E850-DTU(4440-GPRS)

User Manual

This manual may change with the continuous improvement of the product. Please refer to the latest version of the instruction.

Feature

- Support 4-channel analog input, default current detection
- Support 4-channel switch input, default dry contact
- Support 4-channel relay output
- Support socket connection to remote server, support TCP Client
- Support Modbus TCP/RTU protocol
- Support Ebyte cloud, controlled by command
- Support 2 working mode, master and slave, slave can cascade multiple devices through RS485
- Support reload button to restore factory settings, tap the button and press for 5 seconds, the Modbus device address、RS485 baud rate and parity bit restore factory settings
- Hardware watchdog with high reliability
- Multiple indicators to show working status
- The power supply has good functions such as overcurrent, overvoltage and anti-reverse connection

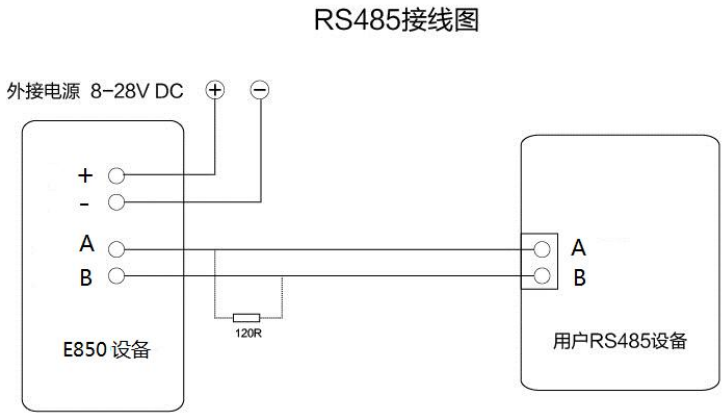
Note: Products can be customized, such as conditional control (how to output based on input state)

1. Quick start

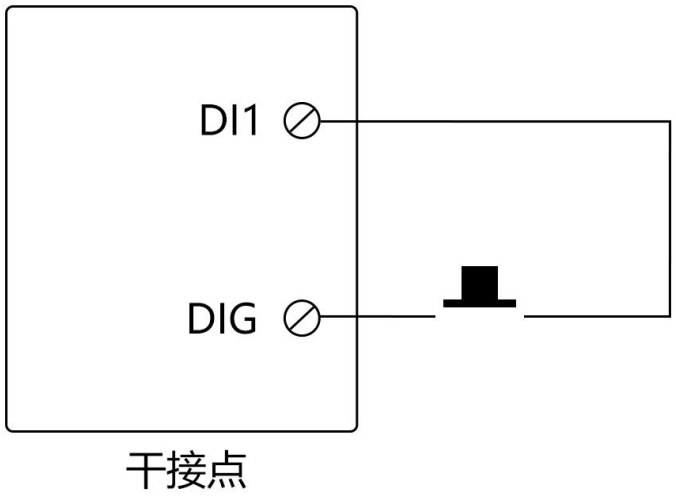
This chapter is a quick introduction to the E850-DTU (4440-GPRS) series. It is recommended that users read this chapter carefully and follow the instructions. It will have a systematic understanding of the product, and users can also select the chapters of interest as needed. . For specific details and instructions, please refer to the following sections.

1.1 Connection

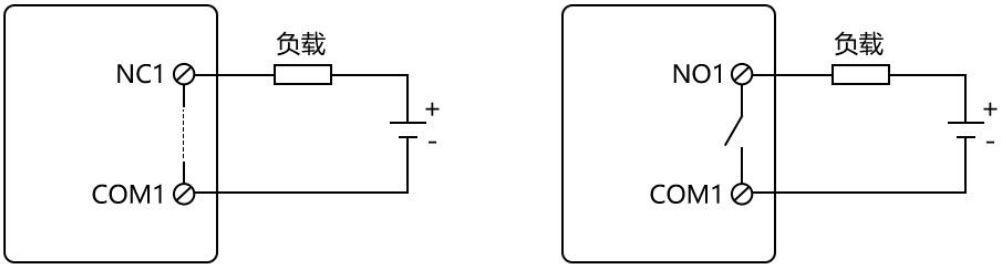
1.1.1 RS485 connection



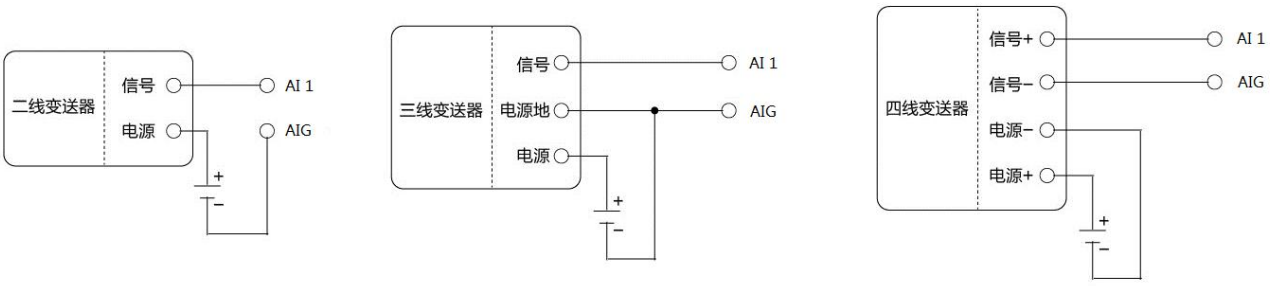
1.1.2 switch input connection



1.1.3 Relay output connection



1.1.4 Analog input connection



模拟量输入连接

1.2 Quick instruction

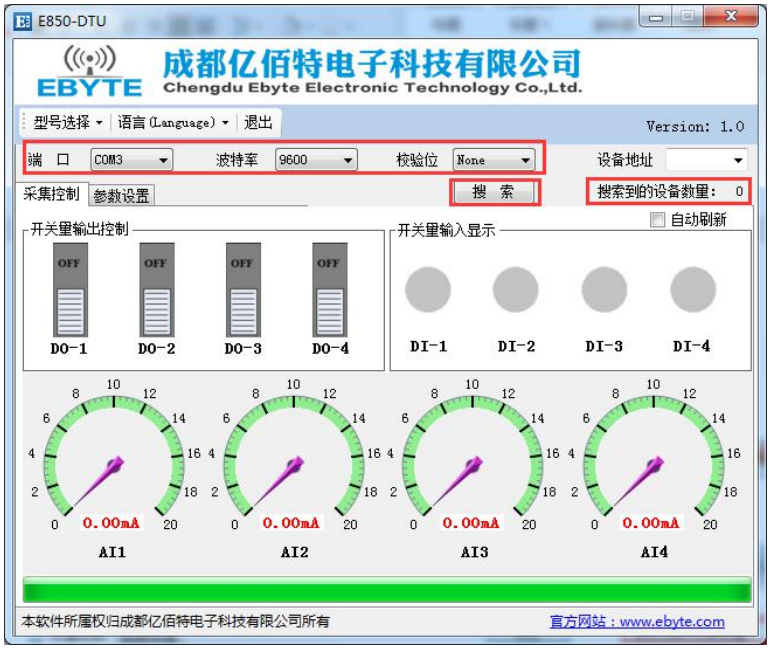
Cable connection: The computer connects to the E850-DTU (4440-GPRS) via USB to RS485.

Network connection: Insert SIM card in power-off state.

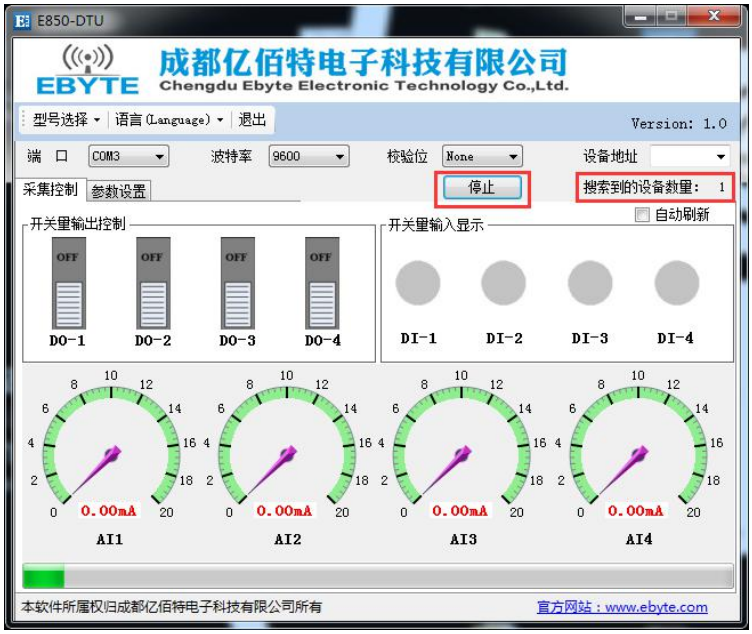
Power supply: E850-DTU (4440-GPRS) working voltage is DC 8-28V.

1.2.1 RS485 control

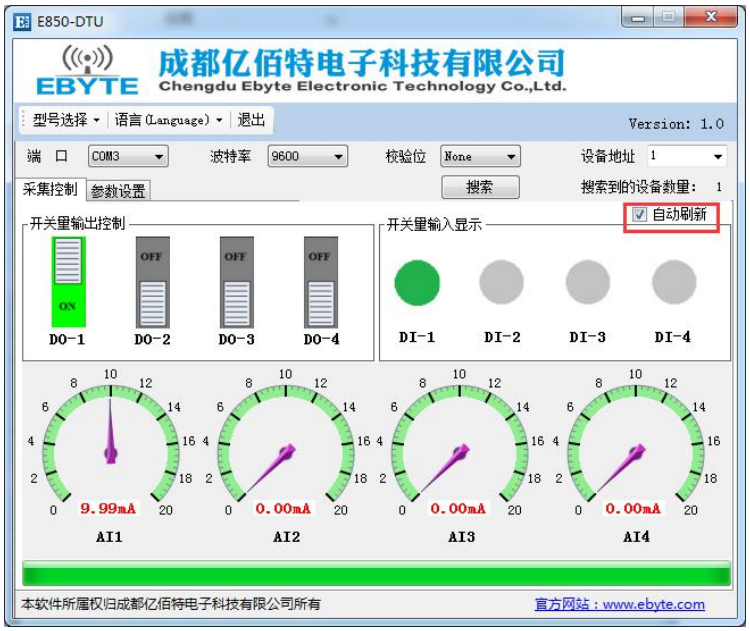
Select the port, click “Search”, and search the device.



After the device is searched, click "Stop".

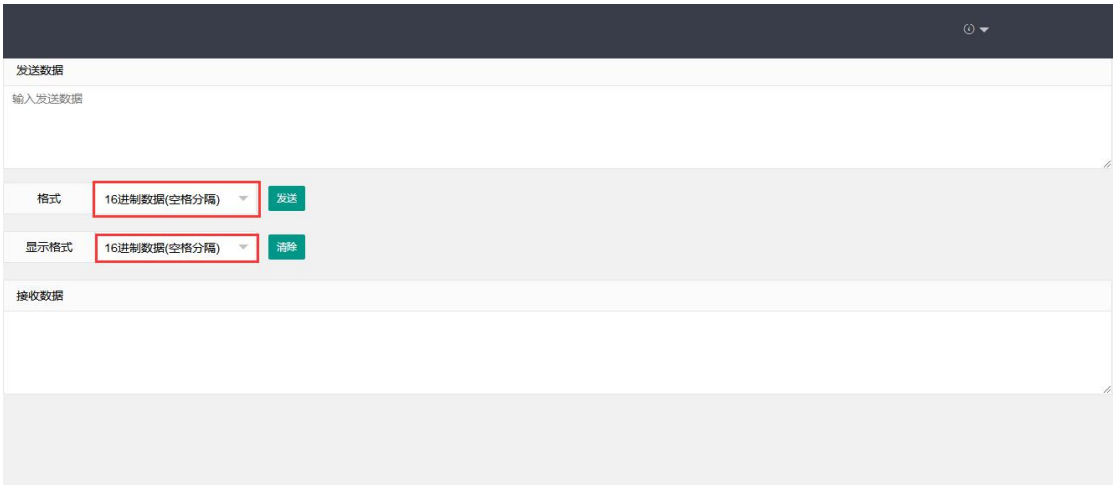


At this time, you can see the current device address, for the "auto refresh" process, you can perform the switch output control, the switch input read, the analog input read.



1.2.2 Network control

At the website: http://yun.cdebyte.com/www/data_direct， The acquisition can be controlled by commands, and the command supports the Modbus TCP/RTU protocol.



2. Introduction

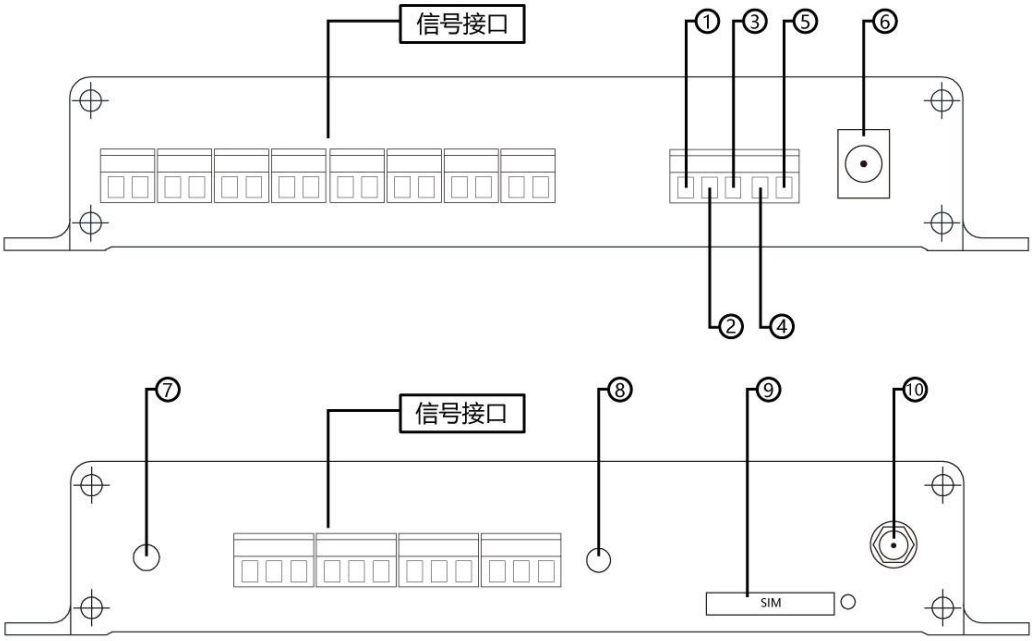
E850-DTU (4440-GPRS) is a network IO product that supports 4 switch inputs (default dry contact), 4 analog inputs, and 4 relay outputs. Support Modbus TCP/RTU protocol. The product is highly easy to use and can be easily and quickly integrated into your system for GPRS-based remote control.

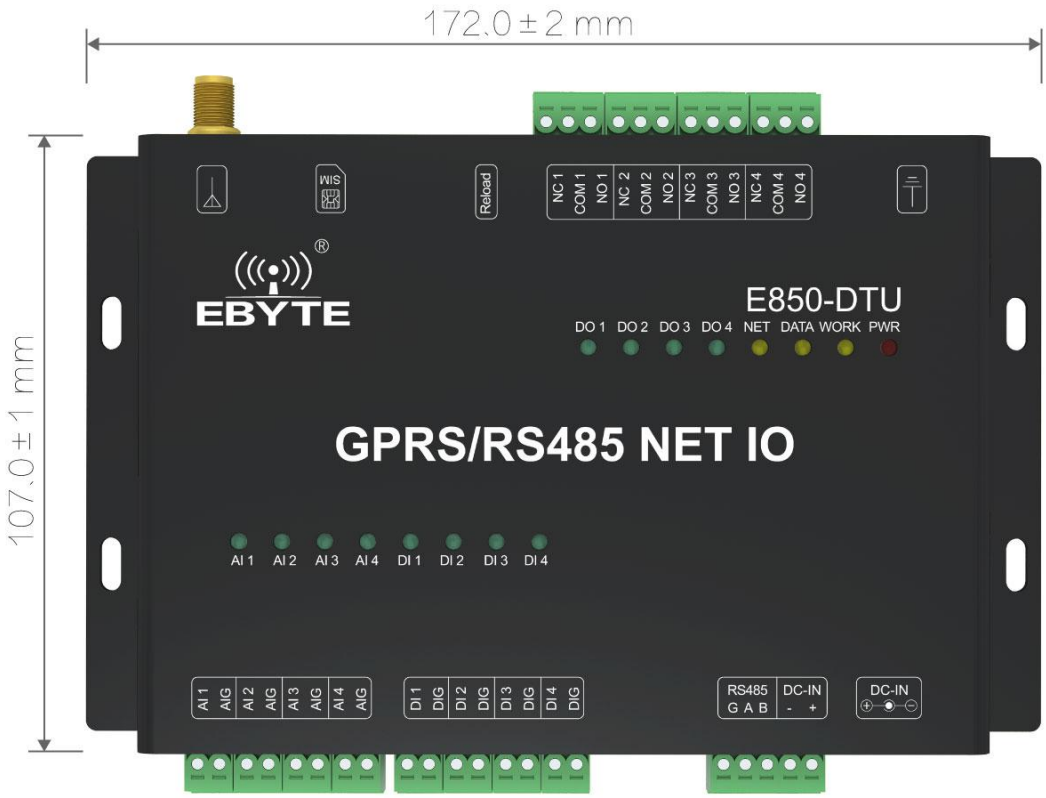
2.1 Parameters

	Item	Description
Wireless parameter	Standard frequency band	850/900/1800/1900MHz
Hardware parameter	Size (H*W*D)	172*107*29mm
	Weight	-
	Working temperature	-20℃~+70℃
	Storage temperature	-40℃~+85℃
	Working humidity	5%~95%
	Storage humidity	1%~95%
	Working voltage	8V~28V
	Current collection range	0mA~20mA

	Accuracy	0.2%
	Data interface	RS485：1200~115200bps
Software parameter	Network type	GPRS
	Configuration command	Modbus TCP/RTU
	Network protocol	Modbus TCP/RTU
	Working mode	Master mode, slave mode
	Data transmission mode	TCP Client

2.2 Size and interface





No.	Definition	Function	Description
1	RS485 G	Signal ground	Signal ground, can be no onnected
2	RS485 A	RS485 A	RS485 A connects with device A
3	RS485 B	RS485 B	RS485 B connects with device B
4	DC-IN -	Power input negative	Power ground
5	DC-IN +	Power input positive	Power input, DC 8V~28V, recommended 12V/24V
6	DC-IN	DC 5.5*2.1mm	Power input, DC 8V~28V, recommended 12V/24V
7	Ground screw	Ground	Ground
8	Reload	Reset button	Long press 5s to work
9	SIM Card slot	SIM Card slot	Insert SIM card to connect network
10	ANT	Antenna interface	GPRS antenna
Signal interface			
1	AI 1	Analog input 1	Analog input pin, forming an input with AIG
2	AIG	Analog input ground	Can be paired with AI 1
3	AI 2	Analog input 2	Analog input pin, forming an input with AIG
4	AIG	Analog input ground	Can be paired with AI 2
5	AI 3	Analog input 3	Analog input pin, forming an input with AIG
6	AIG	Analog input ground	Can be paired with AI 3
7	AI 4	Analog input 4	Analog input pin, forming an input with AIG
8	AIG	Analog input ground	Can be paired with AI 4
9	DI 1	switch input 1	Forming dry contact with DIG
10	DIG	switch input ground	Can be paired with DI 1
11	DI 2	switch input 2	Forming dry contact with DIG

12	DIG	switch input ground	Can be paired with DI 2
13	DI 3	switch input 3	Forming dry contact with DIG
14	DIG	switch input ground	Can be paired with DI 3
15	DI 4	switch input 4	Forming dry contact with DIG
16	DIG	switch input ground	Can be paired with DI 4
17	NC 1	Relay 1 normally closed pin	Can be paired with common end of relay 1
18	COM 1	Common end of relay 1	Can be paired with other pins of relay 1
19	NO 1	Relay 1 normally open pin	Can be paired with common end of relay 1
20	NC 2	Relay 2 normally closed pin	Can be paired with common end of relay 2
21	COM 2	Common end of relay 2	Can be paired with other pins of relay 2
22	NO 2	Relay 2 normally open pin	Can be paired with common end of relay 2
23	NC 3	Relay 3 normally closed pin	Can be paired with common end of relay 3
24	COM 3	Common end of relay 3	Can be paired with other pins of relay 3
25	NO 3	Relay 3 normally open pin	Can be paired with common end of relay 3
26	NC 4	Relay 4 normally closed pin	Can be paired with common end of relay 4
27	COM 4	Common end of relay 4	Can be paired with other pins of relay 4
28	NO 4	Relay 4 normally open pin	Can be paired with common end of relay 4
LED			
1	AI 1	Analog input channel 1 indication	Green LED, large enough input (≥0.5mA) lights up
2	AI 2	Analog input channel 2 indication	Green LED, large enough input (≥0.5mA) lights up
3	AI 3	Analog input channel 3 indication	Green LED, large enough input (≥0.5mA) lights up
4	AI 4	Analog input channel 4 indication	Green LED, large enough input (≥0.5mA) lights up
5	DI 1	switch input channel 1 indication	Green LED, lights up when DI 1 and DIG are shorted
6	DI 2	switch input channel 2 indication	Green LED, lights up when DI 2 and DIG are shorted
7	DI 3	switch input channel	Green LED, lights up when DI 3 and DIG are shorted

		3 indication	
8	DI 4	switch input channel 4 indication	Green LED, lights up when DI 4 and DIG are shorted
9	DO 1	Relay 1 output indication	Green LED, lights up when NO 1 and COM 1 close
10	DO 2	Relay 2 output indication	Green LED, lights up when NO 1 and COM 1 close
11	DO 3	Relay 3 output indication	Green LED, lights up when NO 1 and COM 1 close
12	DO 4	Relay 4 output indication	Green LED, lights up when NO 1 and COM 1 close
13	NET	Network indication	Yellow LED, Constantly bright after entering network
14	DATA	Serial data indication	Yellow LED, Lights up when the RS485 interface has data transmission (blinking)
15	WORK	Work/Reset indication	Yellow LED, Regular blinking / quick blinking after successful reset
16	PWR	Power indication	Red LED, Constantly bright

Note: Grounding: It is recommended to connect the case to the ground

2.3 Reload button description

Long press 5S is valid, after the reset is successful, the WORK led blinks quickly, and the Modbus device address, RS485 serial port baud rate and parity bit are restored to the factory settings.

3. Modbus

3.1 Register Address Table

Register Address Table（Function code: 0x01H、0x05H、0x0FH、0x03H、0x06H、0x10H)					
Register address	Number of registers	Register properties	Register type	Register value range	Function code supported
00017 (0x0010)	1	DO1 switch output	Read/Write	0x0000 or 0xFF00（0x05 function code）0-1（0x01、0x0F function code）	0x01 0x05 0x0F
00018 (0x0011)	1	DO2 switch output	Read/Write		
00019 (0x0012)	1	DO3 switch output	Read/Write		
00020 (0x0013)	1	DO4 switch output	Read/Write		
Reserve					
10017 (0x0010)	1	DI1 switch intput	Read only	0-1	0x02
10018 (0x0011)	1	DI2 switch intput	Read only		

10019 (0x0012)	1	DI3 switch intput	Read only		
10020 (0x0013)	1	DI4 switch intput	Read only		
30017 (0x0010)	1	AI1 input value, unit (uA)	Read only	0-20000	0x03 0x04
30018 (0x0011)	1	AI2 input value, unit (uA)	Read only		
30019 (0x0012)	1	AI3 input value, unit (uA)	Read only		
30020 (0x0013)	1	AI4 input value, unit (uA)	Read only		
Reserve					
40049 (0x0030)	1	DI1 Pulse counting value	Read only	0-65535	0x03
40050 (0x0031)	1	DI2 Pulse counting value	Read only	0-65535	
40051 (0x0032)	1	DI3 Pulse counting value	Read only	0-65535	
40052 (0x0033)	1	DI4 Pulse counting value	Read only	0-65535	
Reserve					
40065 (0x0040)	1	DI1-DI4 Pulse counting to zero	Write only	0x00 - 0x0F	0x06
Reserve					
40078 (0x004D)	1	Device address	Read/Write	1 - 247	0x03 0x06 0x10
40079 (0x004E)	1	Baud rate	Read/Write	0 - 7	
40080 (0x004F)	1	Parity bit	Read/Write	0 - 2	
40081(0x0050)	1	Master or slave mode	Read/Write	0 - 1	
40082 (0x0051)	1	Automatic reporting of switch quantity	Read/Write	0 - 2	
40083 (0x0052)	1	switch output time setting (ms)	Read/Write	300-65535	
40084(0x0053)	1	Analog quantity range setting	Read/Write	0 - 1	
40085 (0x0054)	1	switch quantity restart output state setting	Read/Write	0x00 - 0x10	
Reserve					
40100 (0x0063)	22	Server IP or domain name (domain name is ASCII)	Read/Write	--	0x03 0x06 0x10
40122 (0x0079)	1	Server port	Read/Write	1 - 65535	
40123 (0x007A)	1	Protocol type (UDP、TCP)	Read/Write	0 - 1	
40124(0x007B)	22	User defineddd package	Read/Write	--	
40146(0x0091)	1	Registration package mechanism	Read/Write	0 - 4	
40147(0x0092)	22	Heartbeat package	Read/Write	--	
40169 (0x00A8)	1	Heartbeat package time	Read/Write	0 - 65535	
40170	1	Cloud transparent	Read/Write	0 - 1	

(0x00A9)		transmission			
40171 (0x00AA)	11	IMEI value	Read only	--	0x03
40182 (0x00B5)	11	SN value	Read only	--	
40193 (0x00C0)	20	LBS	Read only	--	
Reserve					
40300(0x012B)	1	Version number	Read only	--	0x03

3.2 Modbus address table

Modbus address table	
1 (default)	1
2	2
3	3
…	…
245	245
246	246
247	247

3.3 RS485 Serial port baud rate value table

Serial port baud rate value table	
0	1200
1	2400
2	4800
3 (default)	9600
4	19200
5	38400
6	57600
7	115200

3.4 RS485 Serial port parity bit value table

Serial port parity bit value table	
0 (default)	No parity
1	Even parity
2	Odd parity

3.5 Configure parameters through the host computer

Select “Parameter Setting” to read parameters and write parameters. For specific functions, see the product function description below.

4. Function

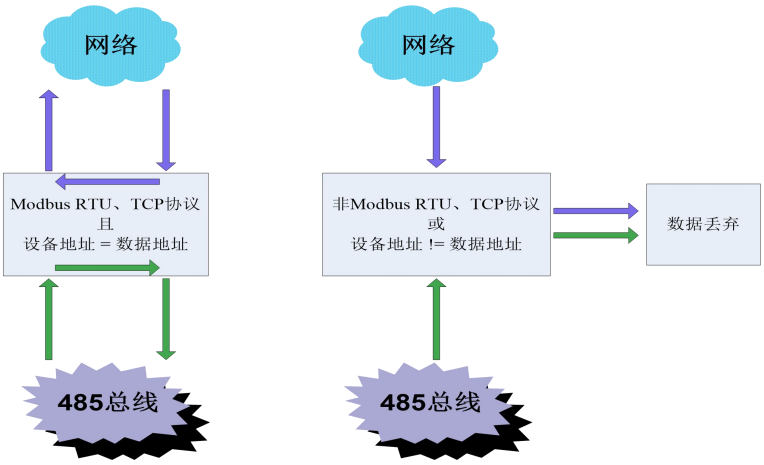
4.1 Working mode

The working mode is master mode and slave mode, configured by Modbus register 40081 (0x0050). When the register value is 0, it is in master mode; when the register value is 1, it is slave mode, the default slave mode.

4.1.1 Slave mode

In slave mode (register value 0x01), the data sent to the device by the network or 485 bus (sender) conforms to Modbus RTU, Modbus TCP protocol, and the address in the data is the device address, and the device will reply with the same protocol. If the data sent by the network or 485 bus to the device does not comply with the Modbus RTU, Modbus TCP protocol, or even if the Modbus RTU and Modbus TCP protocols are met, but the data address is not the device address, the data will be discarded.

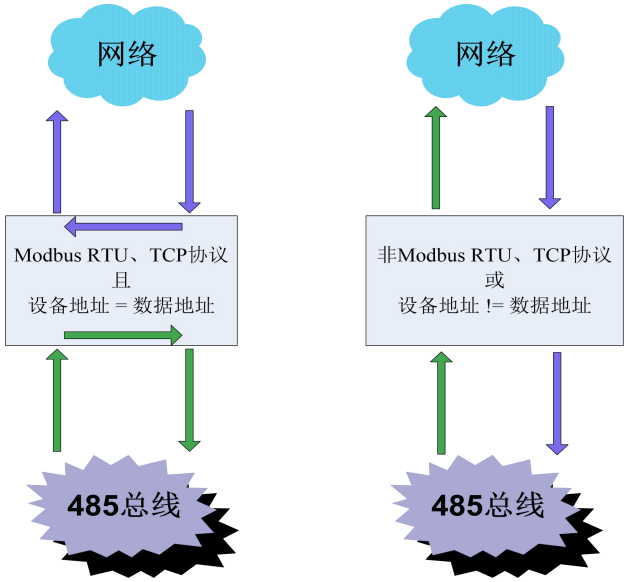
In the slave mode, the device can directly connect to the device in the master mode through the 485 bus. When the slave is not connected to the network, the network can also access the data of the slave through the master.



4.1.2 Master mode

In Master mode (register value is 0x00), the data sent by the network or 485 bus (sender) to the device conforms to the Modbus RTU, Modbus TCP protocol, and the address in the data is the device address, and the device responds to the sender with the same protocol. If the data sent by the network or 485 bus to the device does not comply with the Modbus RTU, Modbus TCP protocol, or even if the Modbus RTU and Modbus TCP protocols are met, but the data address is not the device address, the data transmitted from the 485 bus will be transmitted to the network, the data transmitted from the network will be transmitted to the network 485 bus.

This function of the host mode enables device cascading and data transfer between the 485 bus and the network.



4.2 IO basic function

4.2.1 switch DO output

- **Read switch DO output**

Function code: 01, Read coil state

Address range：00017(0x0010)~00020(0x0013)

Note: When the device relay is passively output, and the coil is not power on, the NC and COM of the relay are normally closed, the NO and the COM are normally open, the value is 0; when the coil is power on, the phenomenon is reversed, the relay NC and COM are disconnected, the NO and the COM are closed, and the value is 1. The relay state can be queried by command.

For example:

When reading the 4-channel switch output state, assume that the return value is 03, corresponding to the binary bit 0000 0011, the lower four bits represent the switch output state, which in turn is DO4, DO3, DO2, DO1.

Modbus RTU protocol read switch output:

Send	01	01	00 10	00 04	3C 0C
	Device ModBus address	Function code	Start address	Read number of switch quantity	CRC code

Receive	01	01	01	03	11 89
	Device ModBus address	Function code	Bytes returned	Switch Output Value	CRC code

Modbus TCP protocol read switch output:

Send	00 01	00 00	00 06	01	01	00 10	00 04
	Transmission identifier	Protocol identifier	Length	Unit identifier	Function code	Start address	Read number of switch quantity

Receive	00 01	00 00	00 04	01	01	01	03
	Transmission identifier	Protocol identifier	Length	Unit identifier	Function code	Bytes returned	Switch Output Value

- **Control switch quantity DO output**

Function: 05, Write a single coil state, 0F, Write multiple coil states

Address range: 00017(0x0010)~00020(0x0013)

Note: When the device relay is passively output, and the coil is not power on, the NC and COM of the relay are normally closed, the NO and the COM are normally open; when the coil is power on, the phenomenon is reversed, the relay NC and COM are disconnected, the NO and the COM are closed,. The state of relay can be controlled by command.

For example:

Function code 0x05 writes DO2 switch output, making NC2, COM2 disconnect, NO2, COM2 close, writing value FF 00; making NC2, COM2 close, NO2, COM2 disconnect, writing value 00.

Modbus RTU protocol write switch output:

Send	01	05	00 11	FF 00	DC 3F
	Device ModBus address	Function code	Switch quantity address	Write value	CRC code

Receive	01	05	00 11	FF 00	DC 3F
	Device ModBus address	Function code	Switch quantity address	Write value	CRC code

Modbus TCP protocol write switch output:

Send	00 01	00 00	00 06	01	05	00 11	FF 00
	Transmission identifier	Protocol identifier	Length	Unit identifier	Function code	Start address	Read number of switch quantity

Receive	00 01	00 00	00 06	01	05	00 11	FF 00
	Transmission identifier	Protocol identifier	Length	Unit identifier	Function code	Start address	Read number of switch quantity

Function code 0x0F writes DO2, DO3 switch output, making NC2, COM2 disconnected, NO2, COM2 closed; making NC3, COM3 disconnected, NO3, COM3 closed. Write value should be 0x03, corresponding to binary bit 0000 0011

Modbus RTU protocol write switch output:

Send	01	0F	00 11	00 02	01	03	62 95
	Device ModBus address	Function code	Switch quantity address	Write number of switch quantity	Bytes	Write value	CRC code

Receive	01	0F	00 11	00 02	84 0F
	Device ModBus address	Function code	Switch quantity address	Write value	CRC code

Modbus TCP protocol write switch output:

Send	00 01	00 00	00 08	01	0F	00 11	00 02	01	03
	Transmission identifier	Protocol identifier	Length	Unit identifier	Function code	Switch quantity address	Write number of switch quantity	Bytes	Write value

Receive	00 01	00 00	00 06	01	0F	00 11	00 02
	Transmission identifier	Protocol identifier	Length	Unit identifier	Function code	Switch quantity address	Write number of switch quantity

4.2.2 Read switch quantity DI input

Function: 02, Read switch quantity input state

Address range: 10017(0x0010)~10020(0x0013)

Note: The device default dry contact input. When DI and COM are shorted, the read value should be 1; when DI and COM are not shorted, the read value should be 0.

For example:

Reading 4 channel switch input value, DI1 and COM1 are shorted, DI2 and COM2 are not shorted, DI3 and COM3 are shorted, DI4 and COM4 are not shorted. The input switch value is 0x05, corresponding to the binary bit 0000 0101, and the lower four bits represent the switch input value, which are in turn DI4, DI3, DI2, DI1.

Modbus RTU protocol read switch input:

Send	01	02	00 10	00 04	78 0C
	Device ModBus address	Function code	Start address	Write number of switch quantity	CRC code
Receive	01	02	01	05	61 8B
	Device ModBus address	Function code	Bytes returned	Write value	CRC code

Modbus TCP protocol read switch input:

Send	00 01	00 00	00 06	01	02	00 10	00 04
	Transmission identifier	Protocol identifier	Length	Unit identifier	Function code	Switch quantity address	Write number of switch quantity

Receive	00 01	00 00	00 04	01	02	01	05
	Transmission identifier	Protocol identifier	Length	Unit identifier	Function code	Bytes returned	Write value

4.2.3 Read analog quantity AI input

Function code: 03, Read holding register; 04, Read input register

Address range: 30017(0x0010)~30020(0x0013)

Note: Input unit is uA

For example:

Function code 0x03, read AI1 input, assuming AI1 input is 9946uA, the corresponding value should be 0x0x26DA.

Modbus RTU read analog quantity input:

Send	01	03	00 10	00 01	85 CF
	Device ModBus address	Function code	Start address	Write number of switch quantity	CRC code

Receive	01	03	02	26 DA	23 BF
	Device ModBus address	Function code	Bytes returned	Write value	CRC code

Modbus TCP read analog quantity input:

Send	00 01	00 00	00 06	01	03	00 10	00 01

	Transmission identifier	Protocol identifier	Length	Unit identifier	Function code	Analog quantity address	Write number of analog quantity
Receive	00 01	00 00	00 05	01	03	02	26 DA
	Transmission identifier	Protocol identifier	Length	Unit identifier	Function code	Bytes returned	Write value

Function code 0x04, read AI1 input, assuming AI1 input is 9946uA, the corresponding value should be 0x0x26DA

Modbus RTU read analog quantity input:

Send	01	04	00 10	00 01	30 0F
	Device ModBus address	Function code	Start address	Write number of switch quantity	CRC code

Receive	01	04	02	26 DA	22 CB
	Device ModBus address	Function code	Bytes returned	Write value	CRC code

Modbus TCP read analog quantity input:

Send	00 01	00 00	00 06	01	04	00 10	00 01
	Transmission identifier	Protocol identifier	Length	Unit identifier	Function code	Analog quantity address	Write number of analog quantity

Receive	00 01	00 00	00 05	01	04	02	26 DA
	Transmission identifier	Protocol identifier	Length	Unit identifier	Function code	Bytes returned	Write value

4.2.4 Analog quantity input AI range setting

When register 0x40084(0x0053）is 0, analog quantity input range is 0 -- 20mA;

When register 0x40084(0x0053）is 1, analog quantity input range is 4 -- 20mA;

4.3 IO special function

4.3.1 Pulse count and count clear to zero

The pulse count will not be saved after power off, and the pulse level maintenance time must be more than 10ms to be effective. The switch input changes from the disconnected state to the closed state and maintains the closing time of 10ms or more, and then becomes to disconnected state, completing a pulse count.

- **Read pulse count value**

Function code: 03，Read hoding register

Address: 40049 (0x0030)~40052 (0x0033)

Note: The maximum value of the pulse count is 65535

For example：

DI1 has detected 16 pulses at present, DI2 has detected 3 pulses at present, and reads DI1 and DI2 switch input count values.

Modbus RTU protocol read pulse count value:

Send	01	03	00 30	00 02	C4 04
	Device ModBus address	Function code	Start address	Read number	CRC code

Receive	01	03	04	00 10	00 03	BB F7
	Device ModBus address	Function code	Bytes returned	DI1 count value	DI2 count value	CRC code

Modbus TCP protocol read pulse count valu:

Send	00 01	00 00	00 06	01	03	00 30	00 02
	Transmission identifier	Protocol identifier	Length	Unit identifier	Function code	Start address	Read number

Receive	00 01	00 00	00 07	01	03	04	00 10	00 03
	Transmission identifier	Protocol identifier	Length	Unit identifier	Function code	Bytes returned	DI1 count value	DI2 count value

- **Clear to zero pulse count value**

Function code: 06，Writing hoding register

Address range: 40065 (0x0040)

Note: The lower four bits of the register value represent the DI4, DI3, DI2, and DI1 counts respectively. Writing a '1' indicates that the count is cleared and the pulse count is restarted.

For example:

Clear the DI2 and DI4 pulse count values and keep the DI1 and DI3 pulse count values. The value written should be 0x0a and the corresponding binary value is 0000 1010.

Modbus RTU protocol Clear to zero pulse count value

Send	01	06	00 40	00 0a	08 19
	Device ModBus address	Function code	Address	Write value	CRC code

Receive	01	06	00 40	00 0a	08 19
	Device ModBus address	Function code	Address	Write value	CRC code

Modbus TCP protocol Clear to zero pulse count value

Send	00 01	00 00	00 06	01	06	00 40	00 0a
	Transmission identifier	Protocol identifier	Length	Unit identifier	Function code	Address	Write value

Receive	00 01	00 00	00 06	01	06	00 40	00 0a
	Transmission identifier	Protocol identifier	Length	Unit identifier	Function code	Address	Write value

4.3.2 Switch quantity input DI report automatically

Switch quantity input report automatically function is to transmit the changed value when the switch quantity changes. You can choose to transmit via RS485 or GPRS, or you can turn off this function.

The corresponding Modbus register is 40082 (0x0051), and the value corresponds to the function:

Turn off switch quantity input report automatically function

Switch quantity input report automatically via RS485

Switch quantity input report automatically via GPRS

Switch quantity change uploading protocol is as follows, The frame headers 0xAA and 0xBB are fixed, and the values of DI1, DI2, DI3, and DI4 are 0x00, 0x01, and 0xFF.

0x00 represents the switch quantity input disconnected

0x01 represents the switch quantity input closed

0xff represents the switch quantity input not changed

The values of DI1, DI2, DI3, and DI4 in the table indicate that the DI1 and DI2 states are updated to be disconnected, the DI3 state is updated to be closed, and the DI4 state is not changed. The last two bytes are Modbus CRC16 calculated values.

Header	DI1	DI2	DI3	DI4	Modbus CRC
0xAA 0xBB	0x00	0x00	0x01	0xff	0xBD 0xDA

4.3.3 Switch quantity output DO time setting

The switch quantity pulse output time setting is to set the switch output time (relay NO, COM closing time), the corresponding Modbus register is 40083 (0x0052), its value range is 300-65535ms, if the value is lower than 300ms, default switch output closed to the holding state, that is, the switch output is holding as closed. If it is set to 300ms and above, such as 500ms, after the switch output close command is sent, the switch quantity will be closed for 500ms, and then automatically disconnected after 500ms.

4.3.4 Restart switch quantity output state setting

This is the setting of whether the device hodes the state when restarts after power off, or restartes to maintain the specific output state. This function is valid only when the device switch output time setting register value is less than 300ms.

The restart switch quantity output state setting corresponding to Modbus register is 40085 (0x0054), and its value range is 0x00-0x10. When the value of this register is 0x10, it is maintained to the last switch output state after power-off and restart; when the value of this register is 0x00-0x0F, the restart switch output state is decided by the lower four bits, bit4 corresponds to DO4, bit3 corresponds DO3, bit2 corresponds to DO2, bit1 corresponds to DO1. When power is on, DO4 and DO2 are in the closed state (relay NO, COM is closed), DO3, DO1 are disconnected (relay NO, COM is disconnected), the corresponding register value is 00001010, that is 0xa0, "1" is the closed state, and "0" is the disconnected state.

4.4 Network function

4.4.1 Server IP or domain name, port, TCP or UDP settings (Socket)

The server IP or domain name is stored by 22 modbus registers. The first register is used to store the ASCII code length corresponding to the IP or domain name. The remaining registers are used to store the ASCII code value corresponding to the IP or domain name. For example, the IP is 116.62.42.192, the port is 31687, a total of 13 characters, that is, the length is 0x000D, and the IP corresponding ASCII code value is 31 31 36 2E 36 32 2E 34 32 2E 31 39 32. The corresponding modbus register storage value is as follows table. If it is a domain name, it is also converted into ASCII corresponding hexadecimal storage. (Note: The maximum length of the domain name does not exceed 40 ASCII codes).

40101(0x0063)	40101(0x0064) -- 40121(0x0078)
Length	IP or domain name
00 0D	31 31 36 2E 36 32 2E 34 32 2E 31 39 32

Port 31687, corresponding to hexadecimal 7BC7; protocol type (TCP, UDP) is stored using the protocol register, the value 0x0001 corresponds to the TCP protocol, and the value 0x0000 corresponds to the UDP protocol. That is, when the IP is 116.62.42.192 and the port is 31687, the TCP protocol, the unused IP or domain name register can be filled with "0" or not filled. If you need to use the function code "0x10" to write the IP, domain name, port and protocol type at one time. Then the unused registers must be filled with values in order to continuously write Modbus registers. The corresponding register values are as follows:

40100(0x0063)	40101(0x0064) -- 40121(0x0078)	40122(0x0079)	40123(0x007A)
IP or domain name length	IP or domain name value	Server port	TCP protocol
00 0D	31 31 36 2E 36 32 2E 34 32 2E 31 39 32 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	7B C7	00 01

Since the length of the IP or domain name register is longer than the length of the IP or domain name value, the length of the IP or domain name needs to be considered when writing the IP register, that is, how many registers need to be occupied. For example, write the above IP into the modbus register:

Modbus RTU protocol write Socket register:

Send	01	10	00 63	00 18	30	00 0D 31 31 36 2E 36 32 2E 34 32 2E 31 39 32 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 7B C7 00 01	7B F0
	Device ModBus address	Function code	Address	Register length	Bytes	Write value	CRC code

Receive	01	10	00 63	00 18	30 1D
	Device ModBus address	Function code	Address	Register length	CRC code

Modbus TCP protocol write Socket register:

Send	00 01	00 00	00 37	01	10	00 63	00 18	30	00 0D 31 31 36 2E 36 32 2E 34 32 2E 31 39 32 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 7B C7 00 01
	Transmission identifier	Protocol identifier	Length	Unit identifier	Function code	Address	Register length	Bytes	Write value

Receive	00 01	00 00	00 06	01	10	00 63	00 18
	Transmission identifier	Protocol identifier	Length	Unit identifier	Function code	Address	Register length

4.4.2 User defineddd register

The user-defined registration package can be ASCII code or hex, the length of hex cannot be longer than 20 bytes, and the length of ASCII code cannot be longer than 40 bytes. The first register of the user-defined registration packet is used to store the registration packet type. The value 0x0000 indicates that the registration packet is in hex format, and the value 0x0001 indicates that the registration packet is in ASCII format. When the value is 0x0001, the registration packet value is ABCDEFGHIJ, and the corresponding ASCII code value is as below table. The second register of the user-defined registration packet is used to store the length of the registration packet value. The length of the registration packet value is 10, which corresponds to 0x0A in hexadecimal. Like the IP registers, unused register value registers can be filled with "0" or not filled.

40124(0x007B)	40125(0x007C)	40126(0x007D) -- 40145(0x0090)	40146(0x0091)
Type	Length	Registration packet value	Registration packet mechanism
00 01	00 0A	41 42 43 44 45 46 47 48 49 4A 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	00 01

The registration package mechanism has five modes:

Registration package mechanism register value (0x0091)	Description
00 00	Close the registration package mechanism
00 01	Add MAC/IMEI as registration packet data before each packet is sent to the server.
00 02	Add user-defined registration package data before each packet is sent to the server
00 03	Send a MAC/IMEI registration package only when connecting to the server for the first time
00 04	Send a user-defined registration package only the first time connected to the server

Modbus RTU protocol write registration packet register:

Send	01	10	00 7B	00 17	2E	00 01 00 0A 41 42 43 44 45 46 47 48 49 4A 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 01	00 F4
	Device ModBus address	Function code	Address	Register length	Bytes	Write value	CRC code

Receive	01	10	00 7B	00 17	F0 1E
	Device ModBus address	Function code	Address	Register length	CRC code

Modbus TCP protocol write registration packet register:

Send	00 01	00 00	00 33	01	10	00 7B	00 17	2E	00 01 00 0A 41 42 43 44 45 46 47 48 49 4A 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 01

	Transmission identifier	Protocol identifier	Length	Unit identifier	Function code	Address	Register length	Bytes	Write value
Receive	00 01	00 00	00 06	01	10	00 7B	00 17		
	Transmission identifier	Protocol identifier	Length	Unit identifie	Function code	Address	Register length		

4.4.3 Heartbeat package

The heartbeat packet can be ASCII code or hex, the length of hex cannot be longer than 20 bytes, and the length of ASCII code cannot be longer than 40 bytes. The first register of the heartbeat packet is used to store the heartbeat packet data type. The value 0x0000 indicates that the heartbeat packet is in hex format, and the value 0x0001 indicates that the heartbeat packet is in ASCII format. When the value is 0x0000, the heartbeat packet value is 0x00, 0x01, 0x02, 0x03, 0x04, 0x05, 0x06, 0x07, 0x08, 0x09. The second register of the heartbeat packet is used to store the length of the heartbeat packet value. The heartbeat packet value has a length of 10, corresponding to decimal 0x0A. As with the custom registration packet registers, the unused heartbeat packet value registers can be filled with "0" or not filled.

40147(0x0092)	40148(0x0093)	40149(0x0094) -- 40168(0x00A7)
Type	Length	Registration packet value
00 00	00 0A	00 01 02 03 04 05 06 07 08 09 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

Modbus RTU protocol write registration packet register:

Send	01	10	00 92	00 16	2C	00 00 00 0A 00 01 02 03 04 05 06 07 08 09 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	52 9E
	Device ModBus address	Function code	Address	Register length	Bytes	Write value	CRC code

Receive	01	10	00 92	00 16	E0 2A
	Device ModBus address	Function code	Address	Register length	CRC code

Modbus TCPprotocol write registration packet register:

Send	00 01	00 00	00 33	01	10	00 92	00 16	2C	00 00 00 0A 00 01 02 03 04 05 06 07 08 09 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
	Transmission identifier	Protocol identifier	Length	Unit identifier	Function code	Address	Register length	Bytes	Write value

Receive	00 01	00 00	00 06	01	10	00 92	00 16
	Transmission identifier	Protocol identifier	Length	Unit identifie	Function code	Address	Register length

4.4.4 Heartbeat packet time

The heartbeat packet time setting range is 0-65535 seconds. When the heartbeat packet time is set to 0, the heartbeat packet is closed. Set the heartbeat packet length to 5s as follows.

Modbus RTU protocol write heartbeat packet time register:

Send	01	06	00 A8	00 05	C8 29
	Device ModBus address	Function code	Address	Write value	CRC code

Receive	01	06	00 A8	00 05	C8 29
	Device ModBus address	Function code	Address	Write value	CRC code

Modbus TCP protocol write heartbeat packet time register:

Send	00 01	00 00	00 06	01	06	00 A8	00 05
	Transmission identifier	Protocol identifier	Length	Unit identifier	Function code	Address	Register length

Receive	00 01	00 00	00 06	01	06	00 A8	00 05
	Transmission identifier	Protocol identifier	Length	Unit identifier	Function code	Address	Register length

4.4.5 Ebyte cloud transparent transmission

The Ebyte cloud transparent transmission function can be turned on or off. The register corresponding to this function is 40170 (0x00A9), which supports 0x03, 0x06, 0x10 function codes:

When the value of this register is 0x0000, the cloud transparent transmission function is turned off;

When the value of this register is 0x0001, the cloud transparent transmission function is turned on;

Default turn off.

4.4.6 Read IMEI

The IMEI register read start address is 40171 (0x00AA), and the total length of the register is 11. The first register is the IMEI length, and the second register to the eleventh register stores the IMEI value. For example, IMEI:867732035802677, the corresponding register values are as follows. In the registers, the IMEI value is a hexadecimal ASCII value.

40171（0x00AA)	40172（0x00AB) --40181(0x00B4)
IMEI length	IMEI value
00 0F	38 36 37 37 33 32 30 33 35 38 30 32 36 37 37 00 00 00 00 00

4.4.7 Read SN

The SN register read start address is 40182 (0x00B5), and the total length of the register is 11. The first register is the SN length, and the second register to the eleventh register stores the SN value. For example, SN: 18101194228B027, the corresponding register value is as follows. In the register,

the SN value is a hexadecimal ASCII value.

40182（0x00B5）	40183（0x00B6）--40192(0x00BF)
SN length	SN value
00 10	31 38 31 31 30 31 31 39 34 32 32 38 42 30 32 37

4.4.8 Read base station location LBS

The base station location register read start address is 40193 (0x00C0), and the total length of the register is 20. The first register is the base station location information length, and the second register to the twentieth register stores the base station location value in the . For example, the base station location value LAC: 812F, CID: 8056B08, the corresponding register value is as follows. In the register, the base station location value is a hexadecimal ASCII value.

40193（0x00C0）	40194（0x00C1）--40213(0x00D4)
Base station location register length	Base station location register
00 14	4C 41 43 3A 38 31 32 46 2C 43 49 44 3A 38 30 35 36 42 30 38

Important Notes

- All rights to interpret and modify this manual belong to Ebyte.
- This manual will be updated based on the upgrade of firmware and hardware, please refer to the latest version.
- Please refer to our website for new product information.

About us

Technical support: support@cdebyte.com
Documents and RF Setting download link: www.ebyte.com
Thank you for using Ebyte products! Please contact us with any questions or suggestions: info@cdebyte.com
---
Fax: 028-64146160 ext. 821
Web: www.ebyte.com
Address: Innovation Center D347, 4# XI-XIN Road,Chengdu, Sichuan, China

成都亿佰特电子科技有限公司
Chengdu Ebyte Electronic Technology Co.,Ltd.