ZIEHL industrie – elektronik GmbH + Co KG
Daimlerstraße 13, D – 74523 Schwäbisch Hall
+ 49 791 504-0, info@ziehl.de, www.ziehl.de

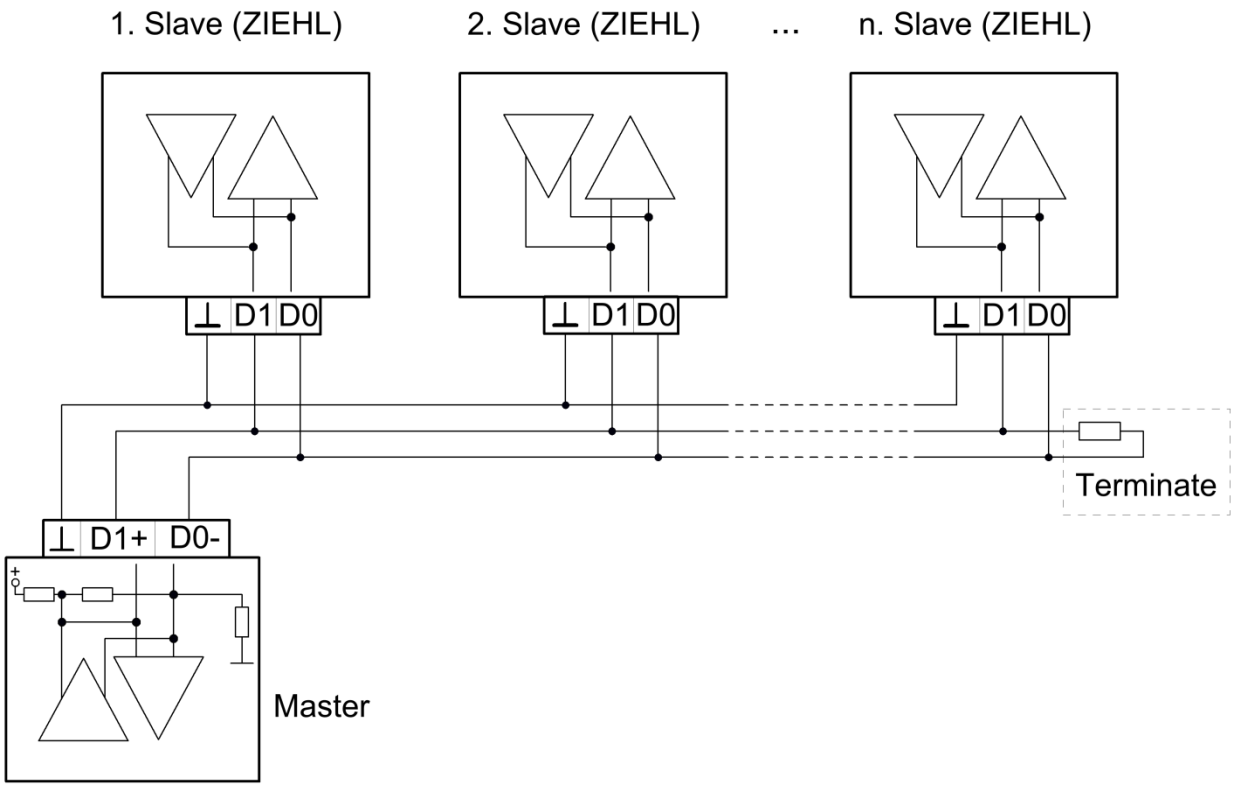
Operating Manual TMU104V

Stand: 2021-11-18 / tw

- RS485 interface with Modbus communication protocol

Table of contents

1 Connection diagram



Pin designation	Modbus	EIA/TIA-485
- wire	D0	A
+ wire	D1	B

2 Important information

Please read the general operating manual for the TMU104V carefully and observe the safety instructions.

3 Interface parameter

Protocol	Baud rate	Data bits	Parity	Stop bit
Modbus, RS485	4800, 9600, 19200	8	even, odd, none	1 (with Parity none: 2) ➢ since firmware -02: adjustable

The interface parameter are factory adjusted to Modbus, 9600 Baud, 8 Bits, even Parity, 1 Stop bit.
Only the Modbus-Protocol is documented in this manual.

The RTU-Mode is being used.
The TMU104V acts as a Slave in the BUS-System with an address that can be set from 1 to 247.
The parameters setting is described in the TMU104V's manual.

4 Telegram structure

Slave-address (1 .. 247)	Function	Data	CRC-16 checksum
1 Byte	1 Byte	n- Bytes	2 Byte

5 Supported function codes

Function code	Label	usage
3 (03H)	Read Holding Registers	Daten aus den Registern lesen
16 (10H)	Write Multiple Registers	Daten in die Register schreiben

5.1 Function code 3 (03H) – read register values

Request (Master)

Byte no.	Meaning		1. Example	2. Example
1	Slave-address		0x01	0x0A
2	Function		0x03	0x03
3	Start-address	Hi-Byte	0x00	0x00
4		Lo-Byte	0x01	0x11
5	Amount-Words (Bytes / 2)	Hi-Byte	0x00	0x00
6		Lo-Byte	0x04	0x02
7	Checksum CRC-16	Lo-Byte	0x15	0x95
8		Hi-Byte	0xC9	0x75

Response (Slave / TMU104V)

Byte no.	Meaning		1. Example	2. Example
1	Slave- address		0x01	0x0A
2	Function		0x03	0x03
3	Amount Bytes (n) (Words x 2)		0x08	0x04
4	1. Word (2 Bytes)	Hi-Byte	0x00	0x02
5		Lo-Byte	0x32	0x5A
6	2. Word	Hi-Byte	0x00	0xFF

7	(2 Bytes)	Lo-Byte	0x3C	0xFB
8	3. Word	Hi-Byte	0x00	
9	(2 Bytes)	Lo-Byte	0x46	
10	n- Words	Hi-Byte	0x00	
11	(2 Bytes)	Lo-Byte	0x50	
:	:			
3 + (n + 1)	Checksum	Lo-Byte	0x37	0x61
3 + (n + 2)	CRC-16	Hi-Byte	0xF8	0x2B

5.2 Function code 16 (10H) – write register values

Request (Master)

Byte no.	Meaning		1. Example	2. Example
1	Slave-address		0x01	0x0A
2	Function		0x10	0x10
3	Start-	Hi-Byte	0x00	0x00
4	address	Lo-Byte	0x07	0x10
5	Amount-Words	Hi-Byte	0x00	0x00
6	(Bytes / 2)	Lo-Byte	0x04	0x02
7	Amount Bytes (n)		0x08	0x04
8	1. Register	Hi-Byte	0x00	0x00
9		Lo-Byte	0x5A	0x00
10	2. Register	Hi-Byte	0xFF	0x00
11		Lo-Byte	0xFB	0x64
12	3. Register	Hi-Byte	0x00	
13		Lo-Byte	0x0A	
14	4. Register	Hi-Byte	0x00	
15		Lo-Byte	0x14	
:	:			
7 + (n + 1)	Checksum	Lo-Byte	0x68	0xD6
7 + (n + 2)	CRC-16	Hi-Byte	0x62	0x6C

Response (Slave / TMU104V)

Byte no.	Meaning		1. Example	2. Example
1	Slave-address		0x01	0x0A
2	Function		0x10	0x10
3	Start-	Hi-Byte	0x00	0x00
4	address	Lo-Byte	0x07	0x10
5	Amount Words (n)	Hi-Byte	0x00	0x02
6	(Bytes / 2)	Lo-Byte	0x04	0x02
7	Checksum	Lo-Byte	0x70	0x40
8	CRC-16	Hi-Byte	0x0B	0x16

5.3 Modbus register tables

Register of function code 3 (03H) – read register values			
Adr.	Data type	Description	Value range
0000	Signed Int	Sensor input, sensortyp	-1 = Pt100 -2 = Pt1000 -3 = KTY83 -4 = KTY84 -5 = Thermocouple Typ B -6 = Thermocouple Typ E -7 = Thermocouple Typ J -8 = Thermocouple Typ K -9 = Thermocouple Typ L -10 = Thermocouple Typ N -11 = Thermocouple Typ R -12 = Thermocouple Typ S -13 = Thermocouple Typ T
0001	Signed Int	Line resistance oft he sensor input	-1 = 3-Leiter 0…999 = 2- Line resistance -> 0,0 … 99,9 Ω
0002	Signed Int	Sensor output, typ	-1 = Pt100
0003	Signed Int	Min.- measured value	-1999…8500 = -199,9…850,0 °C
0004	Signed Int	Max.- measured value	-1999…8500 = -199,9…850,0 °C
0005	Signed Int	simulated temperature value at OUT 1	-1999…8500 = -199,9…850,0 °C
0006	Signed Int	simulated temperature value at OUT 2	-1999…8500 = -199,9…850,0 °C
0007	Signed Int	simulated temperature value at OUT 3	-1999…8500 = -199,9…850,0 °C
0008	Signed Int	simulated temperature value at OUT 4	-1999…8500 = -199,9…850,0 °C
0009	Signed Int	measured value sensor input	-1999…8500 = -199,9…850,0 °C 32767 = sensor short circuit 32766 = sensor disconnection
000A	Signed Int	Firmware-Version	00-00, (Lo-Byte, Hi-Byte)

Register of function code 16 (10H) – write register values			
Adr.	Data type	Description	Value range
0000	Signed Int	Sensor input, sensortyp	-1 = Pt100 -2 = Pt1000 -3 = KTY83 -4 = KTY84 -5 = Thermocouple Typ B -6 = Thermocouple Typ E -7 = Thermocouple Typ J -8 = Thermocouple Typ K -9 = Thermocouple Typ L -10 = Thermocouple Typ N -11 = Thermocouple Typ R -12 = Thermocouple Typ S -13 = Thermocouple Typ T
0001	Signed Int	Line resistance oft he sensor input	-1 = 3-Conductor 0…999 = 2- Line resistance -> 0,0 … 99,9 Ω
0002	Signed Int	Sensor output, typ	-1 = Pt100
0003	Signed Int	Min.- measured value	1 = reset Min.- measured value
0004	Signed Int	Max.- measured value	1 = reset Max.- measured value
0005	Signed Int	Simulate temperature value at OUT 1	-1999…8500 = -199,9…850,0 °C
0006	Signed Int	Simulate temperature value at OUT 2	-1999…8500 = -199,9…850,0 °C
0007	Signed Int	Simulate temperature value at OUT 3	-1999…8500 = -199,9…850,0 °C
0008	Signed Int	Simulate temperature value at OUT 4	-1999…8500 = -199,9…850,0 °C

0009	Signed Int	Sensor-simulation	bit0 = OUT 1, bit1 = OUT 2, bit2 = OUT 3, bit3 = OUT 4 Example: 0x0003 = Simulation of OUT 1 + OUT2 0x0000 = Simulation off

6 Fault reports

The telegram sended by the Master is checked by the Slave (TMU104V).
In the case of an error a fault report will be generated and sent back to the Master. The 7th Bit of the function byte will be set to 1.

Fehlertelegram:

Byte No.	Meaning		1. Example	2. Example
1	Slave-address		0x01	0x0A
2	Function		0x81	0x90
3	Error code		0x02	0x03
4	Checksum	Lo-Byte	0xC1	0x7D
5	CRC-16	Hi-Byte	0x91	0xC3

The following error codes are possible:

- 1 (01H) Invalid function
- 2 (02H) Invalid start address
- 3 (03H) Invalid data value
- 4 (04H) Slave-device error

Errors which will be not recognized by the Slave (telegram will be discarded):

- Wrong Checksum CRC-16
- Unknown Slave- address

7 Checksum CRC-16

The checksum will be attached to every Modbus-telegram and is used for the detection of transmission errors. The length of the checksum is 2 Bytes and is calculated using all bytes that were used in the telegrams transmission. The Lo-Byte is transmitted prior to the Hi-Byte.

Further details can be found in the orginial modbus document at http://www.modbus.org