Modbus TCP – Interface description
Wallbox eM4 Controller / Extender

Contents

1. Configuration

- Communication is via LAN/WLAN
- Wallbox eM4 is server/slave
- Server port: 502 (standard Modbus TCP)

2. Telegram structure

2.1. Read data from server

2.1.1. Client enquiry

Byte	Contents	Description
0	transaction identifier 1	Identification no. for several simultaneously active requests
1	transaction identifier 2	
2	protocol identifier 1	0x00 (MODBUS)
3	protocol identifier 2	0x00 (MODBUS)
4	Number of subsequent telegram bytes (high byte)	0x00 (always 0x00, as max. 255 data bytes)
5	Number of subsequent telegram bytes (low byte)	0x06 (byte 6...11)
6	unit identifier	0xFF (identification via IP address)
7	Function Code	0x03 (read)
8	Start register address (high byte)	
9	Start register address (low byte)	
10	Number of registers to be read (high byte)	0x00
11	Number of registers to be read (low byte)	0x01 ... 0x7E (server can supply max. 126 registers)

2.1.2. Server response

Byte	Contents	Description
0	transaction identifier 1	Identification no. of the associated request
1	transaction identifier 2	
2	protocol identifier 1	0x00 (MODBUS)
3	protocol identifier 2	0x00 (MODBUS)
4	Number of subsequent telegram bytes (high byte)	0x00 (always 0x00, as max. 255 data bytes)
5	Number of subsequent telegram bytes (low byte)	0x05 ... 0xFF (byte 6...n; min. 5 bytes, max. 255 bytes)
6	unit identifier	0xFF (identification via IP address)
7	Function Code	0x03 (read)
8	Number of data bytes	0x02 ... 0xFC (min. 2, max. 252 bytes)
9	Contents start register address (low byte)	
10	Contents start register address (low byte)	
11..n	Additional registers	max. 125 additional registers

2.2. Write data to server

2.2.1. Client enquiry

Byte	Contents	Description
0	transaction identifier 1	Identification no. for several simultaneously active requests
1	transaction identifier 2	
2	protocol identifier 1	0x00 (MODBUS)
3	protocol identifier 2	0x00 (MODBUS)
4	Number of subsequent telegram bytes (high byte)	0x00 (always 0x00, as max. 255 data bytes)
5	Number of subsequent telegram bytes (low byte)	0x09 ... 0xFF (byte 6...n; min. 9 bytes, max. 255 bytes)
6	unit identifier	0xFF (identification via IP address)
7	Function Code	0x10 (write)
8	Start register address (high byte)	
9	Start register address (low byte)	
10	Number of registers to be written (high byte)	0x00

Byte	Contents	Description
11	Number of registers to be written (low byte)	0x01 ... 0x7C (min. 1 register, max. 124 registers)
12	Number of data bytes to be written	0x02 ... 0xF8 (min. 2 bytes, max. 248 bytes)
13	Data byte for start register address (high byte)	
14	Data byte for start register address (low byte)	
15...n	Data byte for additional registers	

2.2.2. Server response

Byte	Contents	Description
0	transaction identifier 1	Identification no. of the associated enquiry
1	transaction identifier 2	
2	protocol identifier 1	0x00 (MODBUS)
3	protocol identifier 2	0x00 (MODBUS)
4	Number of subsequent telegram bytes (high byte)	0x00 (always 0x00, as max. 255 data bytes)
5	Number of subsequent telegram bytes (low byte)	0x06 (byte 6...11)
6	unit identifier	0xFF (identification via IP address)
7	Function Code	0x10 (write)
8	Start register address (high byte)	
9	Start register address (low byte)	
10	Number of registers written (high byte)	0x00
11	Number of registers written (low byte)	0x01 ... 0x7C (client can write max. 124 registers)

2.3. Error (server)

Error code	Description
0x01	Use of an unsupported function code
0x02	Use of an invalid register address
0x03	Use of unauthorised data values, e.g. an unauthorised number of registers
0x06	Server busy (max. number of simultaneous transactions reached)

The error codes are not supported. In the event of an error or transmission error, the server will not respond.

3. Register

Parameters / Values whose content extends over more than one register are handled in big endian order (MSR first).

3.1. API address space register

Start	End	Start	End	Contents
0x0001	0x00FF			General properties of the MODBUS endpoint (revision API, controller (ESP32, SBC), server/client)
0x0100	0x20FF			Properties and function of products
		0x0100	0x01FF	Stand-alone: Product Group: Product 1
		0x0200	0x02FF	Group: Product 2
		0x0300	0x03FF	Group: Product 3
		…	…	Group: Product n
		0x2000	0x20FF	Group: Product 32
0x2100	0x2FFF			(reserved)
0x3000	0x4FFF			Properties and function of outlet
		0x3000	0x30FF	Stand-alone: Outlet 1 = Outlet left Group: Outlet 1
		0x3100	0x31FF	Stand-alone: Outlet 2 = Outlet right Group: Outlet 2
		0x3200	0x32FF	Group: Outlet 3
		...	...	Group: Outlet n
		0x4F00	0x4FFF	Group: Outlet 32

3.2. General properties of the Modbus endpoint

Describes the general properties of the MODBUS endpoint with which MODBUS communication takes place

Address space (MSB) 0x00...

Start address (LSB)	Number of registers	Contents		
0x..01	1	Revision API	R	HighByte → Major (e.g. 0x01) LowByte → Minor (e.g. 0x05)
0x..02	1	Controller type	R	0x0000 → ESP32 0x0001 → SBC
0x..03	1	Node type	R	0x0000 → Server

R → Read only access W → Write only access RW → Read and write access

3.3. Products

Describes the properties of products with outlet (charging point control level).

Address space (MSB) 0x01... - 0x20..

Start address (LSB)	Number of registers	Contents		
0x..00	16	Type	R	SAP material number of the wallbox 32 ASCII characters, two characters per register
0x..10	16	S/N	R	S/N of the wallbox 32 ASCII characters, two characters per register
0x..20	1	Construction	R	bit 15...12 → 0x0 (reserved) bit 11... 8 → 0x0= one outlet; 0x1= two outlets bit 7... 4 → 0x0= cable; 0x1= socket bit 3... 0 → 0x0= single-phase; 0x1= three-phase
0x..21	1	Outlet#	R	Number of the assigned outlets (1...32) bit 15 ... 8 → Outlet left (stand-alone 0x1) bit 7 ... 0 → Outlet right (stand-alone 0x2)
0x..22	1	FW revision	R	bit 15...12 → Major bit 11... 8 → Minor bit 7 ... 0 → Patch

Start address (LSB)	Number of registers	Contents		
0x..23	1	I_{rated}	R	Rated current of the product 6 ... 32 A 0x003C ... 0x0140 → 10x current in (A)
0x..24	1	$I_{default}$ $I_{default} \leq I_{rated}$	R	Maximum current of the product 6 ...32 A due to the installation 0x003C ... 0x0140 → 10x current in (A)
0x..26	1	Voltage AR4100	R	Voltage 0 ... 12 V at control input AR4100 0x0000 ... 0x0078 → 10x voltage in (V) In group: can only be read by controllers

R → Read only access W → Write only access RW → Read and write access

Example:

1. $I_{default}$ Stand-alone → Register 0x0124
2. Construction of outlets product 6 of a group → Register 0x0620

3.4. Outlet

Describes the properties of outlets.

Address space (MSB) 0x30... - 0x4f..

Start address (LSB)	Number of registers	Contents		
0x..00	1	Product#	R	Number of the assigned product (1...32) Stand-alone → 0x0001
0x..01	6	Electricity meter (3x uint32)	R	10x phase current in A; resolution 0.1 A bit 95 ... 64 → Phase 1 bit 63 ... 32 → Phase 2 bit 31 ... 0 → Phase 3
0x..07	6	Voltage meter (3x uint32)	R	10x phase voltage in V; resolution 0.1V bit 95 ... 64 → Phase 1 bit 63 ... 32 → Phase 2 bit 31 ... 0 → Phase 3
0x..0D	2	Active power meter (unit32)	R	Active power outlet in W 0x00000000 ... 0xFFFFFFFF → 0 ... 4294967.295 kW

Start address (LSB)	Number of registers	Contents		
0x..0F	2	Active energy meter reading (unit32)	R	100x outlet meter reading in kWh; resolution 0.0 1kWh0 ... 999999.99 kWh 0x00000000 ... 0xFFFFFFFF → 0 ... 42949672.95 kWh
0x..31	1	Outlet status (unit16)	R	0x00A0 → Outlet blocked, EV is recognised 0x00A1 → Outlet is waiting for EV 0x00A2 → Outlet reserved 0x00B0 → EV recognised, authentication failed 0x00B1 → EV recognised, authentication 0x00B2 → Outlet can provide energy for charging 0x00B3 → EV has ended or interrupted charging 0x00C2 → Outlet provides energy for charging (request by EV) 0x00E0 → Outlet blocked, EV is not recognised 0x00E2 → Outlet in boot process 0x00Fx → Error
0x..32	1	Icmax (unit16)	RW	EMS default maximum current for the outlet; resolution 0.1 A Icmax ≤ Idefault 0x0000, 0x003C ... 0x0140 → 0.0 A, 6.0 ... 32.0 A
0x..33	1	Ic (unit16)	R	Max. current that the EV may draw; resolution 0.1 A 0x0000, 0x003C ... 0x0140 → 0.0 A, 6.0 ... 32.0 A

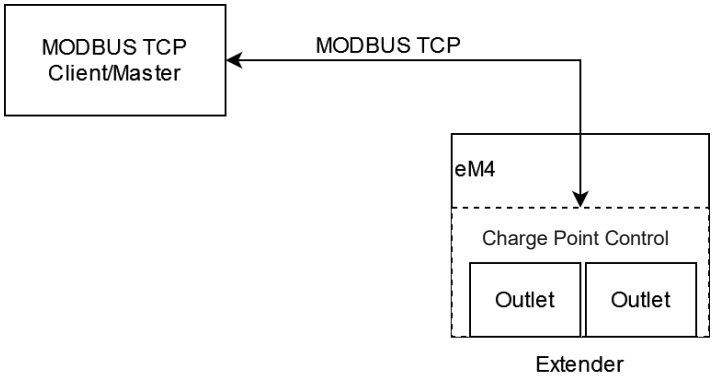
R → Read only access W → Write only access RW → Read and write access

Example:

1. Query voltage meter right TWIN stand-alone (i.e. outlet 2) → Register 0x3107
2. Query the product number to which this outlet (here outlet 6) is assigned → Register 0x3500

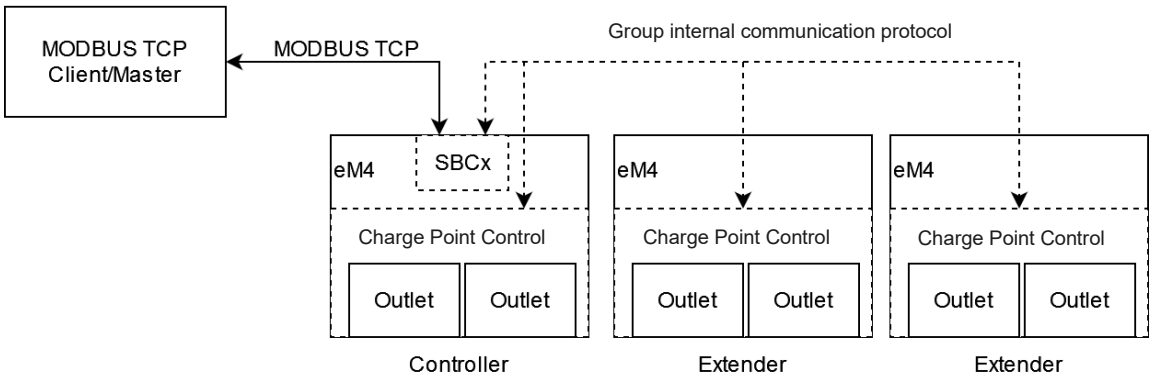
4. Application examples

4.1. Extender as stand-alone



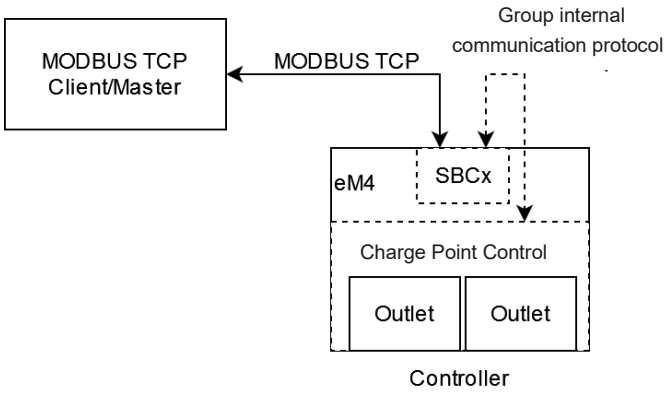
- Charge point control is MODBUS end point
- Endpoint supports 3.2 (endpoint properties), 3.3 for the product and 3.4 for each outlet

4.2. Group with controller and extender



- SBCx the controller is MODBUS end point
- Endpoint supports 3.2 (endpoint properties), 3.3 for each product and 3.4 for each outlet

4.3. Controller as stand-alone



- SBCx the controller is MODBUS end point
- Endpoint supports 3.2 (endpoint properties), 3.3 for the product and 3.4 for each outlet