Battery Monitoring System

(Battery monitoring system for larger UPS systems)

MODBUS MANUAL

Version: 2.0
Date: 08.06.2022
Document No: 01062022

Robotina d.o.o.
www.robotina.com
info@robotina.com

Table of content

1. Introduction

The purpose of the BTMS Controller MODBUS TCP/IP communications protocol is to allow that all measured data to be efficiently transferred between a MODBUS Master Station (SCADA or BMS) and a BT-C.

Data Encoding

MODBUS uses a 'big-endian' representation for addresses and data items. This means that when a numerical quantity larger than a single byte is transmitted, the MOST significant byte is sent first. So, for example:

type		value	1st byte	2nd byte	3rd byte	4th byte
int	16 bits	0x1234	0x12	0x34	/	/
long	32 bits	0x12345678	0x12	0x34	0x56	0x78

Example

Read UPS Voltage (Address: 1, Type: int, Scale: 0.01):

Sent message:

0x00	0x01	0x00	0x00	0x00	0x06	0x01	0x03	0x00	0x01	0x00	0x02
Transaction ID		Protocol ID		Length		Unit ID	Function code	Address		Quantity of registers	

Received message:

0x00	0x01	0x00	0x00	0x00	0x06	0x01	0x03	0x04	0x00	0x12	0xD6	0x87
Transaction ID		Protocol ID		Length		Unit ID	Function code	Byte count	Register values			

0x0012D687 = 1234567 => 12345,67 V

More detailed information about the Modbus TCP/IP protocol can be found here: https://www.simplymodbus.ca/TCP.htm.

The free Modbus communication testing application QModMaster can be obtained here: https://sourceforge.net/projects/qmodmaster/.

2. Modbus registers

UPS

var	address	type	scale	description
ups_status	0	int	-	UPS status (0 = disabled, 1 = OK (all strings OK), 2 = ERR (at least one string error))
ups_voltage	1	long	0.01	UPS voltage (average of strings voltages) [0.01 V]
ups_current	3	long	0.01	UPS current (sum of string currents) [0.01 A]; + = charge, - = discharge
ups_soc	5	int	1	UPS state of charge (average of strings SOC) [%]

String

si = string index (0-7)

var	address	type	scale	description
string_status	20 + 2440 × si	int	-	String status (0 = disabled, 1 = OK, 2 = Error)
string_voltage	21 + 2440 × si	long	0.01	String voltage [0.01 V]
string_current	23 + 2440 × si	long	0.01	String current [0.01 A]; + = charge, - = discharge
string_soc	25 + 2440 × si	int	1	String state of charge [%]
string_ballance	26 + 2440 × si	long	0.01	String balancing state [%]
string_state	28 + 2440 × si	int	-	0=floating charge, 1=equalizing charge, 2=discharge, 3=idle, 5=abnormal
string_alarm	29 + 2440 × si	int	-	bit coded; b0=current hi (charging), b1=current lo (discharging), b2=voltage hi, b3=voltage lo, b4=SOC lo, b5=SOH lo, b6=hall disconnected
string_cell_count	30 + 2440 × si	int	1	Number of string cells
string_ambient_temperature	31 + 2440 × si	long	0.1	String ambient temperature [0.1 °C]
string_ambient_humidity	33 + 2440 × si	int	1	String ambient relative humidity [% RH]
string_relay_status	34 + 2440 × si	int		String relay status (0=open, 1=closed)

Cell

si = string index (0...7) ci = cell index (0...119)

var	address	type	scale	description
cell_status	60 + 2440 × si + 20 × ci	int	-	Cell status (0=disabled, 1=OK, 2=error)
cell_voltage	61 + 2440 × si + 20 × ci	long	0.001	Cell voltage [0.001 V]
cell_resistance	63 + 2440 × si + 20 × ci	long	0.001	Cell resistance [0.001 mΩ]
cell_temperature	65 + 2440 × si + 20 × ci	long	0.1	Cell temperature [0.1 °C]
cell_soc	67 + 2440 × si + 20 × ci	int	1	Cell state of charge [%]
cell_soh	68 + 2440 × si + 20 × ci	int	1	Cell state of health [%]
cell_alarm	69 + 2440 × si + 20 × ci	int	-	Cell alarm (bit coded): b0=voltage hi, b1=voltage lo, b2=resistance hi, b3=SOC lo, b4=SOH lo, b5=temperature hi
cell_remaining_time	70 + 2440 × si + 20 × ci	long	0.1	Cell remaining time [0.1 h]
cell_status	60 + 2440 × si + 20 × ci	int	-	Cell status (0=disabled, 1=OK, 2=error)
cell_voltage	61 + 2440 × si + 20 × ci	long	0.001	Cell voltage [0.001 V]
cell_resistance	63 + 2440 × si + 20 × ci	long	0.001	Cell resistance [0.001 mΩ]

Legend

type	
int	16 bits signed (-32768...32767)
long	32 bits signed (-2147483648...2147483647)
scale	
0.01	12345 → 123.45

The information in this manual is subject to change without notice. Robotina assumes no responsibility for any errors that may appear in this document.

In no case is Robotina liable for incidental or consequential damages arising from the use of this manual or the software or hardware described in this document.