Overview

This document describes the specification of the PZEM-004T AC communication module, the module is mainly used for measuring AC voltage, current, active power, frequency, power factor and active energy, the module is without display function, the data is read through the TTL interface.

PZEM-004T-10A: Measuring Range 10A (Built-in Shunt)
PZEM-004T-100A: Measuring Range 100A (external transformer)

1. Function description

1.1 Voltage

1.1.1 Measuring range: 80 ~ 260V

1.1.2 Resolution: 0.1V

1.1.3 Measurement accuracy: 0.5%

1.2 Current

1.2.1 Measuring range: 0 ~ 10A (PZEM-004T-10A); 0 ~ 100A (PZEM-004T-100A)

1.2.2 Starting measure current: 0.01A (PZEM-004T-10A); 0.02A (PZEM-004T-100A)

1.2.3 Resolution: 0.001A

1.2.4 Measurement accuracy: 0.5%

1.3 Active power

1.3.1 Measuring range: 0 ~ 2.3kW (PZEM-004T-10A); 0 ~ 23kW (PZEM-004T-100A)

1.3.2 Starting measure power: 0.4W

1.3.3 Resolution: 0.1W

1.3.4 Display format:

< 1000W, it display one decimal, such as: 999.9W

≥ 1000W, it display only integer, such as: 1000W

1.3.5 Measurement accuracy: 0.5%
1.4 Power factor

1.4.1 Measuring range: 0.00〜1.00
1.4.2 Resolution: 0.01
1.4.3 Measurement accuracy: 1%

1.5 Frequency

1.5.1 Measuring range: 45Hz〜65Hz
1.5.2 Resolution: 0.1Hz
1.5.3 Measurement accuracy: 0.5%

1.6 Active energy

1.6.1 Measuring range: 0〜9999.99kWh
1.6.2 Resolution: 1Wh
1.6.3 Measurement accuracy: 0.5%
1.6.4 Display format:

<10kWh, the display unit is Wh (1kWh=1000Wh), such as: 9999Wh

≥10kWh, the display unit is kWh, such as: 9999.99kWh

1.6.5 Reset energy: use software to reset.

1.7 Over power alarm

Active power threshold can be set, when the measured active power exceeds the threshold, it can alarm.

1.8 Communication interface

RS485 interface.

2 Communication protocol

2.1 Physical layer protocol

Physical layer use UART to RS485 communication interface
Baud rate is 9600, 8 data bits, 1 stop bit, no parity

2.2 Application layer protocol
The application layer use the Modbus-RTU protocol to communicate. At present, it only supports function codes such as 0x03 (Read Holding Register), 0x04 (Read Input Register), 0x06 (Write Single Register), 0x41 (Calibration), 0x42 (Reset energy), etc.

0x41 function code is only for internal use (address can be only 0xF8), used for factory calibration and return to factory maintenance occasions, after the function code to increase 16-bit password, the default password is 0x3721

The address range of the slave is 0x01 ~ 0xF7. The address 0x00 is used as the broadcast address, the slave does not need to reply the master. The address 0xF8 is used as the general address, this address can be only used in single-slave environment and can be used for calibration etc. operation.

2.3 Read the measurement result

The command format of the master reads the measurement result is (total of 8 bytes):

Slave Address + 0x04 + Register Address High Byte + Register Address Low Byte + Number of Registers High Byte + Number of Registers Low Byte + CRC Check High Byte + CRC Check Low Byte.

The command format of the reply from the slave is divided into two kinds:

Correct Reply: Slave Address + 0x04 + Number of Bytes + Register 1 Data High Byte + Register 1 Data Low Byte + ... + CRC Check High Byte + CRC Check Low Byte

Error Reply: Slave address + 0x84 + Abnormal code + CRC check high byte + CRC check low byte

Abnormal code analyzed as following (the same below)

- 0x01, Illegal function
- 0x02, Illegal address
- 0x03, Illegal data
- 0x04, Slave error

The register of the measurement results is arranged as the following table

<table>
<thead>
<tr>
<th>Register address</th>
<th>Description</th>
<th>Resolution</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x0000</td>
<td>Voltage value</td>
<td>1LSB correspond to 0.1V</td>
</tr>
<tr>
<td>0x0001</td>
<td>Current value <strong>low</strong> 16 bits</td>
<td>1LSB correspond to 0.001A</td>
</tr>
<tr>
<td>0x0002</td>
<td>Current value <strong>high</strong> 16 bits</td>
<td>1LSB correspond to 0.1W</td>
</tr>
<tr>
<td>0x0003</td>
<td>Power value <strong>low</strong> 16 bits</td>
<td>1LSB correspond to 0.1W</td>
</tr>
<tr>
<td>0x0004</td>
<td>Power value <strong>high</strong> 16 bits</td>
<td>1LSB correspond to 1Wh</td>
</tr>
<tr>
<td>0x0005</td>
<td>Energy value <strong>low</strong> 16 bits</td>
<td>1LSB correspond to 0.1Hz</td>
</tr>
<tr>
<td>0x0006</td>
<td>Energy value <strong>high</strong> 16 bits</td>
<td>1LSB correspond to 0.01</td>
</tr>
<tr>
<td>0x0007</td>
<td>Frequency value</td>
<td>1LSB correspond to 0.1Hz</td>
</tr>
<tr>
<td>0x0008</td>
<td>Power factor value</td>
<td>1LSB correspond to 0.01</td>
</tr>
<tr>
<td>0x0009</td>
<td>Alarm status</td>
<td>0xFFFF is alarm, 0x0000 is not alarm</td>
</tr>
</tbody>
</table>
For example, the master sends the following command (CRC check code is replaced by 0xHH and 0xLL, the same below)

\[0x01 + 0x04 + 0x00 + 0x00 + 0x00 + 0x0A + 0xHH + 0xLL\]

Indicates that the master needs to read 10 registers with slave address 0x01 and the start address of the register is 0x0000

The correct reply from the slave is as following:

\[0x01 + 0x04 + 0x14 + 0x08 + 0x98 + 0x03 + 0xE8+0x00 + 0x00 +0x08 + 0x98+ 0x00 + 0x00 + 0x00 + 0x00 + 0x00 + 0x00 + 0x00 + 0x00 + 0xHH + 0xLL\]

The above data shows

- Voltage is 0x0898, converted to decimal is 2200, display 220.0V
- Current is 0x000003E8, converted to decimal is 1000, display 1.000A
- Power is 0x00000898, converted to decimal is 2200, display 220.0W
- Energy is 0x00000000, converted to decimal is 0, display 0Wh
- Frequency is 0x01F4, converted to decimal is 500, display 50.0Hz
- Power factor is 0x0064, converted to decimal is 100, display 1.00
- Alarm status is 0x0000, indicates that the current power is lower than the alarm power threshold

### 2.4 Read and modify the slave parameters

At present, it only supports reading and modifying slave address and power alarm threshold

The register is arranged as the following table

<table>
<thead>
<tr>
<th>Register address</th>
<th>Description</th>
<th>Resolution</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x0001</td>
<td>Power alarm threshold</td>
<td>1LSB correspond to 1W</td>
</tr>
<tr>
<td>0x0002</td>
<td>Modbus-RTU address</td>
<td>The range is 0x0001~0x00F7</td>
</tr>
</tbody>
</table>

The command format of the master to read the slave parameters and read the measurement results are same (described in details in Section 2.3), only need to change the function code from 0x04 to 0x03.

The command format of the master to modify the slave parameters is (total of 8 bytes):

\[\text{Slave Address} + 0x06 + \text{Register Address High Byte} + \text{Register Address Low Byte} + \text{Register Value High Byte} + \text{Register Value Low Byte} + \text{CRC Check High Byte} + \text{CRC Check Low Byte}.\]

The command format of the reply from the slave is divided into two kinds:

Correct Response: \[\text{Slave Address} + 0x06 + \text{Number of Bytes} + \text{Register Address Low Byte} + \text{Register Value High Byte} + \text{Register Value Low Byte} + \text{CRC Check High Byte} + \text{CRC Check Low Byte}.\]

Error Reply: \[\text{Slave address} + 0x86 + \text{Abnormal code} + \text{CRC check high byte} + \text{CRC check low byte}.\]
For example, the master sets the slave's power alarm threshold:

0x01 + 0x6 + 0x00 + 0x01 + 0x08 + 0xFC + 0xHH + 0xLL

Indicates that the master needs to set the 0x0001 register (power alarm threshold) to 0x08FC (2300W).

Set up correctly, the slave return to the data which is sent from the master.

For example, the master sets the address of the slave

0x01 + 0x6 + 0x00 + 0x02 + 0x00 + 0x05 + 0xHH + 0xLL

Indicates that the master needs to set the 0x0002 register (Modbus-RTU address) to 0x0005

Set up correctly, the slave return to the data which is sent from the master.

### 2.5 Reset energy

The command format of the master to reset the slave's **energy** is (total 4 bytes):

Slave address + 0x42 + CRC check high byte + CRC check low byte.

Correct reply: slave address + 0x42 + CRC check high byte + CRC check low byte.

Error Reply: Slave address + 0xC2 + Abnormal code + CRC check high byte + CRC check low byte.

### 2.6 Calibration

The command format of the master to calibrate the slave is (total 6 bytes):

0xF8 + 0x41 + 0x37 + 0x21 + CRC check high byte + CRC check low byte.

Correct reply: 0xF8 + 0x41 + 0x37 + 0x21 + CRC check high byte + CRC check low byte.

Error Reply: 0xF8 + 0xC1 + Abnormal code + CRC check high byte + CRC check low byte.

It should be noted that the calibration takes 3 to 4 seconds, after the master sends the command, if the calibration is successful, it will take 3 ~ 4 seconds to receive the response from the slave.

### 2.7 CRC check

CRC check use 16bits format, occupy two bytes, the generator polynomial is X16 + X15 + X2 +1, the polynomial value used for calculation is 0xA001.

The value of the CRC check is a frame data divide all results of checking all the bytes except the CRC check value.

### 3 Functional block diagram
3.1 PZEM-004T-10A Functional block diagram

3.2 PZEM-004T-10A Functional block diagram

4 Wiring diagram

PZEM-004T-10A wiring diagram

U: AC80~260V
I: 0~10A

TTL to USB cable
PC

Picture 4.1 PZEM-004T-10A wiring diagram
5. Other instructions

5.1 The TTL interface of this module is a passive interface, it requires external 5V power supply, which means, when communicating, all four ports must be connected (5V, RX, TX, GND), otherwise it cannot communicate.

5.2 Working temperature

-20°C ~ +60°C.