CO_2NTROL RS485 Sensors

Modbus RTU Programmer's Manual

Firmware version:

COOUM003

Important Notice

Rev.	Revision Date	Author	Change Description
00	15.03.2021	TCa/CS	Initial version
01	02.08.2021	CS	Update to COOUM002
02	02.09.2021	CS	Update to COOUM003

Table of Content

1 MODBUS RTU GENERAL INFORMATION 5

1.1 INTRODUCTION 5
1.2 HAMILTON ARC SENSORS: MODBUS COMMAND STRUCTURE 5
1.2.1 Modbus RTU: Definitions According to Modbus IDA 5
1.2.2 Command Structure 6
1.2.3 Modbus RTU Transmission Mode 6
1.2.4 Modbus RTU Message Framing 7
1.2.5 Modbus RTU CRC Checking 8
1.3 IMPLEMENTATION OF MODBUS RTU IN HAMILTON ARC SENSORS 9
1.4 MODBUS RTU FUNCTION CODES USED FOR ARC SENSORS 9
1.4.1 Modbus function code #3: Read Holding Registers 9
1.4.2 Modbus function code #4: Read Input Registers 10
1.4.3 Modbus Function Code #16: Write Multiple Registers 11
1.5 DATA FORMATS USED IN ARC SENSORS 12
1.5.1 Float 12
1.5.2 Character 13
1.5.3 Decimal 13
1.6 MODBUS RTU ERROR MESSAGES 14

2 CO2NTROL RS485 COMMANDS IN MODBUS RTU 15

2.1 GENERAL 15
2.2 OPERATOR LEVELS AND PASSWORDS 15
2.2.1 Reading / Setting Operator Level 15
2.2.2 Changing Passwords for Operator Level 16
2.3 CONFIGURATION OF THE SERIAL RS485 INTERFACE 16
2.3.1 Device Address 16
2.3.2 Baud Rate 17
2.3.3 Parity and Stop Bits 17
2.4 CONFIGURATION OF THE ANALOG INTERFACES 18
2.4.1 Available Analog Interfaces 18
2.4.2 Available Analog Interface Modes 18
2.4.3 Description of the Analog Interfaces 1 and 2 19
2.4.4 Selection of an Analog Interface Mode 19
2.4.5 Configuration of the 4-20 mA Interface 19
2.4.6 Reading the Internally Calculated Output Current 23
2.5 MEASUREMENT 24
2.5.1 Definition of Measurement Channels and Physical Units 24
2.5.2 Primary Measurement Channel 1 (CO_2) 26
2.5.3 Primary Measurement Channel 6 (Temperature) 27
2.5.4 Definition of the Measurement Status for PMC1 / PMC6 28
2.6 CONFIGURATION OF THE MEASUREMENT 29
2.6.1 Available Parameters 29
2.6.2 PA2: Air Pressure 30
2.6.3 PA3: Humidity 31
2.6.4 PA9: Moving Average 32
2.7 VERIFICATION 34
2.7.1 VPA2 – Air Pressure 35
2.7.2 VPA3 – Humidity 36
2.8 CALIBRATION 37
2.8.1 Available Calibration Points 37
2.8.2 Definitions of Calibration Points 38
2.8.3 Calibration Procedure 41
2.8.4 Reading the Calibration Status 49
2.8.5 Currently active Calibration Parameters 51
2.9 SENSOR STATUS 62
2.9.1 Temperature Ranges 62
2.9.2 Operating Hours, Counters and System Time 64
2.9.3 Warnings 66

2.9.4 Errors....68
2.9.5 Measurement values - Exceeds Temperature range, and CO2 measurement errors ..70
2.9.6 Measurement values – no CO2 measurement within the frontend is running....71
2.9.7 Measurement values – Hardware errors....71
2.9.8 Reading Definition of SIP and CIP....72
2.9.9 Quality Indicator....73
2.10 SENSOR IDENTIFICATION AND INFORMATION....74
2.10.1 General Information....74
2.10.2 Sensor Identification....74
2.10.3 Free User Memory Space....75
2.11 SYSTEM COMMANDS....76
2.11.1 Restore Factory Settings....76

3 APPENDIX....77

3.1 LIST OF TABLES....77
3.2 LIST OF FIGURES....79
3.3 ABBREVIATIONS....79

1 Modbus RTU General Information

1.1 Introduction

This document describes in detail the CO2NTROL RS485 Modbus RTU interface. It is addressed to software programmers.
The general information about Modbus command structures and its implementation in the Hamilton Arc Sensor family is described in detail in chapter 1.

1.2 Hamilton Arc Sensors: Modbus Command Structure

In the present manual, only the specific command structure for the CO2NTROL RS485 Sensor is described. It is valid for the firmware:

COOUM003

Please check the firmware version by reading register 1032.

This present definition of the command structure is an additional document to the Operating Instructions of the specific CO2NTROL RS485 Sensors. Before reading this manual, the operating instructions of the sensors should be read and understood.

1.2.1 Modbus RTU: Definitions According to Modbus IDA

The definitions in chapter 1.2 are an excerpt from the documents:

- "Modbus over serial line - Specification and Implementation Guide V1.02" and
- "Modbus Application Protocol Specification V1.1b"

For more detailed information please consult http://www.modbus.org.

Attention:

- In this manual the register counting starts per definition at address 1. Some Modbus master protocols operate with register-count starting at address 0. Usually, the Modbus master software translates the addressing. Thus, the register address of 2088 will be translated by Modbus master software to 2087 which is sent to the sensor (Modbus slave). This must be observed during programming. Please check the specifications of the Modbus master that you are using.
- Representation of data formats in this document:
  - decimal values are displayed as numbers without any prefix, for example 256
  - hexadecimal values are displayed as: 0x2A
  - ASCII-characters or ASCII strings are displayed as: "Text"

1.2.2 Command Structure

The Modbus application protocol defines a simple **P**rotocol **D**ata **U**nit (PDU) independent of the underlying communication layers:

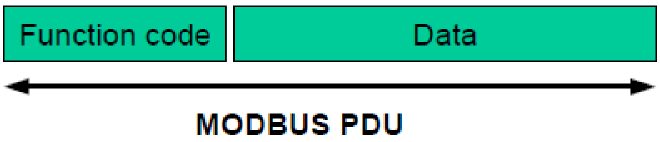


Figure 1 Modbus Protocol Data Unit

The mapping of Modbus protocol on a specific bus or network introduces some additional fields on the **P**rotocol **D**ata **U**nit. The client that initiates a Modbus transaction builds the Modbus PDU, and then adds fields in order to build the appropriate communication PDU.

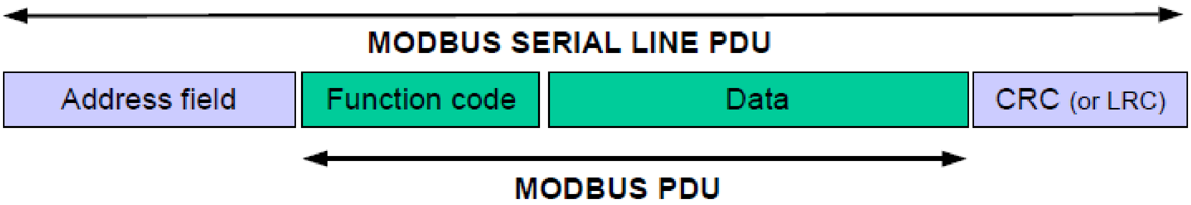


Figure 2 Modbus frame over Serial Line

- On Modbus Serial Line, the Address field only contains the slave address.

Note:

Arc Sensors support only slave addresses 1 to 32.

A master addresses a slave by placing the slave address in the address field of the message. When the slave returns its response, it places its own address in the response address field to let the master know which slave is responding.

- The function code indicates to the server what kind of action to perform. The function code can be followed by a data field that contains request and response parameters.

- The CRC field is the result of a "Redundancy Checking" calculation that is performed on the message contents.

1.2.3 Modbus RTU Transmission Mode

When devices communicate on a Modbus serial line using the RTU (Remote Terminal Unit) mode, each 8-bit byte in a message contains two 4-bit hexadecimal characters. The main advantage of this mode is that its greater character density allows better data throughput than ASCII mode for the same baud rate. Each message must be transmitted in a continuous stream of characters.

The format (11bits) for each byte in RTU mode is:

Coding System:	8 bit binary
Bits per Byte:	1 start bit
	8 data bits, least significant bit sent first
	1 bit for parity completion
	1 stop bit

Remark: The use of no parity requires 2 stop bits.

How characters are transmitted serially:
Each character or byte is sent in this order (left to right):
Least Significant Bit (LSB)…Most Significant Bit (MSB)

With Parity Checking

Start	1	2	3	4	5	6	7	8	Par	Stop

Figure 3 Bit sequence in RTU mode

Frame Checking Field:
Cyclical Redundancy Checking (CRC)

Frame description:

Slave Address	Function Code	Data	CRC
1 byte	1 byte	0 up to 252 byte(s)	2 bytes CRC Low CRC Hi

Figure 4 RTU Message Frame

=> The maximum size of a Modbus RTU frame is 256 bytes.

1.2.4 Modbus RTU Message Framing

A Modbus message is placed by the transmitting device into a frame that has a known beginning and ending point. This allows devices that receive a new frame to begin at the start of the message, and to know when the message is completed. Partial messages must be detected and errors must be set as a result.

In RTU mode, message frames are separated by a silent interval of at least 3.5 character times.

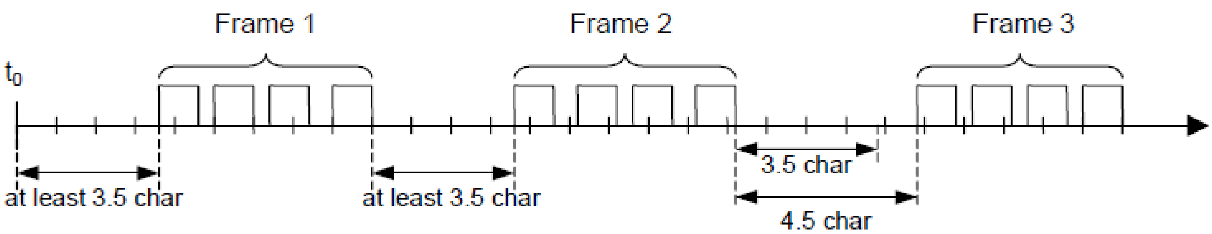


Figure 5 Valid frames with silent intervals

MODBUS message

Start		Address	Function	Data	CRC Check		End
≥ 3.5 char		8 bits	8 bits	N x 8 bits	16 bits		≥ 3.5 char

Figure 6 RTU Message Frame

The entire message frame must be transmitted as a continuous stream of characters.

If a silent interval of more than 1.5 character times occurs between two characters, the message frame is declared incomplete and should be discarded by the receiver.

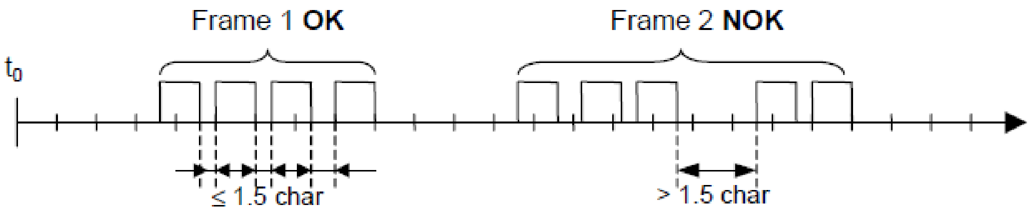


Figure 7 Data transmission of a frame

1.2.5 Modbus RTU CRC Checking

The RTU mode includes an error-checking field that is based on a Cyclical Redundancy Checking (CRC) method performed on the message contents.

The CRC field checks the contents of the entire message. It is applied regardless of any parity checking method used for the individual characters of the message.

The CRC field contains a 16-bit value implemented as two 8-bit bytes.

The CRC field is appended to the message as the last field in the message. When this is done, the low-order byte of the field is appended first, followed by the high-order byte. The CRC high-order byte is the last byte to be sent in the message.

The CRC value is calculated by the sending device, which appends the CRC to the message. The receiving device recalculates a CRC during receipt of the message, and compares the calculated value to the current value it received in the CRC field. If the two values are not equal, an error results.

The CRC calculation is started by first pre-loading a 16-bit register to all 1's. Then a process begins of applying successive 8-bit bytes of the message to the current contents or the register. Only the eight bits of data in each character are used for generating the CRC. Start and stop bits and the parity bit do not apply to the CRC.

During generation of the CRC, each 8-bit character is exclusive OR-ed with the register contents. Then the result is shifted in the direction of the least significant bit (LSB), with a zero filled into the most significant bit (MSB) position. The LSB is extracted and examined. If the LSB was a 1, the register is then exclusive OR-ed with a preset, fixed value. If the LSB was a 0, no exclusive OR takes place.

This process is repeated until eight shifts have been performed. After the last (eight) shift, the next 8-bit byte is exclusive OR-ed with the register's current value, and the process repeats for eight more shifts as described above. The final content of the register, after all the bytes of the message have been applied, is the CRC value.

When the CRC is appended to the message, the low-order byte is appended first, followed by the high-order byte.

A detailed introduction to CRC generation can be found in the manual
"MODBUS over Serial Line, Specification and Implementation Guide, V1.02"
in chapter 6.2 "Appendix B - LRC/CRC Generation" form http://www.modbus.org.

1.3 Implementation of Modbus RTU in Hamilton Arc Sensors

According to the official Modbus definition, the start of a command is initiated with a pause of ≥ 3.5 characters. Also the end of a command is indicated with a pause of ≥ 3.5 char.
The device address and the Modbus function code have 8 bits.
The data string consists of n*8 bits. The data string contains the starting address of the register and the number of registers to read/write.
The checksum CRC is 16 bits long.

Table 1 Modbus definition for data transmission

	start	**device address**	**function**	**data**	**Checksum**		**end**
value	no signal during ≥ 3.5 char	1-32	function code according to Modbus specs	data according to Modbus specs	CRC L	CRC H	No signal during ≥ 3.5 char
bytes	≥ 3.5	1	1	n	1	1	≥ 3.5

The RS485 interface is configured as follows:

Table 2 RS485 definitions for Arc Sensors

Modbus RTU implementation in Hamilton Arc Sensors	
Start Bits	1
Data Bits	8
Parity	none
Stop Bit	2
String length	11 Bits
Baud Rate	19200 (default), other baud rate can be configured

1.4 Modbus RTU Function Codes Used for Arc Sensors

Arc Sensors use only 3 Modbus function codes:

- # 3: Read Holding Registers
- # 4: Read Input Registers
- # 16: Write Multiple Registers

These three function codes are described below in detail using excerpts from "Modbus Application Protocol Specification V1.1b "(http://www.modbus.org).

1.4.1 Modbus function code #3: Read Holding Registers

This function code is used to read the contents of a contiguous block of holding registers in a remote device. The Request PDU specifies the starting register address and the number or registers. The PDU Registers are addressed starting at zero. Therefore registers numbered 1 – 16 are addressed as 0 – 15.

The register data in the response message are packed as two bytes per register. For each register, the first byte contains the high order bits and the second contains the low order bits.

Request

Function code	1 Byte	0x03
Starting Address	2 Bytes	0x0000 to 0xFFFF
Quantity of Registers	2 Bytes	1 to 125 (0x7D)

Response

Function code	1 Byte	0x03
Byte count	1 Byte	2 x **N***
Register value	**N*** x 2 Bytes	

*__N__ = Quantity of Registers

Error

Error code	1 Byte	0x83
Exception code	1 Byte	01 or 02 or 03 or 04

Figure 8 Definition of Holding Registers

Request		Response	
Field Name	*(Hex)*	*Field Name*	*(Hex)*
Function	03	Function	03
Starting Address Hi	00	Byte Count	06
Starting Address Lo	6B	Register value Hi (108)	02
No. of Registers Hi	00	Register value Lo (108)	2B
No. of Registers Lo	03	Register value Hi (109)	00
		Register value Lo (109)	00
		Register value Hi (110)	00
		Register value Lo (110)	64

Figure 9 Example of reading holding registers 108 – 110. The contents of register 108 are read as the two byte values 0x022B. The contents of registers 109 and 110 are 0x0000 and 0x0064

1.4.2 Modbus function code #4: Read Input Registers

The function code is used to read from 1 to 125 contiguous input registers in a remote device. The Request PDU specifies the starting register address and the number of registers. The PDU Registers are addressed starting at zero. Therefore input registers numbered 1 – 16 are addressed as 0 – 15.

The register data in the response message are packed as two bytes per register. For each register, the first byte contains the high order bits and the second contains the low order bits.

Request

Function code	1 Byte	0x04
Starting Address	2 Bytes	0x0000 to 0xFFFF
Quantity of Input Registers	2 Bytes	0x0001 to 0x007D

Response

Function code	1 Byte	0x04
Byte count	1 Byte	2 x **N***
Input Registers	**N*** x 2 Bytes	

*__N__ = Quantity of Input Registers

Error

Error code	1 Byte	0x84
Exception code	1 Byte	01 or 02 or 03 or 04

Figure 10 Definition of Input Registers

Request		Response	
Field Name	*(Hex)*	*Field Name*	*(Hex)*
Function	04	Function	04
Starting Address Hi	00	Byte Count	02
Starting Address Lo	08	Input Reg. 9 Hi	00
Quantity of Input Reg. Hi	00	Input Reg. 9 Lo	0A
Quantity of Input Reg. Lo	01		

Figure 11 *Example of reading input register 9. The contents of input register 9 are read as the two byte value 0x000A*

1.4.3 Modbus Function Code #16: Write Multiple Registers

This function code is used to write a block of contiguous registers (1 to 123 registers) in a remote device. The requested values are specified in the request data field. Data is packed as two bytes per register. The response returns the function code, starting address, and quantity of registers written.

Request

Function code	1 Byte	**0x10**
Starting Address	2 Bytes	0x0000 to 0xFFFF
Quantity of Registers	2 Bytes	0x0001 to 0x007B
Byte Count	1 Byte	2 x **N***
Registers Value	**N*** x 2 Bytes	value

***N** = Quantity of Registers

Response

Function code	1 Byte	**0x10**
Starting Address	2 Bytes	0x0000 to 0xFFFF
Quantity of Registers	2 Bytes	1 to 123 (0x7B)

Error

Error code	1 Byte	**0x90**
Exception code	1 Byte	01 or 02 or 03 or 04

Figure 12 Definition of Write Multiple Registers

Request		Response	
Field Name	*(Hex)*	*Field Name*	*(Hex)*
Function	10	Function	10
Starting Address Hi	00	Starting Address Hi	00
Starting Address Lo	01	Starting Address Lo	01
Quantity of Registers Hi	00	Quantity of Registers Hi	00
Quantity of Registers Lo	02	Quantity of Registers Lo	02
Byte Count	04		
Registers Value Hi	00		
Registers Value Lo	0A		
Registers Value Hi	01		
Registers Value Lo	02		

Figure 13 Example of writing the value 0x000A and 0x0102 to two registers starting at address 2

1.5 Data Formats Used in Arc Sensors

1.5.1 Float

Definition: Floating point according to IEEE 754 (Single Precision)

Table 3 Definition Floating point Single Precision (4 bytes resp. 2 Modbus registers)

Explanation:	sign	exponent	mantissa	total
Bit:	31	30 to 23	22 to 0	32
Exponent bias	127			

Example: translate the decimal value 62.85 into binary

Step 1: Conversion of the decimal value into binary fixed-point number

62 / 2	= 31	residue 0	LSB	**0.85** * 2	= 1.70	=> 1	MSB
31 / 2	= 15	residue 1		0.70 * 2	= 1.40	=> 1	
15 / 2	= 7	residue 1		0.40 * 2	= 0.80	=> 0	
7 / 2	= 3	residue 1		0.80 * 2	= 1.60	=> 1	
3 / 2	= 1	residue 1		0.60 * 2	= 1.20	=> 1	
1 / 2	= 0	residue 1	MSB	0.20 * 2	= 0.40	=> 0	
	= **111110**			0.40 * 2	= 0.80	=> 0	LSB
					…		
					= **0.11011001100110011001100**…		

62.85 = 111110.11011001100110011001100…

Step 2: Normalizing (in order to obtain 1 bit on the left side of the fraction point)

111110**.**11011001100110011001100… *2^**0** = 1**.**11110110110011001100110011001100… *2^**5**

Sep 3: Calculation of the dual exponent

2^5 => Exponent **5**
Exponent + Exponent bias = **5** + 127 = **132**

132 / 2	= 66	residue 0	LSB
66 / 2	= 33	residue 0	
33 / 2	= 16	residue 1	
16 / 2	= 8	residue 0	
8 / 2	= 4	residue 0	
4 / 2	= 2	residue 0	
2 / 2	= 1	residue 0	
1 / 2	= 0	residue 1	MSB
	= **10000100**		

Sep 4: Definition of the sign bit

Positive = 0
Negative = 1
= **0**

Step 5: conversion into floating-point

1 Bit Sign + 8 Bit Exponent + 23 Bit Mantissa
0 10000100 11110110110011001100110 (corresponds to 0x427B6666)

One important note for the 23 Bit Mantissa: The first bit (so-called hidden bit) is not represented. The hidden bit is the bit to the left of the fraction point. This bit is per definition always 1 and therefore suppressed.

Example: translate the binary float 0100 0010 0111 1011 0110 0110 0110 0110 to a decimal value

Step 1: Separating the binary value into Sign, Exponent and Mantissa

0 10000100 11110110110011001100110
1 Bit Sign + 8 Bit Exponent + 23 Bit Mantissa

S: **0** binary = **0** (positive sign)
E: **10000100** binary = $1*2^7 + 0*2^6 + 0*2^5 + 0*2^4 + 0*2^3 + 1*2^2 + 0*2^1 + 0*2^0$
= 128 + 0 + 0 + 0 + 0 + 4 + 0 + 0
= **132**
M: **11110110110011001100110** binary = **8087142**

Step 2: Calculate the decimal value

$D = (-1)^S * (1.0 + M/2^{23}) * 2^{E-127}$
$= (-1)^0 * (1.0 + 8087142/2^{23}) * 2^{132-127}$
= 1 * 1.964062452316284 * 32
= **62.85**

1.5.2 Character

Definition: The numerical representation of characters is defined in 8-Bit ASCII-Code-Table (ANSI X3.4-1986). Accordingly, each Modbus register in Arc Sensors can store two ASCII characters.

Example: translate the ASCII-string "2076" to Hex representation

The following interpretation is made according to the ASCII Codes-Table:

"2" => ASCII code table=> 0x32 Low Byte
"**0"** => ASCII code table => 0x30
"**7"** => ASCII code table => 0x37
"**6"** => ASCII code table => 0x36 High Byte

"2076" => **0x36373032**

1.5.3 Decimal

Example: translate Decimal 2227169 to Hex

2227169 / 16	= 139198	residue 1	Low Byte
139198 / 16	= 8699	residue 14 => E	
8699 / 16	= 543	residue 11 => B	
543 / 16	= 33	residue 15 => F	
33/16	= 2	residue 1	
2 / 16	= 0	residue 2	High Byte
	= **0x21FBE1**		

1.6 Modbus RTU Error Messages

Here are listed the Modbus standard error-codes we have implemented in Arc Sensors.

Table 4 Implemented Error-Codes

Error-Code Hex	**Status-Text**
0x00	OK
0x01	Illegal function
0x02	Illegal data address
0x03	Illegal data value
0X04	Slave device failure

See "Modbus_Application_Protocol_V1.1b" (www.modbus.org) for details.

If a slave device failure exception occurs, try to repeat the command that has thrown the exception. If the exception remains, check the sensor status.

2 CO2NTROL RS485 Commands in Modbus RTU

2.1 General

In order to communicate with a CO2NTROL RS485 Sensor over Modbus RTU protocol a Modbus master terminal application software is needed. The Modbus RTU is an open standard and a number of free and commercial application toolkits are available.

This manual contains examples and illustrations from WinTECH Modbus Master ActiveX Control tool: WinTECH (www.win-tech.com) "Modbus Master OCX for Visual Basic". The Modbus Organization (www.modbus.org/tech.php) provides other links to a wide variety of Modbus terminal software.

In the present manual the addressing of the Modbus registers starts at 1. But the Modbus master protocol operates with register addresses starting at 0. Usually, the Modbus master software translates the addressing. Thus, the register address of 2090 will be translated by the Modbus master software to 2089 which is sent to the sensor (Modbus slave).

⚠ Attention:

When configuring and calibrating the sensor, please limit write operations to a reasonable number. More than 100'000 write operations will physically damage the memory of the sensor.

2.2 Operator levels and Passwords

2.2.1 Reading / Setting Operator Level

A CO2NTROL RS485 Sensor can be operated in three different operator levels. Each operator level allows a defined access to a specific set of commands.

Table 5 Definition of operator level and default passwords

Abbreviation	Description	Code (hex)	Password (default) (decimal)
U	User (lowest level)	0x03	0
A	Administrator	0x0C	18111978
S	Specialist	0x30	16021966

At each power up or processor reset, the operator level falls back to the default level U.

The active operator level can be read and written in register 4288.

Start register	Number of registers	Reg1 / Reg2	Reg3 / Reg4	Modbus function code	Read access	Write access
4288	4	Operator Level	Password	3, 4, 16	U/A/S	U/A/S

⚠ Attention:

If the password is wrong, the operator level falls back to operator level U. To make sure that the operator level switch was successful, read back register 4288.

2.2.2 Changing Passwords for Operator Level

The passwords for accessing the operator levels A and S can be modified by S (Specialist) only. U (User) and A (Administrator) have no right to change any password. If they try anyway, an illegal data address exception (0x02) is returned.
The new password will remain stored after power down.

Start register	Number of registers	Reg1 / Reg2	Reg3 / Reg4	Modbus function code	Read access	Write access
4292	4	Level	New password	16	None	S

2.3 Configuration of the serial RS485 Interface

Factory settings for the RS485 interface:

Table 6 RS485 factory settings

Start Bits	1
Data Bits	8
Parity	None
Stop Bits	2
Baud Rate	19200

2.3.1 Device Address

2.3.1.1 Reading and Writing the Device Address

The sensor specific device address can be read and written in register 4096.

Start register	Number of registers	Reg1 / Reg2	Modbus function code	Read access	Write access
4096	2	device address	3, 4, 16	U/A/S	S

The device address can be set by S (Specialist), default value is 1. If the address limits are not met when setting a new address, the former address stays active.

⚠ Attention:

The device address changes immediately, what means that the next Modbus access has to be done using the new address.

2.3.1.2 Reading the Device Address Limits

The device address limits can be read in register 4098.

Start register	Number of registers	Reg1 / Reg2	Reg3 / Reg4	Modbus function code	Read access	Write access
4098	4	Min. device address	Max. device address	3, 4	U/A/S	none

Device address limits are:
Minimal device address: 1
Maximal device address: 32

2.3.1.3 Broadcast

Independent from the selected device address, the sensor responds to broadcasted Modbus commands (address 0).

2.3.2 Baud Rate

2.3.2.1 Reading and Writing the Baud Rate

The baud rate can be read and written in register 4102.

Start register	Number of registers	Reg1 / Reg2	Modbus function code	Read access	Write access
4102	2	Baud rate code (see Table 7)	3, 4, 16	U/A/S	S

The code for the baud rate is defined as follows:

Table 7 Code for the baud rates

Baud rate	4800	9600	19200	38400	57600	115200
Code	2	3	4	5	6	7

The baud rate can be set by S (Specialist), default is 19200.

⚠ Attention:

If the baud rate limits are not met when setting a new baud rate, the former baud rate stays active.
The baud rate **does not** change before the next power up.

2.3.2.2 Reading the Baud Rate Limits

The baud rate limits can be read in register 4104.

Start register	Number of registers	Reg1 / Reg2	Reg3 / Reg4	Modbus function code	Read access	Write access
4104	4	Min. Baud rate code	Max. Baud rate code	3, 4	U/A/S	none

The baud rate limits are:
Minimal baud rate code: 2
Maximal baud rate code: 7

2.3.3 Parity and Stop Bits

Start register	Number of registers	Reg1 / Reg2 (hex)	Modbus function code	Read access	Write access
4108	2	Interface parameter	3, 4, 16	U/A/S	S

The interface parameter is coded as following:
0xAABBCCDD where
AA = no meaning (reading: 0x00)
BB = Parity (0x00: no parity, 0x01: even, 0x02: odd)
CC = Stop bits (0x00: 1 stop bit, 0x04: 2 stop bits)
DD = no meaning (reading 0x00)

The interface parameters do not change before the next power up!
When writing to register 4108 set Bytes AA and DD to 0x00.

⚠ Attention:

If one of the parameter limits is not met, the old configuration stays active! Parity option (even or odd) is only available with one stop bit (max. string length of 11 bits).
The configuration **does not** change before the next power up.

2.4 Configuration of the Analog Interfaces

Note:

The CO2NTROL RS485 Sensor does not have any analog interfaces itself. They are provided by an Arc Wi 2G Adapter BT. But the registers to configure these interfaces are available on the sensor. That means, that these registers can be read and written with or without Arc Wi 2G Adapter BT.

2.4.1 Available Analog Interfaces

Equipped with an Arc Wi 2G Adapter BT, the CO2NTROL RS485 Sensor has two individual physical analog interfaces that have identical functionalities, but can be configured independently from each other.

- Analog Output Interface 1 (AO1)
- Analog Output Interface 2 (AO2)

The number of analog interfaces is defined in register 4320.

Start register	Number of registers	Reg1 / Reg2 (uint)	Modbus function code	Read access	Write access
4320	2	Available analog interfaces	3, 4	U/A/S	none

The answer always is "0x03" meaning that there exists an Analog Interface 1 (AO1) and an Analog Interface 2 (AO2).

2.4.2 Available Analog Interface Modes

With register 4322, the available analog interface modes for AO1 and AO2 are defined

Start register	Number of registers	Reg1 / Reg2 (uint)	Reg3 / Reg4 (uint)	Reg5 / Reg6	Reg7 / Reg8	Modbus function code	Read access	Write access
4322	8	Available Analog Interface Modes for AO1	Available Analog Interface Modes for AO2	reserved	reserved	3,4	U/A/S	none

Register 4322 defines the analog interface modes available for AO1 and AO2. The analog interface modes are described in Table 8.

Table 8 Definition of the analog interface modes

Code (Hex)	Analog Interface Mode	Description
0x00	4-20 mA inactive	Analog interface is deactivated
0x01	4-20 mA fixed	Set to a constant output value for current loop testing See 2.4.5.7
0x02	4-20 mA linear	Linear output of measurement (PMC1 / 6)

The answer is a bitwise combination (OR) of the available modes defined in Table 8.
Reg1/Reg2 and Reg3/Reg4 always return "0x03" meaning that fixed and linear mode are available.
Reg5 to Reg8 return 0. How to select or change the analog interface mode, see 2.4.4.

2.4.3 Description of the Analog Interfaces 1 and 2

Register 4352 / 4480 contain the descriptions of AO1 / AO2 as plain text ASCII:

Start register	Number of registers	Reg1 to Reg8 (16 ASCII characters)	Modbus function code	Read access	Write access
4352	8	Description of AO1	3, 4	U/A/S	none
4480	8	Description of AO2	3, 4	U/A/S	none

2.4.4 Selection of an Analog Interface Mode

The analog interface mode of AO1 / AO2 is selected by programming the analog interface mode in register 4360 / 4488.

Start register	Number of registers	Reg1 / Reg2 (uint)	Modbus function code	Read access	Write access
4360	2	Analog interface mode for AO1	3, 4, 16	U/A/S	S
4488	2	Analog interface mode for AO2	3, 4, 16	U/A/S	S

For available interface modes see register 4322.

Only one bit can be set. Using not allowed interface mode codes will leave the selection unchanged.

2.4.5 Configuration of the 4-20 mA Interface

Note:

The configuration of AO1 / AO2 is only effective if register 4360 / 4488 (active analog interface mode) is set to the value 0x01 or 0x02.

2.4.5.1 Reading the Available Primary Measurement Channels to be mapped to the Analog Output

Start register	Number of registers	Reg1 / Reg2 (uint)	Modbus function code	Read access	Write access
4362	2	Available Primary Measurement Channels for AO1	3, 4	U/A/S	none
4490	2	Available Primary Measurement Channels for AO2	3, 4	U/A/S	none

Table 9 Code for selection of the primary measurement channel

Code (Hex)	Primary Measurement Channel (PMC)
0x01	PMC1
0x20	PMC6

Reading the available Primary Measurement Channels (PMC) always return the hexadecimal value of “0x21” meaning that PMC1 or PMC6 can be mapped to AO1 respectively AO2.

2.4.5.2 Selecting the Primary Measurement Channel to be mapped to the Analog Interface

Start register	Number of registers	Reg1 / Reg2 (uint)	Modbus function code	Read access	Write access
4364	2	Selected PMC for AO1	3, 4, 16	U/A/S	S
4492	2	Selected PMC for AO2	3, 4, 16	U/A/S	S

Write this register to change the mapped measurement channel to AO1 respectively AO2. Make sure that only one bit is set, according to Table 9. Writing 0 or an illegal code will leave the selection unchanged. Only one bit can be set!
Reading this register returns the selected PMC for AO1 respectively AO2 according to Table 9.

The factory setting for register 4364 is "0x01" mapping PMC1 to AO1.
The factory setting for register 4492 is "0x20" mapping PMC6 to AO2.

2.4.5.3 Reading the Minimal and Maximal Possible Physical Output Current

Register 4366/4494 delivers the limits of the physical output current for AO1/AO2.

Start register	Number of registers	Reg1 / Reg2 (float)	Reg3 / Reg4 (float)	Modbus function code	Read access	Write access
4366	4	Min physical output current for AO1 [mA]	Max physical output current for AO1 [mA]	3, 4	U/A/S	none
4494	4	Min physical output current for AO2 [mA]	Max physical output current for AO2 [mA]	3, 4	U/A/S	none

The limits are fixed to:
Minimum is 3.5mA
Maximum is 22 mA

Note:
Currents above 20 and below 4 mA indicate erroneous measurements or errors.

2.4.5.4 Reading the Minimal and Maximal Current for Measurement Value Output

Start register	Number of registers	Reg1 / Reg2 (float)	Reg3 / Reg4 (float)	Reg5 / Reg6 (float)	Modbus function code	Read access	Write access
4370	6	Min output for measurement value for AO1 [mA]	Max output for measurement values for AO1 [mA]	Mid output for measurement values for AO1 [mA]	3, 4	U/A/S	none
4498	6	Min output for measurement value for AO2 [mA]	Max output for measurement values for AO2 [mA]	Mid output for measurement values for AO2 [mA]	3, 4	U/A/S	none

These registers deliver the minimal, maximal and middle output current for AO1 respectively AO2 in mA during normal operation. They are fixed to 4, 20 and 12 mA.

2.4.5.5 Reading the Selected Physical Unit for Analog Interface

Start register	Number of registers	Reg1 / Reg2 (uint)	Modbus function code	Read access	Write access
4376	2	Selected physical unit of AO1 (see Table 14)	3, 4	U/A/S	none
4504	2	Selected physical unit of AO2 (see Table 14)	3, 4	U/A/S	none

Read the selected unit of the selected PMC of AO1 respectively AO2. The value returned is an unsigned integer that represents the unit according to Table 14.
The physical unit for the PMC is defined in Reg. 2090 or 2410 and applies automatically for 4-20 mA output.

2.4.5.6 Defining the Measurement Values for 4 and 20 mA Output

Start register	Number of registers	Reg1 / Reg2 (float)	Reg3 / Reg4 (float)	Reg5 / Reg6 (float)	Modbus function code	Read access	Write access
4378	6	Measurement value at Min Output Current (4 mA) for AO1	Measurement value at Max Output Current (20 mA) for AO1	Measurement value at Mid Output Current (12 mA) for AO1	3, 4, 16	U/A/S	S
4506	6	Measurement value at Min Output Current (4 mA) for AO2	Measurement value at Max Output Current (20 mA) for AO2	Measurement value at Mid Output Current (12 mA) for AO2	3, 4, 16	U/A/S	S

These registers define the relation between measurement value and output current in linear mode. Reg1/Reg2 define the measurement value at 4mA and Reg3/Reg4 define the measurement value at 20mA. Reg5/Reg6 do not affect the 4-20mA output. When writing, write 0 or any random value. When reading, Reg5/Reg6 return half of Min + Max.
The corresponding physical unit can be read in register 4376 / 4504 respectively in the corresponding PMC register (2090 for PMC1 and 2410 for PMC6).

⚠ Attention:

When assigning measurement values to 4-20 mA analog output by using register 4378 / 4506, you need to consider the following:

- The PMC you have mapped to AO1 / AO2 (register 4364 / 4492)
- The physical unit currently in use for the selected PMC (register 2090 for PMC1and register 2410 for PMC6.
- The measurement parameter 2 Air Pressure when %-vol or %-sat is selected as physical unit and PMC1 is mapped to AO1 or AO2.
- The measurement parameter 3 Humidity when %-vol is selected as physical unit and PMC1 is mapped to AO1 or AO2.

Therefore, when the operator redefines any of the above mentioned register, the definitions of the register 4378 / 4506 should be reviewed. If not, the current output at the 4-20 mA interfaces may suddenly be unexpected!

Example:
Register 4364 is set to 1 (PMC1 is mapped to AO1).
Register 2090 is set to 0x00000020 (the unit "%-sat" is assigned to PMC1).
Register 4378 is set to 0 and 20 (4 mA = 0 %-sat, 20 mA = 20 %-sat) as shown in Figure 14.
The measurement parameter 2 Air Pressure is set to 1000 mbar.
The sensor reads currently a value of 100 mbar carbon dioxide, the output at the 4-20 mA accordingly is 12 mA (10 %-sat).

The operator now changes the measurement parameter 2 Air Pressure to 500 mbar. Therefore, the output of AO1 increases to 20 mA (20 %-sat).

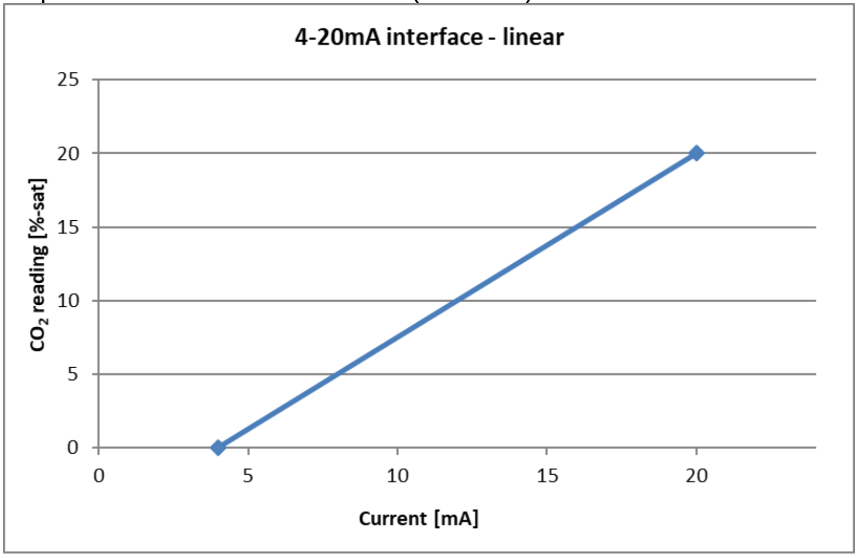


Figure 14 Example of linear 4-20mA output characteristics

2.4.5.7 Defining a Constant Current Output for Testing

Note:

For constant current output, the AO1 / AO2 must be set to analog interface mode 4-20 mA fixed (0x01) (see Table 8):

Start register	Number of registers	Reg1 / Reg2 (float)	Modbus function code	Read access	Write access
4384	2	Constant current output value for AO1 [mA]	3, 4, 16	U/A/S	S
4512	2	Constant current output value for AO2 [mA]	3, 4, 16	U/A/S	S

Values lower than 4mA respectively higher than 20mA will automatically be set within the limits.

2.4.5.8 Defining the Error and Warning Output of the 4-20 mA Interface

Errors and warnings can be mapped to the AO1 / AO2.

Start register	Number of registers	Reg1 / Reg2 (uint)	Reg3 / Reg4 (uint)	Reg5 / Reg6 (uint)	Reg7 / Reg8 (uint)	Modbus function code	Read access	Write access
4386	8	Code of warnings and errors for AO1	Current in case of "warning" for AO1 [mA]	Current in case of "error" for AO1 [mA]	Current in case of "T exceed" for AO1 [mA]	3, 4, 16	U/A/S	S
4514	8	Code of warnings and errors for AO2	Current in case of "warning" for AO2 [mA]	Current in case of "error" for AO2 [mA]	Current in case of "T exceed" for AO2 [mA]	3, 4, 16	U/A/S	S

Table 10 Code for the 4-20 mA interface in case of errors and warnings

Bit #	Code (hex)	Behavior of the 4-20 mA interface in case of errors and warnings
0 (LSB)	0x000001	Error continuous output
Einheitlich gestalten		
16	0x010000	Warning continuous output

If the corresponding bits for the errors and warnings are not set (=0x00), the respective options are inactive.
"T exceed" is always active. What means that in case of a measurement temperature limit violation, the output current will be as the specified value.

The default settings are:

- Code 0x01
- current in case of warnings: 3.5 mA
- current in case of errors: 3.5 mA
- current in case of measurement temperature limits violation: 3.5 mA

Table 11 Example: Read the settings for AO1 in case of warnings and errors

Command: ErrorWarnings AO1		Modbus address: **4386**	Length: **8**	Type: **3** Read
Parameter:	Warning code	Current in case of warning [mA]	Current in case of error [mA]	Current in case of temperature exceed [mA]
Format:	Hex	Float	Float	Float
Value:	**0x010001**	**3.5**	**3.5**	**3.5**

Warning code 0x010001 corresponds to the continuous output current in case of warning (0x010000) and continuous output current in case of error (0x01) of 3.5 mA. The output current in case of temperature exceed is 3.5 mA.

For more information about warnings, errors and temperature limits, see chapter 2.9.

2.4.6 Reading the Internally Calculated Output Current

Reg. 4414 / 4542 provides internal calculated output current of AO1 / AO2. These values are helpful in order to compare against the externally measured electrical current.

Start register	Number of registers	Reg1 / Reg2 (float)	Reg3 / Reg4 (float)	Modbus function code	Read access	Write access
4414	4	Set point [mA] AO1	Internally measured [mA] AO1	3, 4	U/A/S	none
4542	4	Set point [mA] AO2	Internally measured [mA] AO2	3, 4	U/A/S	none

Attention:

The current outputs are provided by the Arc Wi 2G Adapter BT. Therefore, the sensor cannot internally measure any output currents. Reg3/Reg4 always deliver the same value than Reg1/Reg2 even though there is no Arc Wi 2G Adapter BT connected. This is due to compatibility to other Arc Sensors.

2.5 Measurement

Note:

For more information about the measurement theory see the CO2NTROL RS485 Operating Instructions.

2.5.1 Definition of Measurement Channels and Physical Units

The Arc Modbus register structure allows the definition of 6 individual Primary Measurement Channels (PMC) and 16 individual Secondary Measurement Channels (SMC).

Table 12 Definition of PMC1 to 6 and SMC1 to 16

Bit #	Hex value	Description	Definition
0 (LSB)	**0x000001**	**PMC1**	**CO_2**
1	0x000002	PMC2	not available
2	0x000004	PMC3	not available
3	0x000008	PMC4	not available
4	0x000010	PMC5	not available
5	**0x000020**	**PMC6**	**Temperature**
6	0x000040	SMC1	not available
7-19	…	…	not available
20	0x100000	SMC15	not available
21 (MSB)	0x200000	SMC16	not available

In Register 2048, the available PMC and SMC are defined for a specific sensor and a specific operator level.

Start register	Number of registers	Reg1 / Reg2	Modbus function code	Read access	Write access
2048	2	Available measurement channels PMC and SMC (bitwise set)	3, 4	U/A/S	none

Table 13 Example to read register 2048

Command: Avail. meas. channels	Modbus address: **2048**	Length: **2**	Type: **3**	Read
Parameter:	Avail. PMC and SMC			
Format:	Hex			
Value:	**0x21**			

In case of operator level U, A or S, the value 0x21 is returned. In other words: PMC1 and PMC6 are available to the operator U, A or S.

The CO2NTROL RS485 Modbus register structure uses the following physical units.

Table 14 Definition of physical units

Bit #	Hex value	Physical unit	Start register. (8 ASCII characters, length 4 registers, Type 3, read for U/A/S)
0 (LSB)	0x00000001	none	1920
1	0x00000002	K	1924
2	0x00000004	°C	1928
3	0x00000008	°F	1932
4	0x00000010	%-vol	1936
5	0x00000020	%-sat	1940
6	0x00000040	ug/l ppb	1944
7	0x00000080	mg/l ppm	1948
8	0x00000100	g/l	1952
9	0x00000200	PCV	1956
10	0x00000400	mS/cm	1960
11	0x00000800	1/cm	1964
12	0x00001000	mmHg	1968
13	0x00002000	hPa	1972
14	0x00004000	kOhm	1976
15	0x00008000	MOhm	1980
16	0x00010000	pA	1984
17	0x00020000	nA	1988
18	0x00040000	uA	1992
19	0x00080000	mA	1996
20	0x00100000	uV	2000
21	0x00200000	mV	2004
22	0x00400000	V	2008
23	0x00800000	mbar	2012
24	0x01000000	Pa	2016
25	0x02000000	Ohm	2020
26	0x04000000	%/K	2024
27	0x08000000	°	2028
28	0x10000000	e6 c/ml	2032
29	0x20000000	%	2036
30	0x40000000	not used	2040
31 (MSB)	0x80000000	OD	2044

Table 15 Example to read the physical unit in plain text ASCII in register 1952

Command: Unit text		Modbus address: **1952**	Length: **4**	Type: **3** Read
Parameter:	Text			
Format:	8 ASCII characters			
Value:	**"g/l"**			

2.5.2 Primary Measurement Channel 1 (CO_2)

2.5.2.1 Definition of PMC1

Start register	Data size (registers)	Function	Data type	Modbus function code	Read access	Write access
2080	8	Description of PMC1	ASCII chars	3, 4	U/A/S	none
2088	2	Available physical units of PMC1	uint	3, 4	U/A/S	none
2090	2	Selected physical unit for PMC1	uint	16	none	S

In register 2080, a plain text ASCII description of PMC1 is given. PMC1 for CO2NTROL RS485 is called "CO2".

In register 2088, the available physical units for this channel are defined. The available physical units for PMC1: 0x008030F0 → %-vol, %-sat, µg/l ppb, mg/l ppm, mmHg, hPa and mbar

In register 2090, the active physical unit for this channel can be selected, by choosing one of the physical units that are defined in register 2088.
Selecting an invalid unit code will leave the current unit unchanged.

Table 16 Example to set the physical unit of PMC1 to %-vol (0x00000010)

Command: PMC1 set unit		Modbus address: **2090**	Length: **2**	Type: **16** Write
Parameter:	Unit			
Format:	Hex			
Value:	**0x00000010**			

⚠ Attention:

Changing the physical unit of PMC1 has also an influence on the output of the 4-20 mA analog output, as the same physical unit is active for 4-20 mA. All limits of the 4-20 mA analog output have to be redefined after changing the physical unit! See chapter 2.4.5 for more details.

2.5.2.2 Reading the measurement value of PMC1

Register 2090 is also used to read the measurement values of PMC1.

Start reg.	Num-ber of reg.	Reg1 / Reg2	Reg3 / Reg4	Reg5 / Reg6	Reg7 / Reg8	Reg9 / Reg10	Modbus function code	Read access	Write access
2090	10	Selected physical unit	Measure-ment value of PMC1 (1)	Measure-ment status (2)	Min allowed value (1)	Max allowed value (1)	3, 4	U/A/S	none

(1) Value is always in the physical unit defined in register 2090/2.
(2) Definition of the status see chapter 2.5.4. All bits set to zero means no problem.

⚠ Attention:

You cannot read selectively the registers 3 and 4 for the measurement value only. You have to read the entire length of the command (10 registers) and extract the desired information.

2.5.3 Primary Measurement Channel 6 (Temperature)

The CO2NTROL RS485 sensor has a built-in temperature sensor (NTC22kΩ). This temperature sensor is to be used only for monitoring the sensor conditions, not for controlling the process temperature. Due to the infrared technology the temperature measurement is internally compensated.

2.5.3.1 Definition of PMC6

Start register	Data size (registers)	Function	Data type	Modbus function code	Read access	Write access
2400	8	Description of PMC6	ASCII chars	3, 4	U/A/S	none
2408	2	Available physical units of PMC6	uint	3, 4	U/A/S	none
2410	2	Selected physical unit for PMC6	uint	16	none	S

In register 2400, a plain text ASCII description of PMC6 is given. PMC6 is called "T".

In register 2408, the available physical units for this channel are defined. The available physical units for PMC6: 0x0000000E => K, °C and °F

In register 2410, the active physical unit for this channel can be selected, by choosing one of the physical units that are defined in register 2408.
Selecting an invalid unit code will leave the current unit unchanged.

Attention:
Changing the physical unit of PMC6 has also an influence on the output of AO1 / AO2, as the same physical unit is active for the analog outputs. All limits of the 4-20 mA analog output have to be redefined after changing the physical unit! See chapter 2.4.5 for more details.

2.5.3.2 Reading the measurement value of PMC6

Register 2410 is also used to read the measurement values of PMC6.

Start reg.	Num-ber of reg.	Reg1 / Reg2	Reg3 / Reg4	Reg5 / Reg6	Reg7 / Reg8	Reg9 / Reg10	Modbus function code	Read access	Write access
2410	10	Selected physical unit	Measure-ment value of PMC6 (1)	Measure-ment status (2)	Min allowed value (1)	Max allowed value (1)	3, 4	U/A/S	none

(1) Value is always in the physical unit defined in register 2410/2.
(2) Definition of the status see chapter 2.5.4. All bits set to zero means no problem.

Table 17 Example to read register 2410

Command: PMC6 read		Modbus address: **2410**		Length: **10** Type: **3**	Read
Parameter:	Unit	Value	Status	Min limit	Max limit
Format:	Hex	Float	Hex	Float	Float
Value:	**0x04**	**27.42447**	**0x00**	**-10**	**140**

Physical unit is set to °C, PMC6 is 27.42 (°C), Status is 0x00, Min allowed value is -10 °C, Max allowed value is 140 °C.

Attention:
You cannot read selectively the registers 3 and 4 for the measurement value only. You have to read the entire length of the command (10 registers) and extract the desired information.

2.5.4 Definition of the Measurement Status for PMC1 / PMC6

This is the definition of the Measurement Status for the registers 2090 (PMC1) and 2410 (PMC6):

Table 18 Definition of measurement status for Primary Measurement Channels

Bit #	Hex value	Description
0 (LSB)	0x01	Temperature out of user defined measurement temperature range (see register 4624 in chapter 2.9.1)
1	0x02	Temperature out of operating range (see register 4608 in chapter 2.9.1)
2	0x04	Not available
3	0x08	Warning not zero (see register 4736 chapter 2.9.3)
4	0x10	Error not zero (see register 4800 chapter 2.9.4)

2.6 Configuration of the Measurement

This chapter describes the configuration of PMC1 and PMC6 by means of measurement parameters (PA).

2.6.1 Available Parameters

In register 3072, all available parameters (PA) are given.

Start register	Number of registers	Reg1 / Reg2 (bitwise defined)	Modbus function code	Read access	Write access
3072	2	Available parameters (see Table 19)	3, 4	U/A/S	none

Table 19 Bitwise definition of all parameters PA1 to PA16

Bit #	Hex value	Description	Definition
0 (LSB)	0x0001	PA1	Salinity
1	**0x0002**	**PA2**	**Air pressure**
2	**0x0004**	**PA3**	**Humidity**
3	0x0008	PA4	not available
4-6	…	…	not available
7	0x0080	PA8	not available
8	**0x0100**	**PA9**	**Moving average**
9	0x0200	PA10	not available
10-13	…	…	not available
14	0x4000	PA15	not available
15 (MSB)	0x8000	PA16	not available

Table 20 Example to read the available Parameters with operator level S

Command: Available parameters		Modbus address: **3072**	Length: **2**	Type: **3** Read
Parameter:	Measurement parameters			
Format:	Hex			
Value:	**0x107**			

The hex value 0x107 corresponds to 0x01 (PA1) + 0x02 (PA2) + 0x04 (PA3) + 0x0100 (PA9).

Attention:

The measurement Parameter 1 (Salinity) is read only and always 0.

Note:

- PA1 to PA8 use FLOAT as data format for its values
- PA9 to PA16 use UNSIGNED INT as data format for its values

2.6.2 PA2: Air Pressure

The measuring parameter 2 Air Pressure defines the current air pressure and is used to calculate the volume fraction (%-vol or %-sat) of carbon dioxide in gas or liquid phase.

2.6.2.1 Description of PA2 (Air Pressure)

In register 3136, a plain text ASCII description of PA2 is given. It returns “Air pressure”.

Start register	Number of registers	Reg1 to Reg8 16 ASCII characters	Modbus function code	Read access	Write access
3136	8	Description of PA2	3, 4	U/A/S	none

2.6.2.2 Selecting the Physical Unit and Writing the Value for PA2

In register 3144, the available physical units for PA2 are defined. The only one available here is mbar (0x800000). For the definition of the physical units see Table 14.

Start register	Number of registers	Reg1 / Reg2 (bitwise defined)	Modbus function code	Read access	Write access
3144	2	Available physical units for PA2	3, 4	U/A/S	none

By writing to register 3146/4, the active physical unit for PA2 can be selected, by choosing one of the physical units that are defined in register 3144. According to register 3144 only one bit for the physical unit can be set. The value of the parameter can be set as well.

Start register	Number of registers	Reg1 / Reg2 (bitwise defined)	Reg3 / Reg4	Modbus function code	Read access	Write access
3146	4	Select physical unit for PA2	Value for PA2 (10-12'000)	16	none	S

2.6.2.3 Reading All Values for PA2

By reading register 3146/8 the active physical unit, the current value, and the min and max allowed values can be read.
The unit is always mbar (0x800000), the min is 10 (mbar) and the max is 12000 (mbar).

Start register	Number of registers	Reg1 / Reg2	Reg3 / Reg4	Reg5 / Reg6	Reg7 / Reg8	Modbus function code	Read access	Write access
3146	8	Physical unit	Current value	Min value	Max value	3, 4	U/A/S	none

2.6.3 PA3: Humidity

The measuring parameter 3 Humidity defines the current humidity and is used to calculate the volume fraction in %-vol of carbon dioxide in gas phase.
As the physical units %-sat, µg/l ppb and mg/l ppm should only be used for liquid media, the measuring parameter 3 Humidity has no influence on these units.

2.6.3.1 Description of PA3 (Humidity)

In register 3168, a plain text ASCII description of PA3 is given. It returns "Humidity".

Start register	Number of registers	Reg1 to Reg8 16 ASCII characters	Modbus function code	Read access	Write access
3168	8	Description of PA3	3, 4	U/A/S	none

2.6.3.2 Selecting the Physical Unit and Writing the Value for PA3

In register 3176, the available physical units for PA3 are defined. The only one available here is % (0x20000000). For the definition of the physical units see Table 14.

Start register	Number of registers	Reg1 / Reg2 (bitwise defined)	Modbus function code	Read access	Write access
3176	2	Available physical units for PA3	3, 4	U/A/S	none

By writing to register 3178/4, the active physical unit for parameter 3 can be selected, by choosing one of the physical units that are defined in register 3176. According to register 3176 only one bit for the physical unit can be set. The value of the parameter can be set as well.

Start register	Number of registers	Reg1 / Reg2 (bitwise defined)	Reg3 / Reg4	Modbus function code	Read access	Write access
3178	4	Select physical unit for PA3	Value for PA3 (0-100)	16	none	S

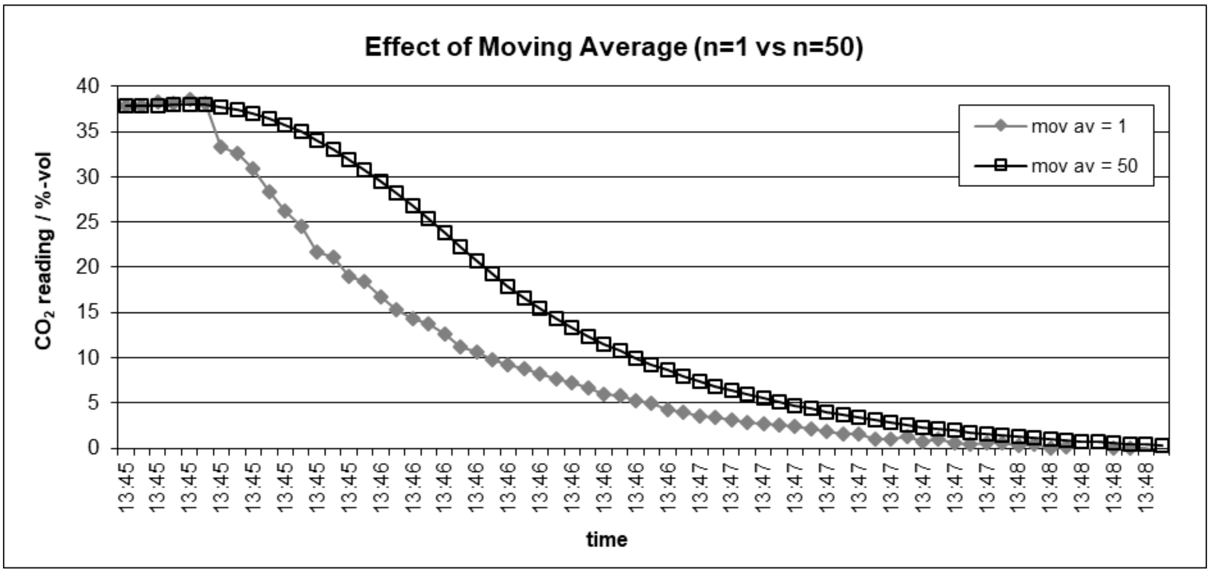
2.6.3.3 Reading all Values for PA3

By reading register 3178/8, the active physical unit, the current value, and the min and max allowed values can be read.
The unit is always % (0x20000000), the min is 0 (%) and the max is 100 (%).

Start register	Number of registers	Reg1 / Reg2	Reg3 / Reg4	Reg5 / Reg6	Reg7 / Reg8	Modbus function code	Read access	Write access
3178	8	Physical unit	Current value	Min value	Max value	3, 4	U/A/S	none

2.6.4 PA9: Moving Average

CO2NTROL RS485 calculates new CO_2 readings with a measurement interval of 1 second. One has the possibility to smoothen the CO_2 and Temperature reading (PMC1 and PMC6) by means of a moving average.
Figure 15 shows a comparison between no moving average (n=1) and a moving average over 50 readings at a measurement interval of 1 second.



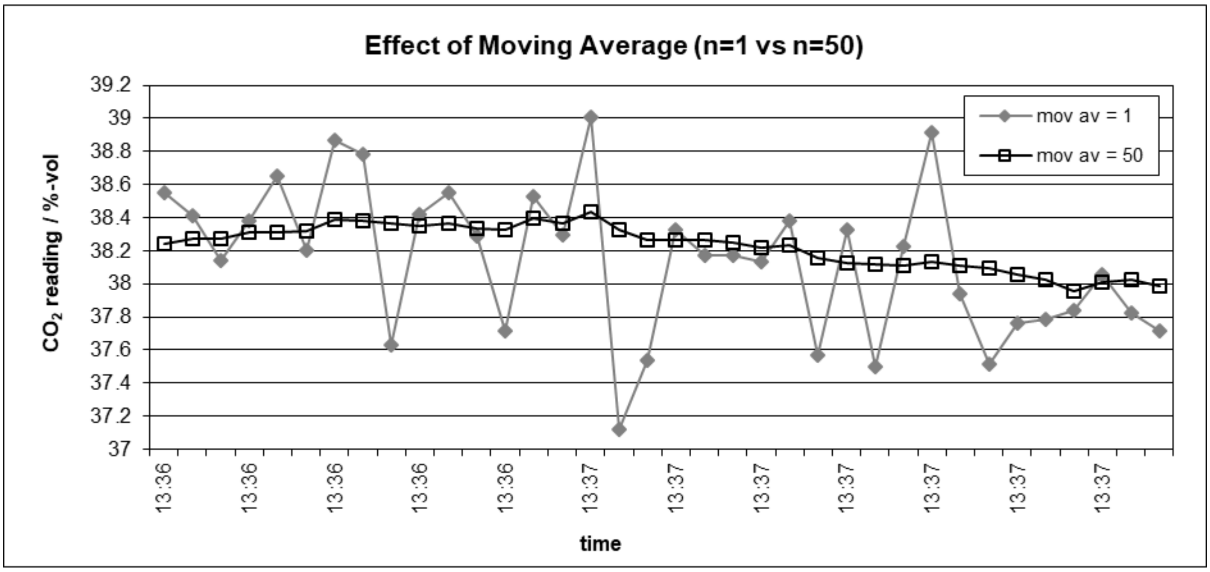


Figure 15 Comparison of the response of CO2NTROL RS485 to a change from 38.5 %-vol to zero CO_2

Using moving average, the short term signal stability can be improved; on the other hand, the response time of the sensor increases with increasing moving average. A moving average over 50 samples results in a response time of at least 50 times the measurement interval of 1 second.

Note:

The moving average defined by PA9 is applied to both PMC1 and PMC6.

2.6.4.1 Description of PA9 (Moving Average)

In register 3360, a plain text ASCII description of PA9 is given. It returns "Moving average".

Start register	Number of registers	Reg1 to Reg8 16 ASCII characters	Modbus function code	Read access	Write access
3360	8	Description of PA9	3, 4	U/A/S	none

2.6.4.2 Selecting the Physical Unit and Writing the Value for PA9

In register 3368, the available physical units for PA9 are defined. The only one available here is "none" (0x01). For the definition of the physical units see Table 14.

Start register	Number of registers	Reg1 / Reg2 (bitwise defined)	Modbus function code	Read access	Write access
3368	2	Available physical units for PA9	3, 4	U/A/S	none

By writing to register 3370/4, the active physical unit for PA9 can be selected, by choosing one of the physical units that are defined in register 3368. According to register 3368 only one bit for the physical unit can be set. The value of the parameter can be set as well.

Start register	Number of registers	Reg1 / Reg2 (bitwise defined)	Reg3 / Reg4	Modbus function code	Read access	Write access
3370	4	Select physical unit for PA9	Value for PA9 (1-150, default: 50)	16	none	S

PA9 can take values between 1 and 150. The value of 1 does not influence the response time of the sensor, the value of 50 increases the response time 50 times the value of the measurement interval.

2.6.4.3 Reading all Values for PA9

By reading register 3370/8, the active physical unit of measurement, the current value, and the min and max values can be read.
The unit is always none (0x01), the min is 1 and the max is 150.

Start register	Number of registers	Reg1 / Reg2	Reg3 / Reg4	Reg5 / Reg6	Reg7 / Reg8	Modbus function code	Read access	Write access
3370	8	Physical unit	Current value	Min value	Max value	3, 4	U/A/S	none

2.7 Verification

The main idea of the verification parameter (VPA), is to verify the current sensor measurement value with the last sensor calibration.
Performing such a verification, measurement parameters that are also defined within the standard calibration points, has to be adjust in the same way as in the last valid standard calibration point. After adjusting the measurement parameters, the current measurement value can be compared with the last valid standard calibration value.

The verification process can be done, adjusting the measurement parameter via the measurement parameter command, the value is stored persistent into the sensor. In case that the connection between sensor and host is interrupted, the current measurement parameter are modified and a possible fermentation process can run with wrong measurement parameter, even after a power up of the sensor.

The feature of verification parameter has to be switched on or off, with the command in register 3684. The verification command is not stored persistent into the sensor, a restart of the sensor the verification command is automatically switched off.

To activate the verification feature, it has to be switched on in register 3684 with the verification command code 0x00000001 (Start Verification) as shown in Table 22. After the activation of the verification feature, only then a write process for VPA2 (Air Pressure) and VPA3 (Humidity) is possible. If this is not the case, a Modbus exception with the error code illegal function 0x01 is generated.

By sending the verification command code 0x00000000 (Stop Verification) in register 3684 the verification feature is deactivated and the sensor parameter values are restored again out from the sensor memory.

Start register	Number of registers	Reg1 / Reg2 (bitwise defined)	Modbus function code	Read access	Write access
3684	2	Verification Command (see Table 21)	3, 4, 16	U/A/S/ D/H	S/D/H

Table 21: Bitwise definition of the verification command

Hex value	Description
0x00000000	Stop Verification
0x00000001	Start Verification
0x00000002 – 0xFFFFFFFF	Not Used

Table 22 Example to start the verification process

Command: Verification Command		Modbus address: **3684**	Length: **2**	Type: **16** Write
Parameter:	Verifi. Com.			
Format:	Hex			
Value:	**0x00000001**			

2.7.1 VPA2 – Air Pressure

In register 3700, a plain text ASCII description of Verification Parameter VPA2 is given. It returns "Air Pressure".

Start register	Number of registers	Reg1 to Reg8 16 ASCII characters	Modbus function code	Read access	Write access
3700	8	Description of Verification Parameter	3, 4	U/A/S/ D/H	none

Table 23 Example to read the description of verification parameter 2

Command: VPA2 – text		Modbus address: **3700**	Length: **8**	Type: **3** Read
Parameter:	Text			
Format:	Character			
Value:	**Air Pressure**			

In register 3708, Verification Parameter Air Pressure value is given.

Start register	Number of registers	Reg1 / Reg2	Reg3 / Reg4	Reg5 / Reg6	Modbus function code	Read access	Write access
3708	6	Not relevant	Unit	Verification Parameter Air Pressure value	3, 4, 16	U/A/S/ D/H	S/D/H

Table 24 Example to set the value of the verification parameter 2 to 200 mbar

Command: VPA2 – Set Value		Modbus address: **3708**	Length: **6**	Type: **16** Write
Parameter:	VPA2 - Not relevant	VPA2 - Unit	VPA2 – Value	
Format:	Hex	Hex	Float	
Value:	**0x00000000**	**0x00800000**	**200**	

2.7.2 VPA3 – Humidity

In register 3700, a plain text ASCII description of Verification Parameter VPA3 is given. It returns "Humidity".

Start register	Number of registers	Reg1 to Reg8 16 ASCII characters	Modbus function code	Read access	Write access
3714	8	Description of Verification Parameter	3, 4	U/A/S/ D/H	none

Table 25 Example to read the description of verification parameter 3

Command: VPA3 – text		Modbus address: **3714**	Length: **8**	Type: **3** Read
Parameter:	Text			
Format:	Character			
Value:	**Humidity**			

In register 3722, Verification Parameter Humidity value is given.

Start register	Number of registers	Reg1 / Reg2	Reg3 / Reg4	Reg5 / Reg6	Modbus function code	Read access	Write access
3722	6	Not relevant	Unit	Verification Parameter Humidity value	3, 4, 16	U/A/S/ D/H	S/D/H

Table 26 Example to set the value of the verification parameter 3 to 10 %

Command: VPA3 – Set Value		Modbus address: **3708**	Length: **6**	Type: **16** Write
Parameter:	VPA3 - Not relevant	VPA3 - Unit	VPA3 – Value	
Format:	Hex	Hex	Float	
Value:	**0x00000000**	**0x20000000**	**10**	

2.8 Calibration

2.8.1 Available Calibration Points

In register 10256, the available number of Calibration Points (CP) for Primary Measurement Channel 1 (PMC1) are defined. As shown in Table 27, six individual CPs are theoretically possible.

Start register	Number of registers	Reg1 / Reg2 (bitwise defined)	Modbus function code	Read access	Write access
10256	2	Available number of CP for PMC1 (see Table 27)	3, 4	U/A/S	none

Table 27 Bitwise definition of CP0 to CP6

Bit #	Hex value	Description	Definition in CO_2NTROL
0 (LSB)	0x01	CP0	not available
1	**0x02**	**CP1**	**Calibration low point**
2	**0x04**	**CP2**	**Calibration high point**
3	0x08	CP3	not available
4	0x10	CP4	not available
5	0x20	CP5	not available
6	**0x40**	**CP6**	**Product calibration**

Table 28 Example to read the available CPs

Command: Available cali points		Modbus address: **10256**	Length: **2**	Type: **3** Read
Parameter:	Points			
Format:	Hex			
Value:	**0x46**			

The hex value 0x46 in Table 28 corresponds to 0x02 (CP1) + 0x04 (CP2) + 0x40 (CP6).

As shown in Figure 16 the CO2NTROL RS485 Sensors allow 2 calibration points (CP1 and CP2) for the standard calibration. As shown in Figure 17 the product calibration with CP6 is used to adjust the standard calibration function to specific process conditions.

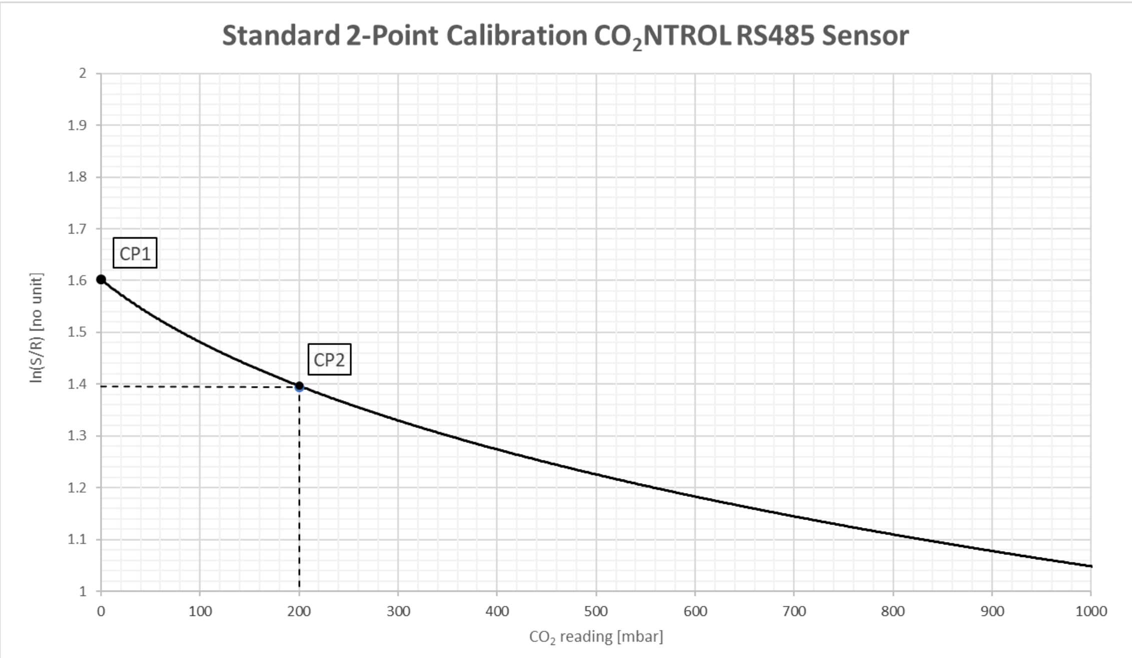


Figure 16 Standard 2-Point Calibration

2.8.2 Definitions of Calibration Points

2.8.2.1 Calibration Points 1 and 2 (Standard Calibration)

The available calibration units for calibration point 1 is defined in register 10322 and for calibration point 2 in register 11042.

Start register	Number of registers	Reg1 / Reg2	Modbus function code	Read access	Write access
10322	2	Available calibration units for CP1, see chapter 2.5.1	3, 4	U/A/S	none
11042	2	Available calibration units for CP2, see chapter 2.5.1	3, 4	U/A/S	none

Table 29 Example to read available calibration units of CP1

Command: Avail. Cali. Units CP1		Modbus address: **10322** Length: **2**	Type: **3**	Read
Parameter:	Avail. Cali. Units			
Format:	Hex			
Value:	**0x00800010**			

According to Table 29 the available physical units for CP1 are mbar and %-vol.

In register 10324 for calibration point 1 and register 11044 for calibration point 2, the active physical calibration unit can be selected, by choosing one of the physical units that are defined in register 10322 for CP1 and register 11044 for CP2. Only one bit can be set.

Start register	Number of registers	Reg1 / Reg2	Modbus function code	Read access	Write access
10324	2	Selected active physical calibration units for CP1, see chapter 2.5.1	16	none	S
11044	2	Selected active physical calibration units for CP2, see chapter 2.5.1	16	none	S

Table 30 Example to set the physical calibration unit of CP1 to %-vol (0x00000010)

Command: CP1 set unit		Modbus address: **10322** Length: **2**	Type: **16**	Write
Parameter:	Unit			
Format:	Hex			
Value:	**0x00000010**			

The limits for the calibration point 1 are defined in register 10318, limits for calibration point 2 in register 11038.

Start register	Number of registers	Reg1 / Reg2	Reg3 / Reg4	Modbus function code	Read access	Write access
10318	4	Min value for CP1 (in the physical unit as defined in register 10324/2)	Max value for CP1 (in the physical unit as defined in register 10324/2)	3, 4	U/A/S	none
11038	4	Min value for CP2 (in the physical unit as defined in register 11044/2)	Max value for CP2 (in the physical unit as defined in register 11044/2)	3, 4	U/A/S	none

Table 31 Example to read the limits of CP1 (active unit is mbar)

Command: Calibration limits CP1		Modbus address: **10318** Length: **4**	Type: **3**	Read
Parameter:	Min value	Max value		
Format:	Float	Float		
Value:	**0.0**	**80.0**		

Table 32 Example to read the limits of CP2 (active unit is mbar)

Command: Calibration limits CP2		Modbus address: **11038**	Length: **4**	Type: **3** Read
Parameter:	Min value	Max value		
Format:	Float	Float		
Value:	**80.0**	**1000.0**		

Table 33 Example to read the limits of CP2 (active unit is %-vol)

Command: Calibration limits CP2		Modbus address: **11038**	Length: **4**	Type: **3** Read
Parameter:	Min value	Max value		
Format:	Float	Float		
Value:	**7.9**	**98.7**		

As shown in Table 33, the min value is 7.9 (%-vol) and the max value is 98.7 (%-vol), which is compensated at current humidity (0 percent), at current ambient pressure (1013 mbar) and at current temperature (26.5 °C).

When initiating the calibration at CP1 and CP2, the measured SR and temperature have to be stable for at least 100 seconds. The stability criteria is defined in register 10278:

Start register	Number of registers	Reg1 / Reg2 (Float)	Reg3 / Reg4 (Float)	Modbus function code	Read access	Write access
10278	4	Max. Drift PMC1 CO2 [mbar/min]	Max. Drift PMC6 Temperature [K/min]	3, 4, 16	U/A/S	S

Table 34 Example to read the calibration stability

Command: Read calibration stability		Modbus address: **10278**	Length: **4**	Type: **3** Read
Parameter:	Max drift CO_2 [mbar/min]	Max drift temp [K/min]		
Format:	Float	Float		
Value:	**2.0**	**0.5**		

Table 35 Example to set the calibration stability

Command: Set calibration stability		Modbus address: **10278**	Length: **4**	Type: **16** Write
Parameter:	Max drift CO_2 [mbar/min]	Max drift temp [K/min]		
Format:	Float	Float		
Value:	**2.0**	**0.5**		

Attention:

The stability criteria defined in register 10278 is valid for CP1 and CP2 only, but NOT for CP6.

2.8.2.2 Calibration Point 6 (Product Calibration)

The available calibration units for calibration point 6 are defined in register 13922.

Start register	Number of registers	Reg1 / Reg2	Modbus function code	Read access	Write access
13922	2	Available calibration units for CP6, see chapter 2.5.1	3, 4	U/A/S	none

Table 36 Example to read available calibration units of CP6

Command: Avail. Cali. Units CP6		Modbus address: **13922**	Length: **2**	Type: **3** Read
Parameter:	Avail. Cali. Units			
Format:	Hex			
Value:	**0x008030F0**			

According to Table 36 the available physical units for CP6 are %-vol, %-sat, µg/l, mg/l, mmHg, hPa and mbar.

In register 13924 for calibration point 6, the active physical calibration unit can be selected, by choosing one of the physical units that are defined in register 13922. Only one bit can be set.

Start register	Number of registers	Reg1 / Reg2	Modbus function code	Read access	Write access
13924	2	Selected active physical calibration units for CP6, see chapter 2.5.1	16	none	S

Table 37 Example to set the physical calibration unit of CP6 to %-vol

Command: CP6 set unit		Modbus address: **13924**	Length: **2**	Type: **16** Write
Parameter:	Unit			
Format:	Hex			
Value:	**0x00000010**			

The limits for calibration point 6 are given in register 13918.

Start register	Number of registers	Reg1 / Reg2	Reg3 / Reg4	Modbus function code	Read access	Write access
13918	4	Min value for CP6 (in the physical unit as defined in register 13924/2)	Max value for CP6 (in the physical unit as defined in register 13924/2)	3, 4	U/A/S	none

Table 38 Example to read the limits of CP6

Command: Calibration limits CP6		Modbus address: **13918**	Length: **4**	Type: **3** Read
Parameter:	Min value	Max value		
Format:	Float	Float		
Value:	**0**	**1000**		

Note:

The physical unit of the limits is the unit of PC6 at the time of initial measurement. In Table 38 it is mbar (0x00800000), the min value is 0 (mbar) and the max value is 1000 (mbar). When changing the calibration physical unit for PC6 (using register 13924), the min and max value will be updated to the new physical unit. Temperature, atmospheric pressure and humidity are compensated.

2.8.3 Calibration Procedure

2.8.3.1 Calibration at CP1 and CP2 (Standard Calibration)

The CO2NTROL RS485 Sensor has a unique calibration routine. When initiating the calibration, the data set of the CO2NTROL RS485 Sensor is automatically traced back within the last 100 seconds and a decision is made immediately if the calibration is successful or not. The operator therefore gets an immediate result. The criteria for a successful calibration are:

- both CP1 and CP2 calibration points have to be performed
- the stability of SR and temperature over the last 100 seconds
- the SR value is in a reasonable window
- the CO_2 content is in the limits defined for CP1 / CP2

Note:

Recommended measurement stabilization time for each calibration point is at least 500 seconds.

Attention:

It is important that the CO2NTROL RS485 Sensor is in a defined calibration media at least 100 seconds BEFORE the calibration is started.

If the temperature is out of the user defined measurement temperature range, the procedure is as follows:

1. Send calibration command to the sensor. The sensor will return "SR reading during calibration is not stable" (Calibration status code 0x00002000) within the CP1 calibration status in register 10312/2 or within the CP2 calibration status in register 11032/2.
2. The sensor temporarily expands the user defined measurement temperature range to the maximum allowed for the sensor. 10 minutes after the last calibration command or after a power up, these settings are reset and the sensor runs with the originally entered values.
3. Send calibration command after at least 100s again to the sensor. If the stability is ok, the sensor returns "calibration successful" (0x00000000 in the status register), if the stability is not ok, the sensor sets the corresponding bit in the calibration status register (reg. 10312/2 or 11032/2).

The calibration is initiated at CP1 by writing to register 10314 and at CP2 by writing to register 11034. The CO_2 value has to be given in the physical unit defined within register 10324 for CP1 and register 11044 for CP2.

By entering 0x00000001 in Reg1 / Reg2 the automatic calibration for the respecting calibration point will be started, which calibrates at CP1 by 0.04 %-vol CO_2 (low point) and at CP2 by 20.0 %-vol CO_2 (high point). In automatic calibration mode the value in Reg3 / Reg4, is not taken in to account.

Start register	Number of registers	Reg1 / Reg2	Reg3 / Reg4	Modbus function code	Read access	Write access
10314	4	Calibration command code, see Table 39	Calibration value	3, 4, 16	U/A/S	S
11034	4	Calibration command code, see Table 39	Calibration value	3, 4 , 16	U/A/S	S

Table 39 Calibration command code

Bit #	Hex value	Description
none	0x00000000	Not valid (leads to a modbus exception)
0 (LSB)	0x00000001	Automatic calibration setting mode
1	0x00000002	Manual value setting mode
2	0x00000004	Not valid (leads to a modbus exception)
3-30	…	
31 (MSB)	0x80000000	

Table 40 Example to start the calibration at CP1 in automatic mode

Command: Make calibration CP1		Modbus address: **10314**	Length: **4**	Type: **16** Write
Parameter:	Command Code	Value		
Format:	Hex	Float		
Value:	**0x00000001**	**0.0**		

As shown in Table 40, the command code in Reg 1 / Reg 2 is 0x00000001. The calibration value in automatic mode for CP1 is 0.04 %-vol CO_2. In automatic calibration mode the value in Reg3 / Reg4, in this example 0.0, is not taken in to account.

Table 41 Example to start the calibration at CP2 in automatic mode

Command: Make calibration CP2		Modbus address: **11034**	Length: **4**	Type: **16** Write
Parameter:	Command Code	Value		
Format:	Hex	Float		
Value:	**0x00000001**	**0.0**		

As shown in Table 41 the command code in Reg 1 / Reg 2 is 0x00000001. The calibration value in automatic mode for CP2 is 20.0 %-vol CO_2. In automatic calibration mode the value in Reg3 / Reg4, in this example 0.0, is not taken in to account.

Table 42 Example to start the calibration at CP2 in manual mode

Command: Make calibration CP2		Modbus address: **11034**	Length: **4**	Type: **16** Write
Parameter:	Command Code	Value		
Format:	Hex	Float		
Value:	**0x00000002**	**10.0**		

As shown in Table 42 the command code in Reg 1 / Reg 2 is 0x00000002. The calibration value in Reg3 / Reg4, in this example 10.0, has to be considered and represents the calibration input value.

⚠ Attention:

CP1 is fixed to a calibration in air (0 – 80 mbar carbon dioxide) – low point, and CP2 is fixed to a calibration between 80 mbar and 1000 mbar carbon dioxide – high point.
There is an automatic mode for both CP1 and CP2. In this mode, the operator declares that CP1 is performed in air, and CP2 in 20 %-vol carbon dioxide. The operator does not need to care about the physical units that are currently active.

Example 1:
Register 11044 is set to physical unit "%-vol" (0x00000010).

Table 43 Example 1: Read the limits for CP2

Command: Calibration limits CP2		Modbus address: **11038**	Length: **4**	Type: **3** Read
Parameter:	Min value	Max value		
Format:	Float	Float		
Value:	**7.9**	**98.7**		

As shown in Table 43, the min value is 7.9 (%-vol) and the max value is 98.7 (%-vol), which is compensated at current humidity (0 percent), at current ambient pressure (1013 mbar) and at current temperature (26.5 °C).

Table 44 Example 1: Make CP2 calibration

Command: Make calibration CP2		Modbus address: **11034**	Length: **4**	Type: **16** Write
Parameter:	Command Code	Value		
Format:	Hex	Float		
Value:	**0x00000002**	**22.31**		

As shown in Table 44, the CP2 calibration is performed by telling the CO2NTROL RS485 Sensor, that the carbon dioxide partial pressure in the calibration media is 22.31 %-vol.

Table 45 Example 1: Read CO2 value

Command: PMC 1 read			Modbus address: **2090**	Length: **10**	Type: **3** Read
Parameter:	Unit	Value	Status	Min limit	Max limit
Format:	Hex	Float	Hex	Float	Float
Value:	**0x00000010**	**22.124**	**0x00**	**7.9**	**98.7**

Table 45 shows the carbon dioxide reading after calibration, which is 22.124 %-vol as desired.

Example 2:
Register 11044 is set to physical unit "%-vol" (0x00000010).

Table 46 Example 2: Make CP2 calibration

Command: Make calibration CP2		Modbus address: **11034**	Length: **4**	Type: **16** Write
Parameter:	Command Code	Value		
Format:	Hex	Float		
Value:	**0x00000001**	**0**		

According to Table 46, with the command code 0x00000001 a calibration in automatic mode at CP2 is started.

Table 47 Example 2: Read CO2 value

Command: PMC 1 read			Modbus address: **2090**	Length: **10**	Type: **3** Read
Parameter:	Unit	Value	Status	Min limit	Max limit
Format:	Hex	Float	Hex	Float	Float
Value:	**0x00000010**	**20.78**	**0x00**	**7.9**	**98.7**

As a result of the calibration and shown in Table 47, the carbon dioxide reading is 20.78 %-vol as the temperature reading at the same time is 26.5 °C. The theoretical concentration of CO2 in water with 26.5 °C, having a humidity of 0 % and current ambient pressure (1013 mbar) is 203.3 mbar (20.78 %-vol). Therefore the calibration was successful.

2.8.3.2 Calibration at CP6 (Product Calibration)

The product calibration is a process in order to adjust the measurement of a correctly calibrated CO2NTROL RS485 Sensor to specific process conditions.
The product calibration is a two stage process:

1. An initial measurement is performed while the operator takes a sample of the process solution. At that time point the CO2NTROL RS485 Sensor stores its raw measurement value, temperature and operating hour in the memory.
   While the operator takes the sample to the analytics lab for reference analysis the CO2NTROL RS485 Sensor is still running on its prior standard calibration (CP1 and CP2) while the initial measurement data for the ongoing product calibration is kept in the CO2NTROL RS485 Sensors memory.

2. When the result of the reference analysis is available this value is assigned at a second time point to the former initial measurement data stored in the CO2NTROL RS485 Sensor.
   The sensor is now, after valid assignment, running on a calibration function which is compensated for the correct process conditions. The product calibration (CP6) is now active.

Performing a Cancel command for the product calibration (CP6) brings the sensor back to its still stored standard calibration (CP1 and CP2).
If a product calibration is still active and a standard calibration (CP1 or CP2) is performed the product calibration (CP6) is cancelled.
If the operator needs to overrun an active product calibration by a new product calibration the above process applies in the same way. After initial measurement the CO2NTROL RS485 Sensor is still running on the first product calibration until a valid assignment has been done .

What happens to the CO2NTROL RS485 Sensors calibration function upon product calibration (CP6)?
As shown in figure 18, a product calibration for a CO2NTROL RS485 Sensor corresponds to an offset correction. On active product calibration (CP6) the CO2NTROL RS485 calibration function is calculated from the data of the factory calibration and from the data of the product calibration (CP6).
For more flexibility the limits of the product calibration procedure allow larger deviations than the standard calibration (CP1 and CP2).

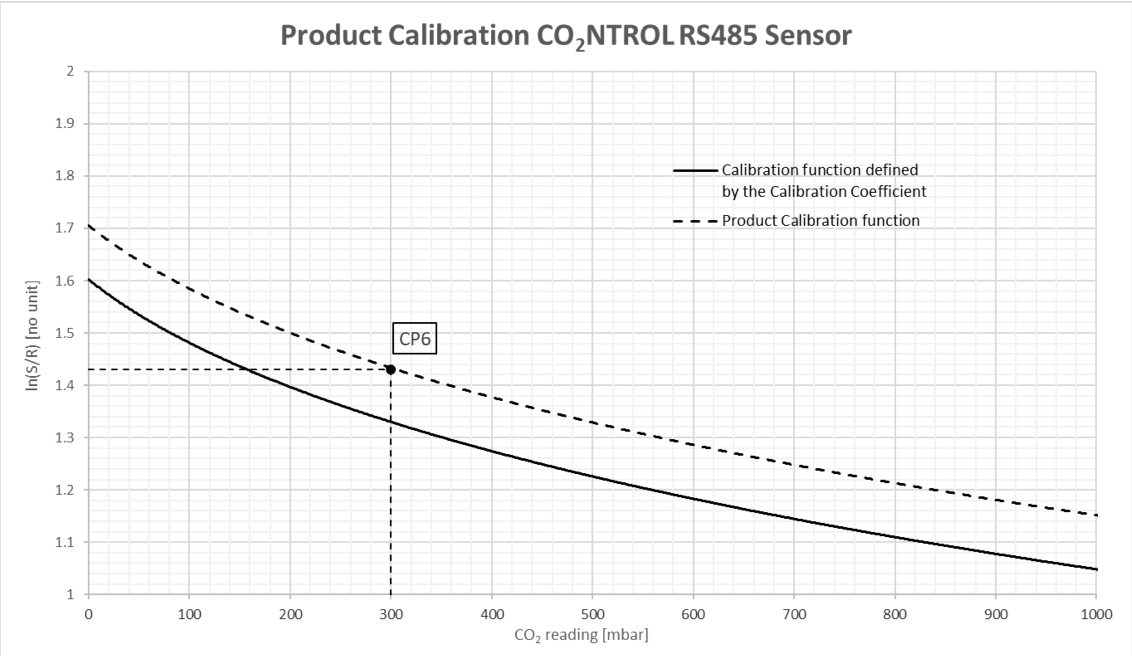


Figure 17 Effect of the product calibration CP6 on an existing factory calibration function

Example:
The operator starts with a standard calibration with calibration points CP1 and CP2:

CP1 → carbon dioxide value: 0 mbar, temperature: 27.77°C, measured SR value: 1.6
CP2 → carbon dioxide value: 200 mbar, temperature: 28.71°C, measured SR value: 1.396

The sensor internally calculates the calibration function, using the calibration points **CP1** and **CP2**. The resulting calibration function, is shown in Figure 17 as a straight line. The calibration function is described by two parameters: the SR value at CP1 readable via register 14640 and the SR value at CP2 readable via register 14672 (see chapter 2.8.5.4).
Some weeks later, the operator believes that the standard calibration function is not correct anymore. As the process is running and he is not able to perform a standard calibration under defined conditions in the lab, he decides to perform a product calibration CP6, in other words adjusting the standard calibration function to the process conditions:

CP6 → carbon dioxide value: 300 mbar, temperature: 28.79°C, measured SR value: 1.43

The sensor internally recalculates the calibration function using the factory calibration points. The new calibration function, is shown in Figure 17 as a dotted line.

Another special feature of the product calibration point is to switch it off and back on again. These functions are called "restore standard calibration" and "restore product calibration".

The sensor's internal criteria for a successful product calibration are:

- the sensor is currently in an environment corresponding to the CO2NTROL RS485 measurement range.
- the carbon dioxide content is within the calibration limits defined for CP6 (see Table 38)

The different functionalities of product calibration (CP6) are accessible through the following sensor commands:

- Initial measurement
- Assignment
- Cancel
- Restore standard calibration
- Restore product calibration

All commands are executed by writing a command to the register 13914. The commands are defined according to Table 48.

Start register	Number of registers	Reg1 / Reg2	Reg3 / Reg4	Modbus function code	Read access	Write access
13914	2	Code as defined in Table 48	Calibration value	3, 4, 16	U/A/S	S

Table 48 Definition of the commands related to the product calibration

Hex Code	Definition of commands
0x00000010	Perform initial measurement
0x00000020	Assign product calibration
0x00000040	Cancel an active product calibration
0x00000080	Restore a standard calibration from an active product calibration
0x00000100	Restore a product calibration from an active standard calibration

2.8.3.2.1 Product calibration: Initial measurement

Upon process sample collection for laboratory analysis the command for initial measurement is sent to the sensor according to Table 49.
This is achieved by writing the command 0x00000010 to register 13914, which performs the initial measurement and stores the corresponding measurement values in the sensor. The calibration value is not considered in this initialization phase and the value is not saved within the sensor.

Table 49 Example to start the product calibration procedure

Command: CP6: Initial measurement		Modbus address: **13914**	Length: **4**	Type: **16** Write
Parameter:	Command	Value		
Format:	Hex	Float		
Value:	**0x00000010**	**0.0**		

As shown in Table 50, after successful initial measurement the corresponding calibration status is "CP6: Initial measurement" (0x20000000). (see Table 59)
The sensor continues measuring using the prior standard calibration.

Table 50 Example to read the current product calibration status

Command: CP6: Initial measurement		Modbus address: **13912**	Length: **2**	Type: **3** Read
Parameter:	Command			
Format:	Hex			
Value:	**0x20000000**			

2.8.3.2.2 Product calibration: Assignment

After successful initial measurement a correct value must be assigned to the initially stored measurement data. As shown in Table 51, is this achieved by writing the carbon dioxide value in the unit of PMC1 during initial measurement (here 300 mbar) to 13914.

Table 51 Example to assign a calibration value to the above performed initial measurement

Command: CP6: Assignment		Modbus address: **13914**	Length: **4**	Type: **16** Write
Parameter:	Command	Value		
Format:	Hex	Float		
Value:	**0x00000020**	**300**		

From now on the sensor is measuring using the here performed product calibration.

According to Table 52 the calibration status of the sensor is 0x50000000 meaning that a correct value has been assigned and that the product calibration is active. (see Table 59)

Table 52 Example to read the current product calibration status, after performing a product calibration.

Command: CP6: Initial measurement		Modbus address: **13912**	Length: **2**	Type: **3** Read
Parameter:	Command			
Format:	Hex			
Value:	**0x50000000**			

2.8.3.2.3 Product calibration: Cancel

To cancel an active product calibration or an active initial measurement the command 0x00000040 is written to register 13914 (see Table 53). The calibration value is not considered in this product calibration cancel phase and the value is not saved within the sensor.

Table 53 Example to cancel an active product calibration or an initial measurement

Command: CP6: Cancel		Modbus address: **13914**	Length: **4**	Type: **16** Write
Parameter:	Command	Value		
Format:	Hex	Float		
Value:	**0x00000040**	**0.0**		

Performing this action the product calibration or any initial measurements are canceled. The values of the prior product calibration are removed from the sensor's memory. From now on the sensor is measuring using its prior CP1 / CP2 standard calibration.

As shown in Table 54 after cancelling a product calibration the sensors calibration status will be reading 0x00000000 again. (see Table 59)

Table 54 Example to read the current product calibration status

Command: CP6: Initial measurement		Modbus address: **13912**	Length: **2**	Type: **3** Read
Parameter:	Command			
Format:	Hex			
Value:	**0x00000000**			

2.8.3.2.4 Product calibration: Restore standard calibration

If a product calibration is active this product calibration can be temporarily switched off by writing the command "restore standard calibration" (0x00000080) (see Table 59) to register 13914 (see Table 55). The calibration value is not considered in this restore standard calibration phase and the value is not saved within the sensor.
Performing this action the values of the product calibration remain stored in the sensor's memory.

Table 55 Example to restore a standard calibration from an active product calibration

Command: CP6: Restore standard		Modbus address: **13914**	Length: **4**	Type: **16** Write
Parameter:	Command	Value		
Format:	Hex	Float		
Value:	**0x00000080**	**0.0**		

From now on the sensor is measuring using its prior CP1 / CP2 standard calibration.
As shown in Table 56 the sensor's calibration status will be reading "CP6 assigned" (0x40000000) (see Table 59) meaning that a valid assignment for a product calibration is available in the sensor's memory.

Table 56 Example to read the current product calibration status after a restoring a standard calibration

Command: CP6: Initial measurement		Modbus address: **13912**	Length: **2**	Type: **3** Read
Parameter:	Command			
Format:	Hex			
Value:	**0x40000000**			

2.8.3.2.5 Product calibration: Restore product calibration

If a valid but inactivated product calibration is available in the sensors memory, the calibration status is reading 0x40000000 ("CP 6 assigned", see Table 56). This stored product calibration can be restored or reactivated by writing command "restore product calibration" 0x00000100 (see Table 59) to register 13914 according to Table 57. The calibration value is not considered in this restore product calibration phase and the value is not saved within the sensor.

Table 57 Example to restore an available product calibration from an active standard calibration

Command: CP6: Restore product	Modbus address: **13914**	Length: **4**	Type: **16**	Write
Parameter:	Command	Value		
Format:	Hex	Float		
Value:	**0x00000100**	**0.0**		

If this command is performed without available product calibration in the sensors memory the sensor will respond with a Modbus exception since this command is not valid.

From now on the sensor is measuring using its prior CP6 product calibration. As shown in Table 58 the sensors calibration status will be reading 0x50000000 (corresponding to "CP6 assigned" and "CP6 active") again. (see Table 59)

Table 58 Example to read the current product calibration status, after a restoring a product calibration

Command: CP6: Initial measurement	Modbus address: **13912**	Length: **2**	Type: **3**	Read
Parameter:	Command			
Format:	Hex			
Value:	**0x50000000**			

2.8.4 Reading the Calibration Status

A calibration is not always successful. In order to analyze what has gone wrong, two different calibration status registers can be read:

- Register 10312 for CP1
- Register 11032 for CP2
- Register 13912 for CP6

Start register	Number of registers	Reg1 / Reg2	Modbus function code	Read access	Write access
10312	2	Status CP1 (see Table 59)	3, 4	U/A/S	none
11032	2	Status CP2 (see Table 59)	3, 4	U/A/S	none
13912	2	Status CP6 (see Table 59)	3, 4	U/A/S	none

Note:

Registers 10312,and 11032contain the same information!

Table 59 Definition of the calibration status for registers 10312, 11032 and 13912

Bit #	Hex value	Calibration Point relevant			Definition
		CP1	CP2	CP6	
None	0x00000000	X	X	X	calibration successful
0 (LSB)	0x00000001				not available
1-5	…				not available
6	0x00000040	X	X		carbon dioxide value to be calibrated at is too low (see register 10318 or register 11038)
7	0x00000080	X	X		carbon dioxide value to be calibrated at is too high (see register 10318 or register 11038)
8	0x00000100	X	X		current temperature reading is too low (see register 4616)
9	0x00000200	X	X		current temperature reading is too high (see register 4616)
10	0x00000400	X	X		The current SR value is too low for the carbon dioxide value to be calibrated.
11	0x00000800	X	X		The current SR value is too high for the carbon dioxide value to be calibrated.
12	0x00001000	X	X		Temperature reading during calibration is not stable (see register 10278)
13	0x00002000	X	X		SR reading during calibration is not stable (see register 10278)
14	0x00004000			X	out of calibration range (wrong CO_2 value entered)
15	0x00008000			X	out of range (SR value out of range)
16	0x00010000				not available
17	0x00020000		X		Calibration not complete, CP1 required
18	0x00040000	X			Calibration not complete, CP2 required
19-27	…	…	…	…	not available
28	0x10000000			X	active
29	0x20000000			X	initial measurement
30	0x40000000			X	assigned
31	0x80000000				not available

Examples:

Table 60 show that the SR value was not stable during the calibration of CP1.

Table 60 Example to read the calibration status "SR reading during calibration is not stable"

Command: Calibration status CP1		Modbus address: **10312**	Length: **2**	Type: **3** Read
Parameter:	Status			
Format:	Hex			
Value:	**0x00002000**			

Table 61 shows that the calibration at CP2 was successful, but CP1 is also required to complete the standard calibration. For a complete successful standard calibration both points has to be calibrated.

Table 61 Example to read the calibration status "Calibration not complete, CP1 required"

Command: Calibration status CP2		Modbus address: **11032**	Length: **2**	Type: **3** Read
Parameter:	Status			
Format:	Hex			
Value:	**0x00020000**			

Table 62 shows the calibration status after initial measurement command under conditions outside the valid calibration range for CP6 (defined in register 13912). The status says: "CP6: out of calibration range" (0x00004000). The sensor is still running on its prior standard calibration.

Table 62 Example to read the calibration status of CP6 with measurement conditions outside the calibration range

Command: Calibration status CP6		Modbus address: **13912**	Length: **2**	Type: **3** Read
Parameter:	Status			
Format:	Hex			
Value:	**0x00004000**			

Calibration status after invalid assignment: The status in Table 63 says: "CP6: out of range" and "CP6: initial measurement" (0x20008000). The initial measurement in this case is still valid and available for further assignment of a product calibration value. The here performed assignment was **not** successful. The sensor remains running on its prior standard calibration.

Table 63 Example to read the calibration status of CP6 after having performed a valid initial measurement at CP6 and an invalid assignment

Command: Calibration status CP6		Modbus address: **13912**	Length: **2**	Type: **3** Read
Parameter:	Status			
Format:	Hex			
Value:	**0x20008000**			

2.8.5 Currently active Calibration Parameters

The available calibration parameters and calibration coefficients are defined in register 10276.

Start register	Number of registers	Reg1 / Reg2	Modbus function code	Read access	Write access
10276	2	Available calibration parameters and calibration coefficients, see Table 64	3, 4	U/A/S	none

Table 64 Bitwise definition of available calibration parameters and calibration coefficients.

Bit #	Hex value	Definition
0 (LSB)	0x00000001	Calibration parameter index 1
1	0x00000002	Calibration parameter index 2
2	0x00000004	Calibration parameter index 3
3	0x00000008	Calibration parameter index 4
4	0x00000010	Calibration parameter index 5
5	0x00000020	Calibration parameter index 6
6	0x00000040	Calibration parameter index 7
7	0x00000080	Calibration parameter index 8
8	0x00000100	Calibration parameter index 9
9-21	…	not available
22	0x0040000	Calibration coefficient 1
23	0x0080000	Calibration coefficient 2
24-31	…	not available

Table 65 Example to read available calibration parameters and calibration coefficients

Command: Avail. Cali. Param. and Coeff.	Modbus address: **10276**	Length: **2**	Type: **3**	Read
Parameter:	Avail. Cali. Units			
Format:	Hex			
Value:	**0x00C001FF**			

Table 66 Description of the available calibration coefficients

Calibration coefficient x#	Definition
1	Zero Point S/R
2	CO2 Point S/R

Table 67 Description of the available calibration parameters

Calibration parameter index#	Definition
1	At the time of a valid calibration, the current SR value is stored within calibration parameter index 1.
2	The assigned calibration value is stored within calibration parameter index 2.
3	At the time of a valid calibration, the current temperature value is stored within calibration parameter index 3.
4	Amount of calibrations Note: The calibration counter is only incremented when CP1 and CP2 are calibrated. In case of a product calibration CP6 the counter value will be incremented after a successful assign step.
5	At the time of a valid calibration, the current operating hours are stored within calibration parameter index 5. Note: The "operating hour" for CP6 is the moment of the "initial measurement".
6	At the time of a valid calibration, the current system time, if started by the operator, is stored within calibration parameter index 6 Note: For CP6, the system time is set during the action "initial measurement".
7	At the time of a valid calibration, the current adjusted measurement parameter salinity is stored within calibration parameter index 7. In that case the value cannot be modified anyway and is fixed.
8	At the time of a valid calibration, the current adjusted measurement parameter 2 Air Pressure (PA2) is stored within calibration parameter index 8. When a new Pressure calibration parameter value is entered, it is stored directly in the calibration point CP1, CP2 or CP6 and temporarily assigned to the regular Air Pressure value PA2 of the measurement parameter set. So that the current carbon dioxide measurement runs with the calibration parameter setting. The Air Pressure PA2 value in the measurement parameter set, is automatically reset to the original Air Pressure value after a calibration or after a sensor restart. (see **Example Calibration Parameter 8: Pressure** in the next chapter)
9	At the time of a valid calibration, the current adjusted measurement parameter 3 Humidity (PA3) is stored within calibration parameter index 9. When a new Humidity calibration parameter value is entered, it is stored directly in the calibration point CP1, CP2 or CP6 and temporarily assigned to the regular Humidity value PA3 of the measurement parameter set. So that the current carbon dioxide measurement runs with the calibration parameter setting. The Humidity value PA3 in the measurement parameter set, is automatically reset to the original Humidity value after a calibration or after a sensor restart. (see Example Calibration Parameter 9: Humidity in the next chapter)

2.8.5.1 CP1 Calibration Parameters

The description of calibration parameter 1 – 9 for CP1, can be read out via register 10328 – 10584.

Start register	Number of registers	Reg1 to Reg8 16 ASCII characters	Calibration Parameter Index and description, see Table 67	Modbus function code	Read access	Write access
10328	8	Measured value	1	3, 4	U/A/S	none
10360	8	Assigned value	2	3, 4	U/A/S	none
10392	8	Temperature	3	3, 4	U/A/S	none
10424	8	Number of calis	4	3, 4	U/A/S	none
10456	8	Operating hours	5	3, 4	U/A/S	none
10488	8	Time stamp	6	3, 4	U/A/S	none
10520	8	Salinity	7	3, 4	U/A/S	none
10552	8	Pressure	8	3, 4	U/A/S	none
10584	8	Humidity	9	3, 4	U/A/S	none

Table 68 Example to read the description of calibration parameter 1

Command: Current interface text		Modbus address: **10328**	Length: **8**	Type: **3** Read
Parameter:	Text			
Format:	Character			
Value:	**Measured value**			

The value and unit of calibration parameter 1 – 9 for CP1, can be read out via register 10336 – 10592.

Start register	Number of registers	Reg1 / Reg2 (return value is always 0)	Reg3 / Reg4 (unit is fixed and cannot be changed)	Reg5 / Reg6	Calibration Parameter Index and description, see Table 67	Modbus function code	Read access	Write access
10336	6	Not relevant	Unit	Value	1	3, 4	U/A/S	none
10368	6	Not relevant	Unit	Value	2	3, 4	U/A/S	none
10400	6	Not relevant	Unit	Value	3	3, 4	U/A/S	none
10432	6	Not relevant	Unit	Value	4	3, 4	U/A/S	none
10464	6	Not relevant	Unit	Value	5	3, 4	U/A/S	none
10496	6	Not relevant	Unit	Value	6	3, 4	U/A/S	none
10528	6	Not relevant	Unit	Value	7	3, 4	U/A/S	none
10560	6	Not relevant	Unit	Value	8	3, 4, 16	U/A/S	S
10592	6	Not relevant	Unit	Value	9	3, 4, 16	U/A/S	S

Table 69 Example to read the unit "none" and SR value 5 of calibration parameter 1

Command: Current interface text	Modbus address: **10336**	Length: **6**	Type: **3**	Read
Parameter:	Cali. Par.1 - Not relevant	Cali. Par.1 - Unit	Cali. Par.1 - Value	
Format:	Hex	Hex	Float	
Value:	**0x00000000**	**0x00000001**	**5**	

Example Calibration Parameter 8: Pressure

As shown in Table 70 the physical unit for calibration parameter 8 is mbar (0x00800000) and is fixed and cannot be changed and the value is 1013 (mbar).

Table 70 Example to read the unit and the value of calibration parameter 8 (pressure) of CP1

Command: Current interface text	Modbus address: **10560**	Length: **6**	Type: **3**	Read
Parameter:	Cali. Par.8 - Not relevant	Cali. Par.8 - Unit	Cali. Par.8 - Value	
Format:	Hex	Hex	Float	
Value:	**0x00000000**	**0x00800000**	**1013**	

As shown in Table 71 the unit is mbar (0x00800000), the value is 1013 (mbar), the min is 10 (mbar) and the max is 12000 (mbar).

Table 71 Example to read measurement parameter 2

Command: Pressure		Modbus address: **3146**	Length: **8**	Type: **3** Read
Parameter:	Unit	Value	Min value	Max value
Format:	Hex	Float	Float	Float
Value:	**0x00800000**	**1013**	**10**	**12000**

Before a new CP1 calibration is initiated, the momentary environmental pressure has to be assigned to the calibration parameter pressure and also the measurement parameter Air Pressure (PA2) is temporary assigned to this value. From that point on the carbon dioxide measurement is using this setting.

As shown in Table 72, the unit for CP1 is set to mbar (0x00800000) and the value to 900 (mbar).

Table 72 Example to set the physical unit of calibration parameter 8 value (pressure) of CP 1

Command: Current interface text		Modbus address: **10560**	Length: **6**	Type: **16** Write
Parameter:	Cali. Par.8 - Not relevant	Cali. Par.8 - Unit	Cali. Par.8 - Value	
Format:	Hex	Hex	Float	
Value:	**0x00000000**	**0x00800000**	**900**	

As shown in Table 73 the unit is mbar (0x00800000), the value is 900 (mbar), the min is 10 (mbar) and the max is 12000 (mbar).

Table 73 Example to read measurement parameter 2

Command: Pressure		Modbus address: **3146**	Length: **8**	Type: **3** Read
Parameter:	Unit	Value	Min value	Max value
Format:	Hex	Float	Float	Float
Value:	**0x00800000**	**900**	**10**	**12000**

A new calibration is initiated at CP1 by writing to register 10314, whether successful or not, the measurement parameter Air Pressure PA2 has to be reset to the origin value of 1013 mbar.

As shown in Table 74 the physical unit is mbar (0x00800000) and is fixed and cannot be changed and the value is 900 (mbar).

Table 74 Example to read the unit and value of calibration parameter 8 (pressure) of CP1

Command: Current interface text		Modbus address: **10560**	Length: **6**	Type: **3** Read
Parameter:	Cali. Par.8 - Not relevant	Cali. Par.8 - Unit	Cali. Par.8 - Value	
Format:	Hex	Hex	Float	
Value:	**0x00000000**	**0x00800000**	**900**	

As shown in Table 75, PA2 is automatically reset to the original Air Pressure value.The unit is mbar (0x00800000), the value is 1013 (mbar), the min is 10 (mbar) and the max is 12000 (mbar).

Table 75 Example to read measurement parameter 2

Command: Pressure		Modbus address: **3146**	Length: **8**	Type: **3** Read
Parameter:	Unit	Value	Min value	Max value
Format:	Hex	Float	Float	Float
Value:	**0x00800000**	**1013**	**10**	**12000**

2.8.5.2 CP2 Calibration Parameters

The description of calibration parameter 1 – 9, can be read out via register starting at 11048 - 11304

Start register	Number of registers	Reg1 to Reg8 16 ASCII characters	Calibration Parameter Index and description, see Table 67	Modbus function code	Read access	Write access
11048	8	Measured value	1	3, 4	U/A/S	none
11080	8	Assigned value	2	3, 4	U/A/S	none
11112	8	Temperature	3	3, 4	U/A/S	none
11144	8	Number of calis	4	3, 4	U/A/S	none
11176	8	Operating hours	5	3, 4	U/A/S	none
11208	8	Time stamp	6	3, 4	U/A/S	none
11240	8	Salinity	7	3, 4	U/A/S	none
11272	8	Pressure	8	3, 4	U/A/S	none
11304	8	Humidity	9	3, 4	U/A/S	none

Table 76 Example to read the description of calibration parameter 2

Command: Current interface text		Modbus address: **11080**	Length: **8**	Type: **3** Read
Parameter:	Text			
Format:	Character			
Value:	**Assigned value**			

The value and unit of calibration parameter 1 – 9 for CP2, can be read out via register 11056 – 11312.

Start register	Number of registers	Reg1 / Reg2 (return value is always 0)	Reg3 / Reg4 (unit is fixed and cannot be changed)	Reg5 / Reg6	Calibration Parameter Index and description, see Table 67	Modbus function code	Read access	Write access
11056	6	Not relevant	Unit	Value	1	3, 4	U/A/S	none
11088	6	Not relevant	Unit	Value	2	3, 4	U/A/S	none
11120	6	Not relevant	Unit	Value	3	3, 4	U/A/S	none
11152	6	Not relevant	Unit	Value	4	3, 4	U/A/S	none
11184	6	Not relevant	Unit	Value	5	3, 4	U/A/S	none
11216	6	Not relevant	Unit	Value	6	3, 4	U/A/S	none
11248	6	Not relevant	Unit	Value	7	3, 4	U/A/S	none
11280	6	Not relevant	Unit	Value	8	3, 4, 16	U/A/S	S
11312	6	Not relevant	Unit	Value	9	3, 4, 16	U/A/S	S

As shown in Table 77 the assigned value of calibration parameter 2 is in the unit mbar and the value is 200 (mbar).

Table 77 Example to read the unit and value of calibration parameter 2 (Assigned value) of CP2

Command: Current interface text	Modbus address: **11088**	Length: **6**	Type: **3**	Read
Parameter:	Cali. Par.2 - Not relevant	Cali. Par.2 - Unit	Cali. Par.2 - Value	
Format:	Hex	Hex	Float	
Value:	**0x00000000**	**0x00800000**	**200**	

Example Calibration Parameter 9: Humidity

As shown in Table 78 the physical unit for calibration parameter 9 is percent (0x20000000) and is fixed and cannot be changed and the value is 100 (percent).

Table 78 Example to read the unit and value of calibration parameter 9 (humidity) of CP2

Command: Current interface text	Modbus address: **11312**	Length: **6**	Type: **3**	Read
Parameter:	Cali. Par.9 - Not relevant	Cali. Par.9 - Unit	Cali. Par.9 - Value	
Format:	Hex	Hex	Float	
Value:	**0x00000000**	**0x20000000**	**100**	

As shown in Table 79 the unit for measurement parameter 3 is percent (0x20000000), the value is 100 (percent), the min is 0 (percent) and the max is 100 (percent).

Table 79 Example to read measurement parameter 3 PA3

Command: Pressure		Modbus address: **3178**	Length: **8**	Type: **3** Read
Parameter:	Unit	Value	Min value	Max value
Format:	Hex	Float	Float	Float
Value:	**0x20000000**	**100**	**0**	**100**

Before a new CP2 calibration is initiated, the momentary environmental humidity has to be assign to the calibration parameter 9 Humidity and also the measurement parameter 3 Humidity is temporary assigned with this value. From that point on the carbon dioxide measurement is using this setting.

As shown in Table 80 the physical unit is set to percent (0x20000000) and the value to 0 (percent).

Table 80 Example to set the unit and value of calibration parameter 9 (humidity) of CP2

Command: Current interface text		Modbus address: **11312**	Length: **6**	Type: **16** Write
Parameter:	Cali. Par.9- Not relevant	Cali. Par.9 - Unit	Cali. Par.9 - Value	
Format:	Hex	Hex	Float	
Value:	**0x00000000**	**0x20000000**	**0**	

As shown in Table 81 the unit is percent (0x20000000), the value is 0 (percent), the min is 0 (percent) and the max is 100 (percent).

Table 81 Example to read measurement parameter 3

Command: Pressure		Modbus address: **3178**	Length: **8**	Type: **3** Read
Parameter:	Unit	Value	Min value	Max value
Format:	Hex	Float	Float	Float
Value:	**0x20000000**	**0**	**0**	**100**

A new calibration is initiated at CP2 by writing to register 11312, whether successful or not, the measurement parameter 3 Humidity has to be reset to the origin value of 100 %.

As shown in Table 82 the physical unit is percent (0x20000000) and is fixed and cannot be changed and the value is 0 (percent).

Table 82 Example to read the unit and value of calibration parameter 9 (humidity) of CP2

Command: Current interface text		Modbus address: **11312**	Length: **6**	Type: **3** Read
Parameter:	Cali. Par.9 - Not relevant	Cali. Par.9 - Unit	Cali. Par.9 - Value	
Format:	Hex	Hex	Float	
Value:	**0x00000000**	**0x20000000**	**0**	

As shown in Table 83, PA3 is automatically reset to the original Humidity value. The unit is percent (0x20000000), the value is 100 (percent), the min is 0 (percent) and the max is 100 (percent).

Table 83 Example to read measurement parameter 3

Command: Pressure		Modbus address: **3178**	Length: **8**	Type: **3** Read
Parameter:	Unit	Value	Min value	Max value
Format:	Hex	Float	Float	Float
Value:	**0x20000000**	**100**	**0**	**100**

2.8.5.3 CP6 Calibration Parameters

The description of calibration parameter 1 – 9 for CP6, can be read out via registers 13928 – 14184.

Start register	Number of registers	Reg1 to Reg8 16 ASCII characters	Calibration Parameter Index and description, see Table 67	Modbus function code	Read access	Write access
13928	8	Measured value	1	3, 4	U/A/S	none
13960	8	Assigned value	2	3, 4	U/A/S	none
13992	8	Temperature	3	3, 4	U/A/S	none
14024	8	Number of calis	4	3, 4	U/A/S	none
14056	8	Operating hours	5	3, 4	U/A/S	none
14088	8	Time stamp	6	3, 4	U/A/S	none
14120	8	Salinity	7	3, 4	U/A/S	none
14152	8	Pressure	8	3, 4	U/A/S	none
14184	8	Humidity	9	3, 4	U/A/S	none

Table 84 Example to read the description of calibration parameter 3

Command: Current interface text	Modbus address: **13992**	Length: **8**	Type: **3**	Read
Parameter:	Text			
Format:	Character			
Value:	**Temperature**			

The value and unit of calibration parameter 1 – 9 for CP6, can be read out via register 13936 – 14192.

Start register	Number of registers	Reg1 / Reg2 (return value is always 0)	Reg3 / Reg4 (unit is fixed and cannot be changed)	Reg5 / Reg6	Calibration Parameter Index and description, see Table 67	Modbus function code	Read access	Write access
13936	6	Not relevant	Unit	Value	1	3, 4	U/A/S	none
13968	6	Not relevant	Unit	Value	2	3, 4	U/A/S	none
14000	6	Not relevant	Unit	Value	3	3, 4	U/A/S	none
14032	6	Not relevant	Unit	Value	4	3, 4	U/A/S	none
14064	6	Not relevant	Unit	Value	5	3, 4	U/A/S	none
14096	6	Not relevant	Unit	Value	6	3, 4	U/A/S	none
14128	6	Not relevant	Unit	Value	7	3, 4	U/A/S	none
14160	6	Not relevant	Unit	Value	8	3, 4, 16	U/A/S	S
14192	6	Not relevant	Unit	Value	9	3, 4, 16	U/A/S	S

As shown in Table 85 the assigned unit is °C ant the assigned value is 26.2 (°C).

Table 85 Example to read the unit and value of calibration parameter 3 (Temperature) of CP6

Command: Current interface text		Modbus address: **14000**	Length: **6**	Type: **3** Read
Parameter:	Cali. Par.3 - Not relevant	Cali. Par.3 - Unit	Cali. Par.3 - Value	
Format:	Hex	Hex	Float	
Value:	**0x00000000**	**0x00000004**	**26.2**	

Note:

For CP6, the system time is set during the action “initial measurement”. The system time is explained in chapter 2.9.2.

According to Table 86 the Time Stamp value has no unit (0x00000001), the value is 1614089002 (no unit), means the initial measurement of the product calibration has been performed on February 23th 2021 at 15:03.

Table 86 Example to read the unit and value of calibration parameter 6 (Time Stamp) of CP6

Command: Time Stamp CP6		Modbus address: 14096	Length: 6	Type: **3** Read
Parameter:	Cali. Par.6 - Not relevant	Cali. Par.6 - Unit	Cali. Par.6 - Value	
Format:	Hex	Hex	Float	
Value:	**0x00000000**	**0x00000001**	**1614089002**	

2.8.5.4 Calibration Coefficients

The description of calibration coefficients Zero Point S/R and CO2 Point S/R can be read out via register 14632 and 14664.

Start register	Number of registers	Reg1 to Reg8 16 ASCII characters	Modbus function code	Read access	Write access
14632	8	Zero Point S/R	3, 4	U/A/S	none
14664	8	CO2 Point S/R	3, 4	U/A/S	none

Table 87 Example to read the description of calibration coefficient of Zero Point S/R

Command: Current interface text	Modbus address: **14632**	Length: **8**	Type: **3**	Read
Parameter:	Text			
Format:	Character			
Value:	**Zero Point S/R**			

The value and unit of calibration coefficients Zero Point S/R and CO2 Point S/R can be read out via register 14640 and 14672.

Start register	Number of registers	Reg1 / Reg2 (return value is always 0)	Reg3 / Reg4 (unit is fixed and cannot be changed)	Reg5 / Reg6	Modbus function code	Read access	Write access
14640	6	Not relevant	Unit	Value	3, 4	U/A/S	none
14672	6	Not relevant	Unit	Value	3, 4	U/A/S	none

Table 88 Example to read register 14672

Command: Calculated cali values	Modbus address: **14672**	Length: **6**	Type: **3**	Read
Parameter:	Not relevant	Unit	Value	
Format:	Hex	Hex	Float	
Value:	**0x00000000**	**0x00000001**	**1.451**	

The example in Table 88 shows that the calibration coefficient CO2 Point S/R has no unit (0x00000001) and that the value is 1.451 (no unit).

2.9 Sensor Status

2.9.1 Temperature Ranges

In registers 4608, 4612, 4616 and 4624 four different temperature ranges are defined:

- Operation – is the maximum temperature range to which the sensor can be exposed to during operation and storage. If the current temperature is out of this range, the corresponding bit in the measurement status register of PMC1 and PMC6 is set. (see chapter 2.5.4.)
- Measurement – is the maximum allowable range where carbon dioxide measurement is possible.
- Calibration – in this range the sensor can be calibrated.
- User defined Measurement – the specialist (operator level S) can adjust the range in which carbon dioxide reading is active. The user defined measurement temperature range is a sub range of the measurement temperature range.

Note:

When performing a calibration i.e. CP1 or CP2, not CP6, the user defined measurement temperature range is temporarily set to the values of the measurement temperature range from register 4612. After 10 minutes after the last calibration command or after a power up, the user defined measurement temperature range in register 4624 is reset to the values the user has defined.

Start register	Number of registers	Reg1 / Reg2	Reg3 / Reg4	Modbus function code	Read access	Write access
4608	4	Operating temperature min [°C]	Operating temperature max [°C]	3, 4	U/A/S	none
4612	4	Measurement temperature min [°C]	Measurement temperature max [°C]	3, 4	U/A/S	none
4616	4	Calibration temperature min [°C]	Calibration temperature max [°C]	3, 4	U/A/S	none
4624	4	User defined measurement temperature min [°C]	User defined measurement temperature max [°C]	3, 4, 16	U/A/S	S

Table 89 Example to read the operating temperature values min and max

Command: Operating T range		Modbus address: **4608**	Length: **4**	Type: **3** Read
Parameter:	Operating T min [°C]	Operating T max [°C]		
Format:	Float	Float		
Value:	**-10**	**140**		

Table 90 Example to read the measurement temperature values min and max

Command: Measurement T range		Modbus address: **4612**	Length: **4**	Type: **3** Read
Parameter:	Measurement T min [°C]	Measurement T max [°C]		
Format:	Float	Float		
Value:	**-10**	**60**		

Table 91 Example to read the calibration temperature values min and max

Command: Calibration T range		Modbus address: **4616**	Length: **4**	Type: **3** Read
Parameter:	Calibration T min [°C]	Calibration T max [°C]		
Format:	Float	Float		
Value:	**15**	**45**		

In example shown in Table 92 the sensor will not perform carbon dioxide reading and the digital output for carbon dioxide is set to a non valid value of -999 (the current temperature is 25 °C).

Table 92 Example set the user defined measurement temperature

Command: User measurement T		Modbus address: **4624**	Length: **4**	Type: **16** Write
Parameter:	User measurement T min [°C]	User measurement T max [°C]		
Format:	Float	Float		
Value:	**-10**	**5**		

Note:

Temperature reading is active at any time, regardless of the current temperature.

2.9.2 Operating Hours, Counters and System Time

In register 4676 are stored:

- total operating hours
- operating hours above max measurement temperature (see chapter 2.9.1)
- the operating hours above max operating temperature (see chapter 2.9.1)

In register 4682 are stored:

- number of power ups
- number of watchdog resets
- number of writing cycles to flash memory

In register 4688 are stored:

- number of sterilization in place (SIP) (see chapter 2.9.8)
- number of cleaning in place (CIP) (see chapter 2.9.8)

In register 4692 is stored

- number of autoclavings

Note:

Theregister 4692 has no effect for the sensor and is only for the user to trace the record for himself.

In register 8232 is stored

- system time counter.

Note:

When the sensor is powered up, the system time is set to 0. A value between 0 and 2^{32} can be written into this register. From this value, the sensor increments this value every second.
We recommend to use as base date the so-called UNIX timestamp (hint: www.epochconverter.com) which starts at 1st of January 1970 GMT. When a calibration is performed the system time value will be copied to the register 10488 for CP1, 11208 for CP2 and 104088 for CP6 (after the action "initial measurement"). With this copied value, the absolute time of calibration can be recovered, even if the sensor has powered down in the meantime.
Be sure to update this register if needed after every power up of the sensor.

Start register	Number of registers	Reg1 / Reg2	Reg3 / Reg4	Reg3 / Reg4	Modbus function code	Read access	Write access
4676	6	Operating hours [h]	Operating hours above max measurement temperature [h]	Operating hours above max operating temperature [h]	3, 4	U/A/S	none
4682	6	Number of Power ups	Number of Watchdog resets	Number of Writing cycles to flash memory	3, 4	U/A/S	none
4688	4	Number of SIP cycles	Number of CIP cycles	-	3, 4	U/A/S	none
4692	2	No of autoclavings			3, 4, 16	U/A/S	S
8232	2	System Time Counter			3, 4, 16	U/A/S	S

Table 93 Example to read the operating hours

Command: Operating hours		Modbus address: **4676**	Length: **6**	Type: **3** Read
Parameter:	Operating hours [h]	Operating hours above max measurement temperature [h]	Operating hours above max operating temperature [h]	
Format:	Float	Float	Float	
Value:	**168.3667**	**0**	**0**	

Table 94 Example to read Power ups and Watchdog

Command: Power & Watchdog		Modbus address: **4682**	Length: **6**	Type: **3** Read
Parameter:	Number of Power ups	Number of Watchdog resets	Number of Writing cycles to flash memory	
Format:	Decimal	Decimal	Decimal	
Value:	**34**	**1**	**16**	

Table 95 Example to read SIP and CIP cycles

Command: SIP & CIP		Modbus address: **4688**	Length: **4**	Type: **3** Read
Parameter:	SIP cycles	CIP cycles		
Format:	Decimal	Decimal		
Value:	**0**	**0**		

Table 96 Example to read the number of autoclavings

Command: Autoclaving		Modbus address: **4692**	Length: **2**	Type: **3** Read
Parameter:	Autoclavings			
Format:	Decimal			
Value:	**7**			

Table 97 Example to write the number of autoclavings

Command: Autoclaving		Modbus address: **4692**	Length: **2**	Type: **16** Write
Parameter:	Autoclavings			
Format:	Decimal			
Value:	**14**			

Table 98 Example to write the system time into the sensor

Command: System Time		Modbus address: **8232**	Length: **2**	Type: **16** Write
Parameter:	System Time			
Format:	Decimal			
Value:	**1614147385**			

Table 99 Example to read the system time

Command: System Time		Modbus address: **8232**	Length: **2**	Type: **3** Read
Parameter:	System Time			
Format:	Decimal			
Value:	**1614178800**			

Note:

The deviation of the system time, if not updated by the operator, is less than one minute per 24h.

2.9.3 Warnings

A "Warning" is a notification message which still allows further functioning of the system. This message alerts the operator of a possible problem that could lead to uncertain results.

2.9.3.1 Currently Active Warnings

The currently active warnings are stored in register 4736.

Start register	Number of registers	Reg1 / Reg2	Reg3 / Reg4	Reg5 / Reg6	Reg7 / Reg8	Modbus function code	Read access	Write access
4736	8	Active warnings measurement (bitwise defined)	Active warnings calibration (bitwise defined)	Active warnings interface (bitwise defined)	Active warnings hardware (bitwise defined)	3, 4	U/A/S	none

Table 100 Example to read the active warnings

Command: Warning active	Modbus address: **4736**	Length: **8**	Type: **3**	Read
Parameter:	W Measurement	W Cal & Membrane	W Interface	W Hardware
Format:	Hex	Hex	Hex	Hex
Value:	**0x00000000**	**0x00000000**	**0x00000000**	**0x00000000**

2.9.3.2 Definition of Warnings

Table 101 Definition of warnings "measurement"

Bit #	Hex code	Description
0 (LSB)	0x00000001	CO2 reading below lower limit
	The carbon dioxide reading is too low (CO2 < -5mbar). Make a new zero-point calibration.	
1	0x00000002	CO2 reading above upper limit
	The carbon dioxide reading is too high (CO2 > 1050mbar). Make a new calibration at CP2.	
25	0x02000000	T reading below lower limit
	The temperature is below the user defined measurement temperature range (see register 4624). If outside this range, the sensor will not perform carbon dioxide readings.	
26	0x04000000	T reading above upper limit
	The temperature is above the user defined measurement temperature range (see register 4624). If outside this range, the sensor will not perform carbon dioxide readings.	
31	0x80000000	Measurement not running
	Causes that trigger this warning: 1.) Sensor operating voltage range is not between 10-27 VDC or 2.) The temperature measurement is outside the user defined temperature range.	

Table 102 Definition of warnings "calibration"

Bit #	Hex code	Description
0 (LSB)	0x00000001	CO2 calibration recommended
	Perform a calibration in order to ensure reliable measurement.	

Table 103 Definition of warnings "interface"

Bit #	Hex code	Description
		Not available

Table 104 Definition of warnings "hardware"

<table>
<tr><td>Bit #</td><td>Hex code</td><td>Description</td></tr>
<tr><td rowspan="2">0 (LSB)</td><td>0x00000001</td><td>Sensor supply voltage too low</td></tr>
<tr><td colspan="2">The sensor supply voltage is below 10V. Please check your power supply.</td></tr>
<tr><td rowspan="2">1</td><td>0x00000002</td><td>Sensor supply voltage too high</td></tr>
<tr><td colspan="2">The sensor supply voltage is above 27V. Please check your power supply.</td></tr>
<tr><td rowspan="2">2</td><td>0x00000004</td><td>FE Measurement Warning</td></tr>
<tr><td colspan="2">Light source consumes too much power</td></tr>
<tr><td rowspan="2">9</td><td>0x00000200</td><td>Replace Sensor recommended</td></tr>
<tr><td colspan="2">The Sensor Quality Indicator is below 40%. The quality of the sensor is sufficient for reliable measurement, but replacement of the sensor will be needed in near future.</td></tr>
</table>

2.9.4 Errors

An "Error" message indicates a serious problem of the sensor which does not allow further proper functioning of the sensor. This problem must be solved.

2.9.4.1 Currently Active Errors

The currently active errors are stored in register 4800.

Start register	Number of registers	Reg1 / Reg2	Reg3 / Reg4	Reg5 / Reg6	Reg7 / Reg8	Modbus function code	Read access	Write access
4800	8	Active errors measure-ment (bitwise defined)	Active errors calibration (bitwise defined)	Active errors interface (bitwise defined)	Active errors hardware (bitwise defined)	3, 4	U/A/S	none

Table 105 Example to read the active errors

Command: Errors current	Modbus address: **4800**	Length: **8**	Type: **3**	Read
Parameter:	E Measurement	E Cal & Membrane	E Interface	E Hardware
Format:	Hex	Hex	Hex	Hex
Value:	**0x00000000**	**0x00000000**	**0x00000000**	**0x00000000**

2.9.4.2 Definition of Errors

Table 106 Definition of errors "measurement"

Bit #	Hex code	Description
0 (LSB)	0x00000001	CO2 reading failure
	CO2 algorithm error	
1	0x00000002	CO2 exceeds air pressure
	Measured partial pressure of carbon dioxide is higher than the air pressure set by the operator. Reconfigure the air pressure parameter (PA2).	
25	0x02000000	T sensor defective
	The internal temperature sensor is defective.	

Table 107 Definition of errors "calibration"

Bit #	Hex code	Description
		Not available

Table 108 Definition of errors "interface"

Bit #	Hex code	Description
		Not available

Table 109 Definition of errors "hardware"

Bit #	Hex code	Description
2	0x00000004	Temperature reading far below min
	The measured temperature is below the operation temperature (Reg. 4608)	
3	0x00000008	Temperature reading far above max
	The measured temperature is above the operation temperature (Reg. 4608)	
9	0x00000200	Sensor Defective
	Sensor is defective or Sensor Quality Indicator is below 10%. The quality of the sensor is not sufficient for reliable measurement. Sensor needs to be replaced.	
22	0x00400000	EEPROM comm. (I2C) error Userend
	Internal I2C communication error Reset the sensor and try again	
24	0x01000000	Internal communication (I2C) failure Userend
	Internal I2C communication error Reset the sensor and try again	
25	0x02000000	Internal communication failure (frontend)
	No communication between Frontend and Userend. Reset the sensor and try again	
26	0x04000000	Stackoverflow
	Internal memory failure Reset the sensor and try again	

2.9.5 Measurement values - Exceeds Temperature range, and CO2 measurement errors

In case that the current measurement temperature exceeds the user defined temperature range following warning bits are set:

Category	Hex Code	Description
Measurement Warning	0x02000000	T reading below lower limit **OR**
	0x04000000	T reading above upper limit **AND**
	0x80000000	Measurement not running

In case that the current measurement temperature exceeds the operating temperature range, following error / warning bits has to be set:

Category	Hex Code	Description
Hardware Error	0x00000004	Temperature reading far below min **OR**
	0x00000008	Temperature reading far above max **AND**
Measurement Error	0x00000001	CO2 reading failure

In case the current calculated partial pressure exceeds the measurement parameter air pressure, following error bit hast to be set:

Category	Hex Code	Description
Measurement Error	0x00000002	CO2 exceeds air pressure

The measurement output values for these cases are defined in Table 110.

Table 110 Measurement values (PMCs) in case of exceeding the user defined- or operating temperature range

Bit #	Hex code	Description	Definition	Measurement output values
0 (LSB)	0x00000001	PMC1	CO_2 (carbon dioxide value)	-999.0
5	0x00000020	PMC6	T (temperature measurement value)	Current temp. value

2.9.6 Measurement values – no CO2 measurement within the frontend is running

In case that the current supply voltage is not within the defined supply voltage range, following warning bits are set:

Category	Hex Code	Description
Hardware Warning	0x00000001	Sensor supply voltage too low **OR**
	0x00000002	Sensor supply voltage too high **AND**
Measurement Warning	0x80000000	Measurement not running

In case the frontend measurement is not running, following warning bit is set:

Category	Hex Code	Description
Measurement Warning	0x80000000	Measurement not running

The measurement output values for this cases, are defined in Table 111.

Table 111 Measurement values (PMCs) in case no CO2 measurement within the frontend is running

Bit #	Hex code	Description	Definition	Measurement output values
0 (LSB)	0x00000001	PMC1	CO_2 (carbon dioxide value)	-999.0
5	0x00000020	PMC6	T (temperature measurement value)	-999.0

2.9.7 Measurement values – Hardware errors

The measurement output values for this cases, are defined in Table 112.

Table 112 Measurement values (PMCs) in case Hardware errors

Bit #	Hex code	Description	Definition	Measurement output values
0 (LSB)	0x00000001	PMC1	CO_2 (carbon dioxide value)	-999.0
5	0x00000020	PMC6	T (temperature measurement value)	-999.0

2.9.8 Reading Definition of SIP and CIP

The CO2NTROL RS485 Sensor is counting special cleaning events such as sterilizations or cleaning cycles by means of tracking typical temperature profiles (see chapter 2.9.2).

Register 4988 defines the temperature profile for SIP (sterilization in place) and register 4996 the temperature profile for CIP (cleaning in place). For the explanation the following values are given:

CIP temperature min: 80 °C   CIP temperature max: 100 °C   CIP time min: 30 minutes
SIP temperature min: 120 °C   SIP temperature max: 130 °C   SIP time min: 30 minutes

CIP / SIP time maximum values: 180 min

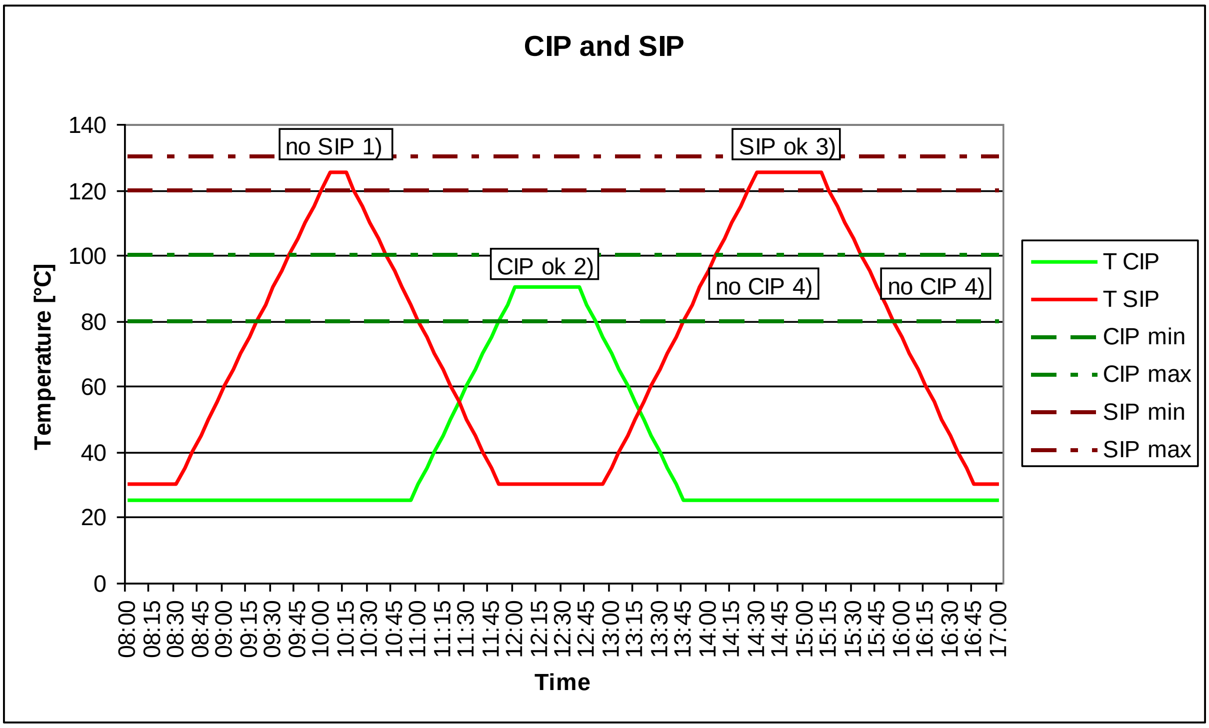


Figure 18 Definition of CIP and SIP cycles

Explanation of Figure 18:
1) no SIP-cycle counted, because time too short (less than 30 minutes).
2) CIP-cycle counted, because time greater than 30 minutes and in CIP temperature range.
3) SIP-cycle counted, because time greater than 30 minutes and in SIP temperature range.
4) no CIP-cycle counted, because of reaching the SIP-min limit.

Start register	Number of registers	Reg1 / Reg2	Reg3 / Reg4	Reg5 / Reg6	Reg7 / Reg8	Modbus function code	Read access	Write access
4988	8	SIP Tempera-ture min	SIP Tempera-ture max	SIP Process time min [min]	Empty	3, 4	U/A/S	S
4996	8	CIP Tempera-ture min	CIP Tempera--ture max	CIP Process time min [min]	Empty	3, 4	U/A/S	S

The unit of the temperatures is according to the selected unit of PMC6 (see 2.5.3.1)

2.9.9 Quality Indicator

2.9.9.1 Reading the Sensor Quality Indicator

In register 5472 with a command length of 2, the sensor quality (0-100%) is given in 10% percent steps. Hamilton recommends the replacement of the sensor at a value less than 40 % (see Table 104).

Start register	Number of registers	Reg1 / Reg2	Modbus function code	Read access	Write access
5472	2	Sensor quality [%]	3, 4	U/A/S	none

Table 113 Example to read the Sensor Quality Indicator with a command length of 2

Command: Sensor cap quality	Modbus address: **5472**	Length: **2**	Type: **3**	Read
Parameter:	Sensor quality [%]			
Format:	Float			
Value:	**100**			

2.9.9.2 Reading the Measurement Quality Indicator

In register 5472 with a command length of 6, the sensor quality (0-100%) in 10% percent steps and the measurement quality is given.

Start register	Number of registers	Reg1 / Reg2	Reg3 / Reg4	Reg5 / Reg6	Modbus function code	Read access	Write access
5472	6	Sensor quality [%]	Not used	Measurement quality, see Table 114	3, 4	U/A/S	none

Table 114 Definition of measurement quality values

Value	Description
100	Good: > 40 % sensor quality (no actions)
50	Bad: < 40 % sensor quality (maintenance required)
0	Defect: < 10 % sensor quality (senser defective)

Table 115 Example to read the sensor- and measurement quality with a command length of 6

Command: Measurement quality	Modbus address: **5472**	Length: **6**	Type: **3**	Read
Parameter:	Sensor quality [%]	Not used	Measurement quality	
Format:	Float	Float	Float	
Value:	**100**	**-999**	**100**	

2.10 Sensor Identification and Information

2.10.1 General Information

General information about the sensor is available as shown in the Table below.

Start register	Number of registers	Reg1 to Reg8 16 ASCII characters	Example of content	Modbus function code	Read access	Write access
1024	8	Userend FW Date	2021-08-31	3, 4	U/A/S	none
1032	8	Userend FW	COOUM003	3, 4	U/A/S	none
1040	8	Userend BL Date	2018-05-07	3, 4	U/A/S	none
1048	8	Userend BL	BL5UX001	3, 4	U/A/S	none
1056	8	Userend Ref	10075122/00	3, 4	U/A/S	none
1064	8	Userend SN	9999	3, 4	U/A/S	none
1072	8	Userend (space holder)	not available	3, 4	U/A/S	none
1080	8	Userend (space holder)	not available	3, 4	U/A/S	none
1088	8	Frontend FW Date	2021-01-13	3, 4	U/A/S	none
1096	8	Frontend FW	COOFJ001	3, 4	U/A/S	none
1104	8	Frontend BL Date	not available	3, 4	U/A/S	none
1112	8	Frontend BL	not available	3, 4	U/A/S	none
1120	8	Frontend Ref	10075843/00	3, 4	U/A/S	none
1128	8	Frontend SN	9999	3, 4	U/A/S	none
1136	8	Frontend (space holder)	not available	3, 4	U/A/S	none
1144	8	Frontend (space holder)	not available	3, 4	U/A/S	none

2.10.2 Sensor Identification

Definition of registers containing sensor identification:

Start register	Number of registers	Reg1 to Reg8 16 ASCII characters	Example of content	Modbus function code	Read access	Write access
1280	8	Sensor Ref	10087810xx/00	3, 4	U/A/S	none
1288	8	Sensor name	CO2NTROL RS485	3, 4	U/A/S	none
1296	8	Sensor Lot	1354271	3, 4	U/A/S	none
1304	8	Sensor Lot date	2021-03-16	3, 4	U/A/S	none
1312	8	Sensor SN	2076	3, 4	U/A/S	none
1320	8	Manufacturer part 1	HAMILTON Bonaduz	3, 4	U/A/S	none
1328	8	Manufacturer part 2	AG Switzerland	3, 4	U/A/S	none
1336	8	Sensor type	ARC CO2 Sensor	3, 4	U/A/S	none
1344	8	Power supply	10 - 27V 1.5W	3, 4	U/A/S	none
1352	8	Pressure range	10..12000mbar	3, 4	U/A/S	none
1360	8	Sensor ID	1008781000-2076	3, 4	U/A/S	none
1368	8	a-length	120	3, 4	U/A/S	none
1376	8	Sensor technology state	TS2.00	3, 4	U/A/S	none
1384	8	Electrical connection	VP 8.0	3, 4	U/A/S	none
1392	8	Process connection	PG 13.5	3, 4	U/A/S	none
1400	8	Sensing material	not available	3, 4	U/A/S	none

2.10.3 Free User Memory Space

These registers can be used to store any customer specific information in the sensor. There are different registers which can be read by everybody, but only specific operators can write them.

Start register	Number of registers	Reg1 to Reg8 16 ASCII characters	Example of content	Modbus function code	Read access	Write access
1536	8	Userspace	*FREE_USERSPACE*	3, 4, 16	U/A/S	U/A/S
1544	8	Userspace	*FREE_USERSPACE*	3, 4, 16	U/A/S	U/A/S
1552	8	Userspace	*FREE_USERSPACE*	3, 4, 16	U/A/S	U/A/S
1560	8	Userspace	*FREE_USERSPACE*	3, 4, 16	U/A/S	U/A/S
1568	8	Userspace	*FREE_USERSPACE*	3, 4, 16	U/A/S	A/S
1576	8	Userspace	*FREE_USERSPACE*	3, 4, 16	U/A/S	A/S
1584	8	Userspace	*FREE_USERSPACE*	3, 4, 16	U/A/S	A/S
1592	8	Userspace	*FREE_USERSPACE*	3, 4, 16	U/A/S	A/S
1600	8	Measuring Point	1008781000-2076	3, 4, 16	U/A/S	S
1608	8	Userspace	*FREE_USERSPACE*	3, 4, 16	U/A/S	S
1616	8	Userspace	*FREE_USERSPACE*	3, 4, 16	U/A/S	S
1624	8	Userspace	*FREE_USERSPACE*	3, 4, 16	U/A/S	S
1632	8	Userspace	*FREE_USERSPACE*	3, 4	U/A/S	none
1640	8	Userspace	*FREE_USERSPACE*	3, 4	U/A/S	none
1648	8	Userspace	*FREE_USERSPACE*	3, 4	U/A/S	none
1656	8	Userspace	*FREE_USERSPACE*	3, 4	U/A/S	none
1664	8	ext. OEM Sensor Name	*FREE_USERSPACE*	3, 4	U/A/S	none
1672	8	ext. OEM PartNumber	*FREE_USERSPACE*	3, 4	U/A/S	none
1680	8	ext. OEM Customer 1	*FREE_USERSPACE*	3, 4	U/A/S	none
1688	8	ext. OEM Customer 2	*FREE_USERSPACE*	3, 4	U/A/S	none
1696	8	Userspace	*FREE_USERSPACE*	3, 4	U/A/S	none
1704	8	Userspace	*FREE_USERSPACE*	3, 4	U/A/S	none
1712	8	Userspace	*FREE_USERSPACE*	3, 4	U/A/S	none
1720	8	Userspace	*FREE_USERSPACE*	3, 4	U/A/S	none
1728	8	Userspace	*FREE_USERSPACE*	3, 4	U/A/S	none
1736	8	Userspace	*FREE_USERSPACE*	3, 4	U/A/S	none
1744	8	Userspace	*FREE_USERSPACE*	3, 4	U/A/S	none
1752	8	Userspace	*FREE_USERSPACE*	3, 4	U/A/S	none

An important register is 1600, as it is the description of the measuring point. The information of this register is to identify individual sensors.

Attention:

The Free User Memory Space is located in a memory which allows in total max 1'000'000 write operations.

2.11 System Commands

2.11.1 Restore Factory Settings

Using register 8192 you can recall the sensor manufacturer values (interfaces, calibration data and passwords), except the SIP and CIP data which remain unchanged. By sending the recall value “911”, all configuration values will be set to default.

Start register	Number of registers	Reg1 / Reg2	Modbus function code	Read access	Write access
8192	2	Recall by value “911”	16	none	S

3 Appendix

3.1 List of tables

Table 1 Modbus definition for data transmission 9
Table 2 RS485 definitions for Arc Sensors 9
Table 3 Definition Floating point Single Precision (4 bytes resp. 2 Modbus registers) 12
Table 4 Implemented Error-Codes 14
Table 5 Definition of operator level and default passwords 15
Table 6 RS485 factory settings 16
Table 7 Code for the baud rates 17
Table 8 Definition of the analog interface modes 18
Table 9 Code for selection of the primary measurement channel 19
Table 10 Code for the 4-20 mA interface in case of errors and warnings 22
Table 11 Example: Read the settings for AO1 in case of warnings and errors 23
Table 12 Definition of PMC1 to 6 and SMC1 to 16 24
Table 13 Example to read register 2048 24
Table 14 Definition of physical units 25
Table 15 Example to read the physical unit in plain text ASCII in register 1952 25
Table 16 Example to set the physical unit of PMC1 to %-vol (0x00000010) 26
Table 17 Example to read register 2410 27
Table 18 Definition of measurement status for Primary Measurement Channels 28
Table 19 Bitwise definition of all parameters PA1 to PA16 29
Table 20 Example to read the available Parameters with operator level S 29
Table 21: Bitwise definition of the verification command 34
Table 22 Example to start the verification process 34
Table 23 Example to read the description of verification parameter 2 35
Table 24 Example to set the value of the verification parameter 2 to 200 mbar 35
Table 25 Example to read the description of verification parameter 3 36
Table 26 Example to set the value of the verification parameter 3 to 10 % 36
Table 27 Bitwise definition of CP0 to CP6 37
Table 28 Example to read the available CPs 37
Table 29 Example to read available calibration units of CP1 38
Table 30 Example to set the physical calibration unit of CP1 to %-vol (0x00000010) 38
Table 31 Example to read the limits of CP1 (active unit is mbar) 38
Table 32 Example to read the limits of CP2 (active unit is mbar) 39
Table 33 Example to read the limits of CP2 (active unit is %-vol) 39
Table 34 Example to read the calibration stability 39
Table 35 Example to set the calibration stability 39
Table 36 Example to read available calibration units of CP6 40
Table 37 Example to set the physical calibration unit of CP6 to %-vol 40
Table 38 Example to read the limits of CP6 40
Table 39 Calibration command code 41
Table 40 Example to start the calibration at CP1 in automatic mode 42
Table 41 Example to start the calibration at CP2 in automatic mode 42
Table 42 Example to start the calibration at CP2 in manual mode 42
Table 43 Example 1: Read the limits for CP2 43
Table 44 Example 1: Make CP2 calibration 43
Table 45 Example 1: Read CO2 value 43
Table 46 Example 2: Make CP2 calibration 43
Table 47 Example 2: Read CO2 value 43
Table 48 Definition of the commands related to the product calibration 45
Table 49 Example to start the product calibration procedure 46
Table 50 Example to read the current product calibration status 46
Table 51 Example to assign a calibration value to the above performed initial measurement 46
Table 52 Example to read the current product calibration status, after performing a product calibration. 46
Table 53 Example to cancel an active product calibration or an initial measurement 47
Table 54 Example to read the current product calibration status 47
Table 55 Example to restore a standard calibration from an active product calibration 47

Table 56 Example to read the current product calibration status after a restoring a standard calibration 47
Table 57 Example to restore an available product calibration from an active standard calibration 48
Table 58 Example to read the current product calibration status, after a restoring a product calibration 48
Table 59 Definition of the calibration status for registers 10312, 11032 and 13912 49
Table 60 Example to read the calibration status "SR reading during calibration is not stable" 50
Table 61 Example to read the calibration status "Calibration not complete, CP1 required".................. 50
Table 62 Example to read the calibration status of CP6 with measurement conditions outside the calibration range........ 50
Table 63 Example to read the calibration status of CP6 after having performed a valid initial measurement at CP6 and an invalid assignment 50
Table 64 Bitwise definition of available calibration parameters and calibration coefficients. 51
Table 65 Example to read available calibration parameters and calibration coefficients 51
Table 66 Description of the available calibration coefficients 51
Table 67 Description of the available calibration parameters 52
Table 68 Example to read the description of calibration parameter 1 53
Table 69 Example to read the unit "none" and SR value 5 of calibration parameter 1........ 54
Table 70 Example to read the unit and the value of calibration parameter 8 (pressure) of CP1 54
Table 71 Example to read measurement parameter 2 55
Table 72 Example to set the physical unit of calibration parameter 8 value (pressure) of CP 1 55
Table 73 Example to read measurement parameter 2 55
Table 74 Example to read the unit and value of calibration parameter 8 (pressure) of CP1 55
Table 75 Example to read measurement parameter 2 55
Table 76 Example to read the description of calibration parameter 2 56
Table 77 Example to read the unit and value of calibration parameter 2 (Assigned value) of CP2 57
Table 78 Example to read the unit and value of calibration parameter 9 (humidity) of CP2 57
Table 79 Example to read measurement parameter 3 PA3 58
Table 80 Example to set the unit and value of calibration parameter 9 (humidity) of CP2.................... 58
Table 81 Example to read measurement parameter 3 58
Table 82 Example to read the unit and value of calibration parameter 9 (humidity) of CP2 58
Table 83 Example to read measurement parameter 3 58
Table 84 Example to read the description of calibration parameter 3 59
Table 85 Example to read the unit and value of calibration parameter 3 (Temperature) of CP6 60
Table 86 Example to read the unit and value of calibration parameter 6 (Time Stamp) of CP6 60
Table 87 Example to read the description of calibration coefficient of Zero Point S/R.......................... 61
Table 88 Example to read register 14672........ 61
Table 89 Example to read the operating temperature values min and max 62
Table 90 Example to read the measurement temperature values min and max 62
Table 91 Example to read the calibration temperature values min and max 62
Table 92 Example set the user defined measurement temperature........ 63
Table 93 Example to read the operating hours........ 65
Table 94 Example to read Power ups and Watchdog 65
Table 95 Example to read SIP and CIP cycles........ 65
Table 96 Example to read the number of autoclavings 65
Table 97 Example to write the number of autoclavings 65
Table 98 Example to write the system time into the sensor 65
Table 99 Example to read the system time 65
Table 100 Example to read the active warnings........ 66
Table 101 Definition of warnings "measurement" 66
Table 102 Definition of warnings "calibration"........ 66
Table 103 Definition of warnings "interface" 67
Table 104 Definition of warnings "hardware"........ 67
Table 105 Example to read the active errors........ 68
Table 106 Definition of errors "measurement" 68
Table 107 Definition of errors "calibration"........ 68
Table 108 Definition of errors "interface" 68
Table 109 Definition of errors "hardware" 69
Table 110 Measurement values (PMCs) in case of exceeding the user defined- or operating temperature range 70

Table 111 Measurement values (PMCs) in case no CO2 measurement within the frontend is running 71
Table 112 Measurement values (PMCs) in case Hardware errors 71
Table 113 Example to read the Sensor Quality Indicator with a command length of 2 73
Table 114 Definition of measurement quality values..73
Table 115 Example to read the sensor- and measurement quality with a command length of 6 73

3.2 List of figures

Figure 1 Modbus Protocol Data Unit 6
Figure 2 Modbus frame over Serial Line 6
Figure 3 Bit sequence in RTU mode 7
Figure 4 RTU Message Frame...7
Figure 5 Valid frames with silent intervals 7
Figure 6 RTU Message Frame..7
Figure 7 Data transmission of a frame 8
Figure 8 Definition of Holding Registers...10
Figure 9 Example of reading holding registers 108 – 110. The contents of register 108 are read as the two byte values 0x022B. The contents of registers 109 and 110 are 0x0000 and 0x0064.....................10
Figure 10 Definition of Input Registers 10
Figure 11 Example of reading input register 9. The contents of input register 9 are read as the two byte value 0x000A..11
Figure 12 Definition of Write Multiple Registers 11
Figure 13 Example of writing the value 0x000A and 0x0102 to two registers starting at address 2.......11
Figure 14 Example of linear 4-20mA output characteristics..22
Figure 15 Comparison of the response of CO2NTROL RS485 to a change from 38.5 %-vol to zero CO_2 32
Figure 16 Standard 2-Point Calibration...37
Figure 17 Effect of the product calibration CP6 on an existing factory calibration function 44
Figure 18 Definition of CIP and SIP cycles 72

3.3 Abbreviations

AO	Analog Output Interface
CO_2	Carbon Dioxide
CP	Calibration Point
ECS	Electrochemical Sensor Interface
PA	Parameter
PMC	Primary Measurement Channel
SMC	Secondary Measurement Channel
MC	Measurement Channel
SIP	Sterilization In Place
CIP	Cleaning In Place

Hamilton Bonaduz AG
Via Crusch 8
CH-7402 Bonaduz
Switzerland

Tel. +41 58 610 10 10
Fax +41 58 610 00 10

contact.pa.ch@hamilton.ch
www.hamiltoncompany.com

September 2021
P/N: 111003564/02