EAST Group Co., Ltd.	Document No.		Version	V1.9
	Security classification	Unclassified	Effective date	2022.01
	Prepared by	Software Department		

EA660 G4 Modbus Communication Protocol

EAST Group Co., Ltd.	Document No.		Version	V1.9
	Security classification	Unclassified	Effective date	2022.01
	Prepared by	Software Department		

Serial No.	Version	Modification	Modification time	Remarks
1	**Ver 1.0**	Determine the basic electrical quantity.	2018-11-14	Gaop
2	**Ver 1.1**	Modify the function code tables 02, 04 and 03.	2018-11-20	Gaop
3	**Ver 1.2**	Add the remote signaling volume and modify the name of remote signaling volume.	2019-01-23	Gaop
4	**Ver 1.3**	Add the remote signaling volumes 249 and 432.	2019-03-28	Gaop
5	**Ver 1.4**	Modify the alarm volume communication anomaly and configuration error	2019-04-09	Gaop
6	**Ver 1.5**	Modify 02 function code table of remote signaling volume	2020-06-02	Taoyb
7	**Ver 1.6**	Modify the alarm volume	2020-06-09	Taoyb
8	**Ver 1.7**	Add the information on the lithium batteries	2021-04-07	Taoyb
9	**Ver 1.8**	Modify 447 and 448 in 02 function code table of remote signaling volume	2021-11-8	Zhang
10	**Ver 1.9**	Add TCP/IP protocol communication description	2022-01-20	Zhang

Table of Contents

I. Relevant description of protocol..... 1

1. Introduction to protocol..... 1

2. Interface mode..... 1

3. Protocol format..... 1

3.1 Frame format of RTU mode..... 1

3.2 Frame format of TCP/IPF mode..... 3

4. Classification of response information..... 5

5. Function code..... 7

II. Communication content..... 8

1. Remote measurement (function code 0x04)..... 8

2. Remote signaling volume (function code 0x02)..... 11

3. Read equipment parameters (function code 0x03)..... 28

4. Set equipment parameters (function code 0x06/0x10)..... 30

4.1 0x06 Command frame format..... 30

4.2 0x10 Command frame format..... 31

Appendix A CRC check..... 32

Appendix B High and low byte table..... 35

I. Relevant description of protocol

1. Introduction to protocol

This document is prepared to normalize the Modbus interface requirement of EA660 G4 UPS equipment for RS485 or TCP/IP connection.

The response mode is used in this communication, in which the host initiates the request (sends telemetry information), and the slave executes and responds to the request. It is required to distinguish the slave through the address. The address range from 1 to 247 is settable for the slave.

2. Interface mode

1) RS485 interface: Asynchronous and half duplex;

Baud rate: 9600 bps, settable

Data length: 8 bits, fixed

Parity check bit: None, fixed

Stop bit: 1 bit, fixed

2) TCP / IP (RJ45) interface

3. Protocol format

This protocol supports both RTU Modbus mode and TCP/IP Modbus mode.

3.1 Frame format of RTU mode

When the communication is made by a controller through Modbus in the form of RTU mode, every 8-bit byte in the information contains two 4-bit hexadecimal characters. The format of each byte in RTU mode is shown as follows:

Coding system: 8-bit binary;
Start bit: 1 bit
Data bit: 8 data bits, low bit first;
Odd/even check: 1 bit for odd check or even check; 1 stop bit in the case of no parity check;
Stop bit: 1 bit
Error check area: cyclic redundancy check (CRC)

The request frame format of RTU mode is shown as follows:

Start	Equipment address	Function code	Register starting address	Number of registers	CRC low byte of	CRC high byte	End
Idle time of At least 3.5	8 bit	8 bit	16 bit	16 bit	8 bit	8 bit	Idle time of At least 3.5

characters							characters

10-bit transmission is used as the character transmission format in RTU mode, in which the data bits cover 8 bits; the bit sequence is: (10 bits due to no parity check bit)

Start bit	1	2	3	4	5	6	7	8	Stop bit (odd/even check bit)	Stop bit

The response frame format of RTU mode is:

Start	Equipment address	Function code	data	CRC low byte	CRC high byte	End
Idle time of at least 3.5 characters	8 bit	8 bit	8N bits	8 bit	8 bit	Idle time of at least 3.5 characters

The time interval of at least 3.5 characters is required to send the message. After the last transmitted character, the time interval of at least 3.5 characters is required to indicate the end of the message. A new message can start after this time interval.

The whole message frame must be transmitted as a continuous flow. If the idle time between two characters is more than 1.5 characters before completion of a frame, the frame will be considered to be wrong. Then it is required to stop receiving the frame and to restart such reception, i.e. it is required to ensure the time interval between two frames is at least more than 3.5 characters. The time interval of 1.5 characters and 3.5 characters are dependent on the specific communication baud rate. The calculation method is as follows: if the communication baud rate is 9600, then

The time interval of 1.5 Characters = (1/9600) ×11×1.5×1000 = 1.72 ms

The time interval of 3.5 Characters = (1/9600) ×11×3.5×1000 = 4.01 ms

[Example]***

Request frame information: request for the data of computer 1, the location: 1 register provided with the starting address of 0002

	Address	Function code	Register starting address		Number of registers		CRC check	
Data	0x01	0x03	0x00	0x02	0x00	0x01	0x25	0xCA
Number of bytes	1	1	2		2		2	

Response frame information: the response frame of computer 1

	Address	Function code	Number of data bytes returned	Data content		CRC check	
Data	0x01	0x03	0x02	0x12	0x22	0xE9	0x5C
Number	1	1	1	2		2	

of bytes					

3.2 Frame format of TCP/IPF mode

According to the definition of national standard GB/T19582, the request frame format is shown as follows:

MBAP message header	Function code	Starting address	Number of registers
7 bytes	1 byte	2 bytes	2 bytes

The MBAP message header consists of 7 bytes, and the specific format is as follows:

Field name	Byte length	Description	Data sources	Response format
Transaction identifier	2 bytes	Identification of Modbus request for/response to transactions	Host setting and distribution	Copy and return
Protocol identifier	2 bytes	Fixed to be 0, indicating Modbus Protocol	Host setting and distribution	Copy and return
Length	2 bytes	The number of subsequent bytes, the unit identifier and the total length of subsequent data	Host setting and distribution	Slave setting
Unit identifier	1 byte	Identification of remote slave stations connected on serial links or other buses	Host setting and distribution	Copy and return

The response frame format is shown as follows:

MBAP message header	Function code	Data byte length	Data content
7 bytes	1 byte	1 byte	1*n/2*n

Length response in MBAP message header = number of actual register data + total number of data bytes of 1 byte + function code of 1 byte + unit identifier of 1 byte = 3 + number of actual register data, which represents the total number of bytes after this byte.

Note: when the function code is 02, the byte length of data content is 1* n, and others are 2 * n

Example of message format:

The request frame format of TCP/IP mode is shown as follows:

	Description	Number of bytes	Example
MBAP header	Transaction identifier Hi	1	0x15
	Transaction identifier Lo	1	0x01
	Protocol identifier	2	0x0000
	Length	2	0x0006
	Unit identifier	1	0xff
MODBUS request	Function code	1	0x03

	Starting address	2	0x0005
	Number of registers	2	0x0001

The request response frame format of TCP/IP mode is shown as follows:

	Description	Number of Bytes	Example
MBAP header	Transaction identifier Hi	1	0x15
	Transaction identifier Lo	1	0x01
	Protocol identifier	2	0x0000
	Length	2	0x0005
	Unit identifier	1	0xff
MODBUS response	Function code	1	0x03
	Number of register data bytes	1	0x01
	Register content	1*2	…..

4. Classification of response information

The host sends a query request to the slave equipment and expects a normal response. During this query, the following 4 events are possible:

(1) If the slave receives the query without communication error and the information is normally processed, a normal response event will be returned.

(2) The slave can not receive query data due to communication error. Consequently, no response is returned. In this case, the host will determine the overtime query depending on the processing program.

(3) If any (CRC) communication error is found after the slave receives the query, no response will be returned. In this case, the host will determine the overtime query depending on the processing program.

(4) If the slave receives the query without communication error, but the information can not be processed (due to reading the non-existent register address or the wrong number of registers, etc.), the nature of the error will be reported to the host.

The response information different from that in the normal response is sent by two areas to report errors to the host:

Function code area: During normal response, the response function code area of the slave is provided with the function code of the original query. MSB of all function codes is 0 (its value is lower than 80H). In case of abnormal response, the slave sets the MSB of the function code to be 1, which causes the value of the function code is greater than 80H and thus higher than the value of normal response. In this way, the host application can identify abnormal response events and check the data area of abnormal code.

Data area: In the normal response, the data area contains data or statistical value (given according to the query requirement). In the abnormal response, the data area is provided with an abnormal code, which indicates the conditions and reasons for the abnormal response from the slave.

Abnormal codes and their meanings are shown in the following table:

Code	Name	Meaning
0x01	Illegal function code	The slave receives a function code that can not be executed. After a query order is given, this code indicates no program function.
0x02	Illegal data address	The received data address is unallowable for the slave, such as wrong

		starting address of the register, and wrong number of registers queried.
0x03	Illegal data value	The received data value is unallowable for the slave;
0x04	Server failure	The server is failed during execution
0x05	Confirm	The server accepts service calling, but it takes a relatively long time to complete the service. Therefore, the server only returns the confirmation of receiving one service calling.
0x06	Busy server	Server data may not be ready

[Example]***

RTU mode:

Order information: request for the data of computer 1, the location: 2 registers with the starting address of 0066

	Address	Function code	Register starting address		Number of registers		CRC check	
Data	0x01	0x03	0x00	0x66	0x00	0x02	0x24	0x14

Response information: in the response frame of computer 1, the wrong starting address of the register causes an illegal data address in the returned information.

	Address	Function code	Data content	CRC check	
Data	0x01	0x83	0x02	0xC0	0xF1

TCP/IP mode:

Request:

	MBAP header	Function code	Register starting address		Number of registers	
Data	-----	0x03	0x00	0x66	0x00	0x02

Response:

	Length in MBAP header	Function code	Error code
Data	3	0x83	0x03

Note: the length in MBAP header = header unit identifier + 1 byte function code + 1 byte error

5. Function code

Function code	Name	Function
0x02	Reading discrete magnitude input	Read status information and alarm information
0x03	Reading holding register	Read current parameter setting information
0x04	Reading input register	Read current analog quantity
0x06	Writing single holding register	Write single setting parameter
0x10	Writing multiple holding registers	Write multiple setting parameters

II. Communication content

1. Remote measurement (function code 0x04)

1) Upper computer request command format:

Definition	Address	Function code	Starting register address		Number of registers		CRC check	
Data	ADDR	04H	High order	Low order	High order	Low order	Low order	High order
Number of bytes	1	1	2		2		2	

Host request command format in TCP/IP mode:

Definition	MBAP header	Function code	Starting register address		Number of registers	
Data	---	04H	High order	Low order	High order	Low order
Number of bytes	7	1	2		2	

2) Normal response format:

Definition	Address	Function code	Number of reply data bytes	Returned data		CRC check	
Data	ADDR	04H	DATA_BYTES	High order	Low order	Low order	High order
Number of bytes	1	1	1	2 * number of registers		2	

Normal response format in TCP/IP mode:

Definition	MBAP header	Function code	Number of reply data bytes	Returned data	
Data	---	04H	DATA_BYTES	High order	Low order
Number of bytes	7	1	1	2* DATA_BYTES	

3) Abnormal response format:

Definition	Address	Error code	Exception code	CRC check	
data	ADDR	84H	ERR_CODE	Low order	High order
Number of bytes	1	1	1	2	

Abnormal response format in TCP / IP mode:

Definition	MBAP header	Error code	Exception code
Data	---	84H	ERR_CODE
Number of bytes	7	1	1

Remote measurement register table:

Address		Data content	Data length /format	Type	Unit	Coefficient	Remarks
HEX	DEC						
0x0000	0	Bypass phase A voltage	2Bytes	SHORT	V	0.1	
0x0001	1	Bypass phase B voltage	2Bytes	SHORT	V	0.1	
0x0002	2	Bypass phase C voltage	2Bytes	SHORT	V	0.1	
0x0003	3	Bypass phase A current	2Bytes	SHORT	A	0.1	
0x0004	4	Bypass phase B current	2Bytes	SHORT	A	0.1	
0x0005	5	Bypass phase C current	2Bytes	SHORT	A	0.1	
0x0006	6	Bypass phase A frequency	2Bytes	SHORT	Hz	0.1	
0x0007	7	Bypass phase B frequency	2Bytes	SHORT	Hz	0.1	
0x0008	8	Bypass phase C frequency	2Bytes	SHORT	Hz	0.1	
0x0009	9	Reserved	2Bytes				
0x000A	10	Reserved	2Bytes				
0x000B	11	Reserved	2Bytes				
0x000C	12	Input phase A voltage	2Bytes	SHORT	V	0.1	
0x000D	13	Input phase B voltage	2Bytes	SHORT	V	0.1	
0x000E	14	Input phase C voltage	2Bytes	SHORT	V	0.1	
0x000F	15	Input phase A current	2Bytes	SHORT	A	0.1	
0x0010	16	Input phase B current	2Bytes	SHORT	A	0.1	
0x0011	17	Input phase C current	2Bytes	SHORT	A	0.1	
0x0012	18	Input phase A frequency	2Bytes	SHORT	Hz	0.1	
0x0013	19	Input phase B frequency	2Bytes	SHORT	Hz	0.1	
0x0014	20	Input phase C frequency	2Bytes	SHORT	Hz	0.1	
0x0015	21	Reserved	2Bytes				
0x0016	22	Reserved	2Bytes				
0x0017	23	Reserved	2Bytes				
0x0018	24	Output phase A voltage	2Bytes	SHORT	V	0.1	
0x0019	25	Output phase B voltage	2Bytes	SHORT	V	0.1	
0x001A	26	Output phase C voltage	2Bytes	SHORT	V	0.1	
0x001B	27	Output phase A current	2Bytes	SHORT	A	0.1	
0x001C	28	Output phase B current	2Bytes	SHORT	A	0.1	
0x001D	29	Output phase C current	2Bytes	SHORT	A	0.1	
0x001E	30	Output phase A frequency	2Bytes	SHORT	Hz	0.1	
0x001F	31	Output phase B	2Bytes	SHORT	Hz	0.1	

		frequency					
0x0020	32	Output phase C frequency	2Bytes	SHORT	Hz	0.1	
0x0021	33	Reserved	2Bytes				
0x0022	34	Reserved	2Bytes				
0x0023	35	Reserved	2Bytes				
0x0024	36	Output phase A apparent power	2Bytes	SHORT	kVA	0.1	
0x0025	37	Output phase B apparent power	2Bytes	SHORT	kVA	0.1	
0x0026	38	Output phase C apparent power	2Bytes	SHORT	kVA	0.1	
0x0027	39	Output phase A active power	2Bytes	SHORT	kW	0.1	
0x0028	40	Output phase B active power	2Bytes	SHORT	kW	0.1	
0x0029	41	Output phase C active power	2Bytes	SHORT	kW	0.1	
0x002A	42	Reserved	2Bytes				
0x002B	43	Reserved	2Bytes				
0x002C	44	Reserved	2Bytes				
0x002D	45	Output phase A load percentage	2Bytes	SHORT	%	1	
0x002E	46	Output phase B load percentage	2Bytes	SHORT	%	1	
0x002F	47	Output phase C load percentage	2Bytes	SHORT	%	1	
0x0030	48	Battery status	2Bytes	INT16	/	1	0: Idle 1: Charging 2: Discharge 3: Equalized charging 4: Floating charge 5: Sleep 6: Unconnected
0x0031	49	Battery voltage	2Bytes	SHORT	V	0.1	
0x0032	50	Reserved	2Bytes				
0x0033	51	Battery current	2Bytes	SHORT	A	0.1	
0x0034	52	Reserved	2Bytes				
0x0035	53	Battery temperature	2Bytes	SHORT	℃	0.1	
0x0036	54	Number of battery cells	2Bytes	SHORT	/	1	
0x0037	55	Battery capacity	2Bytes	SHORT	AH	1	
0x0038	56	Remaining discharge time of battery	2Bytes	SHORT	min	1	
0x0039	57	Remaining capacity of battery	2Bytes	SHORT	%	1	
0x003A	58	Reserved	2Bytes				
0x003B	59	Reserved	2Bytes				
0x003C	60	Reserved	2Bytes				
0x003D	61	Rated capacity	2Bytes	SHORT	kVA	1	
0x003E	62	Reserved	2Bytes				

0x003F	63	Rated input voltage	2Bytes	SHORT	V	1	
0x0040	64	Rated input frequency	2Bytes	SHORT	Hz	1	
0x0041	65	Rated output voltage	2Bytes	SHORT	V	1	
0x0042	66	Rated output frequency	2Bytes	SHORT	Hz	1	
0x0043	67	Energy flow line status register	2Bytes	SHORT	/	1	It is listed in the sequence from low to high, in which each section occupies 2bit, 00 represents stationary, 01 represents forward flow (to the lower right), and 02 represents reverse flow (to the upper left)
0x0044	68	Reserved	2Bytes				
0x0045	69	Reserved	2Bytes				
0x0046	70	Model identification register	2Bytes	SHORT	/	1	0: Single and single , 1: Three singles, 2: Three and three
0x0047	71	Power supply mode	2Bytes	SHORT	/	1	0: No power supply; 1: Mains power supply; 2:Battery powered; 3: Combined power supply; 4: Bypass power supply
0x0048	72	Maximum voltage of battery cell	2Bytes	INT16	mV		
0x0049	73	Minimum voltage of battery cell	2Bytes	INT16	mV		
0x004A	74	Maximum temperature of battery cell	2Bytes	INT16	℃		
0x004B	75	Minimum temperature of battery cell	2Bytes	INT16	℃		

2. Remote signaling volume (function code 0x02)

1) MCU request command format:

Definition	Address	Function code	Starting discrete magnitude address		Number of discrete magnitude		CRC check	
Data	ADDR	02H	High order	Low order	High order	Low order	Low order	High order
Number of	1	1	2		2		2	

bytes					

Host request command format in TCP/IP mode:

Definition	MBAP header	Function code	Starting discrete magnitude address		Number of discrete magnitude	
Data	----	02H	High order	Low order	High order	Low order
Number of bytes	7	1	2		2	

2) DSP normal response format:

Definition	Address	Function code	Number of reply data bytes	Returned discrete magnitude state	CRC check	
Data	ADDR	02H	DATA_BYTES		Low order	High order
Number of bytes	1	1	1		2	

Normal response format in TCP / IP mode:

Definition	MBAP header	Function code	Number of reply data bytes	Returned discrete magnitude state
Data	---	02H	DATA_BYTES	---
Number of bytes	7	1	1	DATA_BYTES

3) DSP abnormal response format:

Definition	Address	Error code	Exception code	CRC check	
Data	ADDR	82H	ERR_CODE	Low order	High order
Number of bytes	1	1	1	2	

Abnormal response format in TCP/IP mode:

Definition	MBAP header	Error code	Exception code
Data	---	82H	ERR_CODE
Number of bytes	7	1	1

Remote signaling volume register table:

Address		Alarm/fault	Data length /format	Remarks
HEX	DEC			
0x00D0	208	Communication abnormality of bypass module / system board	1bit	The range from 208 to 335 represents monitoring alarms
0x00D1	209	Abnormal rectifier communication	1bit	
0x00D2	210	Abnormal inverter communication	1bit	
0x00D3	211	Reserved	1bit	
0x00D4	212	Reserved	1bit	
0x00D5	213	Reserved	1bit	

0x00D6	214	Reserved	1bit	
0x00D7	215	Reserved	1bit	
0x00D8	216	Reserved	1bit	
0x00D9	217	Bypass module/system board configuration error	1bit	
0x00DA	218	Rectifier configuration error	1bit	
0x00DB	219	Inverter configuration error	1bit	
0x00DC	220	Reserved	1bit	
0x00DD	221	Reserved	1bit	
0x00DE	222	Reserved	1bit	
0x00DF	223	Reserved	1bit	
0x00E0	224	Reserved	1bit	
0x00E1	225	Reserved	1bit	
0x00E2	226	Emergency alarm	1bit	
0x00E3	227	Secondary alarm	1bit	
0x00E4	228	Bypass supply	1bit	
0x00E5	229	Battery powered	1bit	
0x00E6	230	Low voltage of battery (DOD)	1bit	
0x00E7	231	Low voltage of battery (EOD)	1bit	
0x00E8	232	Bypass fault	1bit	
0x00E9	233	Fan failure	1bit	
0x00EA	234	Battery ground fault	1bit	
0x00EB	235	Oil engine mode	1bit	
0x00EC	236	Battery switch disconnected	1bit	
0x00ED	237	Emergency stop	1bit	
0x00EE	238	The output switch of the distribution cabinet is disconnected	1bit	
0x00EF	239	The maintenance switch of distribution cabinet is connected	1bit	
0x00F0	240	The bypass switch of distribution cabinet is disconnected	1bit	
0x00F1	241	AC lightning arrester disconnected	1bit	
0x00F2	242	External transformer over-temperature	1bit	
0x00F3	243	Mains power input air switch disconnect	1bit	
0x00F4	244	The machine output switch disconnected	1bit	
0x00F5	245	Machine service switch connected	1bit	
0x00F6	246	The machine bypass switch disconnected	1bit	
0x00F7	247	Maintenance cover open	1bit	
0x00F8	248	The bypass backflow protection	1bit	

		contactor disconnected		
0x00F9	249	Time sharing power down	1bit	
0x00FA	250	Unconnected to BCB	1bit	
0x00FB	251	Door magnetic alarm	1bit	
0x00FC	252	Water-logging alarm	1bit	
0x00FD	253	Abnormal communication of system 2	1bit	
0x00FE	254	Abnormal system board version	1bit	
0x00FF	255	Abnormal rectifier board version	1bit	
0x0100	256	Abnormal inverter board version	1bit	
0x0101	257	Abnormal communication of bypass board	1bit	
0x0102	258	Noncompliance with module quantity	1bit	
0x0103	259	Abnormal CAN bus	1bit	
0x0104	260	Parallel communication abnormality	1bit	
0x0105	261	Abnormal Bms communication	1bit	
0x0106	262	Noncompliance with the number of Bms battery cabinets	1bit	
0x0107	263	Noncompliance with the number of modules in Bms battery cabinets	1bit	
0x0108	264	Noncompliance with the number of cells in Bms modules	1bit	
0x0109	265	Bypass backflow	1bit	
0x010A ~0x014F	266~335	Reserved	1bit	
0x0150	336	System boards and inverter modules Abnormal CAN communication	1bit	The range from 336 to 463 represents bypass module/system board alarm
0x0151	337	Multiple inverters provided with the same address	1bit	
0x0152	338	Power supply status conflict	1bit	
0x0153	339	No current-sharing inverter module	1bit	
0x0154	340	105% output overload	1bit	
0x0155	341	110% output overload	1bit	
0x0156	342	125% output overload	1bit	
0x0157	343	150% output overload	1bit	
0x0158	344	Failed switch between bypass and inverter	1bit	
0x0159	345	Phase lock failure	1bit	
0x015A	346	System self-test failed	1bit	
0x015B	347	Fast and abnormal switch of output voltage to bypass	1bit	
0x015C	348	Load impulse switch to bypass	1bit	

0x015D	349	Output overload alarm	1bit	
0x015E	350	Repetitive switch locking bypass	1bit	
0x015F	351	Repetitive switch locking inverter	1bit	
0x0160	352	Abnormal CAN communication between system boards	1bit	
0x0161	353	105% system overload	1bit	
0x0162	354	110% system overload	1bit	
0x0163	355	125% system overload	1bit	
0x0164	356	150% system overload	1bit	
0x0165	357	Abnormal parallel current sharing	1bit	
0x0166	358	Adjacent computer request for switching to bypass	1bit	
0x0167	359	System overload alarm	1bit	
0x0168	360	The output phase sequence reversed	1bit	
0x0169	361	Abnormal rack No.	1bit	
0x016A	362	Abnormal connection of parallel line	1bit	
0x016B	363	Module unlocked (SYS)	1bit	
0x016C	364	ECU unprepared	1bit	
0x016D	365	Noncompliance of the number of power modules with the loading capacity	1bit	
0x016E	366	Frequency beyond tracking range	1bit	
0x016F	367	Output sampling exception	1bit	
0x0170	368	Bypass phase voltage overvoltage	1bit	
0x0171	369	Bypass phase voltage under-voltage	1bit	
0x0172	370	Too high bypass frequency	1bit	
0x0173	371	Too low bypass frequency	1bit	
0x0174	372	The bypass phase sequence reversed	1bit	
0x0175	373	Bypass phase loss	1bit	
0x0176	374	Bypass phase voltage imbalance	1bit	
0x0177	375	Fast detection abnormality of bypass voltage	1bit	
0x0178	376	Bypass over-current	1bit	
0x0179	377	ECO bypass voltage overvoltage	1bit	
0x017A	378	ECO bypass voltage under-voltage	1bit	
0x017B	379	ECO bypass frequency over-frequency	1bit	
0x017C	380	ECO bypass frequency under-frequency	1bit	
0x017D	381	Fast power failure of ECO bypass	1bit	
0x017E	382	ECO bypass phase sequence reverse	1bit	
0x017F	383	Neutral line loss of ECO bypass	1bit	
0x0180	384	Bypass E2PROM operation failed	1bit	

0x0181	385	Failed communication between bypass DSP and monitoring	1bit	
0x0182	386	Abnormal bypass DSP software version	1bit	
0x0183	387	Mismatch between bypass software version and hardware version	1bit	
0x0184	388	Bypass fan failure	1bit	
0x0185	389	Open circuit/fuse failure of bypass SCR	1bit	
0x0186	390	short circuit fault of Bypass SCR	1bit	
0x0187	391	Emergency stop	1bit	
0x0188	392	Stop button	1bit	
0x0189	393	Auxiliary power failure of bypass	1bit	
0x018A	394	Bypass backflow	1bit	
0x018B	395	Bypass fuse failure	1bit	
0x018C	396	Bypass radiator over-temperature	1bit	
0x018D	397	Reserved	1bit	
0x018E	398	Reserved	1bit	
0x018F	399	Reserved	1bit	
0x0190	400	Abnormal fast detection of output voltage	1bit	
0x0191	401	Output voltage overvoltage	1bit	
0x0192	402	Output voltage under-voltage	1bit	
0x0193	403	Too high output frequency	1bit	
0x0194	404	Too low output frequency	1bit	
0x0195	405	Output voltage imbalance	1bit	
0x0196	406	105% inverter overload	1bit	
0x0197	407	110% inverter overload	1bit	
0x0198	408	125% inverter overload	1bit	
0x0199	409	150% inverter overload	1bit	
0x019A	410	125% bypass overload	1bit	
0x019B	411	135% bypass overload	1bit	
0x019C	412	150% bypass overload	1bit	
0x019D	413	200% bypass overload	1bit	
0x019E	414	Inverter overload alarm	1bit	
0x019F	415	Bypass overload alarm	1bit	
0x01A0	416	Standby time pre-alarm	1bit	
0x01A1	417	Residual capacity pre-alarm	1bit	
0x01A2	418	Battery maintenance reminder	1bit	
0x01A3	419	End of battery discharge time	1bit	

0x01A4	420	End of battery discharge voltage	1bit	
0x01A5	421	Battery over-temperature	1bit	
0x01A6	422	Battery low-temperature	1bit	
0x01A7	423	Battery self-test failed	1bit	
0x01A8	424	Battery discharge timeout	1bit	
0x01A9	425	Reserved	1bit	
0x01AA	426	Reserved	1bit	
0x01AB	427	Reserved	1bit	
0x01AC	428	Reserved	1bit	
0x01AD	429	Reserved	1bit	
0x01AE	430	Reserved	1bit	
0x01AF	431	Reserved	1bit	
0x01B0	432	Maintenance bypass air switch connected	1bit	
0x01B1	433	Output air switch disconnected	1bit	
0x01B2	434	Dial code in test state	1bit	
0x01B3	435	Keeping at inverter output	1bit	
0x01B4	436	Keeping at bypass output	1bit	
0x01B5	437	Manually opened bypass	1bit	
0x01B6	438	Abnormal SPI communication between two system boards in the rack	1bit	
0x01B7	439	Abnormal parallel line 1	1bit	
0x01B8	440	Abnormal parallel line 2	1bit	
0x01B9	441	Abnormal heartbeat between racks	1bit	
0x01BA	442	Parallel line connection failure	1bit	
0x01BB	443	Abnormal communication between system board and bypass can	1bit	
0x01BC	444	Noncompliance of the number of rated racks with the number of current racks	1bit	
0x01BD	445	Noncompliance of the number of rated modules with the actual number of modules	1bit	
0x01BE	446	Dual system board backup IO exception	1bit	
0x01BF	447	LBS line communication abnormality	1bit	
0x01C0	448	LBS CAN communication failure	1bit	
0x01C1	449	Reserved	1bit	
0x01C2	450	Reserved	1bit	
0x01C3	451	Reserved	1bit	
0x01C4	452	Reserved	1bit	
0x01C5	453	Reserved	1bit	
0x01C6	454	Reserved	1bit	

0x01C7	455	Reserved	1bit	
0x01C8	456	Reserved	1bit	
0x01C9	457	Reserved	1bit	
0x01CA	458	Reserved	1bit	
0x01CB	459	Reserved	1bit	
0x01CC	460	Reserved	1bit	
0x01CD	461	Reserved	1bit	
0x01CE	462	Reserved	1bit	
0x01CF	463	Reserved	1bit	
0x01D0	464	Input voltage overvoltage	1bit	The range from 464 to 591 represents rectifier module alarm.
0x01D1	465	Input voltage under-voltage	1bit	
0x01D2	466	Input frequency over-frequency	1bit	
0x01D3	467	Input frequency under-frequency	1bit	
0x01D4	468	Input phase sequence reversed	1bit	
0x01D5	469	Input phase loss	1bit	
0x01D6	470	Input voltage imbalance	1bit	
0x01D7	471	Abnormal fast detection of input voltage	1bit	
0x01D8	472	Input current over-current	1bit	
0x01D9	473	Input current imbalance	1bit	
0x01DA	474	Input zero line unconnected	1bit	
0x01DB	475	Input fuse failure	1bit	
0x01DC	476	Input limit power	1bit	
0x01DD	477	Frequent switching of mains battery	1bit	
0x01DE	478	Input overload	1bit	
0x01DF	479	Reserved	1bit	
0x01E0	480	Abnormal battery connection (un connected or reversely connected)	1bit	
0x01E1	481	Battery over-temperature	1bit	
0x01E2	482	Battery self-test failed	1bit	
0x01E3	483	Battery overvoltage	1bit	
0x01E4	484	Battery DOD under-voltage	1bit	
0x01E5	485	Battery EOD under-voltage	1bit	
0x01E6	486	Battery overcharge	1bit	
0x01E7	487	Battery low-temperature	1bit	
0x01E8	488	Battery hardware overvoltage fault	1bit	
0x01E9	489	Battery charging over-current	1bit	

0x01EA	490	Battery discharge over-current	1bit	
0x01EB	491	Open circuit of battery fuse/charger switch	1bit	
0x01EC	492	Short circuit of Charger switch	1bit	
0x01ED	493	Battery discharge timeout	1bit	
0x01EE	494	Battery connected reversely	1bit	
0x01EF	495	Reserved	1bit	
0x01F0	496	Positive bus voltage overvoltage	1bit	
0x01F1	497	Negative bus voltage overvoltage	1bit	
0x01F2	498	Positive bus voltage under-voltage	1bit	
0x01F3	499	Negative bus voltage under-voltage	1bit	
0x01F4	500	Voltage imbalance between positive and negative buses	1bit	
0x01F5	501	Bus hardware overvoltage fault	1bit	
0x01F6	502	Exceed bus overvoltage times	1bit	
0x01F7	503	Bus capacitor service life less than 1 year	1bit	
0x01F8	504	Instantaneous under-voltage of positive bus	1bit	
0x01F9	505	Instantaneous under-voltage of negative bus	1bit	
0x01FA	506	Bus short circuit	1bit	
0x01FB	507	Bus voltage overvoltage	1bit	
0x01FC	508	Bus voltage under-voltage	1bit	
0x01FD	509	Abnormal bus boost	1bit	
0x01FE	510	Bus capacitance fault pre-alarm	1bit	
0x01FF	511	Reserved	1bit	
0x0200	512	Input AC soft start failed	1bit	
0x0201	513	Battery DC soft start failure	1bit	
0x0202	514	BUS DC/DC soft start failure	1bit	
0x0203	515	Input phase locking failed	1bit	
0x0204	516	Frequent switching between mains power supply and battery	1bit	
0x0205	517	Exceed soft start times of rectifier	1bit	
0x0206	518	Wave-by-wave current limiting fault of rectifier hardware	1bit	
0x0207	519	Rectifier hardware over-current fault	1bit	
0x0208	520	Wave-by-wave current limiting alarm of rectifier hardware	1bit	
0x0209	521	Input PFC soft start failed	1bit	
0x020A	522	Rectifier over-current	1bit	
0x020B	523	Rectifier mains mode over-current	1bit	

0x020C	524	Rectifier battery mode over-current	1bit	
0x020D	525	Phase locking failure	1bit	
0x020E	526	Exceed bus soft start times	1bit	
0x020F	527	Bus hardware soft start failure	1bit	
0x0210	528	Rectifier IGBT module over-temperature	1bit	
0x0211	529	Rectifier E2PROM reading/writing failed	1bit	
0x0212	530	Failed communication between rectifier DSP and monitoring	1bit	
0x0213	531	Failed communication between rectifier DSP and CPLD	1bit	
0x0214	532	Fan failure	1bit	
0x0215	533	Abnormal rectifier PowerOK	1bit	
0x0216	534	Fan fault pre-alarm	1bit	
0x0217	535	Abnormal software version of rectifier CPLD	1bit	
0x0218	536	Abnormal software version of rectifier DSP	1bit	
0x0219	537	Mismatch between the software version and hardware version of the rectifier	1bit	
0x021A	538	Abnormal auxiliary power supply of rectifier	1bit	
0x021B	539	SPI communication fault between rectifier and inverter	1bit	
0x021C	540	Drive connection failure	1bit	
0x021D	541	Rectifier contactor fault	1bit	
0x021E	542	Battery contactor failure	1bit	
0x021F	543	Emergency stop	1bit	
0x0220	544	Short circuit of charger switch	1bit	
0x0221	545	Open circuit of Charger switch	1bit	
0x0222	546	Soft start failure of Charger	1bit	
0x0223	547	Charger overvoltage	1bit	
0x0224	548	Charger hardware overvoltage fault	1bit	
0x0225	549	Charger under-voltage	1bit	
0x0226	550	Charger over-current	1bit	
0x0227	551	Charger over-temperature	1bit	
0x0228	552	Wave-by-wave current limiting fault of charger hardware	1bit	
0x0229	553	Wave-by-wave current limiting alarm of charger hardware	1bit	
0x022A	554	Charger hardware over-current fault	1bit	
0x022B	555	Reserved	1bit	
0x022C	556	Reserved	1bit	

0x022D	557	Reserved	1bit	
0x022E	558	Reserved	1bit	
0x022F	559	Reserved	1bit	
0x0230	560	Hardware over-current fault of balance bridge arm	1bit	
0x0231	561	Wave-by-wave current limiting fault of balance bridge arm hardware	1bit	
0x0232	562	Wave-by-wave current limiting alarm of balance bridge arm hardware	1bit	
0x0233	563	Fast over-current of balance bridge arm	1bit	
0x0234	564	Balance bridge arm over-current	1bit	
0x0235	565	Balance bridge arm over-temperature	1bit	
0x0236	566	Module unlocked (PFC)	1bit	
0x0237	567	Dial code in test state	1bit	
0x0238	568	Reserved	1bit	
0x0239	569	Reserved	1bit	
0x023A	570	Reserved	1bit	
0x023B	571	Reserved	1bit	
0x023C	572	Reserved	1bit	
0x023D	573	Reserved	1bit	
0x023E	574	Reserved	1bit	
0x023F	575	Reserved	1bit	
0x0240	576	Reserved	1bit	
0x0241	577	Reserved	1bit	
0x0242	578	Reserved	1bit	
0x0243	579	Reserved	1bit	
0x0244	580	Reserved	1bit	
0x0245	581	Reserved	1bit	
0x0246	582	Reserved	1bit	
0x0247	583	Reserved	1bit	
0x0248	584	Reserved	1bit	
0x0249	585	Reserved	1bit	
0x024A	586	Reserved	1bit	
0x024B	587	Reserved	1bit	
0x024C	588	Reserved	1bit	
0x024D	589	Reserved	1bit	
0x024E	590	Reserved	1bit	
0x024F	591	Reserved	1bit	
0x0250	592	short circuit of Bus	1bit	The range from 592 to 719 represents the inverter module

				alarm.
0x0251	593	Bus abnormality	1bit	
0x0252	594	Bus overvoltage	1bit	
0x0253	595	Bus under-voltage	1bit	
0x0254	596	Bus voltage imbalance	1bit	
0x0255	597	Reserved	1bit	
0x0256	598	Reserved	1bit	
0x0257	599	Reserved	1bit	
0x0258	600	Reserved	1bit	
0x0259	601	Reserved	1bit	
0x025A	602	Reserved	1bit	
0x025B	603	Reserved	1bit	
0x025C	604	Reserved	1bit	
0x025D	605	Reserved	1bit	
0x025E	606	Reserved	1bit	
0x025F	607	Reserved	1bit	
0x0260	608	Inverter voltage overvoltage	1bit	
0x0261	609	Inverter voltage under-voltage	1bit	
0x0262	610	Inverter voltage imbalance	1bit	
0x0263	611	Excessive DC component (R S T)	1bit	
0x0264	612	105% inverter module overload	1bit	
0x0265	613	110% inverter module overload	1bit	
0x0266	614	125% inverter module overload	1bit	
0x0267	615	150% inverter module overload	1bit	
0x0268	616	Short circuit of inverter output	1bit	
0x0269	617	Inverter module overload alarm	1bit	
0x026A	618	Inverter over-frequency	1bit	
0x026B	619	Inverter under-frequency	1bit	
0x026C	620	Reserved	1bit	
0x026D	621	Reserved	1bit	
0x026E	622	Reserved	1bit	
0x026F	623	Reserved	1bit	
0x0270	624	Open circuit fault of bypass SCR	1bit	
0x0271	625	Short circuit fault of Bypass SCR	1bit	
0x0272	626	125% bypass overload	1bit	
0x0273	627	135% bypass overload	1bit	
0x0274	628	150% bypass overload	1bit	

0x0275	629	200% bypass overload	1bit	
0x0276	630	Bypass overload alarm	1bit	
0x0277	631	Bypass over-temperature	1bit	
0x0278	632	Bypass overvoltage	1bit	
0x0279	633	Bypass under-voltage	1bit	
0x027A	634	Bypass over-frequency	1bit	
0x027B	635	Bypass under-frequency	1bit	
0x027C	636	Bypass phase sequence reversed	1bit	
0x027D	637	Reserved	1bit	
0x027E	638	Reserved	1bit	
0x027F	639	Reserved	1bit	
0x0280	640	soft start failure of inverter	1bit	
0x0281	641	Phase locking failure	1bit	
0x0282	642	Frequent switching between bypass and inverter	1bit	
0x0283	643	Exceed soft start times of inverter	1bit	
0x0284	644	Abnormal parallel current sharing	1bit	
0x0285	645	Capture fault	1bit	
0x0286	646	Load impact	1bit	
0x0287	647	Adjacent machine request for switching to bypass	1bit	
0x0288	648	Abnormal parallel line	1bit	
0x0289	649	Drive connection failure	1bit	
0x028A	650	Synchronous square wave anomaly	1bit	
0x028B	651	Inverter self-test failed	1bit	
0x028C	652	Reserved	1bit	
0x028D	653	Reserved	1bit	
0x028E	654	Reserved	1bit	
0x028F	655	Reserved	1bit	
0x0290	656	Inverter radiator over-temperature	1bit	
0x0291	657	Inversion E2PROM operation failed	1bit	
0x0292	658	Failed communication between inverter DSP and monitoring	1bit	
0x0293	659	Failed communication between inverter DSP and system board	1bit	
0x0294	660	Failed communication between inverter DSP and CPLD	1bit	
0x0295	661	Inverter hardware over-current fault	1bit	
0x0296	662	Inverter fuse failure	1bit	
0x0297	663	Emergency stop	1bit	

0x0298	664	Inverter contactor fault	1bit	
0x0299	665	Wave-by-wave current limiting fault of inverter hardware	1bit	
0x029A	666	Wave-by-wave current limiting alarm of inverter hardware	1bit	
0x029B	667	Abnormal power supply of inverter	1bit	
0x029C	668	Abnormal version of inverter CPLD software	1bit	
0x029D	669	Abnormal DSP software version of inverter	1bit	
0x029E	670	Mismatch between the software version and the hardware version of inverter	1bit	
0x029F	671	Zero ground fault	1bit	
0x02A0	672	Open circuit fault of inverter relay	1bit	
0x02A1	673	Short circuit fault of inverter relay	1bit	
0x02A2	674	Inverter capacitor fault	1bit	
0x02A3	675	Inverter fan fault	1bit	
0x02A4	676	SPI communication fault between rectifier and inverter	1bit	
0x02A5	677	Fast over-current of inverter	1bit	
0x02A6	678	PowerOK exception of inverter	1bit	
0x02A7	679	Module unlocked (INV)	1bit	
0x02A8	680	Inverter fan fault pre-alarm	1bit	
0x02A9	681	Inverter capacitor fault pre-alarm	1bit	
0x02AA	682	Dial code in test state	1bit	
0x02AB	683	Keeping at inverter output	1bit	
0x02AC	684	Keeping at bypass output	1bit	
0x02AD	685	Reserved	1bit	
0x02AE	686	Reserved	1bit	
0x02AF	687	Reserved	1bit	
0x02B0	688	Output overvoltage	1bit	
0x02B1	689	Output under-voltage	1bit	
0x02B2	690	Reserved	1bit	
0x02B3	691	Reserved	1bit	
0x02B4	692	Reserved	1bit	
0x02B5	693	Reserved	1bit	
0x02B6	694	Reserved	1bit	
0x02B7	695	Reserved	1bit	
0x02B8	696	Reserved	1bit	
0x02B9	697	Reserved	1bit	
0x02BA	698	Reserved	1bit	

0x02BB	699	Reserved	1bit	
0x02BC	700	Reserved	1bit	
0x02BD	701	Reserved	1bit	
0x02BE	702	Reserved	1bit	
0x02BF	703	Reserved	1bit	
0x02C0	704	Reserved	1bit	
0x02C1	705	Reserved	1bit	
0x02C2	706	Reserved	1bit	
0x02C3	707	Reserved	1bit	
0x02C4	708	Reserved	1bit	
0x02C5	709	Reserved	1bit	
0x02C6	710	Reserved	1bit	
0x02C7	711	Reserved	1bit	
0x02C8	712	Reserved	1bit	
0x02C9	713	Reserved	1bit	
0x02CA	714	Reserved	1bit	
0x02CB	715	Reserved	1bit	
0x02CC	716	Reserved	1bit	
0x02CD	717	Reserved	1bit	
0x02CE	718	Reserved	1bit	
0x02CF	719	Reserved	1bit	
0x02D0	720	Primary protection against overvoltage of the whole set	1bit	
0x02D1	721	Primary protection against under-voltage of the whole set	1bit	
0x02D2	722	Primary protection of charging over-current	1bit	
0x02D3	723	Primary protection of discharge over-current	1bit	
0x02D4	724	Charging high-temperature primary protection	1bit	
0x02D5	725	Charging low-temperature primary protection	1bit	
0x02D6	726	Primary protection against BMU communication fault	1bit	
0x02D7	727	RFU primary protection	1bit	
0x02D8	728	Primary protection of temperature imbalance	1bit	
0x02D9	729	Primary protection of unbalanced cell	1bit	
0x02DA	730	Primary protection against too Low SOC	1bit	
0x02DB	731	Primary protection at lower insulation level	1bit	
0x02DC	732	Primary protection of single	1bit	

		overvoltage		
0x02DD	733	Primary protection of single under-voltage	1bit	
0x02DE	734	Primary protection against excessive discharge temperature	1bit	
0x02DF	735	Primary protection against too low discharge temperature	1bit	
0x02E0	736	Secondary protection against overvoltage of the whole set	1bit	
0x02E1	737	Secondary protection against under-voltage of the whole set	1bit	
0x02E2	738	Secondary protection of charging over-current	1bit	
0x02E3	739	Secondary protection of discharge over-current	1bit	
0x02E4	740	Charging high-temperature secondary protection	1bit	
0x02E5	741	Charging low-temperature secondary protection	1bit	
0x02E6	742	Secondary protection against BMU communication fault	1bit	
0x02E7	743	RFU secondary protection	1bit	
0x02E8	744	Secondary protection of temperature imbalance	1bit	
0x02E9	745	Secondary protection of unbalanced cell	1bit	
0x02EA	746	Secondary protection against too low SOC	1bit	
0x02EB	747	Secondary protection at low insulation level	1bit	
0x02EC	748	Secondary protection of single overvoltage	1bit	
0x02ED	749	Secondary protection of single under-voltage	1bit	
0x02EE	750	Secondary protection against excessive discharge temperature	1bit	
0x02EF	751	Secondary protection against too low discharge temperature	1bit	
0x02F0	752	Reserved	1bit	
0x02F1	753	Reserved	1bit	
0x02F2	754	Reserved	1bit	
0x02F3	755	Reserved	1bit	
0x02F4	756	Reserved	1bit	
0x02F5	757	Reserved	1bit	
0x02F6	758	Reserved	1bit	
0x02F7	759	Reserved	1bit	
0x02F8	760	Reserved	1bit	
0x02F9	761	Reserved	1bit	
0x02FA	762	Reserved	1bit	

0x02FB	763	Reserved	1bit	
0x02FC	764	Reserved	1bit	
0x02FD	765	Reserved	1bit	
0x02FE	766	Reserved	1bit	
0x02FF	767	Reserved	1bit	
0x0300	768	Overvoltage alarm of the whole set	1bit	
0x0301	769	Under-voltage alarm of the whole set	1bit	
0x0302	770	Charging over-current alarm	1bit	
0x0303	771	Discharge over-current alarm	1bit	
0x0304	772	Charging high-temperature alarm	1bit	
0x0305	773	Charging low-temperature alarm	1bit	
0x0306	774	BMU communication fault alarm	1bit	
0x0307	775	RFU alarm	1bit	
0x0308	776	Temperature imbalance alarm	1bit	
0x0309	777	Cell imbalance alarm	1bit	
0x030A	778	Alarm for too low SOC	1bit	
0x030B	779	Alarm at low insulation level	1bit	
0x030C	780	Single overvoltage alarm	1bit	
0x030D	781	Single under-voltage alarm	1bit	
0x030E	782	Alarm for excessive discharge temperature	1bit	
0x030F	783	Alarm for too low discharge temperature	1bit	

3. Read equipment parameters (function code 0x03)

1) Upper computer request command format:

Definition	Address	Function code	Starting register address		Number of registers		CRC check	
Data	ADDR	03H	High order	Low order	High order	Low order	Low order	High order
Number of bytes	1	1	2		2		2	

Host request command format in TCP/IP mode:

Definition	MBAP header	Function code	Starting register address		Number of registers	
Data	---	03H	High order	Low order	High order	Low order
Number of bytes	7	1	2		2	

2) Normal response format:

Definition	Address	Function code	Number of reply data bytes	Returned data		CRC check	
Data	ADDR	03H	DATA_BYTES	High order	Low order	Low order	High order
Number of bytes	1	1	1	2 * number of registers		2	

Normal response format in TCP/IP mode:

Definition	MBAP header	Function code	Number of reply data bytes	Returned data	
Data	---	03H	DATA_BYTES	High order	Low order
Number of bytes	7	1	1	2* DATA_BYTES	

3) Abnormal response format:

Definition	Address	Error code	Exception code	CRC check	
Data	ADDR	83H	ERR_CODE	Low order	High order
Number of bytes	1	1	1	2	

Abnormal response format in TCP/IP mode:

Definition	MBAP header	Error code	Exception code
Data	---	83H	ERR_CODE
Number of bytes	7	1	1

Setting register table:

Address HEX	Register content	Data length / format	Remarks

0x8000	Reserved	2 bytes	
0x8001	Reserved	2 bytes	
0x8002	Buzzer silencing	2 bytes	Support reading/setting. 1: Buzzer silencing; 0: Buzzer operating normally
0x8003	Reserved	2 bytes	
0x8004	Reserved	2 bytes	
0x8005	Reserved	2 bytes	
0x8006	Reserved	2 bytes	
0x8007	Reserved	2 bytes	
0x8008	Start-up and shutdwon	2 bytes	Support settings. 1: shutdown / shutdown off output; 2: switch from shutdown to bypass; 3: startup;
0x8009	System Unix clock_Lo	2 bytes	Support reading/setting. 32-bit Unix clock, and two registers must be set simultaneously
0x800A	System Unix clock_Hi	2 bytes	
0x800B	Reserved	2 bytes	
0x800C	Reserved	2 bytes	
0x800D	Reserved	2 bytes	
0x800E	Reserved	2 bytes	

4. Set equipment parameters (function code 0x06/0x10)

4.1 0x06 Command frame format

1) Upper computer request command format:

Definition	Address	Function code	Register address		Register value		CRC check	
Data	ADDR	06H	High order	Low order	High order	Low order	Low order	High order
Number of bytes	1	1	2		2		2	

Host request command format in TCP/IP mode:

Definition	MBAP header	Function code	Register address		Register value	
Data	---	06H	High order	Low order	High order	Low order
Number of bytes	7	1	2		2	

2) Normal response format:

Definition	Address	Function code	Register address		Register value		CRC check	
Data	ADDR	06H	High order	Low order	High order	Low order	Low order	High order
Number of bytes	1	1	2		2		2	

Normal response format in TCP / IP mode:

Definition	MBAP header	Function code	Register address		Register value	
Data	---	06H	High order	Low order	High order	Low order
Number of bytes	7	1	2		2	

3) Abnormal response format:

Definition	Address	Error code	Exception code	CRC check	
Data	ADDR	86H	ERR_CODE	Low order	High order
Number of bytes	1	1	1	2	

Abnormal response format in TCP / IP mode:

Definition	MBAP header	Error code	Exception code
Data	---	86H	ERR_CODE
Number of bytes	7	1	1

Parameter setting register table: Same as 03 setting register table.

4.2 0x10 Command frame format

1) Upper computer request command format:

Definition	Address	Function code	Starting register address		Number of registers		Number of bytes	Register value	CRC check	
Data	ADDR	10H	High order	Low order	High order	Low order			Low order	High order
Number of bytes	1	1	2		2		1	2 * number of registers	2	

Host request command format in TCP/IP mode:

Definition	MBAP header	Function code	Starting address		Number of registers		Number of bytes	Register value	
Data	---	10H	High order	Low order	High order	Low order		High order	Low order
Number of bytes	7	1	2		2		1	2 * number of registers	

2) Normal response format:

Definition	Address	Function code	Starting register address		Number of registers		CRC check	
Data	ADDR	10H	High order	Low order	High order	Low order	Low order	High order
Number of bytes	1	1	2		2		2	

Normal response format in TCP/IP mode:

Definition	MBAP header	Function code	Register address		Number of registers	
Data	---	10H	High order	Low order	High order	Low order
Number of bytes	7	1	2		2	

3) Abnormal response format:

Definition	Address	Error code	Exception code	CRC check	
Data	ADDR	90H	ERR_CODE	Low order	High order
Number of bytes	1	1	1	2	

Abnormal response format in TCP / IP mode:

Definition	MBAP header	Error code	Exception code	
Data	---	90H	High order	Low order
Number of bytes	7	1	2	

Parameter setting register table: Same as 03 setting register table.

Appendix A CRC check

CRC Cyclic redundancy check

The cyclic redundancy check (CRC) field consists of two bytes and contains a binary 16-bit value. The CRC value attached after the message is calculated by the transmitting equipment. When receiving the header, the receiving equipment recalculates the CRC value and compares the calculated result with the actually received CRC value. The two values are not same, indicating error occurs.

CRC is calculated by first preloading a 16-bit register as per 1 and then calculated subsequently by continuous 8-bit byte in the header. Only 8 data bits in the character are related to the calculation and generation of CRC, in which the start bit, stop bit and check bit are not related to such calculation and generation. During the generation of CRC, each 8-bit character is XOR-related to the value in the register. Then, the result shifts 1 bit towards the least significant bit (LSB), and the position of the most significant bit (MSB) is filled with zero. Then extract and check LSB: if LSB is 1, the value in the register is XOR-related to a fixed preset value; If LSB is 0, no XOR operation is performed. This process will be repeated till 8 shifts are completed. After the last (the eighth) shift and related operation are completed, the next 8-bit byte is XOR-related to the current value of the register. Then repeat for 8 times as described above. After calculation of bytes in all messages, the obtained final value in the register is CRC.

The process of generating CRC:

1. Load a 16-bit register into hex FFFF (all 1) to get CRC register.

2. The first 8-bit byte of the message is XOR-related to the low byte of the 16-bit CRC register. Then the result is placed in the CRC register.

3. Shift the CRC register to the right by 1 bit (towards LSB), fill zero at MSB, and extract and detect LSB.

4. (If LSB is 0): repeat step 3 (another shift)

(If LSB is 1): XOR polynomial value 0xa001 (1010 0000 0001) for CRC register

5. Repeat the steps 3 and 4 till 8 shifts are completed. After this operation is completed, the complete operation of 8-bit bytes will be completed.

6. Repeat the steps 2 to 5 for the next byte in the message, and continue this operation till all messages are processed.

7. The final content in the CRC register is the CRC value.

8. When the CRC value is placed in the message, as described below, the high and low bytes must be exchanged.

Place CRC in the message. When 16-bit CRC (two 8-bit bytes) is transmitted in the message, the low bytes are sent first, and then the high bytes.

Example: The functions of the C language that executes CRC generation are shown below. All possible CRC values are pre installed in two arrays, which can be simply indexed when calculating the message content. One array contains all 256 possible high-order bytes of 16 bit CRC field, and the other array contains the values of low-order bytes. This way of accessing CRC by index provides a faster way to calculate a new CRC than every new character in the message buffer.

Note: this function internally performs high / low CRC byte exchange. This function returns the CRC value that has been exchanged.

In other words, the CRC value returned from this function can be directly placed in the message for transmission.

The function takes two arguments:

unsigned char *puchMsg;Pointer to the buffer containing the binary data message used to generate CRC

unsigned short usDataLen; Number of bytes of message buffer

CRC generation function

```
unsigned short CRC16 (puchmsg, usdatalen) / * function returns CRC as unsigned short*/
unsigned char *puchMsg ; /* Message used to calculate CRC*/
unsigned short usDataLen ; /* Number of bytes in message*/
{
    unsigned char uchCRCHi = 0xFF ; /* CRC high byte initialization*/
    unsigned char uchCRCLo = 0xFF ; /* Low byte initialization of CRC*/
    unsigned uIndex ; /* CRC query table index*/
    While (usdatalen --) / * complete the whole message buffer*/
    {
```

```
    uIndex = uchCRCLo ^ *puchMsgg++ ; /* Calculate CRC*/
    uchCRCLo = uchCRCHi ^ auchCRCHi[uIndex] ;
    uchCRCHi = auchCRCLo[uIndex] ;
}
 return (uchCRCHi << 8 | uchCRCLo) ;
```

Appendix B High and low byte table

High byte table

/*CRC value of high byte*/

static unsigned char auchCRCHi[] = {

0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41,

0x00, 0xC1, 0x81,

0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81,

0x40, 0x01, 0xC0,

0x80, 0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1,

0x81, 0x40, 0x01,

0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01,

0xC0, 0x80, 0x41,

0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,

0x00, 0xC1, 0x81,

0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80,

0x41, 0x01, 0xC0,

0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0,

0x80, 0x41, 0x01,

0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x00,

0xC1, 0x81, 0x40,

0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41,

0x00, 0xC1, 0x81,

0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81,

0x40, 0x01, 0xC0,

0x80, 0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1,

0x81, 0x40, 0x01,

0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x01,

0xC0, 0x80, 0x41,

```
0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
0x00, 0xC1, 0x81,
0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81,
0x40, 0x01, 0xC0,
0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0,
0x80, 0x41, 0x01,
0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01,
0xC0, 0x80, 0x41,
0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41,
0x00, 0xC1, 0x81,
0x40
} ;
```

Low byte table

```
/*CRC value of low byte*/
static char auchCRCLo[] = {
0x00, 0xC0, 0xC1, 0x01, 0xC3, 0x03, 0x02, 0xC2, 0xC6, 0x06, 0x07, 0xC7,
0x05, 0xC5, 0xC4,
0x04, 0xCC, 0x0C, 0x0D, 0xCD, 0x0F, 0xCF, 0xCE, 0x0E, 0x0A, 0xCA, 0xCB,
0x0B, 0xC9, 0x09,
0x08, 0xC8, 0xD8, 0x18, 0x19, 0xD9, 0x1B, 0xDB, 0xDA, 0x1A, 0x1E, 0xDE,
0xDF, 0x1F, 0xDD,
0x1D, 0x1C, 0xDC, 0x14, 0xD4, 0xD5, 0x15, 0xD7, 0x17, 0x16, 0xD6, 0xD2,
0x12, 0x13, 0xD3,
0x11, 0xD1, 0xD0, 0x10, 0xF0, 0x30, 0x31, 0xF1, 0x33, 0xF3, 0xF2, 0x32,
0x36, 0xF6, 0xF7,
0x37, 0xF5, 0x35, 0x34, 0xF4, 0x3C, 0xFC, 0xFD, 0x3D, 0xFF, 0x3F, 0x3E,
0xFE, 0xFA, 0x3A,
0x3B, 0xFB, 0x39, 0xF9, 0xF8, 0x38, 0x28, 0xE8, 0xE9, 0x29, 0xEB, 0x2B,
```

0x2A, 0xEA, 0xEE,

0x2E, 0x2F, 0xEF, 0x2D, 0xED, 0xEC, 0x2C, 0xE4, 0x24, 0x25, 0xE5, 0x27,

0xE7, 0xE6, 0x26,

0x22, 0xE2, 0xE3, 0x23, 0xE1, 0x21, 0x20, 0xE0, 0xA0, 0x60, 0x61, 0xA1,

0x63, 0xA3, 0xA2,

0x62, 0x66, 0xA6, 0xA7, 0x67, 0xA5, 0x65, 0x64, 0xA4, 0x6C, 0xAC, 0xAD,

0x6D, 0xAF, 0x6F,

0x6E, 0xAE, 0xAA, 0x6A, 0x6B, 0xAB, 0x69, 0xA9, 0xA8, 0x68, 0x78, 0xB8,

0xB9, 0x79, 0xBB,

0x7B, 0x7A, 0xBA, 0xBE, 0x7E, 0x7F, 0xBF, 0x7D, 0xBD, 0xBC, 0x7C, 0xB4,

0x74, 0x75, 0xB5,

0x77, 0xB7, 0xB6, 0x76, 0x72, 0xB2, 0xB3, 0x73, 0xB1, 0x71, 0x70, 0xB0,

0x50, 0x90, 0x91,

0x51, 0x93, 0x53, 0x52, 0x92, 0x96, 0x56, 0x57, 0x97, 0x55, 0x95, 0x94,

0x54, 0x9C, 0x5C,

0x5D, 0x9D, 0x5F, 0x9F, 0x9E, 0x5E, 0x5A, 0x9A, 0x9B, 0x5B, 0x99, 0x59,

0x58, 0x98, 0x88,

0x48, 0x49, 0x89, 0x4B, 0x8B, 0x8A, 0x4A, 0x4E, 0x8E, 0x8F, 0x4F, 0x8D,

0x4D, 0x4C, 0x8C,

0x44, 0x84, 0x85, 0x45, 0x87, 0x47, 0x46, 0x86, 0x82, 0x42, 0x43, 0x83,

0x41, 0x81, 0x80,

0x40

};

