

## Precautions before use

1. The equipment must work under safe operating conditions;
2. Please first understand the relevant contents of the user manual before turning on the power;
3. Any operation that violates the safety requirements of this manual is strictly prohibited.

Version format:2.2

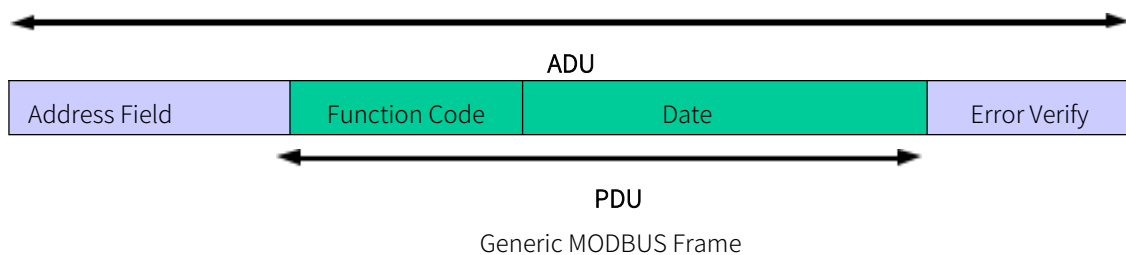
<b>Power Communication Manual .....</b>	<b>1</b>
1.Protocol Description .....	1
2.Function Code Definition .....	2
3.MODBUS Abnormal Response .....	6
4.Register Address Definition .....	7
5. Communication Control Mode .....	11
5.1 Operation Process Of Connecting Upper Computer Software .....	11
5.2 Example (The device address is 0x00) .....	11
5.3 Note .....	13

## Power Communication Manual

This communication adopts the MODBUS-RTU protocol format. The data communication adopts a master/slave mode, with the upper computer as the master and the equipment as the slave. The device will only send data to the upper computer after receiving valid commands from the upper computer. Serial port: RS232 level (or 485) 9600, N, 8, 1 (baud rate is subject to actual setting value).

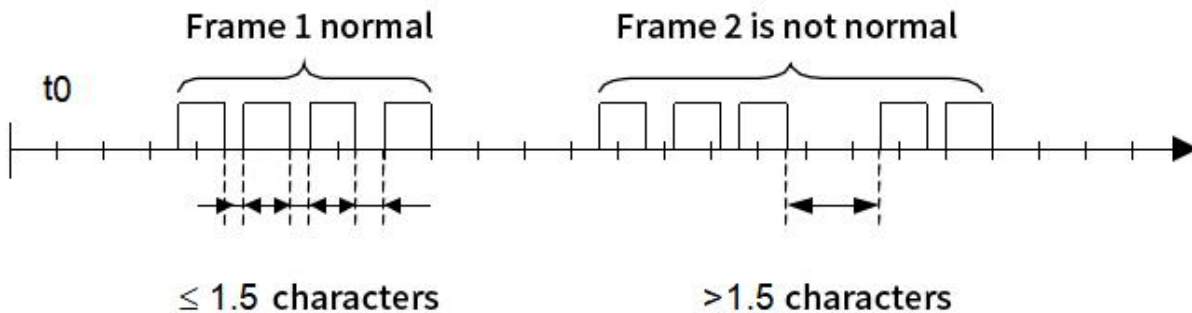
### 1. Protocol Description

The MODBUS protocol defines a simple protocol data unit (PDU) that is independent of the underlying communication layer. MODBUS protocol mapping on a particular bus or network can introduce additional domains on the application data Unit (ADU).



**The entire message frame must be sent in a continuous stream of characters.**

If the idle interval between two characters is greater than 1.5 characters, the message frame is considered incomplete and should be discarded by the receiving node..



## 2.Function Code Definition

		Function code	(Hexadecimal)
16-bits access	Read hold register	03	03
	Write a single register	06	06
	Write multiple registers	16	10

### 2.1 03 (0x03) Read Hold Register

In a remote device, the function code is used to read the contents of a continuous block of the hold register. The request PDU specifies the start register address and number of registers.

The register data in the response message is divided into two bytes per register, and the binary content is adjusted directly in each byte. For each register, the first byte includes the high bit bit, and the second byte includes the low bit bit.

#### Request PDU

Mailing address	1 byte	0x00
Function code	1 byte	0x03
Origin address	2 bytes	0x0000 to 0xFFFF
Number of registers	2 bytes	1 to 125 (0x7D)
CRC verify	2 bytes	Check code

#### Respond PDU

Mailing address	1 byte	0x00
Function code	1 byte	0x03
Number of bytes	1 byte	2×N*
Register value	N*×2 bytes	
CRC verify	2 bytes	Check code

\*N=The number of registers is incorrect.

Mailing address	1 byte	0x00
Error code	1 byte	0x83
Exception code	1 byte	01 or 02 or 03 or 04
CRC verify	2 bytes	Check code

This is an instance of requesting to read registers 107-109:

Request		Respond	
Domain name	(Hexadecimal)	Domain name	(Hexadecimal)
Mailing address	00	Mailing address	00
Function code	03	Function code	03
High origin address	01	Number of bytes	06
Low origin address	07	Register value Hi (107)	02
High register number	00	Register value Lo (107)	2B
Low register number	03	Register value Hi (108)	00
CRC verify the high bit	75	Register value Lo (108)	00
CRC verify the low bit	C6	Register value Hi (109)	00

		Register valueLo (109)	64
		CRC verify the high bit	08
		CRC verify the low bit	EA

Represents the contents of register 107 as two hexadecimal byte values 02 2B, or decimal 555. The contents of registers 108-109 are expressed as hexadecimal 00 00 and 00 64, or decimal 0 and 100, respectively.

## 2.2 06 (0x06) Write a single register

In a remote device, a single hold register is written using this function code.

The request PDU specifies the address of the register to be written. The normal response is the response to the request, and this normal response is returned after the contents of the register are written.

### Request PDU

Mailing address	1 byte	0x00
Function code	1 byte	0x06
Register address	2 bytes	0x0000 to 0xFFFF
Register value	2 bytes	0x0000 to 0xFFFF
CRC verify	1 byte	Check code

### Respond PDU

Mailing address	1 byte	0x00
Function code	1 byte	0x06
Register address	2 bytes	0x0000 to 0xFFFF
Register value	2 bytes	0x0000 to 0xFFFF
CRC verify	1 byte	Check code

### Error

Mailing address	1 byte	0x00
Error code	1 byte	0x86
Exception code	1 byte	01 or 02 or 03 or 04
CRC verify	1 byte	Check code

This is an instance of a request to write hexadecimal 0003 to register 0100:

Request		Respond	
Domain name	(Hexadecimal)	Domain name	(Hexadecimal)
Mailing address	00	Mailing address	00
Function code	06	Function code	06
Register address Hi	00	Register address Hi	00
Register address Lo	01	Register address Lo	01
Register value Hi	00	Register value Hi	00
Register value Lo	03	Register value Lo	03
CRC verify the high bit	C9	CRC verify the high bit	C9

CRC verify the low bit	E6	CRC verify the low bit	E6
------------------------	----	------------------------	----

### 2.3 16 (0x10) Write multiple registers

In a remote device, this function code is used to write consecutive register blocks (1 to about 120 registers). The value to which the request is written is described in the request data field. Each register divides the data into two bytes. The normal response returns the function code, the starting address, and the number of registers written.

#### Request PDU

Mailing address	1 byte	0x00
Function code	1 byte	0x10
Origin address	2 bytes	0x0000 to 0xFFFF
Number of registers	2 bytes	0x0001 to 0x0078
Number of bytes	1 byte	$2 \times N^*$
Register value	$N^* \times 2$ bytes	Value
CRC verify	2 bytes	Check code

\*N=Number of registers

#### Respond PDU

Mailing address	1 byte	0x00
Function code	1 byte	0x10
Origin address	2 bytes	0x0000 to 0xFFFF
Number of registers	2 bytes	1 to 123 (0x7B)
CRC verify	2 bytes	Check code

#### Error

Mailing address	1 byte	0x00
Error code	1 byte	0x90
Exception code	1 byte	01 or 02 or 03 or 04
CRC verify	1 byte	Check code

This is an instance of a request to write hexadecimal 000A and 0102 to two registers starting with 0100:

Request		Respond	
Domain name	(Hexadecimal)	Domain name	(Hexadecimal)
Mailing address	00	Mailing address	00
Function code	10	Function code	10
Origin address Hi	01	Origin address Hi	01
Origin address Lo	00	Origin address Lo	010
Number of registers Hi	00	Number of registers Hi	00
Number of registers Lo	02	Number of registers Lo	02



## AC Power Communication Instructions

Number of bytes	04	CRC verify the high bit	41
Register value Hi	00	CRC verify the low bit	E5
Register value Lo	0A		
Register value Hi	01		
Register value Lo	02		
CRC verify the high bit	5A		
CRC verify the low bit	90		

### 3.MODBUS Abnormal Response

MODBUS Exception Code		
Code	Name	Meaning
01	Illegal function	For the server (or slave station), the function code received in the query is an impermissible operation.This may be because function code only applies to new equipment, and in the selected unit is impossible of achievement.At the same time, it is also indicated that the server (or slave station) processes this request in an error state, for example, because it is unconfigured and requires a return register value.
02	Illegal data address	For the server (or slave station), the data address received in the query is an inadmissible address.In particular, the combination of reference number and transmission length is not valid.For a controller with 100 registers, a request with offset 96 and length 4 will succeed, and a request with offset 96 and length 5 will produce exception code 02.
03	Illegal data value	For server (or slave station), the values included in the query are not allowed values.This value indicates a fault in the remaining structure of the combined request, for example: the implied length is incorrect.It does not mean that, because the MODBUS protocol does not know the significance of any special value of any special register, the data item being committed to storage in the register has a value that is not expected by the application.
04	Slave station equipment failure	An unrecoverable error occurs when the server (or slave station) is trying to perform the requested operation.
05	Verify	Used with programming commands.The server (or slave station) has accepted the request and is processing it, but long durations are required for these operations.Returning this response prevents a timeout error from occurring in the client (or master).The client (or master) can continue to send a polling completion message to determine whether the processing is complete.
06	Slave unit busy	Used with programming commands. The server (or slave station) is processing a long duration program command. When the server (or slave station) is idle, the user (or master) should retransmit the message later.

## 4.Register Address Definition

(W indicates the write command and R indicates the read command)

0x0100	Mailing Address	W/R	0-255
0x0101	Baud rate	W/R	(0)1200, (1)4800, (2)96000, (3)19200, (4)38400, (5)57600, (6)115200
0x0102	Power output setting	W	0x00- Standby state, 0x01- Output state
0x0103	Power state	R	0x00- Standby state, 0x01- Output state, 0xf0-Overcurrent protection, 0xf1-Overheat protection (representing failure)
0x0105	Power on self start	W/R	0x00- Indicates off, 0x01- Indicates open
0x0106	Power grade place	W/R	0- Automatic grade place, 1- Low grade place, 2- High grade place
0x0107	Overcurrent value setting	W/R	Set the overcurrent protection value, ranging from 0.5-600.00(0 to 600A), the resolution is 0.1A
0x0108	Reset	W	1 - reset
0x0109	Remote handover	W	1- local; 2 -remote
Reserve			
0x0123	Output start phase		
0x0124	Output stop phase		
Reserve			
0x0130	Frequency set high 16 bits	W/R	A total of 32 bits of data, the setting range is 0-999999.99(0 to 999KHZ), the resolution is 0.01Hz, if the setting is 100KHZ, 0x0130 is 0x0098,0x0131 is 0x9680
0x0131	Frequency set low 16 bits	W/R	
0x0132	Frequency low end restriction high 16 bit	W/R	Set the range 0 ~ 90%
0x0133	Frequency low end restriction low 16 bit	W/R	Set the range 0 ~ 90%
0x0134	Frequency high end restriction high 16 bit	W/R	Set the range 10% ~ the maximum value
0x0135	Frequency high end restriction low 16 bit	W/R	Set the range 10% ~ the maximum value
Reserve			
0x0140	Amplitude set high 16 bits	W/R	A total of 32 bits of data, set 0-9999.99(0 to 9999V), resolution is 0.01V, if set 750V, 0x0140 is 0x0001,0x0141 is 0x24F8
0x0141	Amplitude set low 16 bits	W/R	
0x0143	Voltage slow rise	W/R	Set the range of 0.1 to 9999.9, the resolution of 0.1 V/mS
0x0145	Voltage slow drop	W/R	Set the range of 0.1 to 9999.9, the resolution of 0.1 V/mS
0x0146	Voltage low end restriction high 16 bit	W/R	A total of 32 bits of data, set 0-9999.99(0 to 9999V), resolution is 0.01V, if set 750V, 0x0146 is 0x0001,0x0147 is 0x24F8
0x0147	Voltage low end restriction low 16 bit	W/R	
0x0148	Voltage high end restriction	W/R	A total of 32 bits of data, set 0-9999.99(0 to 9999V),



	high 16 bit		resolution is 0.01V, if set 750V, 0x0148 is 0x0001, 0x0149 is 0x24F8
0x0149	Voltage high end restriction low 16 bit	W/R	
Reserve			
0x0201	Voltage measurement value high 16 bits	R	Value range 0-1000000, unit depends on power supply (decimal places) <sup>[2]</sup>
0x0202	Voltage measurement value low 16 bits	R	Value range 0-1000000, unit depends on power supply (decimal places) <sup>[2]</sup>
0x0207	Current measurement value high 16 bits	R	Value range 0-1000000, unit depends on power supply (decimal places) <sup>[2]</sup>
0x0208	Current measurement value low 16 bits	R	Value range 0-1000000, unit depends on power supply (decimal places) <sup>[2]</sup>
Reserve			
0x1000	Step step 0 Settings	W/R	Frequency high 16 bits
0x1001	Step step 0 Settings	W/R	Frequency low 16 bits
0x1006	Step step 0 Settings	W/R	Voltage setting
0x1009	Step step 0 Settings	W/R	Hour
0x100A	Step step 0 Settings	W/R	Minute
0x100B	Step step 0 Settings	W/R	Second
0x100C	Step step 0 Settings	W/R	Millisecond
0x100D	Step step 0 Settings	W/R	Microsecond
0x100E	Step step 1 Settings	W/R	Frequency high 16 bits
Step are 40 steps in total, and each step accounts for 14 registers (frequency settings account for two registers and the rest account for one register)			
Reserve			
0x1F01	Start step	W/R	Set the range 0-39, the start step number must be less than the end step number
0x1F02	End step	W/R	Set the range 0-39, the end step number must be greater than the start step number
0x1F03	Cycle index	W/R	Set the range 0-999, with 0 indicating an infinite loop
Reserve			
0x2000	Ladder setting	W/R	Start frequency high 16 bits
0x2001	Ladder setting	W/R	Start frequency low 16 bits
0x2002	Ladder setting	W/R	Stepping frequency high 16 bits
0x2003	Ladder setting	W/R	Stepping frequency low 16 bits
0x200C	Ladder setting	W/R	Start voltage
0x200D	Ladder setting	W/R	Stepping voltage
0x2012	Ladder setting	W/R	Stepping number
0x2013	Ladder setting	W/R	Hour
0x2014	Ladder setting	W/R	Minute
0x2015	Ladder setting	W/R	Second
0x2016	Ladder setting	W/R	Millisecond
0x2017	Ladder setting	W/R	Microsecond
Ladder settings account for a total of 24 registers (frequency settings account for two registers and the rest account for one register)			

Reserve			
0x2F00	Ladder mode setting	W/R	0: Three-phase standard (unified setting of frequency and voltage); 1: Three-phase independent adjustable. <b>Note: When the three-phase standard is used, the actual three-phase output is the A-phase setting.</b>
0x2F01	Cycle index	W/R	Set the range 0-999, with 0 indicating an infinite loop
Reserve			
0x3000	Gradient step 0 Settings	W/R	Start frequency high 16 bits
0x3001	Gradient step 0 Settings	W/R	Start frequency low 16 bits
0x3002	Gradient step 0 Settings	W/R	End frequency high 16 bits
0x3003	Gradient step 0 Settings	W/R	End frequency low 16 bits
0x300C	Gradient step 0 Settings	W/R	Start voltage
0x300D	Gradient step 0 Settings	W/R	End voltage
0x3012	Gradient step 0 Settings	W/R	Hour
0x3013	Gradient step 0 Settings	W/R	Minute
0x3014	Gradient step 0 Settings	W/R	Second
0x3015	Gradient step 0 Settings	W/R	Millisecond
0x3016	Gradient step 0 Settings	W/R	Microsecond
0x3017	Reserve		
0x3018	Gradient step 1 Settings	W/R	Start frequency high 16 bits
<b>Gradient are 40 steps in total, and each step accounts for 24 registers (frequency settings account for two registers and the rest account for one register)</b>			
Reserve			
0x3F01	First step	W/R	Set the range 0-39, the start step number must be less than the end step number
0x3F02	End step	W/R	Set the range 0-39, the end step number must be greater than the start step number
0x3F03	Cycle index	W/R	Set the range 0-999, with 0 indicating an infinite loop
Reserve			

## Remark:

1. Marked <sup>[1]</sup> symbol indicates: Some power supplies do not support this function.
2. Marked <sup>[2]</sup> symbol indicates: If the accuracy is two decimal set the value or the measured value is 6000, this value represents 60.00 (V/A/W). If the accuracy is one decimal set the value or the measured value is 6000, this value represents 600.0 (V/A/W).

## Note:

- 1.Register address is the company unified allocation, different products have different registers.Do not use registers that do not belong to the purchased product to avoid damage to the device.
- 2.When the device is in the high and low gear switch, the device will delay responding to the command.The specific delay time is random, but does not exceed the high and low switching time (high and low switching time varies according to different products, generally between 1.5-4 seconds).
- 3.In the power output state, except for the power supply voltage and current are not allowed to set.
- 4.Each instruction cannot exceed 100 bytes.

5. When the power supply is in local control mode, the remote end is unable to control the device. If this time the control device will receive the device busy instruction, it needs to switch to the remote control mode manually or by using the command.

## 5. Communication Control Mode

### 5.1 Operation Process Of Connecting Upper Computer Software

- 1、Power on (The power supply is in ordinary mode)
- 2、directly set the parameters of the mode you want to run.(Or read whether the parameters of the mode to be run are synchronized with the upper computer)
- 3、Switch to the mode you need to run.
- 4、Power supply output.

### 5.2 Example (The device address is 0x00)

Instruction format

0x10 Write instruction

Send	Device address	Function code	Register address	Number of registers written	Write bytes	Write data	CRC16
byte	1 byte	1 byte	2 bytes	2 bytes	1 byte	N bytes	2 bytes
Back	Device address	Function code	Register address	Number of registers written	CRC16		
byte	1 byte	1 byte	2 bytes	2 byte	2 bytes		

0x03 Read instruction

Send	Device address	Function code	Register address	Number of read registers	CRC16
byte	1 byte	1 byte	2 bytes	2 bytes	2 bytes
Back	Device address	Function code	Read bytes	Read data	CRC16
byte	1 byte	1 byte	1 byte	N bytes	2 bytes

#### 1、 Ordinary mode

##### (1) Power on and wait for the device to start.

At this time, the device can be controlled locally or by the upper computer, using the preemption mechanism, who controls the device first, the device will switch to the corresponding control mode.If the device is under local control and the upper computer controls the device, the upper computer receives the device busy instruction.

**Switch to the upper computer control mode**

Send: 00 10 01 13 00 01 02 00 02 38 62 (0x0113- Local or Remote Control Register 0x0002- Remote Control)

Back: 00 10 01 13 00 01 F0 21

##### (2) Read the set value and synchronize the data of the host computer

**Read frequency setting value (takes up two registers)**

Send: 00 03 01 30 00 02 C4 29 (Read two register values starting at address 0x0130)

Back: 00 03 04 00 01 86 A0 DP 28 (0x0001 0x86A0 decimalism 100000 representative 1000.00Hz)

**Read voltage setting value (takes up two registers)**

Send: 00 03 01 40 00 02 C5 F2 (Read two register values starting at address 0x0140)

Back: 00 03 04 00 00 2E E0 F6 DB (0x2EE0 decimalism 12000 representative 120.00V)

##### (3) Set the required setting value

**Set frequency (takes up two registers)500.00Hz (0x0000 0xC350)**

Send: 00 10 01 30 00 02 04 00 00 C3 50 A9 1B (Write 2 registers starting at 0x0130 and 4 bytes of data is 0x0000 0xC350)

Back: 00 10 01 30 00 02 41 EA (Successfully write 2 registers starting from 0x0130)

**Set voltage (takes up two registers) 30.00V (0x0BB8)**

Send: 00 10 01 40 00 02 04 00 00 0B B8 F9 B1 (Write 2 registers starting at 0x0140 and 4 bytes of data is 0x0000, 0x0BB8)

Back: 00 10 01 40 00 02 40 31 (Successfully write 2 registers starting from 0x0140)

#### (4) Voltage output control

**Turn on power output**

Send: 00 10 01 02 00 01 02 00 01 7B 22 (Write 1 register starting at 0x0102 2 bytes of data to 0x0001)

Back: 00 10 01 02 00 01 A0 24 (Successfully write 2 registers starting from 0x0102)

**Turn off power output**

Send: 00 10 01 02 00 01 02 00 00 BA E2 (Write 1 register starting at 0x0102 and 2 bytes of data is 0x0000)

Back: 00 10 01 02 00 01 A0 24 (Successfully write 2 registers starting from 0x0102)

## 2. Gradient mode

(1)Power on (The device is in ordinary output mode)

(2)The device switches to the remote control mode

(3)Set the gradient data (assuming you want to run 0-2 step cycle 2 times three-phase independent adjustable mode)

**Set gradient mode to three-phase independent mode**

Send: 00 10 3F 00 00 01 02 00 01 A5 03

Back: 00 10 3F 00 00 01 0C 0C

**Set 0 step data (address 0x3000)**

The ABC frequency was 60Hz and the voltage gradient time from 0 to 30V was 2 seconds. The start and stop frequency is set to 60Hz, the start voltage is 0V, the end voltage is 30V, and the gradient time is 0 hours, 0 minutes, 2 seconds, 0 milliseconds, 0 microseconds.

Send: 00 10 30 00 00 17 2E 00 00 17 70 00 00 17 70 00 00 17 70 00 00 17 70 00 00 17 70 00 00 17 70 00 00 0B B8 00 00 0B B8 00 00 0B B8 00 00 00 00 02 00 00 00 00 A3 61

Back: 00 10 30 00 00 17 8E D6

(4)Set the number of start and stop steps and the number of cycles

**Set it to run 3 times from 0 to 2**

Send: 00 10 3F 01 00 03 06 00 00 00 02 00 03 9B 11

Back: 00 10 3F 01 00 03 DC 0D

(5)Switch to gradient output mode

**Mode switching**

Send: 00 10 01 11 00 01 02 00 04 B9 82

Back: 00 10 01 11 00 01 51 E1

Power output control

**Open output**

Send: 00 10 01 40 00 02 04 00 00 0B B8 F9 B1

Back: 00 10 01 02 00 01 A0 24

### 5.3 Note

Because step and step and gradient controls are similar, this article is not intended to describe them.

The upper computer can operate on successive registers, but note that an instruction is not allowed to exceed 100 bytes in length.

The output is stopped when the step and gradient run cycles are completed, and the output of the last step is maintained after the ladder run. Our equipment will not actively report the operation completion instruction after running, and we need to determine whether the operation is completed by querying the voltage and current. This is to prevent communication chaos during 485 networking.

Only when the device is faulty, it automatically reports a command. (Function code is 0xFF)