TRC SERIES

MODBUS

USER MANUAL

REV01- MARCH 2017

CONTENT

TABLES

FIGURES

COMMUNICATION

Main unit is designed to communicate in serial with the other devices over the RS485 data link with Modbus[1] RTU protocol.

The device stores the data in the registers that it can use for communication and makes it possible to access them with the parameters of the protocol.

The device can communicate with different baud rate (9600, 19200, 38400 kbps) options. The data length is 8 bits long.

The device allows communication with some restrictions for different reasons. It allows error interpretations in case of negligence of these restrictions by making error returns. Software that will communicate with Battery & Charger Set should be able to take into account the restrictions and interpret the returns.

This document is about communication method with Modbus RTU protocol, register addresses of TRC Series Battery & Charger sets and related restrictions.

Modbus

Modbus is a serial communication protocol developed by Modicon for use with PLCs in 1979. Modbus can provide communication between 247 devices. The operational logic is based on data exchange between a master device and more devices(slaves) connected to the master device on the same network.

Modbus protocol has many versions; TRC Series use Modbus RTU among these versions.

Data Model

The Modbus data model is created by separating data according to the distinguishing characteristic of the data. Accordingly, 4 basic data tables have occurred.

Table Name	Data Type	Feature	Explanation
Discrete Input	bit	Read Only	It covers only the readable bits of the system.
Coils	bit	Read and Write	It covers the exchangeable bits of the system.
Analog Registers	16-bit	Read Only	It covers the readable analog data of the system.
Writable Registers	16-bit	Read and Write	Reserved modifiable registers of the system.

Tablo 1: Modbus Data Model

[1] Detailed information for modbus: http://www.modbus.org/docs/Modbus_Application_Protocol_V1_1b.pdf

Discrete input values hold values used as single-bit status indicators. However, since there is no such status indicator variable in TRC Series, there is not any defined discrete input in this version.

Coils are defined in the general Modbus as user-changeable values. Relay values in TRC Series are mapped as coils. These values are readable only because they are values that cannot be changed by the user.

Analog registers are 16-bit units and represent units assigned to hold values that cannot be modified by the user. For TRC Series, 10 register in total are assigned to hold 5 different values that are divided into 2-bit registers by data coding method because the lengths are 32-bit (4 byte).

Writable registers are 16-bit registers and hold values that can be changed by user. There are total of 12 writable registers for different variables in TRC Series. Because the information stored here directly affects operation of the device, writing request should not be sent without being sure about the changes to be made during the operation.

Data Codes

Modbus addresses and Modbus data generate data blocks by encoding with "Big-Endian" representations. It solves by sending the big-byte as the first byte for communication of data that does not fit in a byte.

Example:

Register Size	Value	MSB	LSB
16-bit (2 bayt)	**0x2450**	first byte **0x24**	last byte **0x50**

Şekil 1: "Big-Endian" Representation

Adressing Model

Separate blocks are used for each data model in AAGD. Therefore, there is an addressing starting from "0" to "65535" for each data model. This addressing model is configured in accordance with the "IEC-61131 object" standard.

Necessary addresses and data types are shown on the relevant tables.

Functions

The device interprets the data block sent by the mainframe and generates an appropriate response. The second byte of sent data block gives information that will perform to the device.

The vast majority of the basic functions that provide Modbus communication are pre-defined and standardized. Some of these functions can be used for TRC Series, but others are not available because they indicate non-existent registers and functional features.

02 function is used to read discrete inputs but it has an invalid function value because there are no discrete inputs to be read on the device. Modbus can provide up to 65 predefined function but not all of these functions are standard, as well as allowing user-defined functions after 65. Function values can have up to 255 values, starting with 01, allowing a byte in a data frame.

Functions can be used in device:

Function 01 – – The discrete values used to **Read the Coil Values** are processed in bytes and returned as data blocks. More information on this topic can be found under the heading "Discrete Inputs" entry.

Function 03 – The function used to read **Writable Registers** can be used to read a single register or multiple register. The return values return as a data frame.

Function 04 – The function used to read **Analog Registers** allows single or multiple reads.

Function 06 – Single Register Writing function is used to assign a new value to one of the registers that can be written.

Function 16 – Multiple Register Writing function is used to give new values to more than one of registers that can be written. The register address to be written has to be sequential.

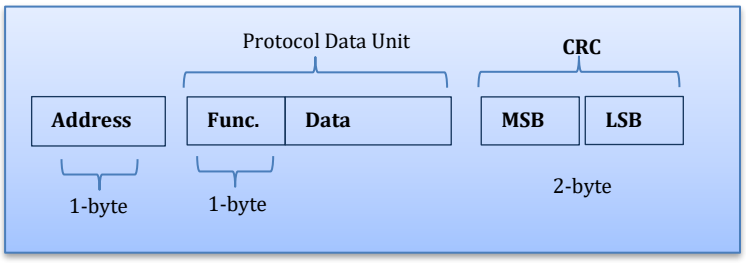
The functions except above functions are not included in this user manual because they are not used for this device.

Data Frame

Modbus provides serial communication by sending and receiving data as block. The first byte of this block carries the address information. So, devices that communicate with Modbus must have value less than 255(0xFF), which is the maximum value that a byte can represent.

The second byte of data frame carries function information, which is evaluated in different ways after the data frame according to the function information.

The last two cells (2 bytes) of each data frame are reserved for CRC (Cyclic Redundancy Check) values, which are special mathematical operations designed for error checking and are frequently referred to in serial communication protocols.



Şekil 2: Modbus Data Frame

Restrictions

In addition to general Modbus restrictions, there are different restrictions on device communication. These restrictions are important about communication and healthy operation of device.

Modbus Restrictions

Modbus protocols can only be realized the communication between devices under certain conditions. For example, if the message is send to the device with a wrong address by a master device, it cannot be returned.

In addition, Modbus also includes serial communication restrictions, so it can send and received 256 bytes data frame. In this device, up to 60 bytes message frames can be carried.

CRC (Cyclic Redundancy Check)

The last two bytes of each message is send with a block reserved for CRC. The control is done by these two bytes if the message block is damaged and changed for any reason.

The CRC is valid used method of fault control method for digital communication is also valid method for Modbus. If a data frame with fault is send, a fault message will be returned.

Wrong and Undefined Functions

Even if the wrong and undefined functions are routed to the correct device address, no action will be taken and an error will be returned. Therefore, it is important to use defined functions in the specified way.

Infraction of the Read-Only Feature

If it is tried to write a value to the only-read data structure, this message will be interpreted but the value will not be written. In such a case, there will be a fault return due to the Modbus protocol.

TRC Series Restrictions

In addition to device protocol dependencies, there are registers that are subject to some restrictions and special writing rules due to capacity of device, usage safety and structural safety.

This tool allows 60 bytes of data to be exchanged. This restriction has been made so that the number of registers is low and communication can run smoothly.

The device can provide 6 baud rates (4800, 7200, 9600, 14400, 19200 and 38400).

Stop bit option can be user defined “1” or “2”.

Parity option is predefined as “none”.

Data bits are predefined and 8-bits long.

Coils

There are 11 coils defined in the device.

Each coil has a bit value and is processed into bytes in the data blocks. Each byte tries to perform 8-bit operation from left to right as if it were the smallest leftmost address value.

No	Address	Name	Explantion	Data Type	Feature
1	**0** (0000)	**Relay 1**	Aural Warning Relay	bit	Read Only
2	**1** (0001)	**Relay 2**	Fan Control Relay	bit	Read Only
3	**2** (0002)	**Relay 3**	Battery Contact Relay	bit	Read Only
4	**3** (0003)	**Relay 4**	Short Circuit Control Relay	bit	Read Only
5	**4** (0004)	**Relay 5**	Over Load Control Relay	bit	Read Only
6	**5** (0005)	**Relay 6**	(-) DC Leakage Control Relay	bit	Read Only
7	**6** (0006)	**Relay 7**	(+) DC Leakage Control Relay	bit	Read Only
8	**7** (0007)	**Relay 8**	AC Over Current Relay	bit	Read Only
9	**8** (0008)	**Relay 9**	AC Low Current Relay	bit	Read Only
10	**9** (0009)	**Relay 10**	DC Over Current Relay	bit	Read Only
11	**10** (000A)	**Relay 11**	DC Lower Current Relay	bit	Read Only

Table 2: Coils

Read Function

01 function is used to read coils. On return of the function a bit is allocated in the data field for each coil. Since the return of the data blocks will be in bytes, the coil values are resolved in the return message.

Example: The coils are wanted to read with using 01 function;

If it is wanted to read 6 coils started from the device address 4 and 0,

Sent Message: **04 01 00 00 00 06 BC 5D -** This is the display in hexadecimal format.

04 – Device address

01 – Function code

0000 – First coil address

0006 – Amount of the total coil wanted to read

BC5D – CRC fault control bytes

Returned Message: **04 01 01 04 50 87** - This is the display in hexadecimal format.

04 – Device address

01 – Function code

01 – The number of bytes of incoming data has

04 – Incoming byte value (0000 0101) Because the coil value is read from 0 to 5, the first two bits from the left are undefined as 0. The lowest coil address is read starting from the left. Accordingly, the 0th coil value is 0, the 1st coil value is 0, the 2nd coil value is 1, the 3rd value is 0, the 4th value is 0, and the 5th value is 1. This knowledge shows that the 2nd and 5th step are active.

5087 - CRC hata kontrol baytları

If the total size of the values to be read is longer than one byte, the data bits after the largest addressed coils are returned as "0". If it is wanted to read 11 coils and assume that the return value is **"04 01 02 04 05 7B 3F":**

04 – Device address

01 – Function Code

02 – The number of bytes of incoming data has

04 – (**0000 0100**) between 7-0 coil values

05 – (**0000 0101**) between 10-8 coil values, it indicates space 5 bits on the left.

Writable Registers

12 writable registers in the device are defined below table.

These registers have both reading and writing features. They hold different values from device addresses to temperature protection. All of them have 16-bit numeric data type.

While function 03 is used to read writable registers, functions 06 and 16, which are write functions, can also be used to change the values of these registers. However, it should not be forgotten that there are areas to influence the communication directly when their values change. For example, if we change the value of address, we must write newly saved value on the device address otherwise it should not be forgotten that it cannot provide communication. To avoid this and similar problems, it is needed to be sure of the values to be written before the write operation.

No	Address	Name	Explanation	Data Type	Feature
1	**0** (0000)	**Address**	Device Address	int **min:1 – max:255**	Read and Write
2	**1** (0001)	**Baud**	Baud Rate	int **min:0 – max:5**	Read and Write
3	**2** (0002)	**Stop Bit**	Stop Bit	int **min:0 – max:1**	Read and Write
4	**3** (0003)	**VCharge**	Charge Voltage Limit	int **min:215 – max:290**	Read and Write
5	**4** (0004)	**ICharge**	Charge Current Limit	int **min:1 – max:30**	Read and Write
6	**5** (0005)	**IDCout**	Output Current Limit	int **min:1 – max:200**	Read and Write
7	**6** (0006)	**VAC_Hi**	AC Over Voltage Protection	int **min:230 – max:270**	Read and Write
8	**7** (0007)	**VAC_Low**	AC Low Voltage Protection	int **min:150 – max:210**	Read and Write
9	**8** (0008)	**VDC_Hi**	DC Over Voltage Protection	int **min:240 – max:290**	Read and Write
10	**9** (0009)	**VDC_Low**	DC Low Voltage Protection	int **min:200 – max:225**	Read and Write
11	**10** (000A)	**FAN_In**	Fan Input Temperature	int **min:20 – max:60**	Read and Write
12	**11** (000B)	**FAN_Out**	Fan Output Temperature	int **min:15 – max:45**	Read and Write

Tablo 3: Writable Registers

If baud rate or parity values are changed, it is needed to readjust the baud rate and parity values of master device to communicate with the slave device again.

Special Register Value

Baudrate Register: 4800, 9600, 19200, 38400 kbps options are adjusted according to values explaned below.

If the Baud register is 0, the baudrate will be 4800, 2 refers to 7200 kbps, 3 refers to 9600 kbps, 4 refers to 14400 kbps, 5 refers to 19200. If a value greater than 5 is written to the baud register, the mode of value is taken according to the number 5.

Stop Bit: If the stop bit register value is 0, then the stop bit is “1”. If the stop bit register is 1, thenn the stop bit value is “2”. If a value greater than 2 is written to the baud register, the mode of value is taken according to the number 2.

Read Function

Function 03 is the read function for the writable registers.

Example:

If the device address is 4 and reading total of 2 registers starting from 5.

Sent Message: **04 03 00 05 00 02 D4 5F**

04 – Device Address

03 – Function Code

0005 – The first register address desired to read

0002 – The amount of the register desired to read

D45F – CRC fault control bytes

Dönen mesaj: 04 03 04 00 BE 00 FA 4F 54

04 – Device Address

03 – Function Code

04 – Number of data byte, it gives information about holding data the next 4 bytes.

00BE – The fisrt register (0005 – ***IDCout***) value (00BE = 190)

00FA – The second register desired to read (0006 – ***VAC_Hi***)

4F54 – CRC fault control bytes

Writing Functions

Two different writing functions are defined before for writing registers in Modbus. These functions are Function 06 and Function 16. While Function 06 allows writing on only one register, Function 16 can provide writing on multiple registers.

Fonksiyon 06 örnek:

Let the device adres is 4 and the fan input value wanted to be "40" degree;

Sent message : 04 06 00 0B 00 28 F8 43

04 – Device Address

06 – Function Code

000B – Register address

0028 – Writing value

F843 - CRC fault control bytes

Received message : 04 06 00 0B 00 28 F8 43 if writing operation is done without any problem, the sent message and received message will be same. If there is a problem a fault message will return.

Function 16 example: This function allows multiple writes for successive registers.

Let device address is 04 and value of 9th register is 25, value of 10th register is 05 and value of 11th register is 17.

Sent message: 04 10 00 09 00 03 06 00 19 00 05 00 11 f7 65 The message length will vary depending on the number of registers desired to be written. According to the example, we send a 15-byte data block.

04 – Device Address

10 – Function Code (16)

0009 – Starting address

0003 – Amount of register

06 – Number of data byte, it will hold the data that will write on the registers followed 6 bytes respectively.

0019 – (**0009**) The value to be written to register at the first address specified.
0005 – (**000A**) The value to be written to register at the second address specified.

0011 – (**000B**) The value to be written to register at the third address specified.

F765 – CRC fault control bytes

Returned Message: **04 10 00 09 00 03 50 5F** – if message is successful

04 – Device Address

10 – Function Code

0009 – Starting address

0003 – Amount of written register

505F – CRC fault control bytes

If the write operation fails in the write functions, a fault message of 5 bytes will be returned.

Analog Registers

There are 5 analog input registers that are defined in the table below. These registers hold only readable input values. Registers holding different qualifiers in each 32-bit length are mapped in 4-byte fields with "Big-Endian" notations.

Two registers must be read together for the exact reading of any of the analog registers. Only a high or only low value of a register can be read if needed.

No	Address	Name	Explanation	Data Type		Feature
1	**0** (0000)	**VAC**	AC Voltage	long	(MSB)	Read Only
2	**1** (0001)				(LSB)	
3	**2** (0002)	**Vout**	Output Voltage	long	(MSB)	Read Only
4	**3** (0003)				(LSB)	
5	**4** (0004)	**Iout**	Output Current	signed long	(MSB)	Read Only
6	**5** (0005)				(LSB)	
7	**6** (0006)	**IBat**	Battery Current	signed long	(MSB)	Read Only
8	**7** (0007)				(LSB)	
9	**8** (0008)	**T**	Temperature	long	(MSB)	Read Only
10	**9** (0009)				(LSB)	

Tablo 4: Analog Registers

Reading Function

Reading function of analog registers is 04.

Örnek: If the device address is 4 and the battery current is wanted to be read;

The begining addres will be 6 and 2 register will be read. (Battery current data type is signed long and the data length is 32-bit)

Sent message : **04 04 00 06 00 02 91 9F**

04 – Device Address

04 – Function Code

0006 – Starting address

0002 – Amount of register desired to read

919F – CRC fault control byte

Returned message : 04 04 04 00 00 00 0E 2F 40

04 – Device Address

04 – Function Code

04 – Number of data byte

0000 – (**0006**) value of first register

000E – (**0007**) value of second register (0E = 14)

INDEX