

Additional operating instructions for R60355.9357

LED Preset Counter

96 × 48 mm, 2 presets

MODBUS Protocol
CR/LF Protocol
Option RS485



Table of Contents

1	Description	3
2	Important notice	3
3	Interface	3
3.1	Optional RS485 board	3
3.2	Device connection recommended by the Modbus Organization [1]	3
3.3	Interface configuration	4
3.4	Programming	4
4	CR/LF protocol	5
4.1	Transmission	5
4.1.1	Transmission principle for data source main counter, batch counter or totalizer	5
4.1.2	Transmission principle for data source main and batch counter / main counter and totalizer	6
5	MODBUS protocol	6
5.1	Master - Slave principle	6
5.2	Transmission principles	6
5.3	Timeout	7
5.4	Message cycle	7
5.5	Byte order	7
5.6	Structure of a message	7
5.7	Function codes	7
5.8	Data values	7
5.8.1	Float format (32 bits)	7
5.8.2	Integer format (32 bits)	8
5.9	CRC (cyclic redundancy check)	8
5.10	Reading several registers	9
5.11	Writing several registers	9
5.12	Identification	10
5.13	Error protocol	10
5.14	Registers	11
5.15	Message transmission examples	14
5.15.1	Example Read	14
5.15.2	Example Write	14
5.15.3	Example Identification	14
6	Parameter sets	15
7	Technical data	15
7.1	Pulse counter	15
7.2	Frequency meter	15
7.3	Timer	15
8	Frequencies (typical)	15
8.1	Pulse counter	15
8.2	Frequency meter	16
9	Help texts	16
10	References	16

1 Description

These operating instructions describe the optional RS485 interfaces and their protocols.

2 Important notice



Read both these operating instructions relating to the optional interfaces and the operating instructions of the basic device before operating the device.

3 Interface

Signal lines are connected via 5 screw terminals with a spacing of 3.5 mm.

3.1 Optional RS485 board

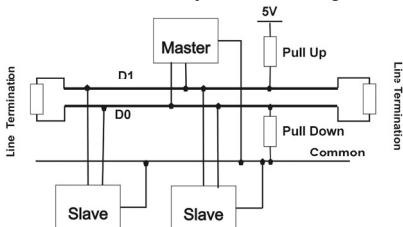
The RS485 interface allows a multipoint connection. Up to 32 devices may be connected to a bus segment.

Connections



The use of low-capacity and low-attenuation twisted pair cable allows achieving extremely reliable connections over long distances (maximum 1000 m).

3.2 Device connection recommended by the Modbus Organization [1]



Description	
MODBUS	Device
D0	B (-)
D1	A (+)
Common	

3.3 Interface configuration

MODBUS (RTU) Slave configuration

Interface	Baud rate	Data format	Address
RS485	9600	1 Start bit 8 Data bits Even Parity 1 Stop bit	1-247

CR/LF Master configuration

Interface	Baud rate	Data format	Address
RS485	9600	1 Start bit 8 Data bits No Parity 1 Stop bit	1-99

3.4 Programming

INTERF.	Main interface programming menu.
PROTOK.	Interface protocols
MODBUS	MODBUS protocol
CR.LF	CR/LF protocol
ADRESS.	Slave Address CR/LF: 1 – 99 MODBUS: 1- 247
1	
PR.TIME	Cycle time (only for CR/LF) 0,5 – 9999,9 sec (ON) 0 sec (OFF)
0.5	
SOURCE	Data sources (only for CR/LF)
MAIN	Main counter
BATCH	Batch counter
TOTAL	Totalizer
MAI.BAT	Main and batch counter
MAI.TOT	Main counter and totalizer

The data sources can be set according to the output operations set for the device. If the concerned counter shows an overflow / underflow, it does not send the data value, but +oooooo / +uuuuuu.

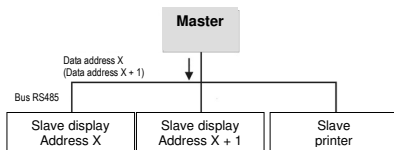
INPUT	Main menu for programming the signal and control inputs
MPI.INP.1	User input 1
MPI.INP.2	User input 2
PRINT	Triggering of the data transmission via the user input; possible cycle time $\geq 0,5$ sec (possible only for CR/LF and cycle time = 0)



Factory settings are highlighted in grey.

4 CR/LF protocol

In the CR/LF protocol, the device can transmit data values to a slave device according to the data source set.



4.1 Transmission



The data bytes are coded in ASCII.

Address 1: Slave address set in the device

Address 2: Slave address incremented by 1

Data value: 6 digits (+ decimal point)

Text: additional transmitted text for data source setting d) or e)

4.1.1 Transmission principle for data source main counter, batch counter or totalizer

Transmission: main or secondary counter

Address 1	Space	Sign	Data value	Carriage Return	Line Feed
2 bytes	1 byte	1 byte	6 or 7 bytes	1 byte	1 byte

Example for data source main counter:

01	[SPACE]	-	123456	[CR]	[LF]
3031 _{hex}	20 _{hex}	2D _{hex}	3132 3334 3536 _{hex}	0D _{hex}	0A _{hex}

05	[SPACE]	+	000000	[CR]	[LF]
3035 _{hex}	20 _{hex}	2B _{hex}	6F6F 6F6F 6F6F _{hex}	0D _{hex}	0A _{hex}

Example for data source totalizer:

01	[SPACE]	+	000.456	[CR]	[LF]
3031 _{hex}	20 _{hex}	2B _{hex}	30 3030 2E34 3536 _{hex}	0D _{hex}	0A _{hex}



Bold: ASCII
Normal: HEX value

4.1.2 Transmission principle for data source main and batch counter / main counter and totalizer

Transmission: main and secondary counter

Address 1	Space	Text	Space	Sign	Data value	Carriage Return	Line Feed
2 bytes	1 byte	4 bytes	1 byte	1 byte	6 or 7 bytes	1 byte	1 byte

Address 2	Space	Text	Space	Sign	Data value	Carriage Return	Line Feed
2 bytes	1 byte	5 bytes	1 byte	1 byte	6 or 7 bytes	1 byte	1 byte

Example for data source main and batch counter:

15	[SPACE]	MAIN	[SPACE]	+	000259	[CR]	[LF]
3135 _{hex}	20 _{hex}	4D41 494E _{hex}	20 _{hex}	2B _{hex}	3030 3032 3539 _{hex}	0D _{hex}	0A _{hex}

16	[SPACE]	BATCH	[SPACE]	+	999999	[CR]	[LF]
3136 _{hex}	20 _{hex}	42 4154 4348 _{hex}	20 _{hex}	2B _{hex}	3939 3939 3939 _{hex}	0D _{hex}	0A _{hex}



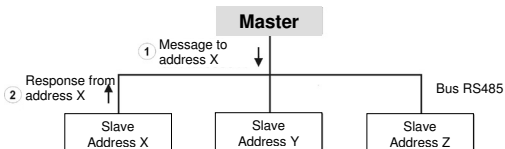
Bold: ASCII
Normal: HEX value

5 MODBUS protocol

The MODBUS data transmission in binary form takes place in the RTU (Remote Terminal Unit) operating mode, via the serial interface (RS485). The protocol is implemented following the specifications (*MODBUS Application Protocol Specification V1.1b* and *MODBUS over Serial Line – Specification and Implementation Guide V1.02.*) of the MODBUS Organization.

5.1 Master - Slave principle

The protocol operates according to the Master-Slave principle. A Master can communicate with one or several Slaves. Only the slave explicitly addressed by the Master is allowed to send data back to the Master.



5.2 Transmission principles

The protocol has two different transmission principles.

Unicast mode

In Unicast mode, the Master addresses a determined Slave. The Slave processes the message and sends an answer back to the Master. Every Slave has a univocal address in the address range from 1 to 247. The transmission principle always consists in a request from the Master and the subsequent response from the Slave. If the Master does not receive a response within a defined period of time (Timeout), the Master can assume that errors occurred during the transmission.

Broadcast mode

In Broadcast mode, the Master sends a write instruction (request) to all Slave devices, which do not generate any response. Broadcast addressing uses address 0.

5.3 Timeout

If the Master device does not receive a response within 0.5 seconds, it can discard the last request.

5.4 Message cycle

A message cycle ≤ 0.5 sec. is supported.

5.5 Byte order

The Modbus protocol uses the Big Endian format.

5.6 Structure of a message

Slave Address	Function code	Data	CRC
1 byte	1 byte	N bytes	2 bytes

Pauses within the transmission of a message

Pauses with a length ≥ 1.5 characters between the single message bytes lead to an error in the data frame.

Start and end identification

A message pause with a length of 3.5 characters is specified as start and end identification of a message. This pause must be respected between the single messages.

5.7 Function codes

The following function codes are supported.

Function code	description
03 _{hex}	Read several registers
10 _{hex}	Write several registers
11 _{hex}	Identification of the Slave

5.8 Data values

The 32 bit data values are transmitted via MODBUS either in the Integer or in the Float format. The register address defines the format to be used.

5.8.1 Float format (32 bits)

The Float values are coded in compliance with the IEEE 754 standard (single floating point) and have the following structure.



$$X = S \times M^E$$



S: Sign
E: Exponent
M: Mantissa
X: Total value

The data conversion in the floating point representation generates conversion errors (rounding error) due to the limited number of mantissa bits.

Machine precision

The relative machine precision indicates the maximum relative error that can be generated during rounding (conversion). For the Float values (single floating point), the machine precision is $\epsilon_0 = 5,96 \cdot 10^{-8}$.

5.8.2 Integer format (32 bits)

The Integer format is used in the representation as a two's complement. The device interprets the transmitted integer value. If the data value is represented in the device with the set number of decimal places, the decimal point is set for the integer in compliance with the set number of decimal places.

Example for a setting to 3 decimal places:

Transmitted integer value: 00000010_{hex}

Interpreted value: 000.016_{dec}

Address X + 1		Address X	
00 _{hex}	00 _{hex}	00 _{hex}	10 _{hex}
Byte 4	Byte 3	Byte 2	Byte 1

5.9 CRC (cyclic redundancy check)

A 16 bit CRC value is calculated according to a defined process for every data block and appended to the data block. For data check purposes, the same calculation process is applied to the data block, including the appended CRC value. If the result is zero, it can be assumed that the data block has not been corrupted.

CRC generation algorithm:

1. Loading of the value FFFF_{hex} in the 16 bit CRC register
2. Exclusive OR operation on the first byte sent or received with the low byte of the CRC register; store the result in the CRC register
3. Shift the CRC register 1 bit to the right and fill the msb with 0
4. If a 0 has been shifted out in step 3, this step is repeated. If a 1 has been shifted out in step 3, an exclusive OR operation is carried out on the CRC register with the value A001_{hex}.
5. Steps 3 and 4 are repeated until 8 bit shifts have been performed.
6. Steps 2 to 5 are repeated for the other bytes sent or received.
7. The result of the CRC generation is now in the CRC register. They are appended to the sent message after a byte exchange.



msb: most significant bit

5.10 Reading several registers

This instruction allows reading 16-bit registers.



XX_{hex} valid value in Hex-Format
LSB: least significant byte
MSB: most significant byte

Instruction (Master):

Field name	Value
Slave address	XX _{hex}
Function Read	03 _{hex}
Start at register address (MSB)	XX _{hex}
Start at register address (LSB)	XX _{hex}
Number of registers (MSB)	XX _{hex}
Number of registers (LSB)	XX _{hex}
CRC value (LSB)	XX _{hex}
CRC value (MSB)	XX _{hex}

Response (Slave):

Field name	Value
Slave address	XX _{hex}
Function Read	03 _{hex}
Number of data bytes	XX _{hex}
Register value 1 (MSB)	XX _{hex}
Register value 1 (LSB)	XX _{hex}
...	...
Register value N (MSB)	XX _{hex}
Register value N (LSB)	XX _{hex}
CRC value (LSB)	XX _{hex}
CRC value (MSB)	XX _{hex}

5.11 Writing several registers

This instruction allows writing 16-bit registers.

Instruction (Master):

Field name	Value
Slave address	XX _{hex}
Function Write	10 _{hex}
Start at register address (MSB)	XX _{hex}
Start at register address (LSB)	XX _{hex}
Number of registers (MSB)	XX _{hex}
Number of registers (LSB)	XX _{hex}
Number of data bytes (2 * number of registers)	XX _{hex}
Register value 1 (MSB)	XX _{hex}
Register value 1 (LSB)	XX _{hex}
...	...
Register value N (MSB)	XX _{hex}
Register value N (LSB)	XX _{hex}
CRC value (LSB)	XX _{hex}
CRC value (MSB)	XX _{hex}

Response (Slave):

Field name	Value
Slave address	XX _{hex}
Function Write	10 _{hex}
Start at register address (MSB)	XX _{hex}
Start at register address (LSB)	XX _{hex}
Number of registers (MSB)	XX _{hex}
Number of registers (LSB)	XX _{hex}
CRC value (LSB)	XX _{hex}
CRC value (MSB)	XX _{hex}

5.12 Identification

This instruction allows reading the identification of the Slave.



Both the Slave ID and the software version are transmitted in ASCII format.

Instruction (Master):

Field name	Value
Slave address	XX _{hex}
Function Identification	11 _{hex}
CRC value (LSB)	XX _{hex}
CRC value (MSB)	XX _{hex}

Response (Slave):

Field name	Value
Slave address	XX _{hex}
Function Identification	11 _{hex}
Number of data bytes (MSB)	00 _{hex}
Number of data bytes (LSB)	11 _{hex}
Slave ID byte 1	XX _{hex}
Slave ID byte 2	XX _{hex}
Slave ID byte 3	XX _{hex}
Slave ID byte 4	XX _{hex}
Slave ID byte 5	XX _{hex}
Slave ID byte 6	XX _{hex}
Slave ID byte 7	XX _{hex}
Slave ID byte 8	XX _{hex}
Status	FF _{hex}
Software Version byte 1	XX _{hex}
Software Version byte 2	XX _{hex}
Software Version byte 3	XX _{hex}
Software Version byte 4	XX _{hex}
Software Version byte 5	XX _{hex}
Software Version byte 6	XX _{hex}
Software Version byte 7	XX _{hex}
Software Version byte 8	XX _{hex}
CRC (LSB)	XX _{hex}
CRC (MSB)	XX _{hex}

5.13 Error protocol

The Slave sends no response if it receives a message sent by the Master including transmission errors.

Detected transmission errors:

- Parity errors
- No stop bit detected
- Error in the data frame
- Overrun error (data buffer overflow)
- CRC error

If the Slave device cannot carry out the uncorrupted message it received, it sends back to the Master an error response including the error code.

Error codes:

Code	Name	Description
01 _{hex}	Function not allowed	The function code is not defined in the device.
02 _{hex}	Address not allowed	The address is not available.
03 _{hex}	Data value not allowed	The received data value cannot be written. Data structure / data length is not correct.
04 _{hex}	Device error	Data value / data format is not correct. The device cannot or can only partially process the instruction.
10 _{hex}	Err1	Set value smaller than 0 is not allowed.
11 _{hex}	Err2	Set value larger than Preset2 is not allowed.

Error response:

Description	Value
Slave address	XX _{hex}
Function	80 _{hex} + Function code
Error code	XX _{hex}
CRC value (LSB)	XX _{hex}
CRC value (MSB)	XX _{hex}

5.14 Registers

All data values extend over 2 MODBUS registers and can only be processed as a whole.

Float register (4 bytes)

Address	Value	Access	Description	Note
0000 _{hex}	Main counter	r/w	Writing any value resets the main counter.	
0002 _{hex}	Secondary counter	r/w	Writing a value resets the main counter and the secondary counter.	
0004 _{hex}	Preset 1	r/w	Preset 1	
0006 _{hex}	Preset 2	r/w	Preset 2	
0008 _{hex}	Multiplication factor	-/w	Multiplication factor	
000A _{hex}	Division factor	-/w	Division factor	
000C _{hex}	Store set value	-/w	Storage of the set value (no setting is performed)	
000E _{hex}	Perform set function	-/w	Setting (writing any value performs the setting)	
0010 _{hex}	Preset 1 sign setting	-/w	Byte 1: Sign value Byte 2: 0 Byte 3: 0 Byte 4: 0	Sign values: 1: + values 2: -- values 3: +/- values
0012 _{hex}	Decimal point setting	r/w	Byte 1: decimal places Byte 2: 0 Byte 3: 0 Byte 4: 0	Decimal places: 0: 0 decimal places 1: 1 decimal place 2: 2 decimal places 3: 3 decimal places 4: 4 decimal places 5: 5 decimal places
0014 _{hex}	Status	r/o	Byte 1: bit1: output 1 bit2: output 2 Byte 2: bit1 - bit4: main counter bit5 - bit8: secondary counter Byte 3: 0 Byte 4: 0	Output: 1: ON 0: OFF Counter status: 0: counter status: regular 1: counter status: overflow 2: counter status: underflow



r/o: Read-only access
r/w: Read and write access
-/w: Write access

Integer register (4 bytes)

Address	Value	Access	Description	Note
8000 _{hex}	Main counter	r/w	Writing any value resets the main counter.	
8002 _{hex}	Secondary counter	r/w	Writing a value resets the main counter and the secondary counter.	
8004 _{hex}	Preset 1	r/w	Preset 1	
8006 _{hex}	Preset 2	r/w	Preset 2	
8008 _{hex}	Multiplication factor	-/w	Multiplication factor	
800A _{hex}	Division factor	-/w	Division factor	
800C _{hex}	Store set value	-/w	Storage of the set value (no setting is performed)	
800E _{hex}	Perform set function	-/w	Setting (writing any value performs the setting)	
8010 _{hex}	Preset 1 sign setting	-/w	Byte 1: Sign value Byte 2: 0 Byte 3: 0 Byte 4: 0	Sign values: 1: + values 2: -- values 3: +/- values
8012 _{hex}	Decimal point setting	r/w	Byte 1: decimal places Byte 2: 0 Byte 3: 0 Byte 4: 0	Decimal places: 0: 0 decimal places 1: 1 decimal place 2: 2 decimal places 3: 3 decimal places 4: 4 decimal places 5: 5 decimal places
8014 _{hex}	Status	r/o	Byte 1: bit1: output 1 bit2: output 2 Byte 2: bit1 - bit4: main counter bit5 - bit8: secondary counter Byte 3: 0 Byte 4: 0	Output: 1: ON 0: OFF Counter status: 0: counter status: regular 1: counter status: overflow 2: counter status: underflow



r/o: Read-only access
r/w: Read and write access
-/w: Write access

General guidelines:

- **Read/Write:** registers must be requested in compliance with the register layout described above. So registers can only be read and written as a whole and according to the authorized access.
- **Write:** writing instructions that cannot be performed because of format or over-range errors generate an error response with the error code *Device error*.
- **Read/Write:** if the device is in the programming menu or in the parameter setting menu, it will not perform any instruction or send back any error response.
- **Read:** the status request informs about overflows or underflows in the main and/or secondary counter and about the outputs switched on.
Status: 00001103_{hex} (output1 and output2 switched on; main counter: overflow; secondary counter: overflow)
- The Loc input has no effect on the functionality of the MODBUS.
- The sign alarm value 1 is relevant only in Trail Mode
- The main and/or secondary counters will only be reset when the reset mode of the counter is programmed to MAN.EL or MAN.RES.

Remarks about the Float registers:

- If the Timer function is programmed with the time format HH:MM:SS, the integer part of the float value will be used in the decimal form for setting the time value.
Examples:
 - Float data value 48DBD8E0_{hex} (450247_{dec}) will be interpreted as 45:02:47 (HH:MM:SS)
 - Float data value 48DBDB60_{hex} (450267_{dec}) cannot be represented. The time display 45:02:67 (HH:MM:SS) is not allowed. A device error is generated. Writing is not performed.
 - In the time format HH:MM:SS, writing the register for programming the decimal places is not possible. Reading the register is still possible.
- In case of Counter setting with input mode A / B and A % B, the multiplication and division factors cannot be set.
- In case of Tachometer setting, the tachometer value of the device cannot be reset. Setting and using a set value is not possible. The sign of Preset 1 cannot be described.

Remarks about the Integer registers:

- If the Timer function is programmed with the time format HH:MM:SS, the transmitted integer value will be interpreted as a time value.
Example:
 - Integer value 00010078_{hex} (65656_{dec}) will be interpreted as 6:56:56 (HH:MM:SS).
 - In the time format HH:MM:SS, writing the register for programming the decimal places is not possible. Reading the register is still possible.
- In case of Counter setting with input mode A / B and A % B, the multiplication and division factors cannot be set.
- In case of Tachometer setting, the tachometer value of the device cannot be reset. Setting and using a set value is not possible. The sign of Preset 1 cannot be described.

5.15 Message transmission examples

5.15.1 Example Read

Register address (Float): 0000_{hex} (main counter)
Set Slave address: 01_{hex}
Read data value: 3F80 0000_{hex} (1_{dec})

Master message:

Slave address	Function	Register address	Number of registers	CRC
01 _{hex}	03 _{hex}	0000 _{hex}	0002 _{hex}	C40B _{hex}

Slave response:

Slave address	Function	Number of bytes	Data value	CRC
01 _{hex}	03 _{hex}	04 _{hex}	3F80 0000 _{hex}	F7CF _{hex}

5.15.2 Example Write

Register address (Integer): 8014_{hex} (status)
Set Slave address: 01_{hex}
Error code: 04_{hex} (device error)

Master message:

Slave address	Function	Register address	Number of registers	Number of bytes	CRC
01 _{hex}	10 _{hex}	8014 _{hex}	0002 _{hex}	04 _{hex}	0DDD _{hex}

Slave response:

Slave address	Function	Error code	CRC
01 _{hex}	90 _{hex}	04 _{hex}	4DC3 _{hex}

5.15.3 Example Identification

Set Slave address: 01_{hex}
Slave ID: 3536 302E 302E 3035_{hex} (560.0.A5)
Software version: 5645 2E30 322E 3031_{hex} (VE.02.01)

Master message:

Slave address	Function	Register address	Number of registers	Number of bytes	CRC
01 _{hex}	10 _{hex}	8014 _{hex}	0002 _{hex}	04 _{hex}	0DDD _{hex}

Slave response:

Slave address	Function	Number of bytes	Slave ID	Status	Software version	CRC
01 _{hex}	90 _{hex}	04 _{hex}	3536 302E 302E 3035 _{hex}	FF _{hex}	5645 2E30 322E 3031 _{hex}	4DC3 _{hex}

6 Parameter sets

The following settings apply to parameter sets 1 to 3.

PR.TIME	1.0
SOURCE	MAIN
PROT.C.	MODBUS
ADRESS.	1

7 Technical data

7.1 Pulse counter

Response time of the outputs at the maximum counting frequency (see 8.1):

Add;Sub;Trail	< 13 ms
for automatic repetition	< 13 ms
A / B; (A-B) / A	< 70 ms

7.2 Frequency meter

Response time of the outputs:

1-channel operation:	< 100 ms @ 40 kHz
	< 160 ms @ 55 kHz
2-channel operation:	< 150 ms @ 40 kHz
	< 250 ms @ 55 kHz

7.3 Timer

Response time of the outputs:

Setting Seconds:	< 15 ms
Setting Minutes:	< 75 ms

Measuring error: < 100 ppm

8 Frequencies (typical)

NOTE: switching level of the inputs

Switching level with AC power supply:

HTL level	Low: 0 ... 4 V DC
	High: 12 ... 30 V DC
5V level	Low: 0 ... 2 V DC
	High: 3.5 ... 30 V DC

Switching level with DC power supply:

HTL level	Low: 0 ... $0.2 \times U_B$
	High: $0.6 \times U_B$... 30 V DC
5V level	Low: 0 ... 2 V DC
	High: 3.5 ... 30 V DC

8.1 Pulse counter

HTL level, rectangular signal shape 1:1

AC power supply	typ. Low	2.5 V
	typ. High	22 V
DC power supply 12V	typ. Low	2 V
	typ. High	10 V
DC power supply 24V	typ. Low	2.5 V
	typ. High	22 V

	Add Sub Trail	AddAr SubAr AddBat SubBat TrailAr	AddTot SubTot
Cnt.Dir	45 kHz	2.4 kHz	2.2 kHz
Up.Dn; Up.Up	25 kHz	2.2 kHz	2.1 kHz
Quad; Quad2	22 kHz	1.0 kHz	1.0 kHz
Quad4	17 kHz	0.8 kHz	0.8 kHz
A / B; (A - B) / A	27 kHz		

5V level, rectangular signal shape 1:1

typ. Low	1,0V
typ. High	4,0V

	Add Sub Trail	AddAr SubAr AddBat SubBat TrailAr	AddTot SubTot
Cnt.Dir	9 kHz	2,1 kHz	2,0 kHz
Up.Dn; Up.Up	9 kHz	2,0 kHz	2,0 kHz
Quad; Quad2	9 kHz	1,0 kHz	1,0 kHz
Quad4	9 kHz	0,8 kHz	0,8 kHz
A / B; (A - B) / A	9 kHz		

8.2 Frequency meter

HTL level, rectangular signal shape 1:1

AC power supply	typ. Low	2.5 V
	typ. High	22 V
DC power supply 12V	typ. Low	2 V
	typ. High	10 V
DC power supply 24V	typ. Low	2.5 V
	typ. High	22 V

5V level, rectangular signal shape 1:1

typ. Low	1.0 V
typ. High	4.0 V

	HTL	5V
A	55 kHz	9 kHz
A - B; A + B; A / B; (A - B) / A	55 kHz	9 kHz
Quad	27 kHz	9 kHz

9 Help texts

INTERF.		MAIN MENU INTERFACES
PROTOD.	MODBUS	MODBUS PROTOCOL
PROTOD.	CR.LF	CRLF PROTOCOL
ADRESS.	1-247 / 1-99	INTERFACE ADDRESS MODBUS / INTERFACE ADDRESS CRLF
PR.TIME	0,5 – 9999,9	SENDING CYCLE CRLF PROTOCOL
SOURCE	MAIN	CRLF DATA MAIN COUNTER
SOURCE	BATCH	CRLF DATA BATCH COUNTER
SOURCE	TOTAL	CRLF DATA TOTALIZER
SOURCE.	MAI.BAT.	CRLF DATA MAIN AND BATCH COUNTER
SOURCE.	MAI.TOT.	CRLF DATA MAIN COUNTER AND TOTALIZER
MP.INP.1	PRINT	MP INPUT1 FUNCTION PRINT
MP.INP.2	PRINT	MP INPUT2 FUNCTION PRINT

10 References

Information about the MODBUS protocol:

[1] **Modbus specification**

MODBUS Application Protocol Specification V1.1b

MODBUS over Serial Line – Specification and Implementation Guide V1.02.

www.modbus.org

Information about RS485:

[2] **ANSI/TIA/EIA-485-A-98**

Contador LED de preselección

96 x 48 mm, 2 preselecciones

Protocolo MODBUS Protocolo CR/LF

Opción RS485



Sumario

1	Descripción	3
2	Informaciones importantes	3
3	Interfaz	3
3.1	Opción tarjeta RS485	3
3.2	Conexión de aparato recomendada por la organización Modbus [1]	3
3.3	Configuración de la interfaz	4
3.4	Programación	4
4	Protocolo CR/LF	5
4.1	Transmisión	5
4.1.1	Principio de transmisión para las fuentes de datos contador principal, contador de lotes o calculadora	5
4.1.2	Principio de transmisión para las fuentes de datos contador principal y de lotes/ contador principal y totalizador	6
5	Protocolo MODBUS	6
5.1	Principio maestro-esclavo	6
5.2	Principios de transmisión	6
5.3	Timeout	7
5.4	Tiempo de ciclo de los mensajes	7
5.5	Sucesión de los bytes	7
5.6	Estructura de un mensaje	7
5.7	Códigos de función	7
5.8	Datos	7
5.8.1.	Formato Float (punto decimal flotante - 32 bits)	7
5.8.2.	Formato Integer (entero - 32 bits)	8
5.9	CRC (cyclic redundancy check)	8
5.10	Lectura de varios registros	9
5.11	Escritura de varios registros	9
5.12	Identificación	10
5.13	Protocolo en caso de error	10
5.14	Registros	11
5.15	Ejemplos de transmisión de mensajes	14
5.15.1	Ejemplo de lectura	14
5.15.2	Ejemplo de escritura	14
5.15.3	Ejemplo de identificación	14
6	Juegos de parámetros	15
7	Características técnicas	15
7.1	Contador de impulsos	15
7.2	Frecuencímetro	15
7.3	Contador horario	15
8	Frecuencias (típicas)	15
8.1	Contador de impulsos ¹	15
8.2	Frecuencímetro	16
9	Textos de ayuda	16
10	Referencias	16

1 Descripción

Estas instrucciones describen las opciones de interfaz RS485 y sus protocolos.

2 Informaciones importantes



Lea atentamente tanto estas instrucciones sobre las opciones de interfaz como las instrucciones de utilización del aparato de base antes de ponerlo en servicio.

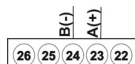
3 Interfaz

Los cables de señal se conectan por medio de 5 bornes roscados al paso de 3,5 mm.

3.1 Opción tarjeta RS485

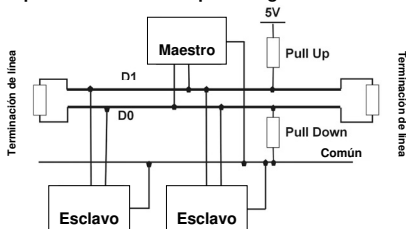
La interfaz RS485 permite una conexión multipuntos. Un máximo de 32 aparatos puede conectarse a un segmento de bus.

Conexión



La utilización de cables de baja capacidad y baja atenuación trenzados por pares (twisted pair) permite realizar conexiones extremadamente fiables en grandes distancias (máximo 1000 m).

3.2 Conexión de aparato recomendada por la organización Modbus [1]



Designación	
MODBUS	Aparato
D0	B (-)
D1	A (+)
Común	

3.3 Configuración de la interfaz

Configuración esclavo MODBUS (RTU)

Interfaz	Velocidad de trans.	Formato de los datos	Dirección
RS485	9600	1 bit inicial 8 bits de datos Paridad par 1 bit final	1-247

Configuración Maestro CR/LF

Interfaz	Velocidad de trans.	Formato de los datos	Dirección
RS485	9600	1 bit inicial 8 bits de datos Sin paridad 1 bit final	1-99

3.4 Programación

INTERF.	Menú principal de programación de la interfaz.
PROTOD.	Protocolos de interface
MODBUS	Protocolo MODBUS
CR.LF	Protocolo CR/LF
ADRESS.	Dirección esclavo CR/LF: 1 – 99 MODBUS: 1- 247
1	
PR.TIME	Tiempo de ciclo (CR/LF solamente) 0,5 – 9999,9 seg (activado) 0 seg (desactivado)
0.5	
SOURCE	Fuente de datos (CR/LF solamente)
MAIN	Contador principal
BATCH	Contador de lotes
TOTAL	Totalizador
MAI.BAT	Contador principal y de lotes
MAI.TOT	Contador principal y totalizador

Las fuentes de datos pueden regularse en función de las operaciones de salida elegidas para el aparato. En caso de rebasamiento de la capacidad del contador por la parte alta (overflow)/por la parte baja (underflow), la interfaz no emite el valor del contador, sino +000000 / +uuuuuu.

INPUT	Menú principal de programación de las entradas de señal y mando
MPI.INP.1	Entrada usuario 1
MPI.INP.2	Entrada usuario 2
PRINT	Inicio de la transmisión de los datos por medio de la entrada usuario. Tiempo de ciclo posible $\geq 0,5$ seg (posible únicamente para CR/LF y tiempo de ciclo = 0)



Los ajustes de fábrica están sobre un fondo gris.

4 Protocolo CR/LF

Con el protocolo CR/LF, el aparato puede transmitir datos a un aparato esclavo en función de la fuente de datos seleccionada.



4.1 Transmisión



Los bytes de datos están codificados en ASCII.

Dirección 1: dirección esclavo regulado en el aparato

Dirección 2: dirección esclavo incrementada de 1

Dato: 6 cifras (+ punto decimal)

Texto: Texto suplementario transmitido si la fuente de datos seleccionada es d) o e)

4.1.1 Principio de transmisión para las fuentes de datos contador principal, contador de lotes o calculadora

Transmisión: contador principal o secundario

Dirección 1	Espacio	Signo	Valor	Vuelta carro	Salto de línea
2 bytes	1 byte	1 byte	6 o 7 bytes	1 byte	1 byte

Ejemplo: fuente de datos contador principal:

01	[SPACE]	-	123456	[CR]	[LF]
3031 _{hex}	20 _{hex}	2D _{hex}	3132 3334 3536 _{hex}	0D _{hex}	0A _{hex}

05	[SPACE]	+	000000	[CR]	[LF]
3035 _{hex}	20 _{hex}	2B _{hex}	6F6F 6F6F 6F6F _{hex}	0D _{hex}	0A _{hex}

Ejemplo: fuente de datos totalizador:

01	[SPACE]	+	000.456	[CR]	[LF]
3031 _{hex}	20 _{hex}	2B _{hex}	30 3030 2E34 3536 _{hex}	0D _{hex}	0A _{hex}



Negrilla : ASCII

Normal : valor HEX

4.1.2 Principio de transmisión para las fuentes de datos contador principal y de lotes/ contador principal y totalizador

Transmisión: contador principal o secundario

Dirección 1	Espacio	Texto	Espacio	Signo	Valor	Vuelta carro	Salto de línea
2 bytes	1 byte	4 bytes	1 byte	1 byte	6 o 7 bytes	1 byte	1 byte

Dirección 2	Espacio	Texto	Espacio	Signo	Valor	Vuelta carro	Salto de línea
2 bytes	1 byte	5 bytes	1 byte	1 byte	6 o 7 bytes	1 byte	1 byte

Ejemplo: fuente de datos contador principal y totalizador:

15	[SPACE]	MAIN	[SPACE]	+	000259	[CR]	[LF]
3135hex	20hex	4D41 494Ehex	20hex	2Bhex	3030 3032 3539hex	0Dhex	0Ahex

16	[SPACE]	BATCH	[SPACE]	+	999999	[CR]	[LF]
3136hex	20hex	42 4154 4348hex	20hex	2Bhex	3939 3939 3939hex	0Dhex	0Ahex



Negrilla : ASCII
Normal : valor HEX

5 Protocolo MODBUS

La transmisión de datos MODBUS en forma binaria se efectúa en modo operativo RTU (Remote Terminal Unit) mediante un interfaz serie (RS485). El protocolo se aplica en la base de las especificaciones (*MODBUS Application Protocol Specification V1.1b* y *MODBUS over Serial Line – Specification and implementation Guide V1.02*.) de la organización MODBUS.

5.1 Principio maestro-esclavo

Este protocolo funciona según el principio maestro-esclavo. Un maestro puede comunicar con uno o varios esclavos. Solamente el esclavo al que el maestro se dirige explícitamente está autorizado a devolver datos al maestro.



5.2 Principios de transmisión

El protocolo dispone de dos principios de transmisión diferentes.

Modo Unicast

En modo Unicast, el maestro se dirige a un esclavo determinado. El esclavo trata el mensaje y envía una respuesta al maestro. Cada esclavo tiene una dirección única en la gama de direcciones de 1 a 247. El principio de transmisión consta siempre de una petición (request) del maestro y del mensaje en respuesta (response) enviada a continuación por el esclavo. Si no llega ninguna respuesta al maestro en un lapso de tiempo dado (Timeout), este último puede admitir que se hayan producido errores en el transcurso de la transmisión.

Modo Broadcast

En modo Broadcast, el maestro emite una orden de escritura (request) a todos los aparatos esclavo, que no generan en este caso ninguna respuesta (response). El direccionamiento Broadcast utiliza la dirección 0.

5.3 Timeout

Si el aparato maestro no recibe ninguna respuesta en un lapso de tiempo de 0,5 segundos, puede rechazar la última demanda.

5.4 Tiempo de ciclo de los mensajes

El sistema soporta un tiempo de ciclo de mensajes $\leq 0,5$ seg.

5.5 Sucesión de los bytes

El protocolo Modbus utiliza el formato Big Endian.

5.6 Estructura de un mensaje

Dirección esclavo	Código de función	Datos	CRC
1 byte	1 byte	N bytes	2 byte

Pausas durante la transmisión de los mensajes

Pausas $\geq 1,5$ longitudes de carácter entre los distintos bytes del mensaje ocasionan un error en la trama de datos.

Detección de inicio y final

Se especifica una pausa de mensajes de una longitud de 3,5 caracteres para detectar el inicio y el final de un mensaje. Esta pausa debe respetarse entre los distintos mensajes.

5.7 Códigos de función

Se aceptan los siguientes códigos de función.

Código de función	Descripción
03 _{hex}	Lectura de varios registros
10 _{hex}	Escritura de varios registros
11 _{hex}	Identificación del esclavo

5.8 Datos

Los datos de 32 bits se transmiten mediante MODBUS o al formato Integer (entero) o al formato Float (punto decimal flotante). El formato de datos que debe utilizarse viene determinado por la dirección del registro.

5.8.1. Formato Float (punto decimal flotante - 32 bits)

Los datos en formato Float se codifican según la norma IEEE 754 (single floating point) y presentan la siguiente estructura.

Dirección X + 1		Dirección X	
SEEEEEEE	EMMMMMMM	MMMMMMMM	MMMMMMMM
byte 4	byte 3	byte 2	byte 1

$$X = S \times M^E$$



S: Signo
E: Exponente
M: Mantisa
X: Valor total

La conversión de datos durante la representación con punto decimal flotante genera errores de conversión (errores de redondeado) debidos al número limitado de los bits de la mantisa.

Precisión máquina

La precisión máquina relativa especifica los errores relativos máximos que pueden sobrevenir durante el redondeado (conversión). Para un valor Float (single floating point), la precisión máquina es de $\epsilon_0 = 5,96 \cdot 10^{-8}$.

5.8.2. Formato Integer (entero - 32 bits)

El formato entero Integer se utiliza en la representación como un complemento de dos. El aparato interpreta el valor entero transmitido. Si el valor está representado en el aparato con el número de decimales especificado, el punto decimal se colocará en el valor entero según este número de decimales.

Ejemplo con 3 decimales:

Entero transmitido: 00000010_{hex}
Valor interpretado: 000,016_{dec}

Dirección X + 1		Dirección X	
00 _{hex}	00 _{hex}	00 _{hex}	10 _{hex}
byte 4	byte 3	byte 2	byte 1

5.9 CRC (cyclic redundancy check)

Se calcula un valor CRC de 16 bits para cada bloque de datos según un procedimiento determinado y se añade al bloque de datos. El mismo procedimiento de cálculo se aplica al bloque de datos, con el valor CRC adjunto, para la comprobación de los datos. Si el resultado obtenido es nulo, se admite que no se corrompió el bloque de datos.

Algoritmo de generación del CRC:

1. Carga del valor FFFF_{hex} en el registro CRC de 16 bits
1. Operación O exclusiva del primer byte de emisión o recepción con el byte de peso más bajo del registro CRC; memorización del resultado en el registro CRC
2. Desfase del registro CRC de un bit hacia la derecha y llenado del msb con 0
3. Si, durante la operación 3, el desfase causó la pérdida de un 0, se repite esta operación. Si, durante la operación 3, el desfase causó la pérdida de un 1, O exclusivo del registro CRC con el valor A001_{hex}.
4. Las operaciones 3 y 4 se repiten hasta realizar 8 desfases de bits.
5. Las operaciones 2 a 5 se repiten para los otros bytes de emisión o recepción.
6. El resultado de la generación del CRC se encuentra ahora en el registro CRC. Se añade al mensaje emitido después de una inversión de bytes.



msb : most significant bit (bit de peso más fuerte)

5.10 Lectura de varios registros

Esta instrucción permite leer registros de 16 bits.



XX_{hex} valor válido al formato Hex

LSB: least significa byte
(byte de peso más bajo)

MSB: most significa byte
(byte de peso más fuerte)

Instrucción (maestro) :

Nombre de campo	Valor
Dirección esclavo	XX _{hex}
Función Lectura	03 _{hex}
Inicio de registro con la dirección (MSB)	XX _{hex}
Inicio de registro con la dirección (LSB)	XX _{hex}
Número de registros (MSB)	XX _{hex}
Número de registros (LSB)	XX _{hex}
Valor CRC (LSB)	XX _{hex}
Valor CRC (MSB)	XX _{hex}

Respuesta (esclavo) :

Nombre de campo	Valor
Dirección esclavo	XX _{hex}
Función Lectura	03 _{hex}
Número de bytes de datos	XX _{hex}
Valor registro 1 (MSB)	XX _{hex}
Valor registro 1 (LSB)	XX _{hex}
...	...
Valor registro N (MSB)	XX _{hex}
Valor registro N (LSB)	XX _{hex}
Valor CRC (LSB)	XX _{hex}
Valor CRC (MSB)	XX _{hex}

5.11 Escritura de varios registros

Esta instrucción permite escribir registros de 16 bits.

Instrucción (maestro):

Nombre de campo	Valor
Dirección esclavo	XX _{hex}
Función Escritura	10 _{hex}
Inicio de registro con la dirección (MSB)	XX _{hex}
Inicio de registro con la dirección (LSB)	XX _{hex}
Número de registros (MSB)	XX _{hex}
Número de registros (LSB)	XX _{hex}
Número de bytes de datos (2 * número de registros)	XX _{hex}
Valor registro 1 (MSB)	XX _{hex}
Valor registro 1 (LSB)	XX _{hex}
...	...
Valor registro N (MSB)	XX _{hex}
Valor registro N (LSB)	XX _{hex}
Valor CRC (LSB)	XX _{hex}
Valor CRC (MSB)	XX _{hex}

Respuesta (esclavo) :

Nombre de campo	Valor
Dirección esclavo	XX _{hex}
Función Escritura	10 _{hex}
Inicio de registro con la dirección (MSB)	XX _{hex}
Inicio de registro con la dirección (LSB)	XX _{hex}
Número de registros (MSB)	XX _{hex}
Número de registros (LSB)	XX _{hex}
Valor CRC (LSB)	XX _{hex}
Valor CRC (MSB)	XX _{hex}

5.12 Identificación

Esta instrucción permite leer la identificación del esclavo.



El identificador del esclavo y la versión del software se transmiten los dos al formato ASCII.

Instrucción (maestro) :

Nombre de campo	Valor
Dirección esclavo	XX _{hex}
Función Identificación	11 _{hex}
Valor CRC (LSB)	XX _{hex}
Valor CRC (MSB)	XX _{hex}

Respuesta (esclavo) :

Nombre de campo	Valor
Dirección esclavo	XX _{hex}
Función Identificación	11 _{hex}
Número de bytes de datos (MSB)	00 _{hex}
Número de bytes de datos (LSB)	11 _{hex}
ID esclavo byte 1	XX _{hex}
ID esclavo byte 2	XX _{hex}
ID esclavo byte 3	XX _{hex}
ID esclavo byte 4	XX _{hex}
ID esclavo byte 5	XX _{hex}
ID esclavo byte 6	XX _{hex}
ID esclavo byte 7	XX _{hex}
ID esclavo byte 8	XX _{hex}
Estado	FF _{hex}
Versión software byte 1	XX _{hex}
Versión software byte 2	XX _{hex}
Versión software byte 3	XX _{hex}
Versión software byte 4	XX _{hex}
Versión software byte 5	XX _{hex}
Versión software byte 6	XX _{hex}
Versión software byte 7	XX _{hex}
Versión software byte 8	XX _{hex}
CRC (LSB)	XX _{hex}
CRC (MSB)	XX _{hex}

5.13 Protocolo en caso de error

Si el esclavo recibe un mensaje enviado por el maestro que contiene errores de transmisión, no envía ninguna respuesta.

Errores de transmisión reconocidos:

- Error de paridad
- No se detecta ningún bit final
- Error en la trama de datos
- Error Overrun (desbordamiento de capacidad del tampón de los datos)
- Error CRC

Si el esclavo no puede ejecutar la instrucción de un mensaje no corrompido que hubiera recibido, envía una respuesta de error, con un código de error, al maestro.

Códigos de error:

Código	Nombre	Descripción
01 _{hex}	Función no autorizada	El código de función no se define en el aparato.
02 _{hex}	Dirección no autorizada	La dirección no está disponible.
03 _{hex}	Valor de dato no autorizado	El valor recibido no puede escribirse. Estructura de datos/longitud de datos incorrecta.
04 _{hex}	Fallo aparato	Valor/formato de datos incorrecto. El aparato no puede tratar la instrucción totalmente o parcialmente.
10 _{hex}	Err1	Valor de precolocación 0 prohibido.
11 _{hex}	Err2	Valor de precolocación superior a Preset2 prohibido.

Respuesta de error:

Designación	Valor
Dirección esclavo	XX _{hex}
Función	80 _{hex} + Código de función
Código de error	XX _{hex}
Valor CRC (LSB)	XX _{hex}
Valor CRC (MSB)	XX _{hex}

5.14 Registros

Todos los valores de datos se extienden en 2 registros MODBUS y sólo pueden leerse y escribirse en forma de un valor global.

Registros Float (punto decimal flotante - 4 bytes)

Dirección	Valor	Acceso	Descripción	Observación
0000hex	Contador principal	r/w	La escritura de cualquier valor recoloca el contador principal.	
0002hex	Contador secundario	r/w	La escritura de un valor recoloca el contador principal y el contador secundario.	
0004hex	Preselección 1	r/w	Preselección 1	
0006hex	Preselección 2	r/w	Preselección 2	
0008hex	Factor de multiplicación	-/w	Factor de multiplicación	
000Ahex	Factor de división	-/w	Factor de división	
000Chex	Memorización del valor de precolocación	-/w	Memorización del valor de precolocación (no efectúa ninguna precolocación)	
000Ehex	Ejecución de la función de precolocación	-/w	Precolocación (la escritura de cualquier valor ejecuta la función de precolocación)	
0010hex	Definición del signo de la preselección 1	-/w	Byte 1: signo Byte 2: 0 Byte 3: 0 Byte 4: 0	Valores de la señal: 1: Valores + 2: Valores -- 3: Valores +/-
0012hex	Definición del punto decimal	r/w	Byte 1: decimales Byte 2: 0 Byte 3: 0 Byte 4: 0	Decimales: 0: 0 Decimales 1: 1 Decimal 2: 2 Decimales 3: 3 Decimales 4: 4 Decimales 5: 5 Decimales
0014hex	Estado	r/o	Byte 1: bit1: salida 1 bit2: salida 2 Byte 2: bit1 - bit4: contador principal bit5 - bit8: contador secundario Byte 3: 0 Byte 4: 0	Salida: 1: activa 0: inactiva Estado del contador: 0: estado: normal 1: estado: overflow 2: estado: underflow



r/o : Lectura solamente
r/w: Lectura y escritura
-/w: Escritura

Registros Integer (entero - 4 bytes)

Dirección	Valor	Acceso	Descripción	Observación
8000 _{hex}	Contador principal	r/w	La escritura de cualquier valor recoloca el contador principal.	
8002 _{hex}	Contador secundario	r/w	La escritura de un valor recoloca el contador principal y el contador secundario.	
8004 _{hex}	Preselección 1	r/w	Preselección 1	
8006 _{hex}	Preselección 2	r/w	Preselección 2	
8008 _{hex}	Factor de multiplicación	-/w	Factor de multiplicación	
800A _{hex}	Factor de división	-/w	Factor de división	
800C _{hex}	Memorización del valor de precolocación	-/w	Memorización del valor de precolocación (no efectúa ninguna precolocación)	
800E _{hex}	Ejecución de la función de precolocación	-/w	Precolocación (la escritura de cualquier valor ejecuta la función de precolocación)	
8010 _{hex}	Definición del signo de la preselección 1	-/w	Byte 1: signo Byte 2: 0 Byte 3: 0 Byte 4: 0	Valores de la señal: 1: Valores + 2: Valores -- 3: Valores +/-
8012 _{hex}	Definición del punto decimal	r/w	Byte 1: decimales Byte 2: 0 Byte 3: 0 Byte 4: 0	Decimales: 0: 0 Decimales 1: 1 Decimal 2: 2 Decimales 3: 3 Decimales 4: 4 Decimales 5: 5 Decimales
8014 _{hex}	Estado	r/o	Byte 1: bit1: salida 1 bit2: salida 2 Byte 2: bit1 - bit4: contador principal bit5 - bit8: contador secundario Byte 3: 0 Byte 4: 0	Salida: 1: activa 0: inactiva Estado del contador: 0: estado: normal 1: estado: overflow 2: estado: underflow



r/o : Lectura solamente
r/w: Lectura y escritura
-/w: Escritura

Observaciones de carácter general:

- **Lectura/escritura:** los registros deben llamarse según la estructura de registros descrita anteriormente. Por tanto, los registros sólo pueden leerse y escribirse en su integralidad y en función de las condiciones de acceso.
- **Escritura:** las instrucciones de escritura que no pueden ejecutarse debido a errores de formato o de rebasamiento del campo autorizado, generan una respuesta de error con el código de error *Fallo aparato*.
- **Lectura/escritura:** si el aparato se encuentra en el menú de programación o en el menú de parametrización, no realiza ninguna instrucción y no envía ninguna respuesta de error.
- **Lectura:** la demanda de estado permite saber si un rebasamiento de la capacidad por la parte alta o por la parte baja tuvo lugar en el contador principal y/o en el contador secundario, y cuáles son las salidas que están activadas.
Estado: 00001103_{hex} (salida 1 y salida 2 activadas; contador principal overflow; contador secundario: overflow)
- La entrada Loc no afecta la funcionalidad del MODBUS.
- El valor de alarma 1 para el signo sólo es pertinente en modo Trail.
- El contador principal y/o el contador secundario sólo pueden recolocarse si el modo de recolocación programado para el contador es MAN.EL o MAN.RES.

Observaciones sobre los registros Float (punto decimal flotante):

- Si la función Contador horario y el formato de tiempo HH:MM:SS están programados, se utiliza la parte entera del valor Float en la forma decimal para la determinación del tiempo.
Ejemplos:
 - El valor Float 48DBD8E0_{hex} (450247_{dec}) se interpreta como 45:02:47 (HH:MM:SS)
 - El valor Float 48DBDB60_{hex} (450267_{dec}) no puede representarse. La indicación de tiempo 45:02:67 (HH:MM:SS) no está autorizada. Se genera un fallo aparato. La escritura no se efectúa.
 - El registro para la programación de los decimales no puede escribirse en el formato de tiempo HH:MM:SS. No obstante, la lectura de este registro sigue siendo posible.
- En el caso del ajuste Counter con los modos de entrada A/B y A % B, no es posible regular los factores de multiplicación y división.
- Para el tacómetro, no es posible recolocar el valor del tacómetro del aparato. No es posible memorizar ni realizar un valor de recolocación. El signo de la preselección 1 no puede definirse.

Observaciones sobre los registros Integer (entero):

- Si la función Contador horario y el formato de tiempo HH:MM:SS están programados, el entero transmitido se interpreta como un valor de tiempo.
Ejemplo:
 - El valor entero 00010078_{hex} (65656_{dec}) se interpreta como 6:56:56 (HH:MM:SS).
 - El registro para la programación de los decimales no puede escribirse en el formato de tiempo HH:MM:SS. No obstante, la lectura de este registro sigue siendo posible.
- En el caso del ajuste Counter con los modos de entrada A/B y A % B, no es posible regular los factores de multiplicación y división.
- Para el tacómetro, no es posible recolocar el valor del tacómetro del aparato. No es posible memorizar ni realizar un valor de recolocación. El signo de la preselección 1 no puede definirse.

5.15 Ejemplos de transmisión de mensajes

5.15.1 Ejemplo de lectura

Dirección del registro (Float): 0000_{hex} (contador principal)
Dirección esclavo definido: 01_{hex}
Valor de dato leído: 3F80 0000_{hex} (1dec)

Mensaje emitido por el maestro:

Dirección esclavo	Función	Dirección registro	Número de registros	CRC
01 _{hex}	03 _{hex}	0000 _{hex}	0002 _{hex}	C40B _{hex}

Respuesta del esclavo:

Dirección esclavo	Función	Número de bytes	Valor de dato	CRC
01 _{hex}	03 _{hex}	04 _{hex}	3F80 0000 _{hex}	F7CF _{hex}

5.15.2 Ejemplo de escritura

Dirección del registro (Integer): 8014_{hex} (estado)
Dirección esclavo definido: 01_{hex}
Código de error: 04_{hex} (fallo aparato)

Mensaje emitido por el maestro:

Dirección esclavo	Función	Dirección registro	Nº de registros	Nº de bytes	CRC
01 _{hex}	10 _{hex}	8014 _{hex}	0002 _{hex}	04 _{hex}	0DDD _{hex}

Respuesta del esclavo:

Dirección esclavo	Función	Código de error	CRC
01 _{hex}	90 _{hex}	04 _{hex}	4DC3 _{hex}

5.15.3 Ejemplo de identificación

Dirección esclavo definido: 01_{hex}
ID esclavo: 3536 302E 302E 3035_{hex} (560.0.A5)
Versión de software: 5645 2E30 322E 3031_{hex} (VE.02.01)

Mensaje emitido por el maestro:

Dirección esclavo	Función	Dirección registro	Nº de registros	Nº de bytes	CRC
01 _{hex}	10 _{hex}	8014 _{hex}	0002 _{hex}	04 _{hex}	0DDD _{hex}

Respuesta del esclavo:

Dirección esclavo	Función	Nº bytes	ID esclavo	Estado	Versión software	CRC
01 _{hex}	90 _{hex}	04 _{hex}	3536 302E 302E 3035 _{hex}	FF _{hex}	5645 2E30 322E 3031 _{hex}	4DC3 _{hex}

6 Juegos de parámetros

Los siguientes ajustes se aplican a los juegos de parámetros 1 a 3.

PR.TIME	1.0
FUENTE	MAIN
PROT.C.	MODBUS
DIRECCION.	1

7 Características técnicas

7.1 Contador de impulsos

Tiempo de respuesta de las salidas a la frecuencia de recuento máxima (véase 8.1):

Add;Sub;Trail	< 13 ms
en repetición automática	< 13 ms
A/B; (A-B)/A	< 70 ms

7.2 Frecuencímetro

Tiempo de respuesta de las salidas:

Modo monocanal:	< 100 ms @ 40 kHz
	< 160 ms @ 55 kHz
Modo bicanal:	< 150 ms @ 40 kHz
	< 250 ms @ 55 kHz

7.3 Contador horario

Tiempo de respuesta de las salidas:

Ajuste segundos:	< 15 ms
Ajuste minutos:	< 75 ms

Error de medida	< 100 ppm
-----------------	-----------

8 Frecuencias (típicas)

NOTA: nivel de conmutación de las entradas

Nivel de conmutación para alimentación AC:

Nivel HTL	Bajo: 0 ... 4 V DC
	Alto: 12 ... 30 V DC

Nivel 5V	Bajo: 0 ... 2 V DC
	Alto: 3,5 ... 30 V DC

Nivel de conmutación para alimentación DC:

Nivel HTL	Bajo: 0 ... 0,2 × U _B
	Alto: 0,6 × U _B ... 30 V DC

Nivel 5V	Bajo: 0 ... 2 V DC
	Alto: 3,5 ... 30 V DC

8.1 Contador de impulsos 1

Nivel HTL, forma de señal rectangular 1:1

Alimentación AC	típ. Bajo 2,5 V
	típ. Alto 22 V
Alimentación DC 12V	típ. Bajo 2 V
	típ. Alto 10 V
Alimentación DC 24V	típ. Bajo 2,5 V
	típ. Alto 22 V

	Add Sub Trail	AddAr SubAr AddBat SubBat TrailAr	AddTot SubTot
Cnt.Dir	45 kHz	2,4 kHz	2,2 kHz
Up.Dn; Up.Up	25 kHz	2,2 kHz	2,1 kHz
Quad; Quad2	22 kHz	1,0 kHz	1,0 kHz
Quad4	17 kHz	0,8 kHz	0,8 kHz
A / B; (A - B) / A	27 kHz		

Nivel 5V, forma por señal rectangular 1:1

típ. Bajo	1,0 V
típ. Alto	4,0 V

	Add Sub Trail	AddAr SubAr AddBat SubBat TrailAr	AddTot SubTot
Cnt.Dir	9 kHz	2,1 kHz	2,0 kHz
Up.Dn; Up.Up	9 kHz	2,0 kHz	2,0 kHz
Quad; Quad2	9 kHz	1,0 kHz	1,0 kHz
Quad4	9 kHz	0,8 kHz	0,8 kHz
A / B; (A - B) / A	9 kHz		

8.2 Frecuencímetro

Nivel HTL, forma de señal rectangular 1:1

Alimentación AC	típ. Bajo 2,5 V típ. Alto 22 V
Alimentación DC 12V	típ. Bajo 2 V típ. Alto 10 V
Alimentación DC 24V	típ. Bajo 2,5 V típ. Alto 22 V

Nivel 5V, forma de señal rectangular 1:1

típ. Bajo 1,0 V
típ. Alto 4,0 V

	HTL	5V
A	55 kHz	9 kHz
A - B; A + B; A / B; (A - B) / A	55 kHz	9 kHz
Quad	27 kHz	9 kHz

9 Textos de ayuda

INTERF.		MENÚ PRINCIPAL INTERFACES
PROTOD.	MODBUS	PROTOCOLO MODBUS
PROTOD.	CR.LF	PROTOCOLO CRLF
ADRESS.	1-247 / 1-99	DIRECCIÓN DE INTERFAZ MODBUS/DIRECCIÓN DE INTERFAZ CRLF
PR.TIME	0,5 – 9999,9	CICLO DE EMISIÓN PROTOCOLO CRLF
SOURCE	MAIN	DATOS CRLF CONTADOR PRINCIPAL
SOURCE	BATCH	DATOS CRLF CONTADOR DE LOTES
SOURCE	TOTAL	DATOS CRLF TOTALIZADOR
SOURCE.	MAI.BAT.	DATOS CRLF CONTADOR PRINCIPAL Y CONTADOR DE LOTES
SOURCE.	MAI.TOT.	DATOS CRLF CONTADOR PRINCIPAL Y TOTALIZADOR
MP.INP.1	PRINT	FUNCIÓN ENTRADA MP 1 PRINT
MP.INP.2	PRINT	FUNCIÓN ENTRADA MP 2 PRINT

10 Referencias

Información sobre el protocolo MODBUS:

[1] **Especificaciones Modbus**

MODBUS Application Protocol Specification V1.1b

MODBUS over Serial Line - Specification and Implementation Guide V1.02.

www.modbus.org

Informaciones sobre RS485:

[2] **ANSI/TIA/EIA-485-A-98**

LED – Vorwahlzähler

96 x 48 mm, 2 Vorwahlen

MODBUS - Protokoll
CR/LF – Protokoll

Option RS485



Inhaltsverzeichnis

1	Beschreibung	3
2	Wichtige Hinweise	3
3	Schnittstelle	3
3.1	Optionsplatine RS485	3
3.2	Von der Modbus - Organisation [1] empfohlener Geräteanschluss	3
3.3	Schnittstellenkonfiguration	4
3.4	Programmierung	4
4	CR/LF-Protokoll	5
4.1	Übertragung	5
4.1.1	Übertragungsprinzip für Datenquelle Haupt-, Batch-, oder Gesamtsummenzähler	5
4.1.2	Übertragungsprinzip für Datenquelle Haupt- und Batchzähler / Haupt- und Gesamtsummenzähler	6
5	MODBUS - Protokoll	6
5.1	Master – Slave - Prinzip	6
5.2	Sendeprozess	6
5.3	Timeout	7
5.4	Nachrichtenzyklus	7
5.5	Byte-Reihenfolge	7
5.6	Aufbau einer Nachricht	7
5.7	Funktionscodes	7
5.8	Datenwerte	7
5.8.1	Float Format (32 Bit)	7
5.8.2	Integer Format (32 Bit)	8
5.9	CRC (cyclic redundancy check)	8
5.10	Lesen mehrerer Register	9
5.11	Schreiben mehrerer Register	9
5.12	Identifizierung	10
5.13	Fehlerprotokoll	10
5.14	Register	11
5.15	Beispiele für Nachrichtenübertragungen	14
5.15.1	Beispiel Lesen	14
5.15.2	Beispiel Schreiben	14
5.15.3	Beispiel Identifizierung	14
6	Parametersätze	15
7	Technische Daten	15
7.1	Impulszähler	15
7.2	Frequenzzähler	15
7.3	Zeitgeber	15
8	Frequenzen (typisch)	15
8.1	Impulszähler	15
8.2	Frequenzgeber	16
9	Hilfsteile	16
10	Referenzen	16

1 Beschreibung

Diese Anleitung beschreibt die Schnittstellen Optionen RS485 und ihre Protokolle.

2 Wichtige Hinweise



Lesen Sie sowohl diese Bedienungsanleitung über die Schnittstellen Optionen als auch die Bedienungsanleitung des Grundgerätes, bevor Sie das Gerät in Betrieb nehmen.

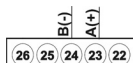
3 Schnittstelle

Signalleitungen werden über eine 5-polige Schraubklemme RM 3,5 angeschlossen.

3.1 Optionsplatine RS485

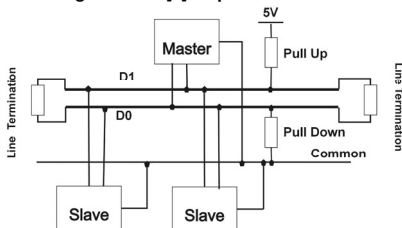
Die RS485 – Schnittstelle ermöglicht eine Mehrpunkt – Verbindung. Es ist möglich bis zu 32 Geräte in einem Bussegment anzuschließen.

Anschlussbelegung



Durch die Verwendung kapazitäts- und dämpfungsarmen, paarig verseiltem (twisted pair) - Kabel lassen sich extrem zuverlässige Verbindungen über eine höhere Distanz (maximal 1000 m) realisieren.

3.2 Von der Modbus - Organisation [1] empfohlener Geräteanschluss



Bezeichnung	
MODBUS	Gerät
D0	B (-)
D1	A (+)
Common	

3.3 Schnittstellenkonfiguration

MODBUS (RTU) Slave Konfiguration

Schnittstelle	Baudrate	Datenformat	Adresse
RS485	9600	1 Startbit 8 Datenbits Even Parity 1 Stoppbit	1-247

CR/LF Master Konfiguration

Schnittstelle	Baudrate	Datenformat	Adresse
RS485	9600	1 Startbit 8 Datenbits No Parity 1 Stoppbit	1-99

3.4 Programmierung

INTERF.	Hauptmenü zum Programmieren der Schnittstelle.
PROTOK.	Schnittstellen-Protokolle
MODBUS	MODBUS Protokoll
CR.LF	CR/LF - Protokoll
ADRESS.	Slave Adresse CR/LF: 1 – 99 MODBUS: 1- 247
1	
PR.TIME	Zykluszeit (nur bei CR/LF) 0,5 – 9999,9 sec (ein) 0 sec (aus)
0.5	
SOURCE	Datenquellen (nur bei CR/LF)
MAIN	Hauptzähler
BATCH	Batchzähler
TOTAL	Gesamtsummenzähler
MAI.BAT	Haupt- und Batchzähler
MAI.TOT	Haupt- und Gesamtsummenzähler

Die Datenquellen lassen sich in Abhängigkeit von den eingestellten Ausgangsoperationen des Gerätes einstellen. Hat der zu übertragende Zähler eine Overflow / Underflow, so wird nicht der Datenwert gesendet sondern +oooooo / +uuuuuu.

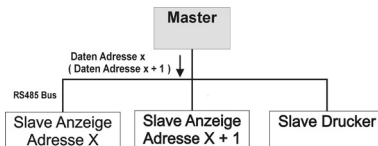
INPUT	Hauptmenü zum Programmieren der Signal- und Steuereingänge
MPI.INP.1	Benutzereingang 1
MPI.INP.2	Benutzereingang 2
PRINT	Auslösen der Datenübertragung über den Benutzereingang; Mögliche Zykluszeit $\geq 0,5$ sec (nur möglich bei CR/LF und Zykluszeit = 0)



Werkseinstellungen sind grau hinterlegt.

4 CR/LF-Protokoll

Im CR/LF – Protokoll kann das Gerät abhängig von der eingestellten Datenquelle Datenwerte an ein Slave-Gerät übertragen.



4.1 Übertragung



Die Datenbytes sind ASCII-kodiert.

Adresse 1: im Gerät eingestellte Slave – Adresse

Adresse 2: um 1 erhöhte Slave – Adresse

Datenwert: 6-stellig (+ Dezimalpunkt)

Text: zusätzlich übertragener Text bei eingestellter Datenquelle d) oder e)

4.1.1 Übertragungsprinzip für Datenquelle Haupt-, Batch-, oder Gesamtsummenzähler

Übertragung: Haupt- oder Nebenzähler

Adresse 1	Leerzeichen	Vorzeichen	Datenwert	Carriage Return	Line Feed
2 Byte	1 Byte	1 Byte	6 oder 7 Byte	1 Byte	1 Byte

Beispiele für Datenquelle Hauptzähler:

01	[SPACE]	-	123456	[CR]	[LF]
3031 _{hex}	20 _{hex}	2D _{hex}	3132 3334 3536 _{hex}	0D _{hex}	0A _{hex}

05	[SPACE]	+	000000	[CR]	[LF]
3035 _{hex}	20 _{hex}	2B _{hex}	6F6F 6F6F 6F6F _{hex}	0D _{hex}	0A _{hex}

Beispiel für Datenquelle Gesamtsummenzähler:

01	[SPACE]	+	000.456	[CR]	[LF]
3031 _{hex}	20 _{hex}	2B _{hex}	30 3030 2E34 3536 _{hex}	0D _{hex}	0A _{hex}



Fett: ASCII

Normal: HEX-Wert

4.1.2 Übertragungsprinzip für Datenquelle Haupt- und Batchzähler / Haupt- und Gesamtsummenzähler

Übertragung: Haupt- und Nebenzähler

Adresse 1	Leerzeichen	Text	Leerzeichen	Vorzeichen	Datenwert	Carriage Return	Line Feed
2 Byte	1 Byte	4 Byte	1 Byte	1 Byte	6 oder 7 Byte	1 Byte	1 Byte

Adresse 2	Leerzeichen	Text	Leerzeichen	Vorzeichen	Datenwert	Carriage Return	Line Feed
2 Byte	1 Byte	5 Byte	1 Byte	1 Byte	6 oder 7 Byte	1 Byte	1 Byte

Beispiel für Datenquelle Haupt- und Batchzähler:

15	[SPACE]	MAIN	[SPACE]	+	000259	[CR]	[LF]
3135 _{hex}	20 _{hex}	4D41 494E _{hex}	20 _{hex}	2B _{hex}	3030 3032 3539 _{hex}	0D _{hex}	0A _{hex}

16	[SPACE]	BATCH	[SPACE]	+	999999	[CR]	[LF]
3136 _{hex}	20 _{hex}	42 4154 4348 _{hex}	20 _{hex}	2B _{hex}	3939 3939 3939 _{hex}	0D _{hex}	0A _{hex}



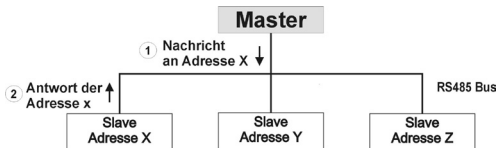
Fett: ASCII
Normal: HEX-Wert

5 MODBUS - Protokoll

Die MODBUS Datenübertragung in binärer Form erfolgt im Betriebsmodus RTU (Remote Terminal Unit) über die serielle Schnittstelle (RS485). Das Protokoll ist in Anlehnung an die Spezifikationen (*MODBUS Application Protocol Specification V1.1b* bzw. *MODBUS over Serial Line – Specification and Implementation Guide V1.02.*) der MODBUS-Organisation implementiert.

5.1 Master – Slave - Prinzip

Das Protokoll arbeitet nach dem Master-Slave Prinzip. Ein Master kann mit einem oder mehreren Slaves kommunizieren. Nur der vom Master explizit adressierte Slave darf Daten an den Master zurücksenden.



5.2 Sendeprinzipien

Das Protokoll kennt zwei unterschiedliche Sendeprinzipien.

Unicast – Modus

Im Unicast - Modus adressiert der Master einen bestimmten Slave. Der Slave verarbeitet die Nachricht und sendet eine Antwort an den Master zurück. Jeder Slave hat eine einzigartige Adresse im Adressbereich von 1 bis 247. Das Sendeprinzip besteht immer aus einer Anfrage (request) vom Master und der darauf gesendeten Antwort – Nachricht (response) des Slaves. Erhält der Master innerhalb einer festgelegten Zeit (Timeout) keine Antwort, kann dieser davon ausgehen, dass Fehler in der Übertragung aufgetreten sind.

Broadcast-Modus

Im Broadcast – Modus sendet der Master einen Schreibbefehl (request) an alle Slave – Geräte, die jedoch keine Antwort (response) generieren. Die Adresse 0 wird zur Broadcast – Adressierung benutzt.

5.3 Timeout

Empfängt das Master – Gerät innerhalb einer 0,5 sec keine Antwort, so kann er die letzte Anfrage verwerfen.

5.4 Nachrichtenzklus

Ein Nachrichtenzklus $\leq 0,5$ sec wird unterstützt.

5.5 Byte-Reihenfolge

Das Modbus – Protokoll benutzt das Big Endian Format.

5.6 Aufbau einer Nachricht

Slave - Adresse	Funktionscode	Daten	CRC
1 Byte	1 Byte	N Bytes	2 Byte

Pausen innerhalb der Nachrichtenübertragung

Pausen $\geq 1,5$ Zeichen Länge zwischen den einzelnen Nachrichtenbytes führen zu einem Fehler im Daten-Frame.

Start- und Ende - Kennung

Als Start- und Ende - Kennung einer Nachricht wird eine Nachrichtenpause von 3,5 Zeichen Länge spezifiziert. Diese Pause muss zwischen den einzelnen Nachrichten eingehalten werden.

5.7 Funktionscodes

Es werden folgende Funktionscodes unterstützt.

Funktionscode	Beschreibung
03 _{hex}	Mehrere Register lesen
10 _{hex}	Mehrere Register beschreiben
11 _{hex}	Identifizierung des Slaves

5.8 Datenwerte

Die 32-Bit Datenwerte werden über MODBUS entweder im Integer oder im Float Format übertragen. Die Register-Adresse legt das zu nutzende Datenformat fest.

5.8.1 Float Format (32 Bit)

Die Float-Werte sind nach dem IEEE 754 Standard (single floating point) kodiert und haben den folgenden Aufbau.



$$X = S \times M^E$$



S: Vorzeichen
E: Exponent
M: Mantisse
X: Gesamtwert

Durch die Datenwandlung in die Gleitkommadarstellung entstehen Wandlungsfehler (Rundungsfehler), deren Ursache in der begrenzten Anzahl an Mantissenbits liegt.

Maschinengenauigkeit

Die relative Maschinengenauigkeit gibt an, welche relativen Fehler beim Runden (Wandeln) maximal entstehen können. Beim Float – Wert (single floating point) liegt die Maschinengenauigkeit $\varepsilon_0 = 5,96 \cdot 10^{-8}$.

5.8.2 Integer Format (32 Bit)

Das ganzzahlige Integer Format wird in der Darstellung als Zweierkomplement eingesetzt. Das Gerät interpretiert den übertragenen Integer – Wert. Wird der Datenwert im Gerät mit den eingestellten Dezimalstellen dargestellt, so wird für den Integer – Wert das Komma entsprechend der eingestellten Dezimalstellen gesetzt.

Beispiel bei eingestellten 3 Dezimalstellen:

Übertragener Integer – Wert: 00000010_{hex}

Interpretierter Wert: 000,016_{dez}

Adresse X + 1		Adresse X	
00 _{hex}	00 _{hex}	00 _{hex}	10 _{hex}
Byte 4	Byte 3	Byte 2	Byte 1

5.9 CRC (cyclic redundancy check)

Für jeden Datenblock wird nach einem bestimmten Verfahren ein 16 Bit CRC-Wert berechnet, der dem Datenblock angefügt wird. Zur Überprüfung der Daten wird dasselbe Berechnungsverfahren auf den Datenblock einschließlich des angefügten CRC-Werts angewandt. Ist das Ergebnis dann Null, kann angenommen werden, dass der Datenblock unverfälscht ist.

Algorithmus zur CRC - Generierung:

1. Laden des 16 Bit CRC - Registers mit dem Wert FFFF_{hex}
2. Exklusiv - Oder Verknüpfung des ersten Sende- bzw. Empfangsbyte mit dem niederwertigen Byte des CRC - Registers; Speicherung des Ergebnis im CRC - Register
3. CRC - Register um 1 Bit nach rechts schieben und Auffüllen des msb mit 0
4. Wurde im Schritt 3 eine 0 hinausgeschoben, so wird dieser Schritt wiederholt. Wurde im Schritt 3 eine 1 hinausgeschoben, wird das CRC - Register mit dem Wert A001_{hex} Exklusiv - Oder verknüpft.
5. Die Schritte 3 und 4 werden solange wiederholt bis 8 Bitverschiebungen erfolgt sind.
6. Für die weiteren Sende- bzw. Empfangsbytes werden die Schritte 2 bis 5 wiederholt.
7. Das Ergebnis der CRC - Generierung befindet sich nun im CRC - Register. Nach einem Byte - Tausch werden sie der Sende-Nachricht angehängt.



msb: most significant bit

5.10 Lesen mehrerer Register

Mit diesem Befehl können 16 Bit Register gelesen werden.



XX_{hex} gültiger Wert im Hex-Format
LSB: least significant byte
MSB: most significant byte

Befehl (Master):

Feldname	Wert
Slave - Adresse	XX _{hex}
Funktion Lesen	03 _{hex}
Start bei Register - Adresse (MSB)	XX _{hex}
Start bei Register – Adresse (LSB)	XX _{hex}
Anzahl der Register (MSB)	XX _{hex}
Anzahl der Register (LSB)	XX _{hex}
CRC-Wert (LSB)	XX _{hex}
CRC-Wert (MSB)	XX _{hex}

Antwort (Slave):

Feldname	Wert
Slave - Adresse	XX _{hex}
Funktion Lesen	03 _{hex}
Anzahl der Daten-Bytes	XX _{hex}
Register Wert 1 (MSB)	XX _{hex}
Register Wert 1 (LSB)	XX _{hex}
...	...
Register Wert N (MSB)	XX _{hex}
Register Wert N (LSB)	XX _{hex}
CRC-Wert (LSB)	XX _{hex}
CRC-Wert (MSB)	XX _{hex}

5.11 Schreiben mehrerer Register

Mit diesem Befehl können 16 Bit Register geschrieben werden.

Befehl (Master):

Feldname	Wert
Slave - Adresse	XX _{hex}
Funktion Schreiben	10 _{hex}
Start bei Register - Adresse (MSB)	XX _{hex}
Start bei Register – Adresse (LSB)	XX _{hex}
Anzahl der Register (MSB)	XX _{hex}
Anzahl der Register (LSB)	XX _{hex}
Anzahl der Daten - Bytes (2 * Anzahl der Register)	XX _{hex}
Register Wert 1 (MSB)	XX _{hex}
Register Wert 1 (LSB)	XX _{hex}
...	...
Register Wert N (MSB)	XX _{hex}
Register Wert N (LSB)	XX _{hex}
CRC-Wert (LSB)	XX _{hex}
CRC-Wert (MSB)	XX _{hex}

Antwort (Slave):

Feldname	Wert
Slave - Adresse	XX _{hex}
Funktion Schreiben	10 _{hex}
Start bei Register - Adresse (MSB)	XX _{hex}
Start bei Register – Adresse (LSB)	XX _{hex}
Anzahl der Register (MSB)	XX _{hex}
Anzahl der Register (LSB)	XX _{hex}
CRC-Wert (LSB)	XX _{hex}
CRC-Wert (MSB)	XX _{hex}

5.12 Identifizierung

Mit diesem Befehl kann die Slave – Identifizierung gelesen werden.



Sowohl die Slave ID als auch die Software-Version werden im ASCII-Format übertragen.

Befehl (Master):

Feldname	Wert
Slave - Adresse	XX _{hex}
Funktion Identifizierung	11 _{hex}
CRC-Wert (LSB)	XX _{hex}
CRC-Wert (MSB)	XX _{hex}

Antwort (Slave):

Feldname	Wert
Slave - Adresse	XX _{hex}
Funktion Identifizierung	11 _{hex}
Anzahl der Daten - Bytes (MSB)	00 _{hex}
Anzahl der Daten - Bytes (LSB)	11 _{hex}
Slave ID Byte 1	XX _{hex}
Slave ID Byte 2	XX _{hex}
Slave ID Byte 3	XX _{hex}
Slave ID Byte 4	XX _{hex}
Slave ID Byte 5	XX _{hex}
Slave ID Byte 6	XX _{hex}
Slave ID Byte 7	XX _{hex}
Slave ID Byte 8	XX _{hex}
Status	FF _{hex}
Software Version Byte 1	XX _{hex}
Software Version Byte 2	XX _{hex}
Software Version Byte 3	XX _{hex}
Software Version Byte 4	XX _{hex}
Software Version Byte 5	XX _{hex}
Software Version Byte 6	XX _{hex}
Software Version Byte 7	XX _{hex}
Software Version Byte 8	XX _{hex}
CRC (LSB)	XX _{hex}
CRC (MSB)	XX _{hex}

5.13 Fehlerprotokoll

Der Slave sendet keine Antwort, wenn er die vom Master gesendete Nachricht mit Übertragungsfehler empfangen hat. Erkannte Übertragungsfehler sind:

- Paritätsfehler
- Kein Stopp – Bit erkannt
- Fehler im Daten – Frame
- Overrun – Fehler (Datenpuffer – Überlauf)
- CRC - Fehler

Kann das Slave - Gerät die fehlerfrei empfangene Nachricht nicht ausführen, so sendet er eine Fehler – Antwort inklusive Fehlercode an den Master zurück.

Fehlercodes:

Code	Name	Bedeutung
01 _{hex}	Funktion nicht erlaubt	Der Funktionscode ist im Gerät nicht definiert.
02 _{hex}	Adresse nicht erlaubt	Adresse steht nicht zur Verfügung
03 _{hex}	Datenwert nicht erlaubt	Der empfangene Datenwert kann nicht geschrieben werden. Datenstruktur / Datenlänge ist nicht korrekt
04 _{hex}	Gerätefehler	Datenwert / -Format ist nicht korrekt. Gerät kann den Befehl nicht oder nicht vollständig bearbeiten.
10 _{hex}	Err1	Setzwert kleiner 0 nicht erlaubt
11 _{hex}	Err2	Setzwert größer Preset2 nicht erlaubt

Fehlerantwort:

Bezeichnung	Wert
Slave - Adresse	XX _{hex}
Funktion	80 _{hex} + Funktionscode
Fehlercode	XX _{hex}
CRC-Wert (LSB)	XX _{hex}
CRC-Wert (MSB)	XX _{hex}

5.14 Register

Alle Datenwerte erstrecken sich über 2 MODBUS-Register und können nur als Gesamtwert angesprochen werden.

Float Register (4 Byte)

Adresse	Wert	Zugriff	Beschreibung	Hinweis
0000 _{hex}	Hauptzähler	r/w	Schreiben eines beliebigen Wertes setzt den Hauptzähler zurück.	
0002 _{hex}	Nebenzähler	r/w	Schreiben eines Wertes setzt Hauptzähler und Nebenzähler zurück.	
0004 _{hex}	Vorwahl 1	r/w	Vorwahl 1	
0006 _{hex}	Vorwahl 2	r/w	Vorwahl 2	
0008 _{hex}	Multiplikationsfaktor	-/w	Multiplikationsfaktor	
000A _{hex}	Divisionsfaktor	-/w	Divisionsfaktor	
000C _{hex}	Speicherung Setzwert	-/w	Speicherung Setzwert (kein Setzen)	
000E _{hex}	Ausführung Setzfunktion	-/w	Setzen (Schreiben eines beliebigen Wertes führt Setzfunktion aus)	
0010 _{hex}	Setzen Vorzeichen Vorwahl 1	-/w	Byte 1: Vorzeichenwert Byte 2: 0 Byte 3: 0 Byte 4: 0	Vorzeichenwerte: 1: + Werte 2: -- Werte 3: +/- Werte
0012 _{hex}	Setzen Dezimalpunkt	r/w	Byte 1:Dezimalstellen Byte 2: 0 Byte 3: 0 Byte 4: 0	Dezimalstellen: 0: 0 Dezimalstellen 1: 1 Dezimalstellen 2: 2 Dezimalstellen 3: 3 Dezimalstellen 4: 4 Dezimalstellen 5: 5 Dezimalstellen
0014 _{hex}	Status	r/o	Byte 1: Bit1: Ausgang 1 Bit2: Ausgang 2 Byte 2: Bit1 - Bit4: Hauptzähler Bit5 - Bit8: Nebenzähler Byte 3: 0 Byte 4: 0	Ausgang: 1: geschaltet 0: nicht geschaltet Zählerstand: 0:Zählerstand: regulär 1:Zählerstand: overflow 2:Zählerstand: underflow



r/o: Zugriff nur lesend
r/w: Zugriff lesend und schreibend
-/w: Zugriff schreibend

Integer Register (4 Byte)

Adresse	Wert	Zugriff	Beschreibung	Hinweis
8000 _{hex}	Hauptzähler	r/w	Schreiben eines beliebigen Wertes setzt den Hauptzähler zurück.	
8002 _{hex}	Nebenzähler	r/w	Schreiben eines Wertes setzt Hauptzähler und Nebenzähler zurück.	
8004 _{hex}	Vorwahl 1	r/w	Vorwahl 1	
8006 _{hex}	Vorwahl 2	r/w	Vorwahl 2	
8008 _{hex}	Multiplikationsfaktor	-/w	Multiplikationsfaktor	
800A _{hex}	Divisionsfaktor	-/w	Divisionsfaktor	
800C _{hex}	Speicherung Setzwert	-/w	Speicherung Setzwert (kein Setzen)	
800E _{hex}	Ausführung Setzfunktion	-/w	Setzen (Schreiben eines beliebigen Wertes führt Setzfunktion aus)	
8010 _{hex}	Setzen Vorzeichen Vorwahl 1	-/w	Byte 1: Vorzeichenwert Byte 2: 0 Byte 3: 0 Byte 4: 0	Vorzeichenwerte: 1: + Werte 2: -- Werte 3: +/- Werte
8012 _{hex}	Setzen Dezimalpunkt	r/w	Byte 1:Dezimalstellen Byte 2: 0 Byte 3: 0 Byte 4: 0	Dezimalstellen: 0: 0 Dezimalstellen 1: 1 Dezimalstellen 2: 2 Dezimalstellen 3: 3 Dezimalstellen 4: 4 Dezimalstellen 5: 5 Dezimalstellen
8014 _{hex}	Status	r/o	Byte 1: Bit1: Ausgang 1 Bit2: Ausgang 2 Byte 2: Bit1 - Bit4: Hauptzähler Bit5 - Bit8: Nebenzähler Byte 3: 0 Byte 4: 0	Ausgang: 1: geschalten 0: nicht geschalten Zählerstand: 0:Zählerstand: regulär 1:Zählerstand: overflow 2:Zählerstand: underflow



r/o: Zugriff nur lesend
 r/w: Zugriff lesend und schreibend
 -/w: Zugriff schreibend

Allgemeine Hinweise:

- **Lesen/Schreiben:** Register müssen nach dem oben beschriebenen Register Layout angefordert werden. Register können also nur vollständig und entsprechend dem erlaubten Zugriff gelesen und geschrieben werden.
- **Schreiben:** Schreibbefehle, die aufgrund von Formatfehlern oder Bereichsüberschreitungen nicht ausgeführt werden können, folgt eine Fehlerantwort mit dem Fehlercode *Gerätefehler*.
- **Lesen / Schreiben:** Ist das Gerät im Programmiermenü oder Parametrieremenü, so werden keine Befehle ausgeführt und auch keine Fehlerantwort zurückgeliefert.
- **Lesen:** Die Statusabfrage gibt Auskunft, ob beim Haupt- und/oder Nebenzähler einen Über- oder ein Unterlauf geschehen ist und welche Ausgänge geschaltet sind.
Status: 00001103_{hex} (Ausgang1 und Ausgang2 geschaltet; Hauptzähler: overflow; Nebenzähler: overflow)
- Der Loc - Eingang beeinflusst die Funktionalität des MODBUS nicht.
- Vorzeichen Alarmwert 1 ist nur relevant im Trail Mode
- Das Zurücksetzen des Haupt- und/oder Nebenzählers erfolgt nur, wenn der Rücksetzmode im Zähler auf MAN.EL oder MAN.RES programmiert wurde.

Bemerkung zu den Float Registern:

- Ist die Funktion Timer und das Zeitformat HH:MM:SS programmiert, so wird der ganzzahlige Anteil des Float-Wertes in dezimaler Schreibweise für das Setzen des Zeitwerts benutzt.
Beispiele:
 - Float-Datenwert 48DBD8E0_{hex} (450247_{dez}) wird interpretiert als 45:02:47 (HH:MM:SS)
 - Float-Datenwert 48DBDB60_{hex} (450267_{dez}) kann nicht dargestellt werden. Die Zeitanzeige 45:02:67 (HH:MM:SS) ist nicht erlaubt. Ein Gerätefehler wird gemeldet. Das Schreiben unterbleibt.
 - Das Register zur Programmierung der Dezimalstellen kann im Zeitformat HH:MM:SS nicht beschrieben werden. Ein Lesen des Registers ist aber möglich.
- Bei der Einstellung Counter und Input - Mode A / B und A % B kann der Multiplikationsfaktor und der Divisionsfaktor nicht eingestellt werden.
- Beim Tacho kann der Tachostand des Gerätes nicht zurückgesetzt werden. Ein Setzwert kann nicht gespeichert und nicht ausgeführt werden. Das Vorzeichen für die Vorwahl 1 kann nicht beschrieben werden.

Bemerkungen zu den Integer Registern:

- Ist die Funktion Timer und das Zeitformat HH:MM:SS programmiert, so wird der übertragene Integer-Wert als Zeitwert interpretiert.
Zum Beispiel:
 - Integer - Wert 00010078_{hex} (65656_{dez}) wird interpretiert als 6:56:56 (HH:MM:SS).
 - Das Register zur Programmierung der Dezimalstellen kann im Zeitformat HH:MM:SS nicht beschrieben werden. Ein Lesen des Registers ist aber möglich.
- Bei der Einstellung Counter und Input - Mode A / B und A % B kann der Multiplikationsfaktor und der Divisionsfaktor nicht eingestellt werden.
- Beim Tacho kann der Tachostand des Gerätes nicht zurückgesetzt werden. Ein Setzwert kann nicht gespeichert und nicht ausgeführt werden. Das Vorzeichen für Vorwahl 1 kann nicht beschrieben werden.

5.15 Beispiele für Nachrichtenübertragungen

5.15.1 Beispiel Lesen

Register Adresse (Float): 0000_{hex} (Hauptzähler)
Eingestellte Slave Adresse: 01_{hex}
Gelesener Datenwert: 3F80 0000_{hex} (1_{dez})

Master Nachricht:

Slave - Adresse	Funktion	Register Adresse	Anzahl Register	CRC
01 _{hex}	03 _{hex}	0000 _{hex}	0002 _{hex}	C40B _{hex}

Slave Antwort:

Slave - Adresse	Funktion	Anzahl Bytes	Datenwert	CRC
01 _{hex}	03 _{hex}	04 _{hex}	3F80 0000 _{hex}	F7CF _{hex}

5.15.2 Beispiel Schreiben

Register Adresse (Integer): 8014_{hex} (Status)
Eingestellte Slave Adresse: 01_{hex}
Fehlercode: 04_{hex} (Gerätefehler)

Master Nachricht:

Slave - Adresse	Funktion	Register Adresse	Anzahl Register	Anzahl Bytes	CRC
01 _{hex}	10 _{hex}	8014 _{hex}	0002 _{hex}	04 _{hex}	0DDD _{hex}

Slave Antwort:

Slave - Adresse	Funktion	Fehlercode	CRC
01 _{hex}	90 _{hex}	04 _{hex}	4DC3 _{hex}

5.15.3 Beispiel Identifizierung

Eingestellte Slave Adresse: 01_{hex}
Slave ID: 3536 302E 302E 3035_{hex} (560.0.A5)
Software Version: 5645 2E30 322E 3031_{hex} (VE.02.01)

Master Nachricht:

Slave - Adresse	Funktion	Register Adresse	Anzahl Register	Anzahl Bytes	CRC
01 _{hex}	10 _{hex}	8014 _{hex}	0002 _{hex}	04 _{hex}	0DDD _{hex}

Slave Antwort:

Slave - Adresse	Funktion	Anzahl Bytes	Slave ID	Status	Software Version	CRC
01 _{hex}	90 _{hex}	04 _{hex}	3536 302E 302E 3035 _{hex}	FF _{hex}	5645 2E30 322E 3031 _{hex}	4DC3 _{hex}

6 Parametersätze

Folgende Einstellungen gelten für die Parametersätze 1 bis 3.

PR.TIME	1.0
SOURCE	MAIN
PROTOD.	MODBUS
ADRESS.	1

7 Technische Daten

7.1 Impulszähler

Ansprechzeit der Ausgänge bei maximaler

Zählfrequenz (siehe 8.1):

Add;Sub;Trail < 13 ms

bei automatischer Wiederholung < 13 ms

A / B; (A-B) / A < 70 ms

7.2 Frequenzzähler

Ansprechzeit der Ausgänge:

1 – Kanal Betrieb: < 100 ms @ 40 kHz

< 160 ms @ 55 kHz

2 – Kanal Betrieb: < 150 ms @ 40 kHz

< 250 ms @ 55 kHz

7.3 Zeitzähler

Ansprechzeit der Ausgänge:

Sekunden – Einstellung: < 15 ms

Minuten – Einstellung: < 75 ms

Messfehler: < 100 ppm

8 Frequenzen (typisch)

HINWEIS: Schaltpegel der Eingänge

Schaltpegel bei AC – Versorgung:

HTL – Pegel Low: 0 ... 4 V DC

High: 12 ... 30 V DC

5V – Pegel Low: 0 ... 2 V DC

High: 3,5 ... 30 V DC

Schaltpegel bei DC – Versorgung:

HTL – Pegel Low: 0 ... 0,2 × U_B

High: 0,6 × U_B ... 30 V DC

5V – Pegel Low: 0 ... 2 V DC

High: 3,5 ... 30 V DC

8.1 Impulszähler

HTL-Pegel, Signalform Rechteck 1:1

AC-Versorgung typ. Low 2,5 V

typ. High 22 V

DC-Versorgung 12V typ. Low 2 V

typ. High 10 V

DC-Versorgung 24V typ. Low 2,5 V

typ. High 22 V

	Add Sub Trail	AddAr SubAr AddBat SubBat TrailAr	AddTot SubTot
Cnt.Dir	45 kHz	2,4 kHz	2,2 kHz
Up.Dn; Up.Up	25 kHz	2,2 kHz	2,1 kHz
Quad; Quad2	22 kHz	1,0 kHz	1,0 kHz
Quad4	17 kHz	0,8 kHz	0,8 kHz
A / B; (A - B) / A	27 kHz		

5V-Pegel, Signalform Rechteck 1:1

typ. Low 1,0V

typ. High 4,0V

	Add Sub Trail	AddAr SubAr AddBat SubBat TrailAr	AddTot SubTot
Cnt.Dir	9 kHz	2,1 kHz	2,0 kHz
Up.Dn; Up.Up	9 kHz	2,0 kHz	2,0 kHz
Quad; Quad2	9 kHz	1,0 kHz	1,0 kHz
Quad4	9 kHz	0,8 kHz	0,8 kHz
A / B; (A - B) / A	9 kHz		

8.2 Frequenzzähler

HTL-Pegel, Signalform Rechteck 1:1

AC-Versorgung		typ. Low	2,5 V
		typ. High	22 V
DC-Versorgung 12V		typ. Low	2 V
		typ. High	10 V
DC-Versorgung 24V		typ. Low	2,5 V
		typ. High	22 V

5V-Pegel, Signalform Rechteck 1:1

typ. Low	1,0 V
typ. High	4,0 V

	HTL	5V
A	55 kHz	9 kHz
A - B; A + B; A / B; (A - B) / A	55 kHz	9 kHz
Quad	27 kHz	9 kHz

9 Hilfstexte

INTERF.		HAUPTMENUE SCHNITTSTELLEN
PROTOK.	MODBUS	MODBUS PROTOKOLL
PROTOK.	CR.LF	CRLF PROTOKOLL
ADRESS.	1-247 / 1-99	SCHNITTSTELLENADRESSE MODBUS / SCHNITTSTELLENADRESSE CRLF
PR.TIME	0,5 – 9999,9	SENDEZYKLUS CRLF PROTOKOLL
SOURCE	MAIN	CRLF DATEN HAUPTZAEHLER
SOURCE	BATCH	CRLF DATEN BATCHZAEHLER
SOURCE	TOTAL	CRLF DATEN GESAMTSUMMENZAEHLER
SOURCE.	MAI.BAT.	CRLF DATEN HAUPT- UND BATCHZAEHLER
SOURCE.	MAI.TOT.	CRLF DATEN HAUPT- UND GESAMTSUMMENZAEHLER
MP.INP.1	PRINT	FUNKTION MP-EINGANG1 PRINT
MP.INP.2	PRINT	FUNKTION MP-EINGANG2 PRINT

10 Referenzen

Informationen zum MODBUS – Protokoll:

[1] **Modbus Spezifikation**

MODBUS Application Protocol Specification V1.1b

MODBUS over Serial Line – Specification and Implementation Guide V1.02.

www.modbus.org

Informationen zu RS485:

[2] **ANSI/TIA/EIA-485-A-98**

Compteur LED à présélection

96 x 48 mm, 2 présélections

Protocole MODBUS Protocole CR/LF

Option RS485



Sommaire

1	Description	3
2	Informations importantes	3
3	Interface	3
3.1	Option carte RS485	3
3.2	Raccordement d'appareil recommandé par l'organisation Modbus [1]	3
3.3	Configuration de l'interface	4
3.4	Programmation	4
4	Protocole CR/LF	5
4.1	Transmission	5
4.1.1	Principe de transmission pour les sources de données compteur principal, compteur de lots ou totalisateur	5
4.1.2	Principe de transmission pour les sources de données compteur principal et de lots / compteur principal et totalisateur	6
5	Protocole MODBUS	6
5.1	Principe maître - esclave	6
5.2	Principes de transmission	6
5.3	Timeout	7
5.4	Temps de cycle des messages	7
5.5	Succession des octets	7
5.6	Structure d'un message	7
5.7	Codes de fonction	7
5.8	Données	7
5.8.1	Format Float (point décimal flottant - 32 bits)	7
5.8.2	Format Integer (entier - 32 bits)	8
5.9	CRC (cyclic redundancy check)	8
5.10	Lecture de plusieurs registres	9
5.11	Ecriture de plusieurs registres	9
5.12	Identification	10
5.13	Protocole en cas d'erreur	10
5.14	Registres	11
5.15	Exemples de transmissions de messages	14
5.15.1	Exemple de lecture	14
5.15.2	Exemple d'écriture	14
5.15.3	Exemple d'identification	14
6	Jeux de paramètres	15
7	Caractéristiques techniques	15
7.1	Compteur d'impulsions	15
7.2	Fréquencemètre	15
7.3	Compteur horaire	15
8	Fréquences (typiques)	15
8.1	Compteur d'impulsions	15
8.2	Fréquencemètre	16
9	Textes d'aide	16
10	Références	16

1 Description

Ces instructions décrivent les options d'interface RS485 et leurs protocoles.

2 Informations importantes



Lisez attentivement aussi bien ces instructions sur les options d'interface que les instructions d'utilisation de l'appareil de base avant de mettre celui-ci en service.

3 Interface

Les câbles de signal sont raccordés au moyen d'un bornier de 5 bornes à visser au pas de 3,5 mm.

3.1 Option carte RS485

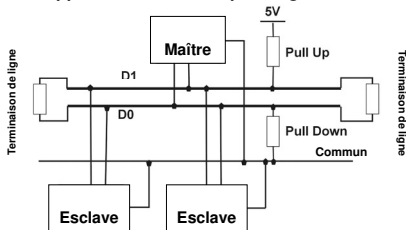
L'interface RS485 permet une liaison multipoint. Un maximum de 32 appareils peut être relié à un segment de bus.

Raccordement



L'utilisation de câbles de faible capacité et à basse atténuation torsadés par paires (twisted pair) permet la réalisation de liaisons extrêmement fiables sur de grandes distances (maximum 1000 m).

3.2 Raccordement d'appareil recommandé par l'organisation Modbus [1]



Désignation	
MODBUS	Appareil
D0	B (-)
D1	A (+)
Common	

3.3 Configuration de l'interface

Configuration Esclave MODBUS (RTU)

Interface	Vitesse de trans.	Format des données	Adresse
RS485	9600	1 bit de début 8 bits de données Parité paire 1 bit de fin	1-247

Configuration Maître CR/LF

Interface	Vitesse de trans.	Format des données	Adresse
RS485	9600	1 bit de début 8 bits de données Pas de parité 1 bit de fin	1-99

3.4 Programmation

INTERF.	Menu principal de programmation de l'interface.
PROTOC.	Protocoles d'interface
MODBUS	Protocole MODBUS
CR.LF	Protocole CR/LF
ADRESS.	Adresse Esclave CR/LF : 1 – 99 MODBUS : 1- 247
1	
PR.TIME	Temps de cycle (CR/LF seulement) 0,5 – 9999,9 sec (activé) 0 sec (désactivé)
0.5	
SOURCE	Sources de données (CR/LF seulement)
MAIN	Compteur principal
BATCH	Compteur de lots
TOTAL	Totalisateur
MAI.BAT	Compteur principal et de lots
MAI.TOT	Compteur principal et totalisateur

Les sources de données peuvent se régler en fonction des opérations de sortie choisies pour l'appareil. En cas de dépassement de la capacité du compteur par le haut (overflow) / par le bas (underflow), l'interface n'émet pas la valeur du compteur, mais +oooooo / +uuuuuu.

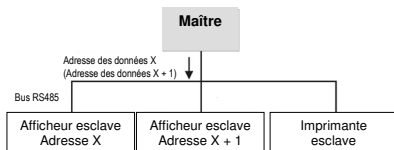
INPUT	Menu principal de programmation des entrées de signal et de commande
MPI.INP.1	Entrée utilisateur 1
MPI.INP.2	Entrée utilisateur 2
PRINT	Déclenchement de la transmission des données au moyen de l'entrée utilisateur. Temps de cycle possible $\geq 0,5$ sec. (possible uniquement pour CR/LF et temps de cycle = 0)



Les réglages d'usine sont indiqués sur fond gris.

4 Protocole CR/LF

Avec le protocole CR/LF, l'appareil peut transmettre des données à un appareil esclave en fonction de la source de données sélectionnée.



4.1 Transmission



Les octets de données sont codés en ASCII.

Adresse 1 : Adresse esclave réglée dans l'appareil

Adresse 2 : Adresse esclave incrémentée de 1

Donnée : 6 chiffres (+ point décimal)

Texte : Texte supplémentaire transmis si la source de données sélectionnée est d) ou e)

4.1.1 Principe de transmission pour les sources de données compteur principal, compteur de lots ou totalisateur

Transmission : compteur principal ou secondaire

Adresse 1	Espace	Signe	Valeur	Retour chariot	Saut de ligne
2 octets	1 octet	1 octet	6 ou 7 octets	1 octet	1 octet

Exemple : source de données compteur principal :

01	[SPACE]	-	123456	[CR]	[LF]
3031hex	20hex	2Dhex	3132 3334 3536hex	0Dhex	0Ahex
05	[SPACE]	+	000000	[CR]	[LF]
3035hex	20hex	2Bhex	6F6F 6F6F 6F6Fhex	0Dhex	0Ahex

Exemple : source de données totalisateur :

01	[SPACE]	+	000.456	[CR]	[LF]
3031hex	20hex	2Bhex	30 3030 2E34 3536hex	0Dhex	0Ahex



Gras : ASCII

Normal : valeur HEX

4.1.2 Principe de transmission pour les sources de données compteur principal et de lots / compteur principal et totalisateur

Transmission : compteur principal ou secondaire

Adresse 1	Espace	Texte	Espace	Signe	Valeur	Retour chariot	Saut de ligne
2 octets	1 octet	4 octets	1 octet	1 octet	6 ou 7 octets	1 octet	1 octet

Adresse 2	Espace	Texte	Espace	Signe	Valeur	Retour chariot	Saut de ligne
2 octets	1 octet	5 octets	1 octet	1 octet	6 ou 7 octets	1 octet	1 octet

Exemple : source de données compteur principal et totalisateur :

15	[SPACE]	MAIN	[SPACE]	+	000259	[CR]	[LF]
3135hex	20hex	4D41 494Ehex	20hex	2Bhex	3030 3032 3539hex	0Dhex	0Ahex

16	[SPACE]	BATCH	[SPACE]	+	999999	[CR]	[LF]
3136hex	20hex	42 4154 4348hex	20hex	2Bhex	3939 3939 3939hex	0Dhex	0Ahex



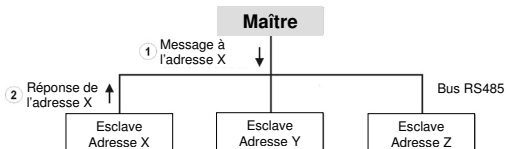
Gras : ASCII
Normal : valeur HEX

5 Protocole MODBUS

La transmission de données MODBUS en forme binaire s'effectue en mode opératoire RTU (Remote Terminal Unit) via une interface série (RS485). Le protocole est implémenté sur la base des spécifications (*MODBUS Application Protocol Specification V1.1b* et *MODBUS over Serial Line – Specification and Implementation Guide V1.02.*) de l'organisation MODBUS.

5.1 Principe maître - esclave

Ce protocole fonctionne selon le principe maître-esclave. Un maître peut communiquer avec un ou plusieurs esclaves. Seul l'esclave auquel le maître s'adresse explicitement est autorisé à retourner des données au maître.



5.2 Principes de transmission

Le protocole dispose de deux principes de transmission différents.

Mode Unicast

En mode Unicast, le maître s'adresse à un esclave déterminé. L'esclave traite le message et renvoie une réponse au maître. Chaque esclave a une adresse unique dans la plage d'adresses de 1 à 247. Le principe de transmission se compose toujours d'une requête (request) du maître et du message en réponse (response) envoyé ensuite par l'esclave. Si aucune réponse ne parvient au maître dans un laps de temps donné (Timeout), ce dernier peut admettre que des erreurs se sont produites au cours de la transmission.

Mode Broadcast

En mode Broadcast, le maître émet un ordre d'écriture (request) à tous les appareils esclave, qui ne génèrent dans ce cas aucune réponse (response). L'adressage Broadcast utilise l'adresse 0.

5.3 Timeout

Si l'appareil maître ne reçoit aucune réponse dans un laps de temps de 0,5 seconde, il peut rejeter la dernière demande.

5.4 Temps de cycle des messages

Le système supporte un temps de cycle des messages $\leq 0,5$ sec.

5.5 Succession des octets

Le protocole Modbus utilise le format Big Endian.

5.6 Structure d'un message

Adresse esclave	Code de fonction	Données	CRC
1 octet	1 octet	N octets	2 octets

Pauses dans la transmission des messages

Des pauses $\geq 1,5$ longueurs de caractère entre les différents octets du message entraînent une erreur dans la trame des données.

Détection de début et de fin

Une pause des messages d'une longueur de 3,5 caractères est spécifiée pour la détection du début et de la fin d'un message. Cette pause doit être respectée entre les différents messages.

5.7 Codes de fonction

Les codes de fonction suivants sont supportés.

Code de fonction	Description
03 _{hex}	Lecture de plusieurs registres
10 _{hex}	Ecriture de plusieurs registres
11 _{hex}	Identification de l'esclave

5.8 Données

Les données de 32 bits sont transmises via MODBUS soit au format Integer (entier) soit au format Float (point décimal flottant). Le format de données à utiliser est déterminé par l'adresse du registre.

5.8.1 Format Float (point décimal flottant - 32 bits)

Les données au format Float sont codées selon la norme IEEE 754 (single floating point) et présentent la structure suivante.

Adresse X + 1		Adresse X	
SEEEEEEE	EMMMMMMM	MMMMMMMM	MMMMMMMM
Octet 4	Octet 3	Octet 2	Octet 1

$$X = S \times M^E$$



S: Signe
E: Exposant
M: Mantisse
X: Valeur totale

La conversion des données lors de la représentation avec point décimal flottant génère des erreurs de conversion (erreurs d'arrondi) dues au nombre limité des bits de la mantisse.

Précision machine

La précision machine relative spécifie les erreurs relatives maximales pouvant survenir lors de l'arrondi (conversion). Pour une valeur Float (single floating point), la précision machine est de $\epsilon_0 = 5,96 \cdot 10^{-8}$.

5.8.2 Format Integer (entier - 32 bits)

Le format entier Integer est utilisé dans la représentation comme un complément à deux. L'appareil interprète la valeur entière transmise. Si la valeur est représentée dans l'appareil avec le nombre de décimales spécifié, le point décimal sera placé sur la valeur entière selon ce nombre de décimales.

Exemple avec 3 décimales :

Entier transmis : 00000010_{hex}

Valeur interprétée : 000,016_{dec}

Adresse X + 1		Adresse X	
00 _{hex}	00 _{hex}	00 _{hex}	10 _{hex}
Octet 4	Octet 3	Octet 2	Octet 1

5.9 CRC (cyclic redundancy check)

Une valeur CRC à 16 bits est calculée pour chaque bloc de données selon une procédure déterminée et ajoutée au bloc de données. La même procédure de calcul est appliquée au bloc de données, avec la valeur CRC jointe, pour la vérification des données. Si le résultat obtenu est nul, il est admis que le bloc de données n'a pas été corrompu.

Algorithme de génération du CRC :

1. Chargement de la valeur FFFF_{hex} dans le registre CRC à 16 bits
2. Opération OU exclusive du premier octet d'émission ou de réception avec l'octet de poids le plus faible du registre CRC ; mémorisation du résultat dans le registre CRC
3. Décalage du registre CRC d'un bit vers la droite et remplissage du msb avec 0
4. Si, lors de l'opération 3, le décalage a provoqué la perte d'un 0, cette opération est répétée. Si, lors de l'opération 3, le décalage a provoqué la perte d'un 1, OU exclusif du registre CRC avec la valeur A001_{hex}.
5. Les opérations 3 et 4 sont répétées jusqu'à la réalisation de 8 décalages de bits.
6. Les opérations 2 à 5 sont répétées pour les autres octets d'émission ou de réception.
7. Le résultat de la génération du CRC se trouve maintenant dans le registre CRC. Il est ajouté au message émis après une inversion d'octets.



msb : most significant bit (bit de poids le plus fort)

5.10 Lecture de plusieurs registres

Cette instruction permet la lecture de registres à 16 bits.



XX_{hex} valeur valide au format Hex

LSB: least significant byte (octet de poids le plus faible)

MSB: most significant byte (octet de poids le plus fort)

Instruction (maître) :

Nom de champ	Valeur
Adresse esclave	XX _{hex}
Fonction Lecture	03 _{hex}
Début au registre avec l'adresse (MSB)	XX _{hex}
Début au registre avec l'adresse (LSB)	XX _{hex}
Nombre de registres (MSB)	XX _{hex}
Nombre de registres (LSB)	XX _{hex}
Valeur CRC (LSB)	XX _{hex}
Valeur CRC (MSB)	XX _{hex}

Réponse (esclave) :

Nom de champ	Valeur
Adresse esclave	XX _{hex}
Fonction Lecture	03 _{hex}
Nombre d'octets de données	XX _{hex}
Valeur registre 1 (MSB)	XX _{hex}
Valeur registre 1 (LSB)	XX _{hex}
...	...
Valeur registre N (MSB)	XX _{hex}
Valeur registre N (LSB)	XX _{hex}
Valeur CRC (LSB)	XX _{hex}
Valeur CRC (MSB)	XX _{hex}

5.11 Ecriture de plusieurs registres

Cette instruction permet l'écriture de registres à 16 bits.

Instruction (maître) :

Nom de champ	Valeur
Adresse esclave	XX _{hex}
Fonction Ecriture	10 _{hex}
Début au registre avec l'adresse (MSB)	XX _{hex}
Début au registre avec l'adresse (LSB)	XX _{hex}
Nombre de registres (MSB)	XX _{hex}
Nombre de registres (LSB)	XX _{hex}
Nombre d'octets de données (2 * nombre de registres)	XX _{hex}
Valeur registre 1 (MSB)	XX _{hex}
Valeur registre 1 (LSB)	XX _{hex}
...	...
Valeur registre N (MSB)	XX _{hex}
Valeur registre N (LSB)	XX _{hex}
Valeur CRC (LSB)	XX _{hex}
Valeur CRC (MSB)	XX _{hex}

Réponse (esclave) :

Nom de champ	Valeur
Adresse esclave	XX _{hex}
Fonction Ecriture	10 _{hex}
Début au registre avec l'adresse (MSB)	XX _{hex}
Début au registre avec l'adresse (LSB)	XX _{hex}
Nombre de registres (MSB)	XX _{hex}
Nombre de registres (LSB)	XX _{hex}
Valeur CRC (LSB)	XX _{hex}
Valeur CRC (MSB)	XX _{hex}

5.12 Identification

Cette instruction permet la lecture de l'identification de l'esclave.



L'identifiant de l'esclave et la version de logiciel sont tous deux transmis au format ASCII.

Instruction (maître) :

Nom de champ	Valeur
Adresse esclave	XX _{hex}
Fonction Identification	11 _{hex}
Valeur CRC (LSB)	XX _{hex}
Valeur CRC (MSB)	XX _{hex}

Réponse (esclave) :

Nom de champ	Valeur
Adresse esclave	XX _{hex}
Fonction Identification	11 _{hex}
Nombre d'octets de données (MSB)	00 _{hex}
Nombre d'octets de données (LSB)	11 _{hex}
ID esclave octet 1	XX _{hex}
ID esclave octet 2	XX _{hex}
ID esclave octet 3	XX _{hex}
ID esclave octet 4	XX _{hex}
ID esclave octet 5	XX _{hex}
ID esclave octet 6	XX _{hex}
ID esclave octet 7	XX _{hex}
ID esclave octet 8	XX _{hex}
Etat	FF _{hex}
Version logiciel octet 1	XX _{hex}
Version logiciel octet 2	XX _{hex}
Version logiciel octet 3	XX _{hex}
Version logiciel octet 4	XX _{hex}
Version logiciel octet 5	XX _{hex}
Version logiciel octet 6	XX _{hex}
Version logiciel octet 7	XX _{hex}
Version logiciel octet 8	XX _{hex}
CRC (LSB)	XX _{hex}
CRC (MSB)	XX _{hex}

5.13 Protocole en cas d'erreur

Si l'esclave reçoit un message envoyé par le maître comportant des erreurs de transmission, il n'envoie pas de réponse.

Erreurs de transmission reconnues :

- Erreur de parité
- Pas de détection de bit de fin
- Erreur dans la trame des données
- Erreur Overrun (dépassement de capacité du tampon des données)
- Erreur CRC

Si l'esclave ne peut pas exécuter l'instruction d'un message non corrompu qu'il a reçu, il envoie une réponse d'erreur, avec un code d'erreur, au maître.

Codes d'erreur :

Code	Nom	Description
01 _{hex}	Fonction non autorisée	Le code de fonction n'est pas défini dans l'appareil.
02 _{hex}	Adresse non autorisée	L'adresse n'est pas disponible.
03 _{hex}	Valeur de donnée non autorisée	La valeur reçue ne peut pas être écrite. Structure des données / longueur des données incorrecte.
04 _{hex}	Défaut appareil	Valeur / format des données incorrect. L'appareil ne peut pas traiter l'instruction en totalité ou en partie.
10 _{hex}	Err1	Valeur de prépositionnement 0 interdite.
11 _{hex}	Err2	Valeur de prépositionnement supérieure à Preset2 interdite.

Réponse d'erreur:

Désignation	Valeur
Adresse esclave	XX _{hex}
Fonction	80 _{hex} + Code de fonction
Code d'erreur	XX _{hex}
Valeur CRC (LSB)	XX _{hex}
Valeur CRC (MSB)	XX _{hex}

5.14 Registres

Toutes les valeurs de données s'étendent sur 2 registres MODBUS et ne peuvent être lus et écrits que sous la forme d'une valeur globale.

Registres Float (point décimal flottant - 4 octets)

Adresse	Valeur	Accès	Description	Remarque
0000 _{hex}	Compteur principal	r/w	L'écriture d'une valeur quelconque repositionne le compteur principal.	
0002 _{hex}	Compteur secondaire	r/w	L'écriture d'une valeur repositionne le compteur principal et le compteur secondaire.	
0004 _{hex}	Présélection 1	r/w	Présélection 1	
0006 _{hex}	Présélection 2	r/w	Présélection 2	
0008 _{hex}	Facteur de multiplication	-/w	Facteur de multiplication	
000A _{hex}	Facteur de division	-/w	Facteur de division	
000C _{hex}	Mémorisation de la valeur de prépositionnement	-/w	Mémorisation de la valeur de prépositionnement (n'effectue pas le prépositionnement)	
000E _{hex}	Exécution de la fonction de prépositionnement	-/w	Prépositionnement (l'écriture d'une valeur quelconque exécute la fonction de prépositionnement)	
0010 _{hex}	Définition du signe de la présélection 1	-/w	Octet 1 : Valeur du signe Octet 2 : 0 Octet 3 : 0 Octet 4 : 0	Valeurs du signe: 1 : Valeurs + 2 : Valeurs -- 3 : Valeurs +/-
0012 _{hex}	Définition du point décimal	r/w	Octet 1 : décimales Octet 2 : 0 Octet 3 : 0 Octet 4 : 0	Décimales : 0 : 0 décimale 1 : 1 décimale 2 : 2 décimales 3 : 3 décimales 4 : 4 décimales 5 : 5 décimales
0014 _{hex}	Etat	r/o	Octet 1 : bit1 : sortie 1 bit2 : sortie 2 Octet 2 : bit1 - bit4 : compteur principal bit5 - bit8 : compteur secondaire Octet 3 : 0 Octet 4 : 0	Sortie : 1 : active 0 : inactive Etat du compteur: 0 : état : normal 1 : état : overflow 2 : état : underflow



r/o : Lecture uniquement
r/w : Lecture et écriture
-/w : Écriture

Registres Integer (entier - 4 octets)

Adresse	Valeur	Accès	Description	Remarque
8000 _{hex}	Compteur principal	r/w	L'écriture d'une valeur quelconque repositionne le compteur principal.	
8002 _{hex}	Compteur secondaire	r/w	L'écriture d'une valeur repositionne le compteur principal et le compteur secondaire.	
8004 _{hex}	Présélection 1	r/w	Présélection 1	
8006 _{hex}	Présélection 2	r/w	Présélection 2	
8008 _{hex}	Facteur de multiplication	-/w	Facteur de multiplication	
800A _{hex}	Facteur de division	-/w	Facteur de division	
800C _{hex}	Mémorisation de la valeur de prépositionnement	-/w	Mémorisation de la valeur de prépositionnement (n'effectue pas le prépositionnement)	
800E _{hex}	Exécution de la fonction de prépositionnement	-/w	Prépositionnement (l'écriture d'une valeur quelconque exécute la fonction de prépositionnement)	
8010 _{hex}	Définition du signe de la présélection 1	-/w	Octet 1 : Valeur du signe Octet 2 : 0 Octet 3 : 0 Octet 4 : 0	Valeurs du signe: 1 : Valeurs + 2 : Valeurs -- 3 : Valeurs +/-
8012 _{hex}	Définition du point décimal	r/w	Octet 1 : décimales Octet 2 : 0 Octet 3 : 0 Octet 4 : 0	décimales : 0 : 0 décimale 1 : 1 décimale 2 : 2 décimales 3 : 3 décimales 4 : 4 décimales 5 : 5 décimales
8014 _{hex}	Etat	r/o	Octet 1 : bit1 : sortie 1 bit2 : sortie 2 Octet 2 : bit1 - bit4 : compteur principal bit5 - bit8 : compteur secondaire Octet 3 : 0 Octet 4 : 0	Sortie : 1 : active 0 : inactive Etat du compteur: 0 : état : normal 1 : état : overflow 2 : état : underflow



r/o : Lecture uniquement
r/w : Lecture et écriture
-/w : Ecriture

Remarques d'ordre général :

- **Lecture/écriture** : les registres doivent être appelés selon la structure de registres décrite ci-dessus. Les registres ne peuvent donc être lus et écrits qu'en intégralité et en fonction des conditions d'accès.
- **Écriture** : les instructions d'écriture qui ne peuvent pas être exécutées du fait d'erreurs de format ou de dépassement de la plage autorisée, génèrent une réponse d'erreur avec le code d'erreur *Défaut appareil*.
- **Lecture/écriture** : si l'appareil se trouve dans le menu de programmation ou dans le menu de paramétrage, il n'exécute aucune instruction et ne renvoie aucune réponse d'erreur.
- **Lecture** : la demande d'état permet de savoir si un dépassement de la capacité par le haut ou par le bas a eu lieu sur le compteur principal et/ou sur le compteur secondaire, et quelles sorties sont activées.
Etat : 00001103_{hex} (sortie1 et sortie 2 activées ; compteur principal overflow ; compteur secondaire : overflow)
- L'entrée Loc n'affecte pas la fonctionnalité MODBUS.
- La valeur d'alarme 1 pour le signe n'est pertinente qu'en mode Trail.
- Le compteur principal et/ou le compteur secondaire ne peuvent être repositionnés que si le mode de repositionnement du compteur programmé est MAN.EL ou MAN.RES.

Remarques sur les registres Float (point décimal flottant) :

- Si la fonction Compteur horaire et le format de temps HH:MM:SS sont programmés, la partie entière de la valeur Float est utilisée dans la forme décimale pour la détermination du temps.
Exemples :
 - La valeur Float 48DBD8E0_{hex} (450247_{dec}) est interprétée comme 45:02:47 (HH:MM:SS)
 - La valeur Float 48DBDB60_{hex} (450267_{dec}) ne peut pas être représentée. L'indication de temps 45:02:67 (HH:MM:SS) n'est pas autorisée. Un défaut appareil est généré. L'écriture n'est pas effectuée.
 - Le registre pour la programmation des décimales ne peut pas être écrit dans le format de temps HH:MM:SS. La lecture de ce registre reste cependant possible.
- Dans le cas du réglage Counter avec les modes d'entrée A / B et A % B, il n'est pas possible de régler les facteurs de multiplication et de division.
- Pour le tachymètre, il n'est pas possible de repositionner la valeur du tachymètre de l'appareil. Il n'est pas possible de mémoriser ni d'exécuter une valeur de prépositionnement. Le signe de la présélection 1 ne peut pas être défini.

Remarques sur les registres Integer (entier) :

- Si la fonction Compteur horaire et le format de temps HH:MM:SS sont programmés, l'entier transmis est interprété comme une valeur de temps.
Exemple :
 - La valeur entière 00010078_{hex} (65656_{dec}) est interprétée comme 6:56:56 (HH:MM:SS).
 - Le registre pour la programmation des décimales ne peut pas être écrit dans le format de temps HH:MM:SS. La lecture de ce registre reste cependant possible.
- Dans le cas du réglage Counter avec les modes d'entrée A / B et A % B, il n'est pas possible de régler les facteurs de multiplication et de division.
- Pour le tachymètre, il n'est pas possible de repositionner la valeur du tachymètre de l'appareil. Il n'est pas possible de mémoriser ni d'exécuter une valeur de prépositionnement. Le signe de la présélection 1 ne peut pas être défini.

5.15 Exemples de transmissions de messages

5.15.1 Exemple de lecture

Adresse du registre (Float) : 0000_{hex} (compteur principal)

Adresse esclave définie : 01_{hex}

Valeur de donnée lue : 3F80 0000_{hex} (1_{dec})

Message émis par le maître :

Adresse esclave	Fonction	Adresse registre	Nombre de registres	CRC
01 _{hex}	03 _{hex}	0000 _{hex}	0002 _{hex}	C40B _{hex}

Réponse de l'esclave :

Adresse esclave	Fonction	Nombre d'octets	Valeur de donnée	CRC
01 _{hex}	03 _{hex}	04 _{hex}	3F80 0000 _{hex}	F7CF _{hex}

5.15.2 Exemple d'écriture

Adresse du registre (Integer) : 8014_{hex} (Etat)

Adresse esclave définie : 01_{hex}

Code d'erreur : 04_{hex} (défaut appareil)

Message émis par le maître :

Adresse esclave	Fonction	Adresse registre	Nbre de registres	Nbre d'octets	CRC
01 _{hex}	10 _{hex}	8014 _{hex}	0002 _{hex}	04 _{hex}	0DDD _{hex}

Réponse de l'esclave :

Adresse esclave	Fonction	Code d'erreur	CRC
01 _{hex}	90 _{hex}	04 _{hex}	4DC3 _{hex}

5.15.3 Exemple d'identification

Adresse esclave définie : 01_{hex}

ID esclave : 3536 302E 302E 3035_{hex} (560.0.A5)

Version de logiciel : 5645 2E30 322E 3031_{hex} (VE.02.01)

Message émis par le maître :

Adresse esclave	Fonction	Adresse registre	Nbre de registres	Nbre d'octets	CRC
01 _{hex}	10 _{hex}	8014 _{hex}	0002 _{hex}	04 _{hex}	0DDD _{hex}

Réponse de l'esclave :

Adresse esclave	Fonction	Nbre octets	ID esclave	Etat	Version logiciel	CRC
01 _{hex}	90 _{hex}	04 _{hex}	3536 302E 302E 3035 _{hex}	FF _{hex}	5645 2E30 322E 3031 _{hex}	4DC3 _{hex}

6 Jeux de paramètres

Les réglages suivants s'appliquent aux jeux de paramètres 1 à 3.

PR.TIME	1.0
SOURCE	MAIN
PROTOK.	MODBUS
ADRESS.	1

7 Caractéristiques techniques

7.1 Compteur d'impulsions

Temps de réponse des sorties à la fréquence de comptage maximale (voir **Fehler! Verweisquelle konnte nicht gefunden werden.**) :

Add;Sub;Trail	< 13 ms
en répétition automatique	< 13 ms
A / B; (A-B) / A	< 70 ms

7.2 Fréquence-mètre

Temps de réponse des sorties :

Mode monocanal :	< 100 ms @ 40 kHz
	< 160 ms @ 55 kHz
Mode bicanal :	< 150 ms @ 40 kHz
	< 250 ms @ 55 kHz

7.3 Compteur horaire

Temps de réponse des sorties :

Réglage secondes :	< 15 ms
Réglage minutes :	< 75 ms

Erreur de mesure	< 100 ppm
------------------	-----------

8 Fréquences (typiques)

NOTA : niveau de commutation des entrées

Niveau de commutation pour alimentation AC :

Niveau HTL	Bas : 0 ... 4 V DC
	Haut : 12 ... 30 V DC

Niveau 5V	Bas : 0 ... 2 V DC
	Haut : 3,5 ... 30 V DC

Niveau de commutation pour alimentation DC :

Niveau HTL	Bas : 0 ... $0,2 \times U_B$
	Haut : $0,6 \times U_B$... 30 V DC

Niveau 5V	Bas : 0 ... 2 V DC
	Haut : 3,5 ... 30 V DC

8.1 Compteur d'impulsions

Niveau HTL, forme de signal rectangulaire 1:1

Alimentation AC	typ. Bas	2,5 V
	typ. Haut	22 V
Alimentation DC 12V	typ. Bas	2 V
	typ. Haut	10 V
Alimentation DC 24V	typ. Bas	2,5 V
	typ. Haut	22 V

	Add Sub Trail	AddAr SubAr AddBat SubBat TrailAr	AddTot SubTot
Cnt.Dir	45 kHz	2,4 kHz	2,2 kHz
Up.Dn; Up.Up	25 kHz	2,2 kHz	2,1 kHz
Quad; Quad2	22 kHz	1,0 kHz	1,0 kHz
Quad4	17 kHz	0,8 kHz	0,8 kHz
A / B; (A - B) / A	27 kHz		

Niveau 5V, forme de signal rectangulaire 1:1

typ. Bas	1,0V
typ. Haut	4,0V

	Add Sub Trail	AddAr SubAr AddBat SubBat TrailAr	AddTot SubTot
Cnt.Dir	9 kHz	2,1 kHz	2,0 kHz
Up.Dn; Up.Up	9 kHz	2,0 kHz	2,0 kHz
Quad; Quad2	9 kHz	1,0 kHz	1,0 kHz
Quad4	9 kHz	0,8 kHz	0,8 kHz
A / B; (A - B) / A	9 kHz		

8.2 Fréquence-mètre

Niveau HTL, forme de signal rectangulaire 1:1

Alimentation AC		typ. Bas	2,5 V
		typ. Haut	22 V
Alimentation DC 12V		typ. Bas	2 V
		typ. Haut	10 V
Alimentation DC 24V		typ. Bas	2,5 V
		typ. Haut	22 V

Niveau 5V, forme de signal rectangulaire 1:1

typ. Bas	1,0 V
typ. Haut	4,0 V

	HTL	5V
A	55 kHz	9 kHz
A - B; A + B; A / B; (A - B) / A	55 kHz	9 kHz
Quad	27 kHz	9 kHz

9 Textes d'aide

INTERF.		MENU PRINCIPAL INTERFACES
PROTOD.	MODBUS	PROTOCOLE MODBUS
PROTOD.	CR.LF	PROTOCOLE CRLF
ADRESS.	1-247 / 1-99	ADRESSE D'INTERFACE MODBUS / ADRESSE D'INTERFACE CRLF
PR.TIME	0,5 – 9999,9	CYCLE D'EMISSION PROTOCOLE CRLF
SOURCE	MAIN	DONNEES CRLF COMPTEUR PRINCIPAL
SOURCE	BATCH	DONNEES CRLF COMPTEUR DE LOTS
SOURCE	TOTAL	DONNEES CRLF TOTALISATEUR
SOURCE.	MAI.BAT.	DONNEES CRLF COMPTEUR PRINCIPAL ET COMPTEUR DE LOTS
SOURCE.	MAI.TOT.	DONNEES CRLF COMPTEUR PRINCIPAL ET TOTALISATEUR
MP.INP.1	PRINT	FONCTION ENTREE MP 1 PRINT
MP.INP.2	PRINT	FONCTION ENTREE MP 2 PRINT

10 Références

Informations sur le protocole MODBUS :

[1] **Spécifications Modbus**

MODBUS Application Protocol Specification V1.1b

MODBUS over Serial Line – Specification and Implementation Guide V1.02.

www.modbus.org

Informations sur RS485 :

[2] **ANSI/TIA/EIA-485-A-98**

Contatore a preselezione a LED

96 x 48 mm, 2 preselezioni

Protocollo MODBUS Protocollo CR/LF

Opzione RS485



Indice

Contatore a preselezione a LED	1
1 Descrizione	3
2 Avvertenze importanti	3
3 Interfaccia	3
3.1 Platina opzionale RS485	3
3.2 Collegamento consigliato dall'Organizzazione Modbus [1]	3
3.3 Configurazione dell'interfaccia	4
3.4 Programmazione	4
4 Protocollo CR/LF	5
4.1 Trasmissione	5
4.1.1 Principio di trasmissione per fonte di dati contatore principale, contatore di partite o totalizzatore	5
4.1.2 Principio di trasmissione per fonte di dati contatore principale e contatore di partite/contatore principale e totalizzatore	6
5 Protocollo MODBUS	6
5.1 Principio Master – Slave	6
5.2 Principi di trasmissione	6
5.3 Timeout	7
5.4 Ciclo d'informazione	7
5.5 Ordine dei byte	7
5.6 Struttura di un messaggio	7
5.7 Codici di funzione	7
5.8 Valori dei dati	7
5.8.1 Formato Float (virgola mobile - 32 bit)	7
5.8.2 Formato Integer (integro - 32 bit)	8
5.9 CRC (cyclic redundancy check)	8
5.10 Lettura di più registri	9
5.11 Scrittura di più registri	9
5.12 Identificazione	10
5.13 Protocollo errore	10
5.14 Registri	11
5.15 Esempi per la trasmissione di messaggi	14
5.15.1 Esempio di lettura	14
5.15.2 Esempio di scrittura	14
5.15.3 Esempio d'identificazione	14
6 Gruppi di parametri	15
7 Dati tecnici	15
7.1 Contatore di impulsi	15
7.2 Frequenziometro	15
7.3 Contatore	15
8 Frequenze (tipiche)	15
8.1 Contatore di impulsi	15
8.2 Frequenziometro	16
9 Testi di aiuto	16
10 Referenze	16

1 Descrizione

Questo manuale descrive le opzioni di interfaccia RS485 e i rispettivi protocolli.

2 Avvertenze importanti



Leggere sia questo manuale d'uso riguardo alle opzioni di interfaccia che il manuale d'uso dell'apparecchio base, prima di utilizzare l'apparecchio.

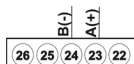
3 Interfaccia

Le linee di segnale sono collegate tramite una morsettiera con passo 3,5 mm a 5 morsetti.

3.1 Platina opzionale RS485

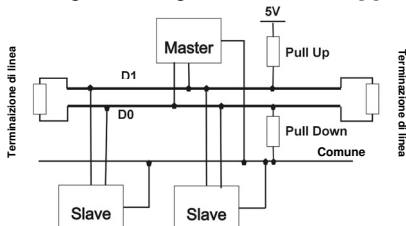
L'interfaccia RS485 – permette un collegamento a più punti. E' possibile allacciare fino a 32 apparecchi in un segmento di bus.

Collegamento



Tramite l'utilizzo di un cavo a capacità e attenuazione ridotta con doppiini intrecciati (twisted pair), si possono realizzare collegamenti molto affidabili per una lunga distanza (massimo 1000 m).

3.2 Collegamento consigliato dall'Organizzazione Modbus [1]



Definizione	
MODBUS	Apparecchio
D0	B (-)
D1	A (+)
Comune	

3.3 Configurazione dell'interfaccia

Configurazione Slave MODBUS (RTU)

Interfaccia	Vel. di trasm.	Formato dati	Indirizzo
RS485	9600	1 bit iniziale 8 bit di dati Parità 1 bit d'arresto	1-247

Configurazione Master CR/LF

Interfaccia	Vel. di trasm.	Formato dati	Indirizzo
RS485	9600	1 bit di inizio 8 bit di dati Senza parità 1bit d'arresto	1-99

3.4 Programmazione

INTERF.	Menu principale per la programmazione dell'Interfaccia.
PROTOD.	Protocolli dell'interfaccia
MODBUS	Protocollo MODBUS
CR.LF	Protocollo CR/LF
ADRESS.	Indirizzo Slave CR/LF: 1 – 99 MODBUS: 1- 247
1	
PR.TIME	Tempo di ciclo (solo per CR/LF) 0,5 – 9999,9 sec (accesso) 0 sec (spento)
0.5	
SOURCE	Fonte di dati (solo per CR/LF)
MAIN	Contatore principale
BATCH	Contatore di partite
TOTAL	Totalizzatore
MAI.BAT	Contatore principale e partite
MAI.TOT	Contatore principale e totalizzatore

Le fonti di dati sono regolabili a seconda delle operazioni in uscita impostate nell'apparecchio. Nel caso di un un Overflow / Underflow sul contatore da trasferire non viene trasmesso il valore dei dati, bensì +ooooo / +uuuuu.

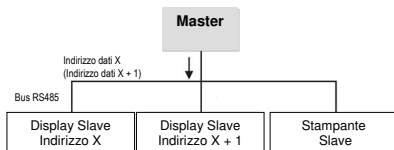
INPUT	Menu principale per la programmazione delle entrate di segnale e di comando
MPI.INP.1	Entrata utente 1
MPI.INP.2	Entrata utente 2
PRINT	Sgancio della trasmissione dei dati attraverso l'entrata utente. Possibile tempo ciclo ≥ 0 , 5 sec (possibile solo con CR/LF e tempo ciclo = 0)



Le regolazioni di fabbrica sono indicate su fondo grigio.

4 Protocollo CR/LF

Nel protocollo CR/LF l'apparecchio può, a seconda della fonte di dati impostata, trasmettere dei dati di valori a un'apparecchio Slave.



4.1 Trasmissione



I byte di dati sono codificati ASCII.

Indirizzo 1: indirizzo slave impostato nell'apparecchio

Indirizzo 2: indirizzo Slave aumentato di 1

Valore dei dati: a 6 cifre (+ punto decimale)

Testo: testo trasmesso in più con l'impostazione della fonte di dati d) o e)

4.1.1 Principio di trasmissione per fonte di dati contatore principale, contatore di partite o totalizzatore

Trasmissione: contatore principale o secondario

Indirizzo 1	Spazio bianco	Segno	Valore dei dati	Carriage Return	Line Feed
2 byte	1 byte	1 byte	6 o 7 byte	1 byte	1 byte

Esempi per fonte di dati contatore principale:

01	[SPACE]	-	123456	[CR]	[LF]
3031hex	20hex	2Dhex	3132 3334 3536hex	0Dhex	0Ahex

05	[SPACE]	+	000000	[CR]	[LF]
3035hex	20hex	2Bhex	6F6F 6F6F 6F6Fhex	0Dhex	0Ahex

Esempio per fonte di dati totalizzatore:

01	[SPACE]	+	000.456	[CR]	[LF]
3031hex	20hex	2Bhex	30 3030 2E34 3536hex	0Dhex	0Ahex



Grassetto: ASCII

Normale: valore HEX

4.1.2 Principio di trasmissione per fonte di dati contatore principale e contatore di partite/contatore principale e totalizzatore

Trasmissione: contatore principale e secondario

Indirizzo 1	Spazio	Testo	Spazio	Segno	Valore dati	Carriage Return	Line Feed
2 byte	1 byte	4 byte	1 byte	1 byte	6 o 7 byte	1 byte	1 byte

Indirizzo 2	Spazio	Testo	Spazio	Segno	Valore dati	Carriage Return	Line Feed
2 byte	1 byte	5 byte	1 byte	1 byte	6 o 7 byte	1 byte	1 byte

Esempio per fonte di dati contatore principale e contatore di partite:

15	[SPACE]	MAIN	[SPACE]	+	000259	[CR]	[LF]
3135hex	20hex	4D41 494Ehex	20hex	2Bhex	3030 3032 3539hex	0Dhex	0Ahex

16	[SPACE]	BATCH	[SPACE]	+	999999	[CR]	[LF]
3136hex	20hex	42 4154 4348hex	20hex	2Bhex	3939 3939 3939hex	0Dhex	0Ahex



Grassetto: ASCII

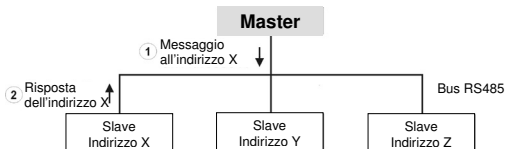
Normale: valore HEX

5 Protocollo MODBUS

La trasmissione dei dati MODBUS in forma binaria avviene nel modo operativo RTU (Remote Terminal Unit) attraverso l'interfaccia seriale (RS485). Il protocollo è implementato in base alle specifiche dell'organizzazione MODBUS (*MODBUS Application Protocol Specification V1.1b* ovvero *MODBUS over Serial Line – Specification and Implementation Guide V1.02*).

5.1 Principio Master – Slave

Il protocollo funziona secondo il principio Master-Slave. Un Master è in grado di comunicare con uno o più Slave. Soltanto lo Slave esplicitamente indirizzato dal Master può ritrasmettere dati al Master.



5.2 Principi di trasmissione

Il protocollo dispone di due principi di trasmissione.

Modo Unicast

Nel modo Unicast il Master indirizza uno Slave specifico. Lo Slave elabora l'informazione e rimanda una risposta al Master. Ogni Slave ha un indirizzo unico nel campo degli indirizzi da 1 a 247. Il principio di trasmissione è composto sempre da una richiesta (request) da parte del Master e il corrispettivo messaggio di risposta (response) inviato dallo Slave. Se il Master entro un tempo prestabilito (Timeout) non riceve risposta, può partire dal presupposto che ci siano stati errori nella trasmissione.

Modo Broadcast

Nel modo Broadcast il Master invia un comando di scrittura (request) a tutti gli apparecchi Slave, che però non generano alcuna risposta (response). L'indirizzo 0 è utilizzato per l'indirizzamento Broadcast.

5.3 Timeout

Se l'apparecchio Master non riceve risposta entro 0,5 secondi, può scartare l'ultima richiesta.

5.4 Ciclo d'informazione

È supportato un ciclo d'informazione ≤ 0,5 sec.

5.5 Ordine dei byte

Il protocollo Modbus utilizza il formato Big Endian.

5.6 Struttura di un messaggio

Indirizzo Slave	Codice di funzione	Dati	CRC
1 byte	1 byte	N bytes	2 byte

Pause all'interno della trasmissione di messaggi

Pause ≥ lunghezza di 1,5 segni tra i singoli byte di un messaggio porta a un errore nella trama dati.

Identificatore d'inizio e di fine

Come Identificatore di inizio e di fine di un messaggio viene specificata una pausa di messaggi della lunghezza di 3,5 segni. Questa pausa deve essere rispettata tra i singoli messaggi.

5.7 Codici di funzione

Sono supportati i seguenti codici di funzione.

Codice di funzione	Descrizione
03hex	Leggere più registri
10hex	Scrivere più registri
11hex	Identificazione degli Slave

5.8 Valori dei dati

I valori dei dati a 32-Bit sono trasmessi attraverso il MODBUS nel formato intero oppure nel formato float. L'indirizzo di registro determina il formato dei dati da utilizzare.

5.8.1 Formato Float (virgola mobile - 32 bit)

I valori Float sono codificati secondo lo standard IEEE 754 (single floating point) e sono impostati nel seguente modo.

Indirizzo X + 1		Indirizzo X	
SEEEEEEE	EMMMMMMM	MMMMMMMM	MMMMMMMM
byte 4	byte 3	byte 2	byte 1

$$X = S \times M^E$$



S: Segno
E: Esponente
M: Mantissa
X: Valore totale

Attraverso la conversione di dati nella rappresentazione a virgola mobile, si sviluppano errori di conversione (errori dovuti ad arrotondamento), la cui causa è dovuta al numero limitato di bit di mantissa.

Precisione macchina

La precisione macchina relativa indica quali errori relativi possono sorgere al massimo nell'arrotondare (convertire). Nel valore Float (single floating point) la precisione macchina è di $\epsilon_0 = 5,96 \cdot 10^{-8}$.

5.8.2 Formato Integer (intero - 32 bit)

Il formato intero viene utilizzato nella rappresentazione come complemento a due. L'apparecchio interpreta il valore intero trasmesso. Se il valore dati dell'apparecchio è rappresentato con i decimali impostati, allora la virgola viene posizionata, per il valore intero, a seconda del numero di decimali impostato.

Esempio con regolazione di 3 decimali:

Valore intero trascritto : 00000010_{hex}

Valore interpretato: 000,016_{dec}

Indirizzo X + 1		Indirizzo X	
00 _{hex}	00 _{hex}	00 _{hex}	10 _{hex}
byte 4	byte 3	byte 2	byte 1

5.9 CRC (cyclic redundancy check)

Per ogni blocco di dati è calcolato un valore CRC a 16 bit secondo un specifico procedimento, che viene aggiunto al blocco dei dati. Per la verifica dei dati è adoperato lo stesso procedimento di calcolo sul blocco dati compreso dell'aggiunto valore CRC. Se il risultato è uguale a zero, si può presupporre che il blocco dei dati non è adulterato.

Algoritmo per la generazione del CRC:

1. Caricamento del registro CRC a 16 bit con il valore FFFF_{hex}
2. Operazione OR Esclusivo del primo byte da inviare, ossia ricevuto, con il byte di valore basso del registro CRC; archiviazione del risultato nel registro CRC
3. Spostamento del registro CRC di un bit a destra e riempimento del msb con 0
4. Se nel punto 3 è stato perso uno 0, viene ripetuto il punto 3. Se nel punto 3 è stato perso un 1, il registro CRC viene associato con il valore A001_{hex} tramite una operazione OR Esclusivo.
5. I punti 3 e 4 vengono ripetuti finché sono stati avvenuti 8 spostamenti di bit.
6. Per i successivi bit da inviare o ricevuti, vengono ripetuti i punti 2 da 5.
7. Il risultato della generazione CRC adesso si trova nel registro CRC e dopo uno scambio di byte viene aggiunto all'informazione inviata.



msb: most significant bit (bit più significativo)

5.10 Lettura di più registri

Con questo comando si possono leggere registri a 16 bit.



XX_{hex} valore valido nel formato Hex

LSB: least significant byte
(byte meno significativo)

MSB: most significant byte
(byte più significativo)

Comando (Master):

Nome del campo	Valore
Indirizzo Slave	XX _{hex}
Funzione lettura	03 _{hex}
Inizio sull'indirizzo di registro (MSB)	XX _{hex}
Inizio sull'indirizzo di registro (LSB)	XX _{hex}
Numero di registri (MSB)	XX _{hex}
Numero di registri (LSB)	XX _{hex}
Valore CRC (LSB)	XX _{hex}
Valore CRC (MSB)	XX _{hex}

Risposta (Slave):

Nome del campo	Valore
Indirizzo Slave	XX _{hex}
Funzione lettura	03 _{hex}
Numero di byte di dati	XX _{hex}
Valore registro 1 (ms.)	XX _{hex}
Valore registro 1 (LSB)	XX _{hex}
...	...
Valore registro N (MSB)	XX _{hex}
Valore registro N (LSB)	XX _{hex}
Valore CRC (LSB)	XX _{hex}
Valore CRC (MSB)	XX _{hex}

5.11 Scrittura di più registri

Con questo comando si possono scrivere registri a 16 bit:

Comando (Master):

Nome del campo	Valore
Indirizzo Slave	XX _{hex}
Funzione scrittura	10 _{hex}
Inizio sull'indirizzo di registro (MSB)	XX _{hex}
Inizio sull'indirizzo di registro (LSB)	XX _{hex}
Numero di registri (MSB)	XX _{hex}
Numero di registri (LSB)	XX _{hex}
Numero di byte di dati (2 * Numero di registri)	XX _{hex}
Valore registro 1 (MSB)	XX _{hex}
Valore registro 1 (LSB)	XX _{hex}
...	...
Valore registro N (MSB)	XX _{hex}
Valore registro N (LSB)	XX _{hex}
Valore CRC (LSB)	XX _{hex}
Valore CRC (MSB)	XX _{hex}

Risposta (Slave):

Nome del campo	Valore
Indirizzo Slave	XX _{hex}
Funzione scrittura	10 _{hex}
Inizio sull'indirizzo di registro (MSB)	XX _{hex}
Inizio sull'indirizzo di registro (LSB)	XX _{hex}
Numero di registri (MSB)	XX _{hex}
Numero di registri (LSB)	XX _{hex}
Valore CRC (LSB)	XX _{hex}
Valore CRC (MSB)	XX _{hex}

5.12 Identificazione

Con questo comando si può leggere l'identificazione dello slave.



Sia l'ID Slave sia la versione-Software sono trasmessi in formato ASCII.

Comando (Master):

Nome del campo	Valore
Indirizzo Slave	XX _{hex}
Funzione Identificazione	11 _{hex}
Valore CRC (LSB)	XX _{hex}
Valore CRC (MSB)	XX _{hex}

Risposta (Slave):

Nome del campo	Valore
Indirizzo Slave	XX _{hex}
Funzione Identificazione	11 _{hex}
Numero di byte di dati (MSB)	00 _{hex}
Numero di byte di dati (LSB)	11 _{hex}
ID Slave Byte 1	XX _{hex}
ID Slave Byte 2	XX _{hex}
ID Slave Byte 3	XX _{hex}
ID Slave Byte 4	XX _{hex}
ID Slave Byte 5	XX _{hex}
ID Slave Byte 6	XX _{hex}
ID Slave Byte 7	XX _{hex}
ID Slave Byte 8	XX _{hex}
Stato	FF _{hex}
Versione Software Byte 1	XX _{hex}
Versione Software Byte 2	XX _{hex}
Versione Software Byte 3	XX _{hex}
Versione Software Byte 4	XX _{hex}
Versione Software Byte 5	XX _{hex}
Versione Software Byte 6	XX _{hex}
Versione Software Byte 7	XX _{hex}
Versione Software Byte 8	XX _{hex}
CRC (LSB)	XX _{hex}
CRC (MSB)	XX _{hex}

5.13 Protocollo errore

Lo slave non invia alcuna risposta, se ha ricevuto l'informazione del Master con errori di trasmissione.

Gli errori di trasmissione riconosciuti sono:

- Errore di parità
- Nessun riconoscimento di bit d'arresto
- Errore nella trama dati
- Errore Overrun (overflow della memoria tampone dei dati)
- Errore CRC

Se l'apparecchio Slave non può eseguire l'informazione ricevuta correttamente, allora rinvia una risposta di errore compreso codice errore al Master.

Codici di errore:

Codice	Nome	Significato
01 _{hex}	Funzione non consentita	Il codice di funzione non è definito nell'apparecchio.
02 _{hex}	Indirizzo non consentito	L'indirizzo non è disponibile
03 _{hex}	Valore dati non consentito	Il valore dati ricevuto non può essere scritto. Struttura dati/lunghezza dati non corretta.
04 _{hex}	Errore apparecchio	Valore/formato dati non corretto. L'apparecchio non può/può solo in parte elaborare il comando.
10 _{hex}	Err1	Valore di preimpostazione inferiore a 0 non consentito
11 _{hex}	Err2	Valore di preimpostazione superiore a Preset2 non consentito

Risposta di errore:

Definizione	Valore
Indirizzo Slave	XX _{hex}
Funzione	80 _{hex} + Codice di funzione
Codice di errore	XX _{hex}
Valore CRC (LSB)	XX _{hex}
Valore CRC (MSB)	XX _{hex}

5.14 Registri

Tutti i valori di dati si estendono su 2 registri MODBUS e possono esser considerati soltanto come un valore globale.

Registri Float (4 byte)

Indirizzo	Valore	Accesso	Descrizione	Osservazioni
0000 _{hex}	Contatore principale	r/w	La scrittura di un qualsiasi valore ripristina il contatore principale	
0002 _{hex}	Contatore secondario	r/w	La scrittura di un valore ripristina i contatori principale e secondario.	
0004 _{hex}	Preselezione 1	r/w	Preselezione 1	
0006 _{hex}	Preselezione 2	r/w	Preselezione 2	
0008 _{hex}	Fattore di moltiplicazione	-/w	Fattore di moltiplicazione	
000A _{hex}	Fattore di divisione	-/w	Fattore di divisione	
000C _{hex}	Memorizzazione valore di preimpostazione	-/w	Memorizzazione del valore di preimpostazione (non effettua la preimpostazione)	
000E _{hex}	Esecuzione funzione di preimpostazione	-/w	Preimpostazione (la scrittura di un qualsiasi valore effettua la funzione di preimpostazione)	
0010 _{hex}	Determinazione segno della preselezione 1	-/w	Byte 1: valore del segno Byte 2: 0 Byte 3: 0 Byte 4: 0	Valori del segno: 1: valori + 2: valori -- 3: valori+/-
0012 _{hex}	Determinazione punto decimale	r/w	Byte 1: Numero di decimali Byte 2: 0 Byte 3: 0 Byte 4: 0	Decimali: 0: 0 decimali 1: 1 decimale 2: 2 decimali 3: 3 decimali 4: 4 decimali 5: 5 decimali
0014 _{hex}	Stato	r/o	Byte 1: bit1: uscita 1 bit2: uscita 2 Byte 2: bit1 - bit4: contatore principale bit5 - bit8: contatore secondario Byte 3: 0 Byte 4: 0	Uscita: 1: attivata 0: non attivata Lettura del contatore: 0: lettura contatore : regolare 1: lettura contatore: overflow 2: lettura contatore: underflow



r/o: accesso solo lettura
r/w: accesso lettura e scrittura
-/w: accesso scrittura

Registri Integer (intero - 4 byte)

Indirizzo	Valore	Accesso	Descrizione	Osservazioni
8000 _{hex}	Contatore principale	r/w	La scrittura di un qualsiasi valore ripristina il contatore principale	
8002 _{hex}	Contatore secondario	r/w	La scrittura di un valore ripristina i contatori principale e secondario.	
8004 _{hex}	Preselezione 1	r/w	Preselezione 1	
8006 _{hex}	Preselezione 2	r/w	Preselezione 2	
8008 _{hex}	Fattore di moltiplicazione	-/w	Fattore di moltiplicazione	
800A _{hex}	Fattore di divisione	-/w	Fattore di divisione	
800C _{hex}	Memorizzazione valore di preimpostazione	-/w	Memorizzazione del valore di preimpostazione (non effettua la preimpostazione)	
800E _{hex}	Esecuzione funzione di preimpostazione	-/w	Preimpostazione (la scrittura di un qualsiasi valore effettua la funzione di preimpostazione)	
8010 _{hex}	Determinazione segno della preselezione 1	-/w	Byte 1: valore del segno Byte 2: 0 Byte 3: 0 Byte 4: 0	Valori del segno: 1: valori + 2: valori -- 3: valori+/-
8012 _{hex}	Determinazione punto decimale	r/w	Byte 1: Numero di decimali Byte 2: 0 Byte 3: 0 Byte 4: 0	Decimali: 0: 0 decimali 1: 1 decimale 2: 2 decimali 3: 3 decimali 4: 4 decimali 5: 5 decimali
8014 _{hex}	Stato	r/o	Byte 1: bit1: uscita 1 bit2: uscita 2 Byte 2: bit1 - bit4: contatore principale bit5 - bit8: contatore secondario Byte 3: 0 Byte 4: 0	Uscita: 1: attivata 0: non attivata Lettura del contatore: 0: lettura contatore : regolare 1: lettura contatore: overflow 2: lettura contatore: underflow



r/o: accesso solo lettura
 r/w: accesso lettura e scrittura
 -/w: accesso scrittura

Indicazioni generali:

- **Lettura/scrittura:** i registri devono essere richiesti secondo la struttura di registro sopra descritta. I registri quindi possono essere letti e scritti soltanto in maniera completa e conforme all'accesso consentito.
- **Scrittura:** i comandi di scrittura, che non possono essere eseguiti a causa di errori di formato o di superamenti di campo, generano una risposta di errore con il codice errore *Errore Apparecchio*.
- **Lettura / scrittura:** Se l'apparecchio è nel menu di programmazione o nel menu di parametrizzazione, non viene eseguito nessun comando e non viene fornita nessuna risposta di errore.
- **Lettura:** L'interrogazione di stato da informazioni sul fatto se sono avvenuti overflow o underflow nel contatore principale e/o secondario, e quali uscite sono attivate.
Stato: 00001103_{hex} (Uscita1 e uscita2 attivate; contatore principale: overflow; contatore secondario: overflow)
- L'entrata Loc non influisce sulla funzionalità del MODBUS.
- Il segno del valore d'allarme 1 è rilevante soltanto nel modo Trail
- Si può ripristinare il contatore principale e/o secondario soltanto se il modo di ripristino nel contatore è stato programmato su MAN.EL o MAN.RES.

Note sui registri Float:

- Se la funzione Timer è programmata con il formato di tempo HH:MM:SS, la parte intera del valore float viene utilizzata in modo di scrittura decimale per la determinazione del tempo.
Esempio:
 - Valore Float 48DBD8E0_{hex} (450247_{dec}) viene interpretato come 45:02:47 (HH:MM:SS)
 - Valore Float 48DBDB60_{hex} (450267_{dec}) non può essere rappresentato. L'indicazione di tempo 45:02:67 (HH:MM:SS) non è consentita. Viene indicato un errore apparecchio. La scrittura viene meno.
 - Il registro per la programmazione delle decimali non può essere scritto nel formato di tempo HH:MM:SS. La lettura del registro è però possibile.
- Con l'impostazione Counter e il modo d'entrata A / B e A % B, il fattore di moltiplicazione e il fattore di divisione non possono essere impostati.
- In modo tachimetro, il tachimetro dell'apparecchio non può essere ripristinato. Non può essere memorizzato ed eseguito un valore di preimpostazione. Il segno per la preselezione 1 non può essere descritto.

Note sui registri Integer:

- Se la funzione Timer è programmata con il formato di tempo HH:MM:SS, il valore intero trasmesso viene interpretato come valore di tempo.
Per esempio:
 - Valore intero 00010078_{hex} (65656_{dec}) viene interpretato come 6:56:56 (HH:MM:SS).
 - Il registro per la programmazione dei decimali non può essere descritto nel formato di tempo HH:MM:SS. E' possibile però la lettura del registro.
- Con l'impostazione Counter e il modo d'entrata A / B e A % B, il fattore di moltiplicazione e il fattore di divisione non possono essere impostati.
- In modo tachimetro, il tachimetro dell'apparecchio non può essere ripristinato. Non può essere memorizzato ed eseguito un valore di preimpostazione. Il segno per la preselezione 1 non può essere descritto.

5.15 Esempi per la trasmissione di messaggi

5.15.1 Esempio di lettura

Indirizzo registro (Float): 0000_{hex} (contatore principale)
Indirizzo Slave impostato: 01_{hex}
Valore di dati letto : 3F80 0000_{hex} (1_{dec})

Messaggio Master:

Indirizzo Slave	Funzione	Indirizzo Registro	Numero dei registri	CRC
01 _{hex}	03 _{hex}	0000 _{hex}	0002 _{hex}	C40B _{hex}

Risposta Slave:

Indirizzo Slave	Funzione	Numero di byte	Valore dati	CRC
01 _{hex}	03 _{hex}	04 _{hex}	3F80 0000 _{hex}	F7CF _{hex}

5.15.2 Esempio di scrittura

Indirizzo registro (Integro): 8014_{hex} (Stato)
Indirizzo Slave impostato : 01_{hex}
Codice di errore: 04_{hex} (errore apparecchio)

Messaggio Master:

Indirizzo Slave	Funzione	Indirizzo registro	Numero di registri	Numero di byte	CRC
01 _{hex}	10 _{hex}	8014 _{hex}	0002 _{hex}	04 _{hex}	0DDD _{hex}

Risposta Slave:

Indirizzo Slave	Funzione	Codice errore	CRC
01 _{hex}	90 _{hex}	04 _{hex}	4DC3 _{hex}

5.15.3 Esempio d'identificazione

Indirizzo Slave impostato: 01_{hex}
ID Slave: 3536 302E 302E 3035_{hex} (560.0.A5)
Versione Software: 5645 2E30 322E 3031_{hex} (VE.02.01)

Messaggio Master:

Indirizzo Slave	Funzione	Indirizzo registro	Numero di registri	Numero di byte	CRC
01 _{hex}	10 _{hex}	8014 _{hex}	0002 _{hex}	04 _{hex}	0DDD _{hex}

Risposta Slave:

Indirizzo Slave	Funzione	Numero di byte	ID Slave	Stato	Versione Software	CRC
01 _{hex}	90 _{hex}	04 _{hex}	3536 302E 302E 3035 _{hex}	FF _{hex}	5645 2E30 322E 3031 _{hex}	4DC3 _{hex}

6 Gruppi di parametri

Le seguenti impostazioni valgono per i gruppi di parametri 1 - 3.

PR.TIME	1.0
SOURCE	MAIN
PROT.C.	MODBUS
ADRESS.	1

7 Dati tecnici

7.1 Contatore di impulsi

Tempo di risposta delle uscite con frequenza di contatore massima (vedi 8.1):

Add;Sub;Trail	< 13 ms
Per ripetizione automatica	< 13 ms
A / B; (A-B) / A	< 70 ms

7.2 Frequenziometro

Tempo di risposta delle uscite:

Modo monocanale:	< 100 ms @ 40 kHz
	< 160 ms @ 55 kHz
Modo bicanale:	< 150 ms @ 40 kHz
	< 250 ms @ 55 kHz

7.3 Contatore

Tempo di risposta delle uscite:

Impostazione - Secondi:	< 15 ms
Impostazione - Minuti:	< 75 ms

Errore di misura:	< 100 ppm
-------------------	-----------

8 Frequenze (tipiche)

NOTA: livello di commutazione delle entrate

Livello di commutazione con alimentazione AC:

Livello HTL	basso: 0 ... 4 V DC
	alto: 12 ... 30 V DC

Livello 5V	basso: 0 ... 2 V DC
	alto: 3,5 ... 30 V DC

Livello di commutazione con alimentazione DC:

Livello HTL	basso: 0 ... $0,2 \times U_B$
	alto: $0,6 \times U_B$... 30 VDC

Livello 5V	basso: 0 ... 2 V DC
	alto: 3,5 ... 30 V DC

8.1 Contatore di impulsi

Livello HTL, forma di segnale rettangolare 1:1

Alimentazione AC	tip. basso	2,5 V
	tip. alto	22 V
Alimentazione DC 12V	tip. basso	2 V
	tip. alto	10 V
Alimentazione DC 24V	tip. basso	2,5 V
	tip. alto	22 V

	Add Sub Trail	AddAr SubAr AddBat SubBat TrailAr	AddTot SubTot
Cnt.Dir	45 kHz	2,4 kHz	2,2 kHz
Up.Dn; Up.Up	25 kHz	2,2 kHz	2,1 kHz
Quad; Quad2	22 kHz	1,0 kHz	1,0 kHz
Quad4	17 kHz	0,8 kHz	0,8 kHz
A / B; (A - B) / A	27 kHz		

Livello 5V, forma di segnale rettangolare 1:1

tip. basso	1,0 V
tip. alto	4,0 V

	Add Sub Trail	AddAr SubAr AddBat SubBat TrailAr	AddTot SubTot
Cnt.Dir	9 kHz	2,1 kHz	2,0 kHz
Up.Dn; Up.Up	9 kHz	2,0 kHz	2,0 kHz
Quad; Quad2	9 kHz	1,0 kHz	1,0 kHz
Quad4	9 kHz	0,8 kHz	0,8 kHz
A / B; (A - B) / A	9 kHz		

8.2 Frequenziometro

Livello HTL, forma di segnale rettangolare 1:1

Alimentazione AC	tip. basso	2,5 V
	tip. alto	22 V
Alimentazione DC 12V	tip. basso	2 V
	tip. alto	10 V
Alimentazione DC 24V	tip. basso	2,5 V
	tip. alto	22 V

Livello 5V, forma di segnale rettangolare 1:1

tip.basso	1,0 V
tip.alto	4,0 V

	HTL	5V
A	55 kHz	9 kHz
A - B; A + B; A / B; (A - B) / A	55 kHz	9 kHz
Quad	27 kHz	9 kHz

9 Testi di aiuto

INTERF.		MENU PRINCIPALE INTERFACCIE
PROTOD.	MODBUS	PROTOCOLLO MODBUS
PROTOD.	CR.LF	PROTOCOLLO CRLF
ADRESS.	1-247 / 1-99	INDIRIZZO INTERFACCIA MODBUS / INDIRIZZO INTERFACCIA CRLF
PR.TIME	0,5 – 9999,9	PROTOCOLLO CICLO D'INVIO CRLF
SOURCE	MAIN	DATI CRLF CONTATORE PRINCIPALE
SOURCE	BATCH	DATI CRLF CONTATORE DI PARTITE
SOURCE	TOTAL	DATI CRLF TOTALIZZATORE
SOURCE.	MAI.BAT.	DATI CRLF CONTATORE PRINCIPALE E DI PARTITE
SOURCE.	MAI.TOT.	DATI CRLF CONTATORE PRINCIPALE E TOTALIZZATORE
MP.INP.1	PRINT	FUNZIONE ENTRATA MP1 PRINT
MP.INP.2	PRINT	FUNZIONE ENTRATA MP2 PRINT

10 Referenze

Informazioni sul protocollo MODBUS:

[1] **Specifica Modbus**

MODBUS Application Protocol Specification V1.1b

MODBUS over Serial Line – Specification and Implementation Guide V1.02.

www.modbus.org

Informazioni su RS485:

[2] **ANSI/TIA/EIA-485-A-98**

Trumeter (Europe)

Trumeter, Pilot Mill, Alfred Street, Bury, Lancashire
BL9 9EF, United Kingdom
Tel: +44 161 674 0960
Email: sales.uk@trumeter.com

Trumeter (The Americas)

Trumeter, 6601 Lyons Rd, SUITE H-7, Coconut Creek, Florida
FL 33073, USA
Tel: +1 954 725 6699
Email: sales.usa@trumeter.com

Asia Pacific Distributor

Innovative Design Technologies Sdn. Bhd, Lot 5881, Lorong Iks Bukit Minyak 1
Taman Perindustrian Iks, 14000 Bukit Tengah
Penang, Malaysia
Web: www.idtworld.com
Tel: + 604 5015700
Email: info@idtworld.com