MODBUS COMMUNICATION			Code ID. 10001	Name: Chris Jin
RAEDIAN NEO AC Wallbox Modbus RTU				Approved:
Revision	Revised by	Approved	Description	
0	Chris Jin		first draft	

Contents

1. Purpose **3**

2. Scope **3**

3. Reference **3**

4. Overview **3**

5. Modbus Protocol **4**
5.1 Modbus Protocol 4
5.1.1 Modbus Request Frame 4
5.1.2 Message Types 4
5.1.3 Exception Responses 4
5.2 Supported Function Codes 5
5.2.1 Function Code 3 (Read Holding Registers) 5
5.2.2 Function Code 16 (Write Multiple Register) 6
5.2.3 Function Code 6 (Write Single Register) 7
5.3 Reading and Writing to Registers 7

6. Mapping Tables **10**

1. Purpose

This document describes the mapping from smart charging data to Modbus RTU and how to read and write to registers in RAEDIAN NEO AC Wallbox charging station. The Modbus register map is intended to not conflict with most of the smart meter.

2. Scope

This document applies to RAEDIAN NEO AC Wallbox charging station act as Modbus Slave or Modbus Slave Device.

3. Reference

NEO NEO AC Wallbox

EMS Energy Management System i.e. Solar Inverter

4. Overview

The RAEDIAN NEO AC Wallbox is developed to support:

- Modbus RTU(Available Firmware V2.1.X)
- Modbus Master and Modbus Slave(as known as Modbus Master and Modbus Slave)
- As Modbus Slave to provide smart charging data for NEO AC Wallbox Master.
- The Modbus Master shall be configured by the application software or APP to identify the Slave devices:
  - Modbus Slave address of each Slave devices
  - Type of each Slave devices
  - Other configurations

5. Modbus Protocol

5.1 Modbus Protocol

Modbus is a Master-Slave communication protocol that can support up to **247** Secondaries organized as a multidrop bus. The communication is half duplex. Services on Modbus are specified by function codes.

The function codes are used to read or write 16-bit registers. All charging data, such as active energy, voltage or firmware version, is represented by one or more such registers. For further information about the relation between register number and charging data, refer to “Mapping Tables” in chapter 6.

The Modbus protocol is specified in its entirety in Modbus Application Protocol Specification V1.1b. The document is available at http://www.modbus.org

5.1.1 Modbus Request Frame

A Modbus request frame generally has the following structure:

Slave Address	Function Code	Data	Error Check

- Slave address: Modbus Slave address, 1 byte.
- Function code: Decides the service to be performed.
- Data: Dependent on the function code. The length varies.
- Error check: CRC-16, 2 bytes.

5.1.2 Message Types

The network messages can be query-response or broadcast type. The query response command sends a query from the Master to an individual Slave and is generally followed by a response.

The broadcast command sends a message to all Secondaries and is never followed by a response. Broadcast is supported by function code 6 and 16.

5.1.3 Exception Responses

If an error should occur while processing a request, then the NEO gives an exception response that contains an exception code.

In the exception response the function code is set to the function code of the request plus 0x80.

An exception frame has the following structure:

Slave Address	Function Code	Exception Code	Error Check

The exception codes that are used are listed in the following table:

Code	Exception	Definition

01	Illegal function	A function code that is not supported has been used.
02	Illegal data address	The requested register is outside the allowed range.
03	Illegal data value	The structure of a received message is incorrect.
04	Slave device failure	Processing the request fail due to an internal error in the meter.

5.2 Supported Function Codes

The following function codes are supported:

- Function code 3 (Read holding registers)
- Function code 6 (Write single register)
- Function code 16 (Write multiple registers)

5.2.1 Function Code 3 (Read Holding Registers)

Function code 3 is used to read measurement values or other information from the electricity meter. It is possible to read up to **125** consecutive registers at a time. This means that multiple values can be read in one request.

Request frame:

Slave Address	Function Code	Address	No. of Registers	Error Check

Response frame:

Slave Address	Function Code	Byte Count	Register Values	Error Check

Example: Read total energy import, etc. …

Request:

Slave address	0x01
Function code	0x03
Start address, high byte	0x50
Start address, low byte	0x00
No. of registers, high byte	0x00
No. of registers, low byte	0x18
Error check (CRC), high byte	0x54
Error check (CRC), low byte	0xC0

Response:

Slave address	0x01
Function code	0x03
Byte count	0x30
Value of register 0x5000, high byte	0x00

Value of register 0x5000, low byte	0x15
…	0x18
Value of register 0x5017, high byte	0xFF
Value of register 0x5017, low byte	0xFF
Error check (CRC), high byte	0xXX
Error check (CRC), low byte	0xXX

5.2.2 Function Code 16 (Write Multiple Register)

Function code 16 is used to modify settings in the meter, such as date/time, to control output and to reset values, such as power fail counter. It is possible to write up to **123** consecutive registers in a single request. This means that several settings can be modified and/or several reset operations can be performed in a single request.

Request frame:

Slave Address	Function Code	Start Address	No. of Registers	Byte Count	Register Values	Error Check

Response frame:

Slave Address	Function Code	Start Address	No. of Registers	Error Check

Example: set Date/Time to November 11, 2010,12:13:14

Request:

Slave address	0x01
Function code	0x10
Start address, high byte	0x8A
Start address, low byte	0x00
No. of registers, high byte	0x00
No. of registers, low byte	0x03
Byte count	0x06
Value of register 0x8A00, high type	0x0A
Value of register 0x8A00, low type	0x0B
Value of register 0x8A01, high type	0x0B
Value of register 0x8A01, low type	0x0C
Value of register 0x8A02, high type	0x0D
Value of register 0x8A02, low type	0x0E
Error check (CRC), high byte	0x8C
Error check (CRC), low byte	0x82

Response:

Slave address	0x01

Function code	0x10
Register address, high byte	0x8A
Register address, low byte	0x00
No. of registers, high byte	0x00
No. of registers, low byte	0x03
Error check (CRC), high byte	0xAA
Error check (CRC), low byte	0x10

5.2.3 Function Code 6 (Write Single Register)

Function code 6 can be used as an alternative to function code 16 if there is only one register to be written. It can, for example be used to reset the power fail counter.

Request frame:

Slave Address	Function Code	Register Address	Register Value	Error Check

Response frame:

Using function code 6, the response frame is an echo of the request frame.

Example: reset power fail counter.

Request:

Slave address	0x01
Function code	0x06
Register address, high byte	0x8F
Register address, low byte	0x00
No. of registers, high byte	0x00
No. of registers, low byte	0x01
Error check (CRC), high byte	0x62
Error check (CRC), low byte	0xDE

5.3 Reading and Writing to Registers

Readable Register

The readable range in the Modbus mapping are registers 1000–8EFF (hexadecimal). Reading any registers within this range will result in a normal Modbus response. It is possible to read any number of registers between 1 and 125, i.e., it is not necessary to read all registers of a quantity listed on one line in the s. Any attempt to read outside this range will result in an illegal data address exception (Modbus exception code 2).

Multi-Register Values

For quantities that are represented as more than 1 register, the most significant byte is found in the high byte of the first (lowest) register. The least significant byte is found in the low byte of the last (highest) register.

Unused Registers

Unused registers within the mapping range, for example missing quantities in the connected charging station, will result in a normal Modbus response but the value of the register will be set to “invalid”.

For quantities with data type “unsigned”, the value will be FFFFh in all registers.

For quantities with data type “signed”, the value is the highest value possible to express. That means that a quantity that is represented by only one register will have the value 7FFFh. A quantity that is represented by 2 registers will have the value 7FFFFFFFh, and so on.

Writing to Registers

Writing to registers is only permitted to the registers listed as writable in the mapping tables. Attempting to write to a register that is listed as writable but that is not supported by the charging station will not result in an error indication.

Confirm Set Values

After setting a value in the charging station, it is recommended that you read the value to confirm the result, since it is not possible to confirm if a write was successful from the Modbus response.

Available register for Confirm Set Values, is the writable “0x8100 Set charging current”, which can be read and confirm by reading “0x800E Current charging current limit”

5.4 RS 485 Configuration

5.4.1 Baud Rate

The default setting of the RS-485 communication interface:

Item	Value	Description
Baud Rate	9600	Data rate for serial data transmission (in bits per second (bps)).
Data bits	8	Number of data bits for each character.
Parity	None	In serial transmission, parity is a method of detecting errors. An extra data bit is sent with each data character, arranged so that the number of 1 bit in each character, including the parity bit, is always odd or always even. If a byte is received with the wrong number of 1s, then it must have been corrupted. However, an even number of errors can pass the parity check. **None** (**N**) - no parity method is used.
Stop bits	1	Stop bits sent at the end of every character allow the receiving signal hardware to detect the end of a character and to re-synchronize with the character stream. Electronic devices usually use one stop bit. Two stop bits are required if slow electromechanical devices are used.
Flow control	None	In many circumstances a transmitter might be able to send data faster than the receiver is able to process it. To cope with this, serial lines often incorporate a "handshaking" method, usually distinguished between hardware and software handshaking.

6. Mapping Tables

The purpose of this section is to explain the relation between register number and smart charging data in NEO AC Wallbox (Modbus Slave). Recommended polling interval for Readable Data is 30s~90s.

Start Reg	Name	Size	Res.	Unit	Data Type	Attr.
Readable Data						
8000	Serial Number	4			UNSIGNED	RO
8004	Firmware version	2			UNSIGNED	RO
8006	Max rated/settable current	2	0.001	A	UNSIGNED	RO
800A	Cable lock	2			UNSIGNED	RO
800C	Charging state	2			UNSIGNED	RO
800E	Current charging current limit	2	0.001	A	UNSIGNED	RO
8010	Charging current phase 1	2	0.001	A	UNSIGNED	RO
8012	Charging current phase 2	2	0.001	A	UNSIGNED	RO
8014	Charging current phase 3	2	0.001	A	UNSIGNED	RO
8016	Voltage phase 1	2	0.1	V	UNSIGNED	RO
8018	Voltage phase 2	2	0.1	V	UNSIGNED	RO
801A	Voltage phase 3	2	0.1	V	UNSIGNED	RO
801C	Active power	2	1	W	UNSIGNED	RO
801E	Energy delivered in charging session	2	1	Wh	UNSIGNED	RO
Writeable Data						
8100	Set charging current limit	2	0.001	A	UNSIGNED	WO
8102	Set charging phase	1			UNSIGNED	WO
8105	Start/Stop Charging Session	1			UNSIGNED	WO

6.1 8000H – Serial Number

This register contains the serial number of the charging station.

Address	Name	Size	Res.	Unit	Data Type	Attr.
8000H	Serial number	4			UNSIGNED	RO

6.2 8004H – Firmware version

This register contains the firmware version of the charging station.

Address	Name	Size	Res.	Unit	Data Type	Attr.
8004H	Firmware version	2			UNSIGNED	RO

6.3 8006H – Max Rated/Settable current

This register contains the maximum current value that can be supported by the hardware of the charging station, and settable by RAEDIAN App

Address	Name	Size	Res.	Unit	Data Type	Attr.
8006H	Max supported current	2	0.001	A	UNSIGNED	RO

Example:

Value 10000(DEC): The maximum charging current of the charging station is 10000 mA = 10 A.

Note: Settable charging current could only be set at 1A interval, e.g. Value 16154(DEC) will be identified as 16A, 154mA will be ignored.

6.4 8008H – Error code

This register contains the error code of the charging station.

Address	Name	Size	Res.	Unit	Data Type	Attr.
8008H	Error code	2			UNSIGNED	RO

Supported values:

- 0: No error
- >1: Specific error code. (refer to user manual)

6.5 800AH – Socket lock state

This register contains the state of the charging cable, and socket lock.

Address	Name	Size	Res.	Unit	Data Type	Attr.
800AH	Socket Lock state	2			UNSIGNED	RO

Supported values:

- 0x0000: No cable is plugged.

- 0x0001: Cable is connected to the charging station unlocked
- 0x0011: Cable is connected to the charging station locked
- 0x0101: Cable is connected to the charging station and the electric vehicle, locked in charging station
- 0x0111: Cable is connected to the charging station and the electric vehicle, unlocked in charging station.

6.6 800CH – Charging state

This register contains the state of the charging station.

Address	Name	Size	Res.	Unit	Data Type	Attr.
4000CH	Charging state	2			UNSIGNED	RO

Charging State contains 4 bytes, A3 A2 A1 A0 in order.

A0- Availability

Reserved

A1-Charging

Bit7: 0- Charging rated current

1- Under set charging current limit

Bit 6~0

0x00: State A: Idle

0x01: State B1: EV Plug in, pending authorization

0x02: State B2: EV Plug in, EVSE ready for charging(PWM)

0x03: State C1: EV Ready for charge, S2 closed(no PWM)

0x04: State C2: Charging Contact closed, energy delivering.

0x05: Other

A2, A3 are reserved for future purpose

6.7 800EH – Current charging current limit

This register contains the maximum charging current of the charging station.

Address	Name	Size	Res.	Unit	Data Type	Attr.
800EH	Max charging current	2	0.001	A	UNSIGNED	RO

Example:

Value 10000(DEC): The maximum charging current of the charging station is 10000 mA = 10 A.

6.8 8010H – Charging current (L1/2/3)

This register contains the measured current value on phase 1/2/3 in mA.

Address	Name	Size	Res.	Unit	Data Type	Attr.
8010H	Charging current L1	2	0.001	A	UNSIGNED	RO
8012H	Charging current L2	2	0.001	A	UNSIGNED	RO
8014H	Charging current L3	2	0.001	A	UNSIGNED	RO

Example:

Value 9450(DEC): The charging current on phase 1 is 9450mA = 9.450A

6.9 8016H – Voltage (L1/2/3)

This register contains the measured voltage value on each phase in volts.

Address	Name	Size	Res.	Unit	Data Type	Attr.
8016H	Voltage phase 1	2	0.1	V	UNSIGNED	RO
8018H	Voltage phase 2	2	0.1	V	UNSIGNED	RO
801AH	Voltage phase 3	2	0.1	V	UNSIGNED	RO

Example:

Value 2265(DEC): The measured voltage value on phase is 226.5 V.

6.10 801CH – Active power

This register contains the active power in W.

Address	Name	Size	Res.	Unit	Data Type	Attr.
801CH	Active power	2	1	W	UNSIGNED	RO

Example:

Value 11331(DEC): The active power of the charging station is 11331 W = 11.331 kW.

6.11 801EH – Energy delivered in charging session

This register contains the transferred energy of the current charging session.

Address	Name	Size	Res.	Unit	Data Type	Attr.
801EH	Charged energy	2	1	Wh	UNSIGNED	RO

Example:

Value 21058(DEC): The transferred energy of the current charging session is 21.058 kWh.

6.12 8100H – Set Charging Current Limit

This register sets the maximum charging current. The value will be used immediately and stays valid until a reset or rewriting the register.
This value can be read back from register 400EH.

Address	Name	Size	Res.	Unit	Data Type	Attr.
8100H	Set charging current limit	2	1	0.001A	UNSIGNED	WO

Example:

Value 12000(DEC): Set the charging current for the charger is 12.000A.(Number after decimal point will be ignored)

6.13 8102H – Set Charging Phase

This register sets the number of phases that will be used for charging. The value stays valid until a reset or rewriting the register.

Address	Name	Size	Res.	Unit	Data Type	Attr.
8102H	Set charging phase	1	1		UNSIGNED	WO

Example:

0x01: Set the charging to single phase

Note: When three phase charger set to charging at single phase, only the wire connected to the L1 terminal will close contact to deliver energy.

0x02: Set the charging to three phases

Note: Invalid for single phase charger

6.14 8105H – Start/Stop Charging Session

Writing this register starts or stops a charge session depending on the value.

Address	Name	Size	Res.	Unit	Data Type	Attr.
8105H	Start/Stop charging session	1	1		UINT16	WO

0x00: Start

0x01: Stop