = *Illegal Data Value*	1 byte	0x03

Note: The exception responses for function code 43 is implemented according to the RFC “RFC Non extended Exception code format of 43 Encapsulated Transport.doc” which is in status “Recommended for approval” at time of writing. This is in contrast with the Modbus specification [1] where the exception responses for function code 43 also have a MEI type field.

23 (0x17) Read / Write Multiple Registers

16 bits read/write register.

Not implemented.

1.4. References

[1] MODBUS Application Protocol Specification V1.1b
[2] MODBUS over serial line specification and implementation guide V1.02
[3] TDE7318, Sunrise user guideline

2. Modbus on Senseair Sunrise

2.1. Modbus settings

Senseair Sunrise supports 9600 baud rate only.

Other Modbus settings are as follows:

Setting	Value
Default slave address	104 (0x68)
Baud rate	9600 bps
Parity	None
DataBits	8
StopBits	1

2.2. Modbus registers

The Modbus registers are mapped in memory and the mapping is interpreted by the sensor at command reception.

The register maps are summarised in Table 1 and Table 2. All registers are 16-bit words. The associated number is the Modbus register number. The register address is calculated as (register number -1). For example, the address of IR4 is 3.

Table 1: Input Registers (IR)

IR#	Addr	Name	Description (read only registers)		
IR1	0x00	ErrorStatus			
			Bit	Error description	Suggested action
			0	**Fatal error** Indicates that initialization of analog front end failed	Try to restart sensor by power on/off. Contact local distributor.
			1	**I2C error** Attempt to read or write to not exiting addresses/registers detected.	Try to restart sensor by power on/off. Check wires, connectors and I2C protocol implementation. Contact local distributor.
			2	**Algorithm error** Corrupt parameters detected.	Try to restart sensor by power on/off. Contact local distributor.
			3	**Calibration error** Indicates that calibration has failed (ABC, zero, background or target calibration).	Try to repeat calibration. Ensure that the environment is stable during calibration.
			4	**Self-diagnostics error** Indicates internal interface failure.	Try to restart sensor by power on/off. Contact local distributor.
			5	**Out of range** Indicates that the measured concentration is outside the sensor's measurement range	Perform suitable CO_2 calibration (zero, background or target calibration). Contact local distributor.
			6	**Memory error** Error during memory operations	Try to restart sensor by power on/off. Contact local distributor.
			7	**No measurement completed** Bit set at startup, cleared after first measurement	0 – First measurement cycle completed 1 – No measurement completed If sensor is used in single measurement mode and powered down between measurements this bit can be used to verify started measurement cycle has finished

			8	**Low internal regulated voltage** Flag is set if sensors regulated voltage is too low, this means supply voltage is lower than 2.8V.	Check power supply.
			9	**Measurement timeout** Flag is set if sensor is unable to complete the measurement in time.	Flag is cleared after a successful measurement. If flag is set permanently try to restart sensor by power on/off. Contact local distributor.
			10	Reserved	
			11	Reserved	
			12	Reserved	
			13	Reserved	
			14	Reserved	
			15	Reserved	
IR2	0x01	Reserved			
IR3	0x02	Reserved			
IR4	0x03	CO_2 Value Filtered Pressure Compensated	Filtered pressure compensated CO_2 concentration. Signed 16 bit value, unit ppm. IR4 is equal to IR10 if pressure compensation is disabled at HR19 (default) Pressure compensated value can exceed 10000ppm. See [3] for details about IIR filtration and pressure compensation.		
IR5	0x04	Temperature	Chip temperature. Signed 16 bit value, unit °C x100. For example, register value = 2223 means 22.23°C.		
IR6	0x05	Reserved			
IR7	0x06	Measurement count	Counter incremented after each measurement, range 0 – 255. Counter value can for example be used by the host system to ensure that the sensor has done a measurement since last time CO_2 value was read.		
IR8	0x07	Measurement cycle time	Measurement cycle time shows current time in present measurement cycle, incremented every 2 seconds. For example, IR8 = 3 it means 6 seconds has passed in current measurement cycle. The value is set to 0 when sensor starts a new measurement. This value can be used by the host system to synchronize readings with sensor measurements.		
IR9	0x08	CO_2 Value Unfiltered Pressure Compensated	Unfiltered pressure compensated CO_2 concentration. Signed 16 bit value, unit ppm. IR9 is equal to IR11 if pressure compensation is disabled at HR19 (default)		
IR10	0x09	CO_2 Value Filtered	Filtered CO_2 concentration. Signed 16 bit value, unit ppm. See [3] for details about IIR filtration.		

IR11	0x0A	CO_2 Value Unfiltered	Unfiltered CO_2 concentration. Signed 16 bit value, unit ppm.
…		Reserved	
IR29	0x1C	FW rev.	Firmware revision. Unsigned 16 bit value. (bit 15 - 8 main) & (bit 7 - 0 sub)
IR30	0x1D	Sensor Id (bit31 – bit16)	Sensor Id. Unsigned 32 bit value.
IR31	0x1E	Sensor Id (bit15 – bit0)	
IR32		Reserved	

Table 2: Holding Registers (HR)

<table>
<tr><th>HR#</th><th>Addr</th><th>Name</th><th>Description (read/write registers)</th></tr>
<tr><td>HR1</td><td>0x00</td><td>Calibration Status</td><td>These bits are set after successful calibrations. The bits need to be cleared/reset by host system, it is recommended to do this before trigging a calibration using the HR2 register.
<table>
<tr><th>Bit</th><th>Description</th></tr>
<tr><td>0</td><td></td></tr>
<tr><td>1</td><td></td></tr>
<tr><td>2</td><td>Factory calibration restored</td></tr>
<tr><td>3</td><td>ABC calibration</td></tr>
<tr><td>4</td><td>Target calibration</td></tr>
<tr><td>5</td><td>Background calibration</td></tr>
<tr><td>6</td><td>Zero calibration</td></tr>
<tr><td>7</td><td></td></tr>
</table></td></tr>
<tr><td>HR2</td><td>0x01</td><td>Calibration Command</td><td>Calibration is initiated by the commands in the table below. See [3] for description of the different calibration modes.
The sensor will perform a calibration based on the first measurement immediately after the calibration command was received. After having performed the calibration all following measurements will use the adjusted calibration parameters
It is recommended that HR1 is cleared before initiating a calibration.
<table>
<tr><th>Command</th><th>Name and description</th></tr>
<tr><td>0x7C02</td><td>Restore factory calibration.
Restores calibration parameters to factory calibration values.</td></tr>
<tr><td>0x7C03</td><td>Forced ABC calibration.
Sensor will perform an ABC calibration after receiving this command if sensor has valid ABC data. The command can be used if one for some reason wants to do an ABC adjustment before one ABC period has passed (when a normal ABC calibration is done).
This command only works if ABC is enabled, see HR14 & HR19.</td></tr>
<tr><td></td><td></td></tr>
<tr><td>0x7C05</td><td>Target calibration.
Calibration using HR3 value as calibration target.</td></tr>
<tr><td>0x7C06</td><td>Background calibration
Calibration using ABC target as calibration target.</td></tr>
<tr><td>0x7C07</td><td>Zero calibration.</td></tr>
</table></td></tr>
</table>

			Calibration using 0 ppm CO_2 as calibration target.
HR3	0x02	Calibration Target	Calibration target used by target calibration (HR2 - 0x7C05 command). Unit ppm.
HR4	0x03	CO_2 Value Override	Default value = 32767 (no override). If a value lower than default is written to the register, both the CO_2 filtered and unfiltered registers will be set to this value.
HR5	0x04	ABC Time	Time passed since last ABC calibration in hours. **If ABC is enabled when sensor is used in single measurement mode and powered down between measurements the host system must increment this value every hour.**
HR6	0x05	ABC Par0	**If ABC is enabled when sensor is used in single measurement mode and powered down between measurements, these registers must be read from the sensor after each measurement and written back to the sensor after each power on (enable) before a new measurement is trigged.**
HR7	0x06	ABC Par1	
HR8	0x07	ABC Par2	
HR9	0x08	ABC Par3	
HR10	0x09	Start Single Measurement	Writing 1 to this register initiates a measurement if the sensor is configured for single measurement mode.
HR11	0x0A	Measurement Mode (EE)	There are two measurement modes to choose between. Register value = 0, continuous mode (default) Register value = 1, single measurement mode. A system reset is required after changing measurement mode.
HR12	0x0B	Measurement Period (EE)	Measurement period in seconds (range from 2 to 65534). Odd numbers will be rounded up to nearest even number. A system reset is required after changing configuration. Default value is 16. Note: Measurement period is only used in continuous measurement mode
HR13	0x0C	Number of samples (EE)	Number of samples in one measurement (range from 1 to 1024). A higher number leads to a better accuracy but also a higher power consumption. A system reset is required after changing configuration. Default is 8 samples. One sample takes < 200ms, this means that “Number of samples” * 0.2s should be less than or equal to measurement period. If time for executing all samples in a measurement is longer than measurement period, sensor will execute all samples and after that start a new measurement. This means that actual measurement period will be longer than measurement period specified in HR12.
HR14	0x0D	ABC period (EE)	Period for ABC cycle in hours (range from 1 to 65534). Default is 180 hours. ABC enabled by writing 1 to 65534 at HR14 **and** bit 1 = 0 at HR19. ABC disabled by writing 0 or 65535 to HR14 **or** bit 1 = 1 at HR19.
HR15	0x0E	Clear ErrorStatus	Write any number to this register to clear ErrorStatus

<table>
<tr><td>HR16</td><td>0x0F</td><td>ABC Target (EE)</td><td>Target value for background and ABC calibrations (ppm CO_2). Default value is 400 (400 ppm CO_2).</td></tr>
<tr><td>HR17</td><td>0x10</td><td>Static IIR filter parameter</td><td>Parameter for static IIR filter, range from 2 – 10. A higher value corresponds to a harder filtration. See [3] for details about IIR filtration.</td></tr>
<tr><td>HR18</td><td>0x11</td><td>SCR</td><td>The SCR register is used to reset the sensor
Register value = 0xFF, sensor will reset/restart itself.</td></tr>
<tr><td>HR19</td><td>0x12</td><td>Meter control (EE)</td><td>Bit field used to enable/disable sensor functions
<table>
<tr><th>Bit</th><th>Description</th></tr>
<tr><td>0</td><td>0 - nRDY enabled (default)
1 - nRDY disabled</td></tr>
<tr><td>1</td><td>0 - ABC enabled (default)
1 - ABC disabled</td></tr>
<tr><td>2</td><td>0 – Static IIR filter enabled (default)
1 - Static IIR filter disabled</td></tr>
<tr><td>3</td><td>0 – Dynamic IIR filter enabled (default)
1 – Dynamic IIR filter disabled
To enable dynamic IIR filter both static IIR filter (bit2) and dynamic IIR filter (bit3) has to be enabled</td></tr>
<tr><td>4</td><td>0 – Pressure compensation enabled
1 – Pressure compensation disabled (default)</td></tr>
<tr><td>5</td><td></td></tr>
<tr><td>6</td><td></td></tr>
<tr><td>7</td><td></td></tr>
</table>
EEPROM mapped register</td></tr>
<tr><td>HR20</td><td>0x13</td><td>MB/I2C address (EE)</td><td>Modbus address, range 1 – 247 (0x01 – 0xF7). Default value is 104 (0x68). A sensor reset is needed to activate the new address.
EEPROM mapped register</td></tr>
<tr><td>…</td><td></td><td>Reserved</td><td></td></tr>
</table>

<table>
<tr><td colspan="4">Registers from HR33 – HR39 are mirrors of registers HR1, HR10 and HR5-HR9. The reason for this is to make it possible to only write respective read one block of read/write registers when sensor is used in single measurement mode and powered down between measurements. See example 3.3 for details</td></tr>
<tr><td>HR33</td><td>0x20</td><td>Calibration Status</td><td>These bits are set after successful calibrations. The sensor never resets the bits so if the host system uses these bits it also must reset
<table>
<tr><th>Bit</th><th>Description</th></tr>
<tr><td>0</td><td></td></tr>
<tr><td>1</td><td></td></tr>
<tr><td>2</td><td>Factory calibration restored</td></tr>
<tr><td>3</td><td>ABC calibration</td></tr>
<tr><td>4</td><td>Target calibration</td></tr>
<tr><td>5</td><td>Background calibration</td></tr>
<tr><td>6</td><td>Zero calibration</td></tr>
<tr><td>7</td><td></td></tr>
<tr><td>8</td><td></td></tr>
<tr><td>9</td><td></td></tr>
<tr><td>10</td><td></td></tr>
<tr><td>11</td><td></td></tr>
<tr><td>12</td><td></td></tr>
<tr><td>13</td><td></td></tr>
<tr><td>14</td><td></td></tr>
<tr><td>15</td><td></td></tr>
</table>
them.</td></tr>
<tr><td>HR34</td><td>0x21</td><td>Start Single Measuremen t</td><td>Writing 1 to this register initiates a measurement if the sensor is configured for single measurement mode.</td></tr>
<tr><td>HR35</td><td>0x22</td><td>ABC Time</td><td>Time passed since last ABC calibration in hours.
If ABC is enabled when sensor is used in single measurement mode and powered down between measurements the host system must increment this value every hour.</td></tr>
<tr><td>HR36</td><td>0x23</td><td>ABC Par0</td><td rowspan="4">If ABC is enabled when sensor is used in single measurement mode and powered down between measurements, these registers must be read from the sensor after each measurement and written back to the sensor after each power on (enable) before a new measurement is trigged.</td></tr>
<tr><td>HR37</td><td>0x24</td><td>ABC Par1</td></tr>
<tr><td>HR38</td><td>0x25</td><td>ABC Par2</td></tr>
<tr><td>HR39</td><td>0x26</td><td>ABC Par3</td></tr>
</table>

HR40	0x27	Filter Par0	**If the sensor is used in single measurement mode with IIR filter enabled and powered down between measurements, these registers has to be read from the sensor after each measurement and written back to the sensor after each power on (enable) before a new measurement is trigged.**
HR41	0x28	Filter Par1	
HR42	0x29	Filter Par2	
HR43	0x2A	Filter Par3	
HR44	0x2B	Filter Par4	
HR45	0x2C	Filter Par5	
HR46	0x2D	Filter Par6	
HR47	0x2E	Barometric air pressure value	Barometric air pressure value. Signed 16 bit, unit 0.1 hPa. Range from 3000 – 13000 (300 – 1300 hPa) . For values outside pressure range error flag “out of range” will be set and compensation will be done with min or max pressure value.
HR48	0x2F	Reserved	

Registers with (EE) after their names use sensors EEPROM, this means that to frequent writes to these registers will lead to a corrupt EEPROM. Total number of EEPROM write cycles should be less than 10000.

When writing multiple (EE) registers in one sequence then this write cycle will be counted as just ONE write cycle out of the 10000 that are allowed writes to the EEPROM.

Writing one register to sensors EEPROM can take up to 25ms, this means time for response will be longer. It is important to wait until response from sensor is received before powering down the sensor, if sensor is powered down when EEPROM write operations are ongoing it may result in corrupt parameters.

Registers marked as “Reserved” can be read and written, however it is strongly recommended to not use these registers.

3. Examples

3.1. Read Error Status and CO_2

Reading input IR1 to IR4 (Error Status, IR2, IR3, CO_2).

Request(hex):

68 04 00 00 00 04 F8 F0
CRC

Response(hex):

68 04 08 00 00 00 00 00 00 05 47 B7 F2
Error status CO_2 CRC

Error status = 0

CO_2 = 1351

For details about CRC calculation see [1].

3.2. Set measurement mode to single measurement mode

Write 1 to HR11. Note that after measurement mode has been written to the sensor it has to be restarted before it will change to the new measurement mode.

Request(hex):

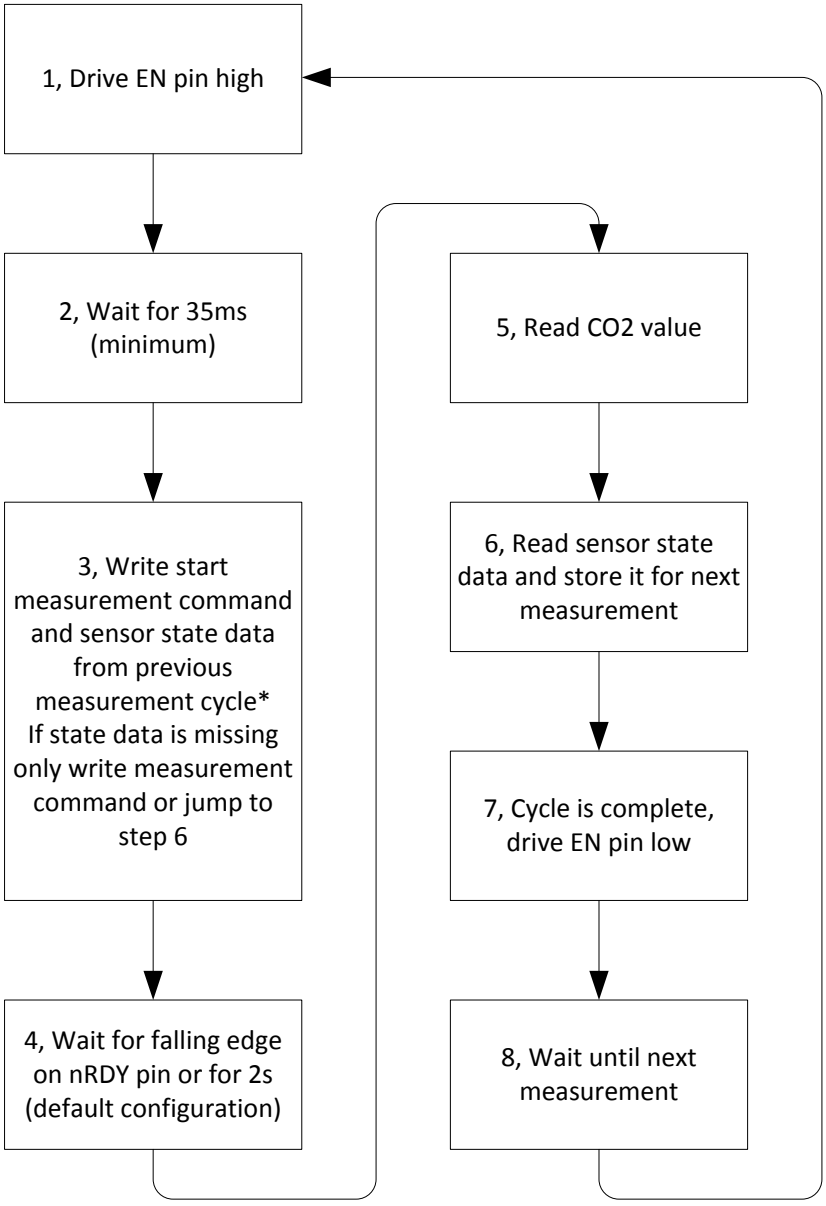
68 10 00 0A 00 01 02 00 01 A5 68

Response(hex):

68 10 00 0A 00 01 28 F2

3.3. Communication sequence for single measurement mode

Example of communication sequence in single measurement mode when sensor is powered up/down by enable pin.



*If start measurement command and state data is written in two separate write sequences, state data must be written before start measurement command

Figure 1, Communication sequence for single measurement mode

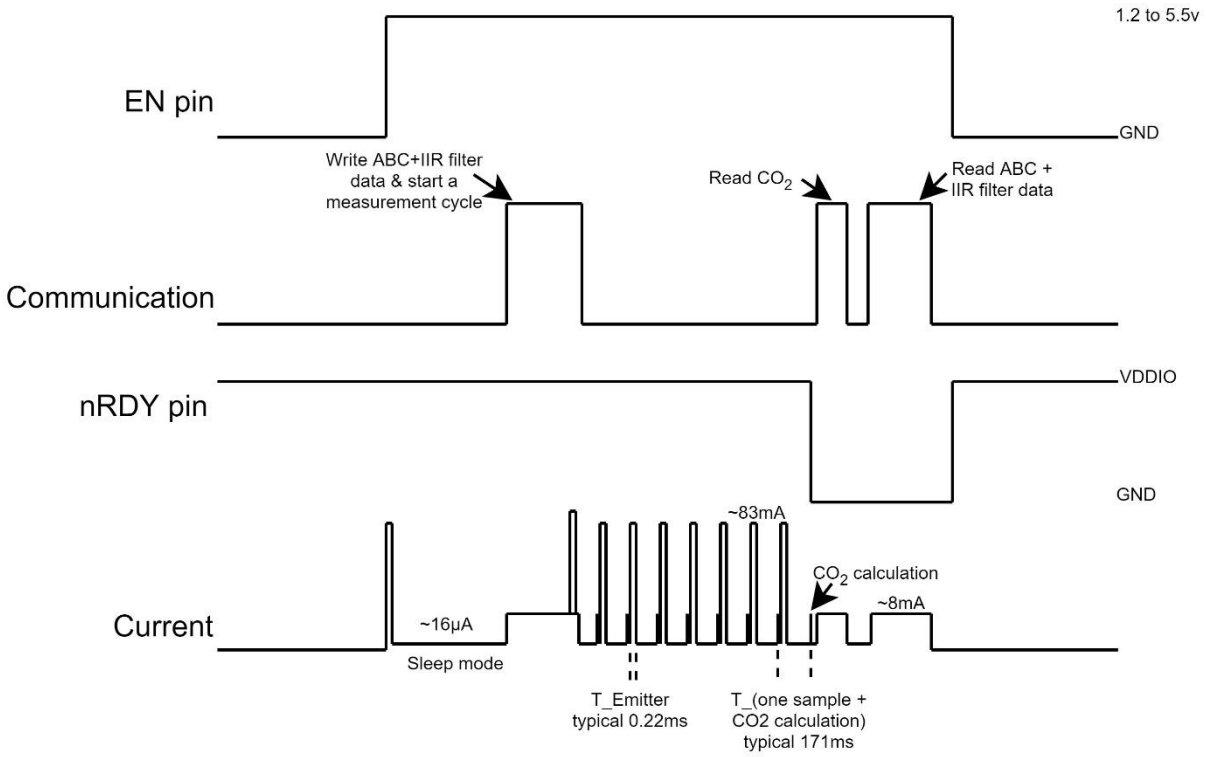


Figure 2, Timing diagram for single measurement mode

1. Drive EN pin high (>1.2V)

2. Wait for minimum 35ms for sensor start-up and stabilization

3. If state data exist go to point 3.1, otherwise go to point 3.2

   3.1. Sensor state data exist

   If pressure compensation is not enabled write start measurement command and state data to HR34 – HR46.

   Request(hex):

   68 10 00 21 00 0D 1A 00 01 00 00 00 00 00 00 7F FF 00 08 00 02 00 01 00 01 97 DC 00 F5 FF 64 00 F5 07 7B

   Response(hex):

   68 10 00 21 00 0D 58 FF

If pressure compensation is enabled, pressure (HR47) can be written together with measurement command and state data.

In this example pressure 1050 hPa is used (1050 x 10 = 10500 = 0x2904)

Request(hex):

68 10 00 21 00 0E 1C 00 01 00 00 00 00 00 7F FF 00 08 00 02 00 01 00 01 97 DC 00 F5 FF 64 00 F5 29 04 DF B4

Response(hex):

68 10 00 21 00 0E 18 FE

3.2. State data does not exist, or it is an initial measurement

If host device has no state data, it is very important that host do not write “0” to HR36-HR46 first time it starts a measurement.

There are 2 options for this condition:

a) Start measurement command to HR34

   If pressure compensation is enabled write pressure to HR47 before writing the start measurement command.

   Write start measurement command

   Request(hex):
   68 10 00 21 00 01 02 00 01 A3 73

   Response(hex):
   68 10 00 21 00 01 58 FA

b) Jump to point 6 immediately to read and save state data from the sensor before the initial measurement.

4. Wait until ready pin goes low or 2s (for default configuration)

5. Read register IR1-IR4 (Error Status, IR2, IR3, CO_2 value).

   Request(hex):

   68 04 00 00 00 04 F8 F0

Response(hex):

68 04 08 00 00 00 00 00 00 05 75 36 27

Error status CO_2

Error status = 0

CO_2 value = 1397

6. Read sensor state data from HR35-HR46 and save it for next measurement.

Request(hex):

68 03 00 22 00 0C EC FC

Response(hex):

68 03 18 00 00 00 00 00 00 7F FF 00 08 00 02 00 01 00 01 97 DC 00 F5 FF 64 00 F5 5A FB

7. Drive EN pin low (<0.4V)

3.4. Enable/Disable ABC

Enable and disable ABC by writing to HR14 and HR19.

Enable ABC:

1. Clear bit1 in HR19

   Start to read HR19.
   Request(hex):
   68 03 00 12 00 01 2D 36

   Response(hex):
   68 03 02 00 F2 65 C8

   Clear bit1 in register and write back
   HR19 = HR19 & 0xFFFD = 0x00F2 & 0xFFFD = 0x00F0

   Write back new HR19 value.
   Request(hex):
   68 10 00 12 00 01 02 00 F0 67 34

   Response(hex):
   68 10 00 12 00 01 A8 F5

2. Read HR14 and verify that it is desired ABC period.
   Request(hex):
   68 03 00 0D 00 01 1C F0

   Response(hex):
   68 03 02 00 B4 E4 3A

3. If HR14 (ABC period) is not is desired period write wanted ABC period to HR14, in this example ABC period is set to 200 hours.
   Request(hex):
   68 10 00 0D 00 01 02 00 C8 64 89

   Response(hex):
   68 10 00 0D 00 01 99 33

Disable ABC:

1. Set bit1 in HR19

   Start to read HR19.
   Request(hex):
   68 03 00 12 00 01 2D 36

   Response(hex):
   68 03 02 00 F0 E4 09

   Set bit1 in register and write back.
   HR19 = HR19 | 0x0002 = 0x00F0 | 0x0002 = 0x00F2

   Write back new HR19 value.
   Request(hex):
   68 10 00 12 00 01 02 00 F2 E6 F5

   Response(hex):
   68 10 00 12 00 01 A8 F5

A possible alternative is to set HR14 to zero.

3.5. Enable/Disable dynamic IIR filter

Enable and disable dynamic IIR filtration by writing to HR19.

Enable dynamic IIR filter.

Start to read HR19.

Request(hex):

68 03 00 12 00 01 2D 36

Response(hex):

68 03 02 00 FF A4 0D

Clear bit2 and bit3 in register and write back

HR19 = HR19 & 0xFFF3 = 0x00FF & 0xFFF3 = 0x00F3

Write back new HR19 value.

Request(hex):

68 10 00 12 00 01 02 00 F3 27 35

Response(hex):

68 10 00 12 00 01 A8 F5

Disable static and dynamic IIR filter.

Start to read HR19.

Request(hex):

68 03 00 12 00 01 2D 36

Response(hex):

68 03 02 00 F3 A4 08

Set bit 2 and bit 3 in HR19 and write back.

HR19 = HR19 | 0x000C = 0x00F3 | 0x000C = 0x00FF

Write back new HR19 value.

Request(hex):

68 10 00 12 00 01 02 00 FF 27 30

Response(hex):

68 10 00 12 00 01 A8 F5

3.6. Background Calibration

Trig a background calibration and reading calibration status and CO_2 value after calibration.

Write background calibration command (0x7C06) to HR2

Request:

68 10 00 01 00 01 02 7C 06 C5 11

Response:

68 10 00 01 00 01 59 30

Read calibration status from HR1

Request:

68 03 00 00 00 01 8D 33

Response:

68 03 02 00 20 E5 95

HR1 = 0x20

To achieve best possible result from calibration it is important that the sensor is in a stable environment.

3.7. Target Calibration

Example how to calibrate against target 500 ppm.

Write calibration target 500 to HR3

Request:

68 10 00 02 00 01 02 01 F4 65 F7

Response:

68 10 00 02 00 01 A9 30

Write calibration command (0x7C05) to HR2

Request:

68 10 00 01 00 01 02 7C 05 85 10

Response:

68 10 00 01 00 01 59 30

Read calibration status from HR1.

Request:

68 03 00 00 00 01 8D 33

Response:

68 03 02 00 10 E5 81

HR1 = 0x10

3.8. Set Modbus address

Set sensors Modbus address to 10.

Request:

68 10 00 13 00 01 02 00 0A E6 A6

Response:

68 10 00 13 00 01 F9 35

Sensor starts to use new address after a sensor reset (reset cmd or power cycle).

3.9. Pressure compensation

Enable and disable pressure compensation by writing to HR19. When pressure compensation is enabled sensor will use value is HR47 to pressure compensate CO2, if no value has been written to HR47 no pressure compensation will be done.

Enable pressure compensation.

Start to read HR19.

Request(hex):

68 03 00 12 00 01 2D 36

Response(hex):

68 03 02 00 FF A4 0D

Clear bit4 in register and write back

HR19 = HR19 & 0xFFEF = 0x00FF & 0xFFEF = 0x00EF

Write back new HR19 value.

Request(hex):

68 10 00 12 00 01 02 00 EF 26 FC

Response(hex):

68 10 00 12 00 01 A8 F5

Disable pressure compensation.

Start to read HR19.

Request(hex):

68 03 00 12 00 01 2D 36

Response(hex):

68 03 02 00 EF A5 C1

Set bit 4 in register and write back.

HR19 = HR19 | 0x0010 = 0x00EF | 0x0010 = 0x00FF

Write back new HR19 value.

Request(hex):

68 10 00 12 00 01 02 00 FF 27 30

Response(hex):

68 10 00 12 00 01 A8 F5

3.10. Write pressure to sensor

Write pressure 997 hPa to sensor HR47. Sensor works with unit 0.1 hPa so first value to write to the sensor has to be calculated

Value to write = 997 * 10 = 9970.

Request:

68 10 00 2E 00 01 02 26 F2 F9 A9

Response:

68 10 00 2E 00 01 68 F9

If pressure compensation is enabled and a value has been written to HR47 sensor will pressure compensate CO_2. If value written to sensor is not in the range 3000 – 13000 sensor will use 3000 or 13000 for the pressure compensation.

Pressure value written to the sensor will be used in the next CO_2 calculation. In continuous measurement mode this means that it can take up to one measurement period before pressure compensated CO_2 value is based on the new pressure value

www.senseair.com