

# プロトコル説明書

# Modbus on Senseair S8

## 1. General

Modbus is a simple, open protocol for both PLC and sensors. Details on Modbus can be found on [www.modbus.org](http://www.modbus.org).

This specification is based on the specification of Modbus implementation on aSense, eSense and Sensor Core and aims to support backwards compatibility with them. The differences between the Modbus specification [1] and the default implementation in the sensor are listed in this document.

## 1.概要

Modbusは取り扱いが簡単な、PLCおよびセンサ用のオープンプロトコルです。Modbusに関する詳細は以下のサイトをご参照下さい。

[www.modbus.org](http://www.modbus.org).

この説明書はaSense、eSenseおよびSenseairCore(センサモジュール)へのModbus実装の仕様に基づいており、これらセンサの後方互換性をサポートすることを目的としています。Modbus仕様[1]とセンサのデフォルト実装の違いは当説明書中に列挙されています。

## Table of contents:

1. General .....	1
2. Byte transmission .....	4
3. Modbus registers on sensor. ....	6
4. Serial line frame and addressing. ....	14
5. Bus timing .....	14
6. Function codes descriptions (PUBLIC). ....	176
7. References .....	232
Appendix A: Application examples .....	254
Appendix B: Compatibility with Senseair Core and aSENSE Modbus definitions. ....	32

## General overview of protocol and implementation in the sensor

### Master – slave:

Only master can initiate a transaction. The sensor is a slave and will never initiate communication. The host system initiates transactions to read CO<sub>2</sub> value from the corresponding register. The host system shall also check status of the sensor periodically (e.g. every two (2) seconds) in order to determine if it is running without faults detected.

### Packet identification:

Any message (packet) starts with a silent interval of 3.5 characters. Another silent interval of 3.5 characters marks message end. Silence interval between characters in the message needs to be kept less than 1.5 characters.

Both intervals are from the end of Stop-bit of previous byte to the beginning of the Start-bit of the next byte.

### Packet length:

According to the Modbus specification [1], the packet length shall be maximum 255 bytes including address and CRC. We cannot support so large packets. Maximum length of packet (serial line PDU including address byte and 2 bytes CRC) supported by the sensor is **39 bytes** (differs from CO<sub>2</sub> Engine models with their 28 bytes). **Packets of larger size are rejected without any answer from sensor**

### Modbus data model:

There are four (4) primary data tables (addressable registers), which may overlay:

- Discrete Input (read only bit).
- Coil (read / write bit).
- Input register (read only 16 bit word, interpretation is up to application).
- Holding register (read / write 16 bit word).

Note: The sensor does not support bitwise access of registers.

### Exception responses:

Slave will send answer to the master only in the case of valid message structure. Nevertheless, it can send exception response because of detection of:

- Invalid function code.
- Invalid data address (requested register doesn't exist in given device).
- Invalid data.
- Error in execution of requested function.

## 目次：

1. 概要 .....	1
2. バイト送信 .....	5
3. センサの Modbus レジスタ .....	7
4. シリアルラインフレームおよびアドレス .....	145
5. バスタイミング .....	145
6. ファンクションコードの説明(PUBLIC) .....	17
7. 参考事項 .....	23
付録 A: アプリケーション例 .....	25
付録 B: SenseairCore(センサモジュール)およびaSenseのModbus定義との互換性 .....	323

## プロトコルの一般的概要とセンサへの実装

### マスター-スレーブ:

マスターのみが処理を開始することができます。センサはスレーブで通信を開始することはできません。ホストシステムが対応するレジスタからCO<sub>2</sub>値を読み出す処理を開始します。ホストシステムはまた、センサがエラー検出なしで動作しているかを判定するために、センサの状態を定期的にチェックします(例: 2秒毎)。

### パケットの識別:

全てのメッセージ(パケット)は3.5文字のサイレントインターバルで始まります。もう一つの3.5文字のサイレントインターバルはメッセージの終りを示します。メッセージ中の文字間のサイレントインターバルは1.5文字未満でなければなりません。何れのインターバルも、前のバイトのストップビットの終りから、次のバイトのスタートビットの始めまでです。

### パケットの長さ:

Modbus仕様[1]に従い、パケットの長さはアドレスおよびCRCを含んで255バイトでなければなりません。大きなパケットはサポートされません。センサがサポートするパケットの最大長さ(アドレスバイトおよび2バイトを含むシリアルラインPDU)は39バイトです(28バイトのCO<sub>2</sub>Engineのモデルとは異なります)。サイズが大きすぎるパケットはセンサからの何の応答もなく拒否されます。

### Modbusデータモデル:

オーバーレイできる四つのプライマリデータテーブルがあります:

- ディスクリード入力(読出しのみのビット)
- コイル(読出し/書込みビット)
- 入力レジスタ(読出しのみ16ビットワード、解釈はアプリケーションによる)
- 保持レジスタ(読出し/書込み16ビットワード)

**注: センサはレジスタのビット単位アクセスをサポートしません。**

### 例外レスポンス:

スレーブは、メッセージ構成が有効な場合にのみマスターに答を送ります。しかしながら、以下の項目を検出した場合には、スレーブは例外レスポンスを送ることができます:

- 無効なファンクションコード
- 無効なデータアドレス(要求されるレジスタが当該のデバイスに存在しない)
- 無効なデータ
- 要求された機能の実行中のエラー

## 2. Byte transmission.

RTU transmission mode is the only mode supported by the sensor.

### 2.1. Byte format:

The format for each byte in RTU mode differs between the sensor default configuration and the description on page 12 of Modbus over serial line specification [2].

	<b>Modbus over serial line specification [2]</b>	<b>Sensor default configuration</b>
Coding system	8-bit binary	8-bit binary
Bits per byte:	1 start bit	1 start bit
Data bits	8 data bits, least significant bit first	8 data bits, least significant bit first
	1 bit for even parity	No parity bit
	1 stop bit	2 stop bits
		1 stop bit for receiving 2 stop bits at transmission

Table 1: Byte format differences

Implementation of 1 stop bit on receive and 2 stop bits at transmit provides compatibility with masters using both 1 and 2 stop bits.

### 2.2. Baud rate (data signalling rate)

9600 bps and 19200 bps are required baud rates and required default baud rate according to MODBUS over serial line specification [2], page 20, is 19200 bps.

Senseair S8 supports 9600 baud rate only.

### 2.3. Physical layer:

The sensor provides CMOS logical levels RxD and TxD lines for serial transmission. It's up to the system integrator to use them for direct communication with master processor or for connection to RS-232 or RS-485 drivers. In the latter case R/T control line may be added on request.

Communication lines are fed directly to the micro-controller of the sensor. Please refer to technical description for electrical specifications of particular model.

## 2. バイト送信

センサがサポートする唯一のモードはRTU送信モードです。

### 2.1 バイトのフォーマット:

RTUモードの各バイトに対するフォーマットは、センサのデフォルト設定と、P12のシリアル通信Modbus仕様[2]の説明との間で異なります。

	シリアル通信Modbus仕様[2]	センサデフォルト設定
コーディングシステム	8ビットバイナリ	8ビットバイナリ
バイト当たりのビット数	1スタートビット	1スタートビット
データビット	8データビット 最下位バイトが最初	8データビット 最下位バイトが最初
	偶数パリティに対して1ビット	パリティビットなし
	1ストップビット	2ストップビット
		受信に1ストップビット 送信時に2ストップビット

表1: バイトフォーマットの違い

受信に1ストップビット、送信時に2ストップビットを付与することにより、1ストップビットおよび2ストップビットを使っているマスタとの互換性が与えられます。

### 2.2 ボーレート

要求されるボーレートは9600bpsおよび19200bpsで、シリアル通信Modbus仕様[2](P20)に従って要求されるデフォルトのボーレートは19200bpsです。

センサエアS8はボーレート9600bpsのみをサポートしています。

### 2.3 物理層:

センサはシリアル通信用に、CMOSロジックレベルRxDおよびTxDラインを備えています。それらをマスタプロセッサとの直接通信に使用するか、あるいは、RS232またはRS485ドライバーとの接続用に使用するかはシステムインテグレーターによります。後者の場合、必要であれば、R/T制御ラインを追加できます。

通信ラインは直接センサのマイコンに送られます。センサの個別の電氣的仕様についての説明を参照して下さい。

### 3. Modbus registers on sensor.

The Modbus registers are mapped in memory, both RAM and EEPROM of the sensor. Mapping is interpreted by sensor firmware at command reception.

Presently, the following restrictive decisions are made:

1. Read only and read / write registers are not allowed to overlay.
2. Bit addressable items (i.e. Coils and Discrete inputs) will not be implemented.
3. Only write single register functional codes are implemented. Multiple write functional codes are not planned for implementation.
4. The total number of registers should be limited. Present decision is to limit number of input registers to 32 and number of holding registers to 32.

Note: the limited buffer space of the sensor puts a limit on how many registers that can be read in one command, currently 6 registers.

5. Larger amount of data should be transferred as file. It is not implemented at the current stage of development.

Maps of registers (All registers are 16 bit word) are summarized in Table 3 and Table 4.

Associated number is Modbus register number: Register address is calculated as (register number -1)

IR#	#	Name																
IR1	0	MeterStatus	DI 16	DI 15	DI 14	DI 13	DI 12	DI 11	DI 10	DI 9	DI 8	DI 7	DI 6	DI 5	DI 4	DI 3	DI 2	DI 1
			DI 1 - Fatal error DI 2 - Offset regulation error DI 3 - Algorithm Error DI 4 - Output Error DI 5 - Self diagnostics error DI 6 - Out Of Range DI 7 - Memory error DI 8 - Reserved <sup>1</sup> DI 9 - Reserved <sup>1</sup> DI 10 - Reserved <sup>1</sup> DI 11 - Reserved <sup>1</sup> DI 12 - Reserved <sup>1</sup> DI 13 - Reserved <sup>1</sup> DI 14 - Reserved <sup>1</sup> DI 15 - Reserved <sup>1</sup> DI 16 - Reserved <sup>1</sup>															
IR2	1	AlarmStatus	DI 16	DI 15	DI 14	DI 13	DI 12	DI 11	DI 10	DI 9	DI 8	DI 7	DI 6	DI 5	DI 4	DI 3	DI 2	DI 1
			DI 17 - Reserved <sup>1</sup> DI 18 - Reserved <sup>1</sup> DI 19 - Reserved <sup>1</sup> DI 20 - Reserved <sup>1</sup> DI 21 - Reserved <sup>1</sup> DI 22 - Reserved <sup>1</sup> DI 23 - Reserved <sup>1</sup> DI 24 - Reserved <sup>1</sup> DI 25 - Reserved <sup>1</sup> DI 26 - Reserved <sup>1</sup> DI 27 - Reserved <sup>1</sup> DI 28 - Reserved <sup>1</sup> DI 29 - Reserved <sup>1</sup> DI 30 - Reserved <sup>1</sup> DI 31 - Reserved <sup>1</sup> DI 32 - Reserved <sup>1</sup>															

### 3. センサのModbusレジスタ

Modbusのレジスタはセンサの、RAMおよびEEPROM両方のメモリ内にマッピングされています。このマッピングはセンサのコマンド受信に対するファームウェアによって解釈されます。

現在、以下の制限が設けられています：

1. 読出しのみレジスタと読出し/書込みレジスタはオーバーレイできません。
2. ビットアドレス化できる項目（即ちコイルおよびDI(discrete input)）は実装されません。
3. 書込み単レジスタのファンクションコードのみが実装されています。複数の書込みファンクションコードは実装される予定はありません。
4. レジスタの総数は制限が必要です。現在決められている制限は、入力レジスタが32、保持レジスタが32です。  
注： センサの制限されているバッファスペースは、ひとつのコマンドで幾つのレジスタの読出しが可能かについての制限を作ります。現在、この制限は六つのレジスタです。
5. 容量の大きいデータはファイルとして送信されなければなりません。この機能は、現在の開発段階では実装されていません。

レジスタのマッピング(全てのレジスタは16ビットワード)は表2および表3にまとめられています。関連付けられている数字はModbusレジスタの数字です:レジスタのアドレスは(レジスタ番号-1)として計算されます。

IR#	#	Name																
IR1	0	MeterStatus	DI 16	DI 15	DI 14	DI 13	DI 12	DI 11	DI 10	DI 9	DI 8	DI 7	DI 6	DI 5	DI 4	DI 3	DI 2	DI 1
			DI 1 - Fatal error	DI 2 - Offset regulation error	DI 3 - Algorithm Error	DI 4 - Output Error	DI 5 - Self diagnostics error	DI 6 - Out Of Range	DI 7 - Memory error	DI 8 - Reserved <sup>1</sup>	DI 9 - Reserved <sup>1</sup>	DI 10 - Reserved <sup>1</sup>	DI 11 - Reserved <sup>1</sup>	DI 12 - Reserved <sup>1</sup>	DI 13 - Reserved <sup>1</sup>	DI 14 - Reserved <sup>1</sup>	DI 15 - Reserved <sup>1</sup>	DI 16 - Reserved <sup>1</sup>
IR2	1	AlarmStatus	DI 16	DI 15	DI 14	DI 13	DI 12	DI 11	DI 10	DI 9	DI 8	DI 7	DI 6	DI 5	DI 4	DI 3	DI 2	DI 1
			DI 17 - Reserved <sup>1</sup>	DI 18 - Reserved <sup>1</sup>	DI 19 - Reserved <sup>1</sup>	DI 20 - Reserved <sup>1</sup>	DI 21 - Reserved <sup>1</sup>	DI 22 - Reserved <sup>1</sup>	DI 23 - Reserved <sup>1</sup>	DI 24 - Reserved <sup>1</sup>	DI 25 - Reserved <sup>1</sup>	DI 26 - Reserved <sup>1</sup>	DI 27 - Reserved <sup>1</sup>	DI 28 - Reserved <sup>1</sup>	DI 29 - Reserved <sup>1</sup>	DI 30 - Reserved <sup>1</sup>	DI 31 - Reserved <sup>1</sup>	DI 32 - Reserved <sup>1</sup>

IR3	2	Output Status	DI 16	DI 15	DI 14	DI 13	DI 12	DI 11	DI 10	DI 9	DI 8	DI 7	DI 6	DI 5	DI 4	DI 3	DI 2	DI 1
			DI 33 - Alarm Output status (inverted due to Open Collector) DI 34 - PWM Output status (1 means full output) DI 35 - Reserved <sup>1</sup> DI 36 - Reserved <sup>1</sup> DI 37 - Reserved <sup>1</sup> DI 38 - Reserved <sup>1</sup> DI 39 - Reserved <sup>1</sup> DI 40 - Reserved <sup>1</sup> DI 41 - Reserved <sup>1</sup> DI 42 - Reserved <sup>1</sup> DI 43 - Reserved <sup>1</sup> DI 44 - Reserved <sup>1</sup> DI 45 - Reserved <sup>1</sup> DI 46 - Reserved <sup>1</sup> DI 47 - Reserved <sup>1</sup> DI 48 - Reserved <sup>1</sup>															
IR4	3	Space CO2	Space CO2															
IR5	4		Reserved for Space Temp, returns "illegal data address" exception															
IR6	5		Reserved, returns "illegal data address" exception															
IR7	6		Reserved, returns "illegal data address" exception															
IR8	7		Reserved, returns "illegal data address" exception															
IR9	8		Reserved, returns "illegal data address" exception															
IR10	9		Reserved, returns "illegal data address" exception															
IR11	10		Reserved, returns "illegal data address" exception															
IR12	11		Reserved, returns "illegal data address" exception															
IR13	12		Reserved, returns "illegal data address" exception															
IR14	13		Reserved, returns "illegal data address" exception															
IR15	14		Reserved, returns "illegal data address" exception															
IR16	15		Reserved, returns "illegal data address" exception															
IR17	16		Reserved, returns "illegal data address" exception															
IR18	17		Reserved, returns "illegal data address" exception															
IR19	18		Reserved, returns "illegal data address" exception															
IR20	19		Reserved, returns "illegal data address" exception															
IR21	20		Reserved, returns "illegal data address" exception															
IR22	21	PWM Output <sup>2</sup>	PWM Output <sup>2</sup>															
IR23	22		Reserved, returns "illegal data address" exception															
IR24	23		Reserved, returns "illegal data address" exception															
IR25	24		Reserved, returns "illegal data address" exception															



IR3	2	Output Status	DI 16	DI 15	DI 14	DI 13	DI 12	DI 11	DI 10	DI 9	DI 8	DI 7	DI 6	DI 5	DI 4	DI 3	DI 2	DI 1
			DI 33 - Alarm Output status (inverted due to Open Collector) DI 34 - PWM Output status (1 means full output) DI 35 - Reserved <sup>1</sup> DI 36 - Reserved <sup>1</sup> DI 37 - Reserved <sup>1</sup> DI 38 - Reserved <sup>1</sup> DI 39 - Reserved <sup>1</sup> DI 40 - Reserved <sup>1</sup> DI 41 - Reserved <sup>1</sup> DI 42 - Reserved <sup>1</sup> DI 43 - Reserved <sup>1</sup> DI 44 - Reserved <sup>1</sup> DI 45 - Reserved <sup>1</sup> DI 46 - Reserved <sup>1</sup> DI 47 - Reserved <sup>1</sup> DI 48 - Reserved <sup>1</sup>															
IR4	3	Space CO2	Space CO2															
IR5	4		Reserved for Space Temp, returns "illegal data address" exception															
IR6	5		Reserved, returns "illegal data address" exception															
IR7	6		Reserved, returns "illegal data address" exception															
IR8	7		Reserved, returns "illegal data address" exception															
IR9	8		Reserved, returns "illegal data address" exception															
IR10	9		Reserved, returns "illegal data address" exception															
IR11	10		Reserved, returns "illegal data address" exception															
IR12	11		Reserved, returns "illegal data address" exception															
IR13	12		Reserved, returns "illegal data address" exception															
IR14	13		Reserved, returns "illegal data address" exception															
IR15	14		Reserved, returns "illegal data address" exception															
IR16	15		Reserved, returns "illegal data address" exception															
IR17	16		Reserved, returns "illegal data address" exception															
IR18	17		Reserved, returns "illegal data address" exception															
IR19	18		Reserved, returns "illegal data address" exception															
IR20	19		Reserved, returns "illegal data address" exception															
IR21	20		Reserved, returns "illegal data address" exception															
IR22	21	PWM Output <sup>2</sup>	PWM Output <sup>2</sup>															
IR23	22		Reserved, returns "illegal data address" exception															
IR24	23		Reserved, returns "illegal data address" exception															
IR25	24		Reserved, returns "illegal data address" exception															

IR26	25		Sensor Type ID High <sup>3</sup>
IR27	26		Sensor Type ID Low <sup>3</sup>
IR28	27		Memory Map version
IR29	28		FW version Main.Sub <sup>4</sup>
IR30	29		Sensor ID High <sup>5</sup>
IR31	30		Sensor ID Low <sup>5</sup>
IR32	31		Reserved, returns "illegal data address" exception

Table 2 : Input Registers

- <sup>1</sup> – Reserved DIs return 0.
- <sup>2</sup> – 0x3FFF represents 100% output. Refer to sensor model's specification for voltage at 100% output.
- <sup>3</sup> – IR26 low byte + IR27 contains Sensor Type ID 3-bytes value.
- <sup>4</sup> – IR29 high byte is FW Main revision, low byte – FW Sub revision.
- <sup>5</sup> – IR30 + IR31 – 4-bytes Sensor's Serial Number.

HR#	#	Name																
HR1	0	Acknowledgement register	DI 16	DI 15	DI 14	DI 13	DI 12	DI 11	DI 10	DI 9	DI 8	DI 7	DI 6	DI 5	DI 4	DI 3	DI 2	DI 1
			CI 1 - Reserved <sup>6</sup> CI 2 - Reserved <sup>6</sup> CI 3 - Reserved <sup>6</sup> CI 4 - Reserved <sup>6</sup> CI 5 - Reserved <sup>6</sup> CI 6 - CO2 background calibration has been performed CI 7 - CO2 nitrogen calibration has been performed CI 8 - Reserved <sup>6</sup> CI 9 - Reserved <sup>6</sup> CI 10 - Reserved <sup>6</sup> CI 11 - Reserved <sup>6</sup> CI 12 - Reserved <sup>6</sup> CI 13 - Reserved <sup>6</sup> CI 14 - Reserved <sup>6</sup> CI 15 - Reserved <sup>6</sup> CI 16 - Reserved <sup>6</sup>															
HR2	1	Special Command Register <sup>7</sup>	Command								Parameter							
			0x7C								0x6 - CO2 background calibration 0x7 - CO2 zero calibration							
HR3	2		Reserved, returns "illegal data address" exception															
HR4	3		Reserved, returns "illegal data address" exception															
HR5	4		Reserved, returns "illegal data address" exception															
HR6	5		Reserved, returns "illegal data address" exception															
HR7	6		Reserved, returns "illegal data address" exception															

IR26	25		Sensor Type ID High <sup>3</sup>
IR27	26		Sensor Type ID Low <sup>3</sup>
IR28	27		Memory Map version
IR29	28		FW version Main.Sub <sup>4</sup>
IR30	29		Sensor ID High <sup>5</sup>
IR31	30		Sensor ID Low <sup>5</sup>
IR32	31		Reserved, returns "illegal data address" exception

表2: 入力レジスタ

(注)1-予約されているDisは0を返します。

(注)2-0x3FFFは出力100%を表します。出力100%での電圧については、各センサの仕様を参照して下さい。

(注)3-IR26ローバイト+IR27は3-バイト値のセンサタイプIDを含みます。

(注)4-IR29ハイバイトはFWのメインリビジョン、ローバイトはFWのサブリビジョン。

(注)5-IR30+IR31は4バイトのセンサシリアル番号。

HR#	#	Name																
HR1	0	Acknowledgement register	DI 16	DI 15	DI 14	DI 13	DI 12	DI 11	DI 10	DI 9	DI 8	DI 7	DI 6	DI 5	DI 4	DI 3	DI 2	DI 1
			CI 1 - Reserved <sup>6</sup> CI 2 - Reserved <sup>6</sup> CI 3 - Reserved <sup>6</sup> CI 4 - Reserved <sup>6</sup> CI 5 - Reserved <sup>6</sup> CI 6 - CO2 background calibration has been performed CI 7 - CO2 nitrogen calibration has been performed CI 8 - Reserved <sup>6</sup> CI 9 - Reserved <sup>6</sup> CI 10 - Reserved <sup>6</sup> CI 11 - Reserved <sup>6</sup> CI 12 - Reserved <sup>6</sup> CI 13 - Reserved <sup>6</sup> CI 14 - Reserved <sup>6</sup> CI 15 - Reserved <sup>6</sup> CI 16 - Reserved <sup>6</sup>															
HR2	1	Special Command Register <sup>7</sup>	Command								Parameter							
			0x7C								0x6 - CO2 background calibration 0x7 - CO2 zero calibration							
HR3	2		Reserved, returns "illegal data address" exception															
HR4	3		Reserved, returns "illegal data address" exception															
HR5	4		Reserved, returns "illegal data address" exception															
HR6	5		Reserved, returns "illegal data address" exception															

HR7	6		Reserved, returns "illegal data address" exception
HR8	7		Reserved, returns "illegal data address" exception
HR9	8		Reserved, returns "illegal data address" exception
HR10	9		Reserved, returns "illegal data address" exception
HR11	10		Reserved, returns "illegal data address" exception
HR12	11		Reserved, returns "illegal data address" exception
HR13	12		Reserved, returns "illegal data address" exception
HR14	13		Reserved, returns "illegal data address" exception
HR15	14		Reserved, returns "illegal data address" exception
HR16	15		Reserved, returns "illegal data address" exception
HR17	16		Reserved, returns "illegal data address" exception
HR18	17		Reserved, returns "illegal data address" exception
HR19	18		Reserved, returns "illegal data address" exception
HR20	19		Reserved, returns "illegal data address" exception
HR21	20		Reserved, returns "illegal data address" exception
HR22	21		Reserved, returns "illegal data address" exception
HR23	22		Reserved, returns "illegal data address" exception
HR24	23		Reserved, returns "illegal data address" exception
HR25	24		Reserved, returns "illegal data address" exception
HR26	25		Reserved, returns "illegal data address" exception
HR27	26		Reserved, returns "illegal data address" exception
HR28	27		Reserved, returns "illegal data address" exception
HR29	28		Reserved, returns "illegal data address" exception
HR30	29		Reserved, returns "illegal data address" exception
HR31	30		Reserved, returns "illegal data address" exception
HR32	31	ABC Period	ABC Period in hours <sup>8</sup>

Table 3: Holding Registers

<sup>6</sup> – Reserved CIs return 0.

<sup>7</sup> – Special Command Register is write-only.

<sup>8</sup> – Writing to ABC\_Period zero value suspends ABC function. ABC samples and ABC time counting will not be reset. To resume ABC function with prior ABC samples and ABC time write to ABC\_Period non-zero value.

HR7	6		Reserved, returns "illegal data address" exception
HR8	7		Reserved, returns "illegal data address" exception
HR9	8		Reserved, returns "illegal data address" exception
HR10	9		Reserved, returns "illegal data address" exception
HR11	10		Reserved, returns "illegal data address" exception
HR12	11		Reserved, returns "illegal data address" exception
HR13	12		Reserved, returns "illegal data address" exception
HR14	13		Reserved, returns "illegal data address" exception
HR15	14		Reserved, returns "illegal data address" exception
HR16	15		Reserved, returns "illegal data address" exception
HR17	16		Reserved, returns "illegal data address" exception
HR18	17		Reserved, returns "illegal data address" exception
HR19	18		Reserved, returns "illegal data address" exception
HR20	19		Reserved, returns "illegal data address" exception
HR21	20		Reserved, returns "illegal data address" exception
HR22	21		Reserved, returns "illegal data address" exception
HR23	22		Reserved, returns "illegal data address" exception
HR24	23		Reserved, returns "illegal data address" exception
HR25	24		Reserved, returns "illegal data address" exception
HR26	25		Reserved, returns "illegal data address" exception
HR27	26		Reserved, returns "illegal data address" exception
HR28	27		Reserved, returns "illegal data address" exception
HR29	28		Reserved, returns "illegal data address" exception
HR30	29		Reserved, returns "illegal data address" exception
HR31	30		Reserved, returns "illegal data address" exception
HR32	31	ABC Period	ABC Period in hours <sup>8</sup>

表3: 保持レジスタ

(注)6-予約されているCIsは0を返します。

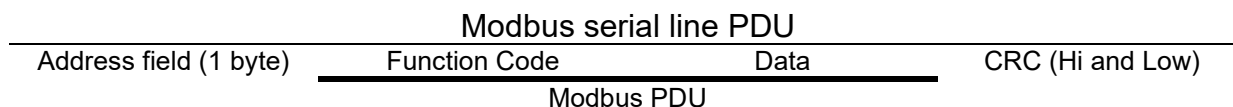
(注)7-特別コマンドレジスタは書込みのみです。

(注)8-ABC\_Periodへの0値の書込みはABC機能を中止させます。ABCのサンプルおよびABCの時間カウントはリセットされません。ABC機能を先のABCサンプルとABC時間で復帰させるためには、ABC\_Periodへゼロ以外の値を書込みます。

## 4. Serial line frame and addressing.

### 4.1. Serial line frame

Modbus over serial line specification [2] distinguishes Modbus Protocol PDU and Modbus serial line PDU in the following way (RTU mode only is under consideration):



### 4.2. Addressing rules

Addressing rules are summarised in the table:

Address	Modbus over serial line V1.0	SenseAir® S8 Sensor
0	Broadcast address	No broadcast commands currently implemented
From 1 to 247	Slave individual address	Slave individual address
From 248 to 253	Reserved	Nothing <sup>1)</sup>
254	Reserved	"Any sensor"
255	Reserved	Nothing <sup>1)</sup>

Table 4: Summarized addressing rules

Notes:

- "Nothing" means that sensor doesn't recognise Modbus serial line PDUs with this address as addressed to the sensor. Sensor does not respond.
- "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. So that this address is for production / test purposes only. It must not be used in the installed network. This is a violation against the Modbus specification [1].

### 4.3. Broadcast address

Modbus specification [1] requires execution of all write commands in the broadcast address mode.

Current status for the sensor:

Only one broadcast command, reset sensor, is planned but not implemented yet.

## 5. Bus timing.

Parameter	Min	Type	Max	Units
Response time-out			180	msec

Table 5: Bus timing

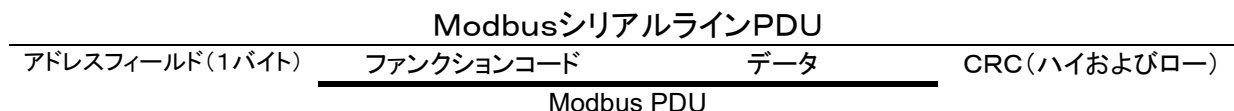
"Response time-out" is defined to prevent master (host system) from staying in "Waiting for reply" state indefinitely. Refer to page 9 of MODBUS over serial line specification [2].

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.

## 4. シリアルラインのフレームおよびアドレス指定

### 4.1 シリアルラインフレーム

シリアル通信Modbus仕様[2]は、ModbusプロトコルPDUとModbusシリアルラインPDUを以下の方法で区別しています(RTUモードのみが考慮中):



### 4.2 アドレス指定のルール

アドレス指定のルールは下表のまとめの通りです:

アドレス	シリアルラインModbus V1.0	センスエアS8センサ
0	ブロードキャストアドレス	現在ブロードキャストアドレスの設定なし
1~247	スレーブ個別アドレス	スレーブ個別アドレス
248~253	保留	なし
254	保留	“何らかのセンサ”
255	保留	なし

表4: アドレス指定ルールまとめ

(注)1-“なし”は、センサがこのアドレスでは、センサがアドレス指定されるようには、ModbusシリアルラインPDUを認識しないことを意味します。従って、センサは応答しません。

(注)2-“何らかのセンサ”は、何らかのスレーブ個別アドレスをもつ何らかのセンサが、通常アドレス指定されるように、アドレス254のシリアルラインPDUを認識することを意味します。応答が得られます。このアドレスは工場での生産/テストの目的でのみ使用されます。現用のネットワーク中では使用してはけません。これはModbus仕様[1]に対しては違反です。

### 4.3 ブロードキャストアドレス

Modbus仕様[1]は、全ての書き込みコマンドの実行がブロードキャストアドレスのモードで行われることを要求しています。

現在のセンサの状態は:

唯一のブロードキャストコマンド、センサリセットが予定されていますが、現在は実装されていません。

## 5 バスタイミング

パラメータ	Min	タイプ	Max	単位
応答タイムアウト			180	msec

表5: バスタイミング

“応答タイムアウト”は、マスタ(ホストシステム)が無期限に“応答待ち”状態に留まることを避けるために定義されます。P9のシリアル通信Modbus仕様[2]を参照して下さい。

スレーブのデバイスでは、“応答タイムアウト”は、“要求されるアクションの処理”、“正常な返答のフォーマット化”および“送信された正常な返答”、あるいは“フォーマット化エラー返答”、“送信された返答のエラー”に費やすことが許される最大の時間を意味します。P10の上述の文書のスレーブの状態ダイアグラムを参照して下さい。

## 6 Function codes 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 = <b>Illegal Function</b>	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 = <b>Illegal Data Address</b>	1 byte	0x02

### 6.1.01 (0x01) Read Coils (one bit read / write registers).

Not implemented.

### 6.2.02 (0x02) Read Discrete Inputs (one bit read only registers).

Not implemented.

### 6.3.03 (0x03) Read Holding Registers (16 bits read / write registers).

Refer to Modbus specification [1].

Quantity of Registers is limited to 32.

Address of Modbus Holding Registers for 1-command reading is limited in range 0x0000..0x001F.

Request PDU

Function code	1 byte	<b>0x03</b>
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	<b>0x03</b>
Byte Count	1 byte	2 x N*
Register Value	N* x 2 bytes	

\* N = Quantity of Registers



## 6 ファンクションコードの説明(PUBLIC)

### 例外レスポンスの説明

受信したコマンドのPDUが間違ったフォーマットである場合:

No Response PDU、(センサは応答しません)

ファンクションコードが実装された何れかのファンクションコードと同じではない場合:

#### 例外レスポンス PDU

Function code	1 byte	Function Code + 0x80
Exception code = <i>Illegal Function</i>	1 byte	0x01

ひとつまたはそれ以上のアドレス指定されたレジスタが割り当てられていない場合(レジスタが保留されている、あるいはレジスタの数がサポートされているレジスタの最大数よりも大きい場合):

#### 例外レスポンス PDU

Function code	1 byte	Function Code + 0x80
Exception code = <i>Illegal Data Address</i>	1 byte	0x02

### 6.1.01 (0x01) 読出しコイル(1ビット読出し/書込みレジスタ)

未実装

### 6.2.02 (0x02) DI(discreteinput) 読出し(1ビット読出しのみレジスタ)

未実装

### 6.3.03 (0x03) 保持レジスタ読出し(16ビット読出し/書込みレジスタ)

Modbus仕様[1]を参照。  
レジスタの数は32に制限。

1に対するModbus保持レジスタのアドレスーコマンド読出しは0x0000~0x001Fに制限されています。

#### リクエストPDU

Function code	1 byte	<b>0x03</b>
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

#### レスポンスPDU

Function code	1 byte	<b>0x03</b>
Byte Count	1 byte	2 x N*
Register Value	N* x 2 bytes	

\* N = レジスタの数

If Address>0x001F or (Address + Quantity)>0x0020:

Exception Response PDU,

Function code	1 byte	<b>0x83</b>
Exception code = <i>Illegal Data Address</i>	1 byte	0x02

If Quantity=0 or Quantity>8:

Exception Response PDU,

Function code	1 byte	<b>0x83</b>
Exception code = <i>Illegal Data Value</i>	1 byte	0x03

#### 6.4.04 (0x04) Read Input Registers (16 bits read only registers).

Refer to Modbus specification [1].

Quantity of Registers is limited to 32.

Address of Modbus Input Registers for 1-command reading is limited in range 0x0000..0x001F.

Request PDU

Function code	1 byte	<b>0x04</b>
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	<b>0x04</b>
Byte Count	1 byte	2 x N*
Register Value	N* x 2 bytes	

\* N = Quantity of Registers

If Address>0x001F or (Address + Quantity)>0x0020:

Exception Response PDU,

Function code	1 byte	<b>0x84</b>
Exception code = <i>Illegal Data Address</i>	1 byte	0x02

If Quantity=0 or Quantity>8:

Exception Response PDU,

Function code	1 byte	<b>0x84</b>
Exception code = <i>Illegal Data Value</i>	1 byte	0x03

#### 6.5.05 (0x05) Write Single Coil (one bit read / write register).

Not implemented.

アドレス>0x001Fまたは(アドレス+数)>0x0020の場合:

例外レスポンス PDU

Function code	1 byte	<b>0x83</b>
Exception code = <i>Illegal Data Address</i>	1 byte	0x02

数=0または数>8の場合:

例外レスポンス PDU

Function code	1 byte	<b>0x83</b>
Exception code = <i>Illegal Data Value</i>	1 byte	0x03

#### 6.4.04 (0x04) 入力レジスタ読出し(16ビット読出しのみレジスタ)

Modbus仕様[1]を参照。

レジスタの数は32に制限。

1に対するModbus入力レジスタのアドレス-コマンド読出しは0x0000~0x001Fに制限されています。

リクエストPDU

Function code	1 byte	<b>0x04</b>
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

レスポンスPDU

Function code	1 byte	<b>0x04</b>
Byte Count	1 byte	2 x N*
Register Value	N* x 2 bytes	

\* N = レジスタの数

アドレス>0x001Fまたは(アドレス+数)>0x0020の場合:

例外レスポンス PDU

Function code	1 byte	<b>0x84</b>
Exception code = <i>Illegal Data Address</i>	1 byte	0x02

数=0または数>8の場合:

例外レスポンス PDU

Function code	1 byte	<b>0x84</b>
Exception code = <i>Illegal Data Value</i>	1 byte	0x03

#### 6.5.05 (0x05) 単コイルの書込み(1ビット読出し/書込みレジスタ)

未実装

## 6.6.06 (0x06) Write Single Register (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..0x001F.

### Request PDU

Function code	1 byte	<b>0x06</b>
Starting Address Hi	1 byte	Address Hi
Starting Address Lo	1 byte	Address Lo
Register Value Hi	1 byte	Value Hi
Register Value Lo	1 byte	Value Lo

### Response PDU (is an echo of the Request)

Function code	1 byte	<b>0x06</b>
Starting Address Hi	1 byte	Address Hi
Starting Address Lo	1 byte	Address Lo
Register Value Hi	1 byte	Value Hi
Register Value Lo	1 byte	Value Lo

If Address>0x001F:

### Exception Response PDU

Function code	1 byte	<b>0x86</b>
Exception code = <i>Illegal Data Address</i>	1 byte	0x02

## 6.7.15 (0x0F) Write Multiple Coils (one bit read / write registers).

Not implemented.

## 6.8.16 (0x10) Write Multiple Registers (16 bits read / write register).

Not implemented.

## 6.9.20 (0x14) Read File record.

Not implemented.

## 6.10.21 (0x15) Write File record.

Not implemented.

## 6.11. 22 (0x16) Mask Write Register (16 bits read / write register).

Not implemented.

## 6.12. 23 (0x17) Read / Write Multiple Registers (16 bits read / write register).

Not implemented.

### 6.6.06 (0x06) 単コイルの書込み(1ビット読出し/書込みレジスタ)

Modbus仕様[1]を参照。

1に対するModbus保持レジスタコマンド読出し/書込みは0x0000～0x001Fに制限されています。

#### リクエストPDU

Function code	1 byte	<b>0x06</b>
Starting Address Hi	1 byte	Address Hi
Starting Address Lo	1 byte	Address Lo
Register Value Hi	1 byte	Value Hi
Register Value Lo	1 byte	Value Lo

#### レスポンスPDU(リクエストのエコー)

Function code	1 byte	<b>0x06</b>
Starting Address Hi	1 byte	Address Hi
Starting Address Lo	1 byte	Address Lo
Register Value Hi	1 byte	Value Hi
Register Value Lo	1 byte	Value Lo

アドレスが>0x001Fの場合:

#### 例外レスポンス PDU

Function code	1 byte	<b>0x86</b>
Exception code = <i>Illegal Data Address</i>	1 byte	0x02

### 6.7.15 (0x0F) 複数コイルの書込み(1ビット読出し/書込みレジスタ)

未実装

### 6.8.16 (0x10) 複数コイルの書込み(16ビット読出し/書込みレジスタ)

未実装

### 6.9.20 (0x14) ファイル記録の読出し

未実装

### 6.10.21 (0x15) ファイル記録の書込み

未実装

### 6.11. 22 (0x16) マスク書込みレジスタ(16ビット読出し/書込みレジスタ)

未実装

### 6.12. 23 (0x17) 複数レジスタの読出し/書込み(16ビット読出し/書込みレジスタ)

未実装

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

Not implemented.

Sensor Type ID, sensor Serial Number can be read through Input Registers.  
(see Table 4 “Input Registers compatibility” for details).

## 7 References

- [1] MODBUS Application Protocol Specification V1.1b
- [2] MODBUS over serial line specification and implementation guide V1.02

## 43 / 14 (0x2B / 0x0E) デバイス識別の読出し

未実装

センサタイプID、センサシリアル番号は入力レジスタで読出し可能。  
(詳細は表4“入力レジスタの互換性”を参照)

## 7 参照事項

- [1] Modbusアプリケーションプロトコル仕様書V1.1b
- [2] シリアル通信Modbus仕様書および実装ガイドV1.02

## Appendix A: Application examples

Prerequisites for the application examples:

1. A single slave (sensor) is assumed (address "any sensor" is used).
2. Values in <..> are hexadecimal.

### CO<sub>2</sub> read sequence:

The sensor is addressed as "Any address" (0xFE).

We read CO<sub>2</sub> value from IR4 using "Read input registers" (function code 04). Hence, Starting address will be 0x0003 (register number-1) and Quantity of registers 0x0001. CRC calculated to 0xC5D5 is sent with low byte first.

We assume in this example that by sensor measured CO<sub>2</sub> value is 400ppm\*.

Sensor replies with CO<sub>2</sub> reading 400ppm (400 ppm = 0x190 hexadecimal).

Master Transmit:

<FE> <04> <00> <03> <00> <01> <D5> <C5>

Slave Reply:

<FE> <04> <02> <01> <90> <AC> <D8>

\* Note that some future models in the Senseair S8 family of sensors may have a different scale factor on the ppm reading. The reading on these models is divided by 10 (i.e. when ambient CO<sub>2</sub> level is 400ppm the sensor will transmit the number 40). In this example the reply from one of these models would be 40 (= 0x28 hexadecimal).

### Sensor status read sequence:

The sensor is addressed as "Any address" (0xFE).

We read status from IR1 using "Read input registers" (function code 04). Hence, Starting address will be 0x0000 (register number-1) and Quantity of registers 0x0001. CRC calculated to 0xC525 is sent with low byte first.

Sensor replies with status 0.

Master Transmit:

<FE> <04> <00> <00> <00> <01> <25> <C5>

Slave Reply:

<FE> <04> <02> <00> <00> <AD> <24>



## 付録 A: アプリケーション例

アプリケーション例に対する前提条件:

1. 単体のスレーブ(センサ)を想定(“何らかのセンサ”アドレスを使用)。
2. <>内の値は16進数。

### CO<sub>2</sub>読出しのシーケンス:

センサは“何らかのアドレス”(0xFE)にアドレス指定されています。  
CO<sub>2</sub>値は、“入力レジスタ読出し”(ファンクションコード04)を使って、IR4から読出します。従って、開始アドレスは0x0003(レジスタ番号-1)、レジスタ数は0x0001になります。0xC5D5に計算されるCRCは最初にローバイトで送信されます。  
この例ではセンサが測定したCO<sub>2</sub>値が400ppm\*と想定します。

センサはCO<sub>2</sub>読み値400ppmで返答します(400ppm=0x190、16進数)。

マスタは以下を送信します:

<FE> <04> <00> <03> <00> <01> <D5> <C5>

スレーブは以下で返答します:

<FE> <04> <02> <01> <90> <AC> <D8>

\*注 将来、S8シリーズの新しいモデルのセンサはppmの読取りに対して異なるスケール要素が採用される可能性があります。それらのモデルでは読み値が10で割られます(即ち、雰囲気CO<sub>2</sub>レベルが400ppmのとき、センサは40の数字を送信します)。この例では、それらのモデルのセンサの返答は40(0x28、16進数)になります。

### センサステータス読出しのシーケンス:

センサは“何らかのアドレス”(0xFE)としてアドレス指定されています。  
“入力レジスタの読出し”(ファンクションコード04)を使って、IR1から読出します。。従って、開始アドレスは0x0000(レジスタ番号-1)、レジスタ数は0x0001になります。0xC5D5に計算されるCRCは最初にローバイトで送信されます。

センサはステータス 0 で返答します。

マスタは以下を送信します:

<FE> <04> <00> <00> <00> <01> <25> <C5>

スレーブは以下で返答します:

<FE> <04> <02> <00> <00> <AD> <24>





## Background calibration sequence:

The sensor is addressed as “Any address” (0xFE).

3. Clear acknowledgement register by writing 0 to HR1. Starting address is 0x0000 and Register value 0x0000. CRC calculated as 0xC59D is sent with low byte first.

Master Transmit:

<FE> <06> <00> <00> <00> <00> <9D> <C5>

Slave Reply:

<FE> <06> <00> <00> <00> <00> <9D> <C5>

4. Write command to start background calibration. Parameter for background calibration is 6 and for nitrogen calibration is 7. We write command 0x7C with parameter 0x06 to HR2. Starting address is 0x0001 and Register value 0x7C06. CRC calculated as 0xC76C is sent with low byte first.

Master Transmit:

<FE> <06> <00> <01> <7C> <06> <6C> <C7>

Slave Reply:

<FE> <06> <00> <01> <7C> <06> <6C> <C7>

5. Wait at least 2 seconds for standard sensor with 2 sec lamp cycle.

6. Read acknowledgement register. We use function 3 “Read Holding register” to read HR1. Starting address is 0x0000 and Quantity of registers is 0x0001. CRC calculated as 0x0590 is sent with low byte first.

Master Transmit:

<FE> <03> <00> <00> <00> <01> <90> <05>

Slave Reply:

<FE> <03> <02> <00> <20> <AD> <88>

Check that bit 5 (CI6) is 1. It is an acknowledgement of that the sensor has performed the calibration operation. The sensor may skip calibration; an example of a reason for this could be unstable signal due to changing CO<sub>2</sub> concentration at the moment of the calibration request.

## バックグラウンド校正のシーケンス:

センサは“何らかのアドレス”(0xFE)にアドレス指定されています。

- HR1に0を書込んで承認(ACK)をクリアします。開始アドレスは0x0000、レジスタ値は0x0000です。0xC59Dと計算されたCRCが最初にローバイトで送信されます。

マスタは以下を送信します:

<FE> <06> <00> <00> <00> <00> <9D> <C5>

スレーブは以下で返答します:

<FE> <06> <00> <00> <00> <00> <9D> <C5>

- バックグラウンド校正をスタートするコマンドを書込みます。バックグラウンド校正のパラメータは6、窒素による校正は7です。コマンド0x7Cをパラメータ0x06でHR1に書込みます。開始アドレスは0x0001、レジスタ値は0x7C06です。0x7C76Cと計算されたCRCが最初にローバイトで送信されます。

マスタは以下を送信します:

<FE> <06> <00> <01> <7C> <06> <6C> <C7>

スレーブは以下で返答します:

<FE> <06> <00> <01> <7C> <06> <6C> <C7>

- ランプ点灯サイクル2秒の標準センサについては、少なくとも2秒待ちます。

- 承認(ACK)レジスタを読出します。HR1を読出すためにはファンクション3“保持レジスタの読出し”を使います。開始アドレスは0x0000、レジスタの数は0x0001です。0x0590と計算されたCRCが最初にローバイトで送信されます。

マスタは以下を送信します:

<FE> <03> <00> <00> <00> <01> <90> <05>

スレーブは以下で返答します:

<FE> <03> <02> <00> <20> <AD> <88>

ビット5(CI6)が1であることを確認して下さい。これはセンサが校正作業を行ったことの承認(ACK)です。センサが校正をスキップする場合があります。その理由の一例としては、校正要求があった時点のCO<sub>2</sub>濃度が変動しており、信号が不安定な場合などです。

## Read ABC parameter, ABC PERIOD:

One of the ABC parameters, ABC PERIOD, is available for modification as it is mapped as a holding register. This example shows how to read ABC PERIOD by accessing HR32.

The sensor is addressed as "Any address" (0xFE).  
Read current setting of ABC PERIOD by reading HR32. We use function code 03 "Read Holding registers". Starting address is 0x001f and Quantity of Registers 0x0001. CRC calculated as 0xC3A1 is sent with low byte first.

Master Transmit:

<FE> <03> <00> <1F> <00> <01> <A1> <C3>

Slave Reply:

<FE> <03> <02> <00> <B4> <AC> <27>

In the slave reply we can see:

Address = 0xFE

Function code = 0x03

Byte count = 0x02

Register value = 0x00B4

CRC = 0x27AC

- We read 2 bytes (1 register of 16 bits)
- 0xB4 hexadecimal = 180 decimal;  
180 hours / 24 equals 7.5 days.
- CRC sent with low byte first

## Disable ABC function:

Disable the ABC function by setting ABC PERIOD to 0.  
The sensor is addressed as "Any address" (0xFE).

Function code 06 "Write Single Register" is used to write to HR32. Register address is 0x001f, register value 0x0000. CRC calculated as 0x03AC is sent with low byte first.

Master transmit:

<FE> <06> <00> <1F> <00> <00> <AC> <03>

Slave reply:

<FE> <06> <00> <1F> <00> <00> <AC> <03>

The reply can be seen as an echo of the transmitted sequence.

## Enable ABC function:

Enable the ABC function by setting ABC PERIOD to any value except 0. Set to 7.5 days in this example.

The sensor is addressed as "Any address" (0xFE).

Function code 06 "Write Single Registers" is used to write to HR32. Register address is 0x001f, register value 0x00B4 (7.5 days x 24 hours = 180<sub>10</sub>; 180<sub>10</sub> = 0xB4<sub>16</sub>).  
CRC calculated as 0x74AC is sent with low byte first.

Master transmit:

<FE> <06> <00> <1F> <00> <B4> <AC> <74>

Slave reply:

<FE> <06> <00> <1F> <00> <B4> <AC> <74>

The reply can be seen as an echo of the transmitted sequence.

## ABCパラメータ、ABCピリオドの読出し:

ABCパラメータのひとつ、ABCピリオドは保持レジスタとしてマップされており、変更が可能です。この例は、HR32にアクセスしてABCピリオドを読出す方法を示しています。

センサは“何らかのアドレス”(0xFE)にアドレス指定されています。  
HR32を読込んで現在のABCピリオドの設定を読出します。ファンクションコード03“保持レジスタの読出し”を使います。開始アドレスは0x001f、レジスタの数は0x0001です。0xC3A1と計算されたCRCが最初にローバイトで送信されます。

マスタは以下を送信します:

<FE> <03> <00> <1F> <00> <01> <A1> <C3>

スレーブは以下で返答します:

<FE< <03> <02> <00> <B4> <AC> <27>

スレーブの返答に以下の内容があります:

アドレス=0xFE

ファンクションコード=0x03

バイトカウント=0x02

レジスタ値=0x00B4

-2バイト(16ビットの1レジスタ)が必要です。

-0xB4、16進数 = 180、10進数)

180時間/24=7.5日

CRC=0x27AC

-CRCが最初にローバイトで送信されます。

## ABC機能の無効化:

ABCピリオドを0に設定することにより、ABC機能を無効化します。

センサは“何らかのアドレス”(0xFE)にアドレス指定されています。

ファンクションコード06“単レジスタの書込み”を使って、HR32を書込みます。レジスタアドレスは0x001f、レジスタの値は0x0000です。0x03ACと計算されたCRCが最初にローバイトで送信されます。

マスタは以下を送信します:

<FE> <06> <00> <1F> <00> <00> <AC> <03>

スレーブは以下で返答します:

<FE> <06> <00> <1F> <00> <00> <AC> <03>

返答は送信されたシーケンスのエコーとして現れます。

## ABC機能の有効化:

ABCピリオドを0以外の値に設定することにより、ABC機能を有効化します。この例では7.5日に設定します。

センサは“何らかのアドレス”(0xFE)にアドレス指定されています。

ファンクションコード06“単レジスタの書込み”を使って、HR32を書込みます。レジスタアドレスは0x001f、レジスタの値は0x00B4(7.5日x24時間=180<sub>10</sub>、180<sub>10</sub>=0xB4<sub>16</sub>)です。0x74ACと計算されたCRCが最初にローバイトで送信されます。

マスタは以下を送信します:

<FE> <06> <00> <1F> <00> <B4> <AC> <74>

スレーブは以下で返答します:

<FE> <06> <00> <1F> <00> <B4> <AC> <74>

返答は送信されたシーケンスのエコーとして現れます。

## Appendix B: Compatibility with Senseair Core and aSENSE Modbus definitions.

	aSENSE	K30	Senseair S8
IR1	MeterStatus	MeterStatus	MeterStatus
IR2	Alarm Status	Alarm Status	Alarm Status
IR3	Output Status	Output Status	Output Status
IR4	Space CO2	Space CO2	Space CO2
IR5	Space Temp	<i>Reserved</i>	<i>Reserved</i>
IR6	Channel 6	<i>Reserved</i>	<i>Reserved</i>
IR7	Channel 7	<i>Reserved</i>	<i>Reserved</i>
IR8	Channel 8	<i>Reserved</i>	<i>Reserved</i>
IR9	Channel 9	<i>Reserved</i>	<i>Reserved</i>
IR10	Channel 10	<i>Reserved</i>	<i>Reserved</i>
IR11	Channel 11	<i>Reserved</i>	<i>Reserved</i>
IR12	Channel 12	<i>Reserved</i>	<i>Reserved</i>
IR13	Meas status	<i>Reserved</i>	<i>Reserved</i>
IR14	Meas status	<i>Reserved</i>	<i>Reserved</i>
IR15	Meas status	<i>Reserved</i>	<i>Reserved</i>
IR16	Case	<i>Reserved</i>	<i>Reserved</i>
IR17	Case	<i>Reserved</i>	<i>Reserved</i>
IR18	Case	<i>Reserved</i>	<i>Reserved</i>
IR19	Case	<i>Reserved</i>	<i>Reserved</i>
IR20	Case	<i>Reserved</i>	<i>Reserved</i>
IR21	Case	<i>Reserved</i>	<i>Reserved</i>
IR22	Output 1	Output 1	Output PWM
IR23	Output 2	Output 2	<i>Reserved</i>
IR24	Output 3	<i>Reserved</i>	<i>Reserved</i>
IR25	Output 4	<i>Reserved</i>	<i>Reserved</i>
IR26	<i>Reserved</i>	<i>Reserved</i>	Type ID High
IR27	<i>Reserved</i>	<i>Reserved</i>	Type ID Low
IR28	<i>Reserved</i>	<i>Reserved</i>	Map version
IR29	<i>Reserved</i>	<i>Reserved</i>	FW version
IR30	<i>Reserved</i>	<i>Reserved</i>	Sensor ID High
IR31	<i>Reserved</i>	<i>Reserved</i>	Sensor ID Low
IR32	<i>Reserved</i>	<i>Reserved</i>	<i>Reserved</i>

Table 6: Input Registers compatibility



## 付録 B: センスエアコア(センサモジュール)とaSENSEのModbus定義の互換性

	aSENSE	K30	Senseair S8
IR1	MeterStatus	MeterStatus	MeterStatus
IR2	Alarm Status	Alarm Status	Alarm Status
IR3	Output Status	Output Status	Output Status
IR4	Space CO2	Space CO2	Space CO2
IR5	Space Temp	Reserved	Reserved
IR6	Channel 6	Reserved	Reserved
IR7	Channel 7	Reserved	Reserved
IR8	Channel 8	Reserved	Reserved
IR9	Channel 9	Reserved	Reserved
IR10	Channel 10	Reserved	Reserved
IR11	Channel 11	Reserved	Reserved
IR12	Channel 12	Reserved	Reserved
IR13	Meas status	Reserved	Reserved
IR14	Meas status	Reserved	Reserved
IR15	Meas status	Reserved	Reserved
IR16	Case	Reserved	Reserved
IR17	Case	Reserved	Reserved
IR18	Case	Reserved	Reserved
IR19	Case	Reserved	Reserved
IR20	Case	Reserved	Reserved
IR21	Case	Reserved	Reserved
IR22	Output 1	Output 1	Output PWM
IR23	Output 2	Output 2	Reserved
IR24	Output 3	Reserved	Reserved
IR25	Output 4	Reserved	Reserved
IR26	Reserved	Reserved	Type ID High
IR27	Reserved	Reserved	Type ID Low
IR28	Reserved	Reserved	Map version
IR29	Reserved	Reserved	FW version
IR30	Reserved	Reserved	Sensor ID High
IR31	Reserved	Reserved	Sensor ID Low
IR32	Reserved	Reserved	Reserved

表6: 入力レジスタの互換性

	aSENSE	K30	Senseair S8
HR1	Acknowledgement register	Acknowledgement register	Acknowledgement register
HR2	Command Register	Command Register	Command Register
HR3	<i>Reserved</i>	<i>Reserved</i>	<i>Reserved</i>
HR4	<i>Reserved</i>	<i>Reserved</i>	<i>Reserved</i>
HR5	<i>Reserved</i>	<i>Reserved</i>	<i>Reserved</i>
HR6	<i>Reserved</i>	<i>Reserved</i>	<i>Reserved</i>
HR7	<i>Reserved</i>	<i>Reserved</i>	<i>Reserved</i>
HR8	<i>Reserved</i>	<i>Reserved</i>	<i>Reserved</i>
HR9	<i>Reserved</i>	<i>Reserved</i>	<i>Reserved</i>
HR10	Set Point Adjustment	<i>Reserved</i>	<i>Reserved</i>
HR11	Set Point Adjustment	<i>Reserved</i>	<i>Reserved</i>
HR12	Set Point Adjustment	<i>Reserved</i>	<i>Reserved</i>
HR13	Set Point Adjustment	<i>Reserved</i>	<i>Reserved</i>
HR14	OUT1 MinLimit	<i>Reserved</i>	<i>Reserved</i>
HR15	OUT1 MaxLimit	<i>Reserved</i>	<i>Reserved</i>
HR16	OUT2 MinLimit	<i>Reserved</i>	<i>Reserved</i>
HR17	OUT2 MaxLimit	<i>Reserved</i>	<i>Reserved</i>
HR18	OUT4 MinLimit	<i>Reserved</i>	<i>Reserved</i>
HR19	OUT4 MaxLimit	<i>Reserved</i>	<i>Reserved</i>
HR20	OUT5 MinLimit	<i>Reserved</i>	<i>Reserved</i>
HR21	OUT5 MaxLimit	<i>Reserved</i>	<i>Reserved</i>
HR22	Override OUT1	<i>Reserved</i>	<i>Reserved</i>
HR23	Override OUT2	<i>Reserved</i>	<i>Reserved</i>
HR24	Override OUT3	<i>Reserved</i>	<i>Reserved</i>
HR25	Override OUT4	<i>Reserved</i>	<i>Reserved</i>
HR26	<i>Reserved</i>	<i>Reserved</i>	<i>Reserved</i>
HR27	<i>Reserved</i>	<i>Reserved</i>	<i>Reserved</i>
HR28	<i>Reserved</i>	<i>Reserved</i>	<i>Reserved</i>
HR29	<i>Reserved</i>	<i>Reserved</i>	<i>Reserved</i>
HR30	<i>Reserved</i>	<i>Reserved</i>	<i>Reserved</i>
HR31	<i>Reserved</i>	<i>Reserved</i>	<i>Reserved</i>
HR32	ABC period	ABC period	ABC period

Table 7: Holding Registers compatibility



Green cells pick out compatible registers.





Pink cells are introduced for Senseair S8 registers.

The product and product specification are subject to change without notice. Contact Senseair to confirm that the information in this product description is up to date.

[www.senseair.com](http://www.senseair.com)

	aSENSE	K30	Senseair S8
HR1	Acknowledgement register	Acknowledgement register	Acknowledgement register
HR2	Command Register	Command Register	Command Register
HR3	Reserved	Reserved	Reserved
HR4	Reserved	Reserved	Reserved
HR5	Reserved	Reserved	Reserved
HR6	Reserved	Reserved	Reserved
HR7	Reserved	Reserved	Reserved
HR8	Reserved	Reserved	Reserved
HR9	Reserved	Reserved	Reserved
HR10	Set Point Adjustment	Reserved	Reserved
HR11	Set Point Adjustment	Reserved	Reserved
HR12	Set Point Adjustment	Reserved	Reserved
HR13	Set Point Adjustment	Reserved	Reserved
HR14	OUT1 MinLimit	Reserved	Reserved
HR15	OUT1 MaxLimit	Reserved	Reserved
HR16	OUT2 MinLimit	Reserved	Reserved
HR17	OUT2 MaxLimit	Reserved	Reserved
HR18	OUT4 MinLimit	Reserved	Reserved
HR19	OUT4 MaxLimit	Reserved	Reserved
HR20	OUT5 MinLimit	Reserved	Reserved
HR21	OUT5 MaxLimit	Reserved	Reserved
HR22	Override OUT1	Reserved	Reserved
HR23	Override OUT2	Reserved	Reserved
HR24	Override OUT3	Reserved	Reserved
HR25	Override OUT4	Reserved	Reserved
HR26	Reserved	Reserved	Reserved
HR27	Reserved	Reserved	Reserved
HR28	Reserved	Reserved	Reserved
HR29	Reserved	Reserved	Reserved
HR30	Reserved	Reserved	Reserved
HR31	Reserved	Reserved	Reserved
HR32	ABC period	ABC period	ABC period

表7： 保持レジスタの互換性

-  グリーンの欄は互換レジスタを表示
-  ピンクの欄はセンスエアS8レジスタに導入

製品および製品仕様は予告なく変更される場合があります。当製品説明書に記載された情報が最新のものであるか否かについてはセンスエア社またはその正規代理店にご照会下さい。

[www.senseair.com](http://www.senseair.com)

