HMC4300

MARINE DIESEL ENGINE CONTROLLER

COMMUNICATION PROTOCOL

CONTENT

SmartGen

SmartGen Registered trademark

No. 28 Xuemei Street, Zhengzhou, Henan, China

Tel: +86-371-67988888/67981888/67992951

   +86-371-67981000(overseas)

Fax: +86-371-67992952

Web: www.smartgen.com.cn/

   www.smartgen.cn/

Email: sales@smartgen.cn

SmartGen reserves the right to change the contents of this document without prior notice.

Table 1 Software Version

Date	Version	Content
2024-2-26	V1.0	Original release.

1   DESCRIPTION

This protocol describes the controller serial ports' read and write command format, and the definition of internal information & data for the third-party to develop and use.

MODBUS communication protocol allows the controller to transfer information and data effectively with PLC, RTU, SCADA system, and DCS of international brands, such as Schneider, Siemens, and Modicon, etc., or the third-party supervisory control system that is   compatible with MODBUS. The system can be set up by installing the central communication master control software (such as KingView, Intouch, FIX, Synall, etc.) on PC (or IPC).

Modbus basic rules:

——All communication transmission should follow the master-slave mode. Data can be transmitted between a master (e.g. PC) and 32 slaves by this mode.

——No request communication can be sent from slave.

——All communication should be transmitted in "frame".

——If master or slave receives frame which contains unknown command, no response will be sent.

Message frame format:

Communication transmission is in asynchronously way, which uses byte (or frame) as data unit. Every frame will be 10-bit(1 stop bit) or 11-bit(2 stop bits) serial data when it is transmitted between master and slaves.

Table 2 Frame format

Item	Bits
Start Bit	1 bit
Data Bit	8 bits
Parity Bit	None
Stop Bit	1 bit
Transmission Baud Rate	9600bps

Error Check Code:

The Error Check Code allows the master or slave to check if the frame or packet has errors. Sometimes, the information occurs imperceptible changes due to electronic noise and other interference when transmitted, and the error check code ensures the error information does not work to increase the system's reliability and efficiency. Error check code adapts CRC-16 method.

The CRC code contains two bytes, and the low-order byte is appended first, followed by the high-order byte.

NOTE: All message frame format is same: address code, function code, data field and CRC code.

The CRC code contains two bytes, which is a 16-bit binary value. The CRC code is calculated by the transmitting device, and follows the entire data/message. The receiving device recalculates the CRC code of the message received, and compares the recalculated value to the actual value it received. If the two values are not equal, an error occurs.

The calculation method of CRC code is started by first preloading a 16-bit register to all 1's. Then a process begins of applying successive 8–bit bytes of the message to the current contents of the register. Only the eight bits of data in each character are used for generating the CRC code. Start and stop bits do not apply to the CRC code.

During generation of the CRC code, each 8-bit character is exclusive OR with the register contents. Then the result is shifted in the direction of the least significant bit (LSB), with a zero filled into the most significant bit (MSB) position. The LSB is extracted and examined. If the LSB was 1, the register is then exclusive OR with a preset, fixed value. If the LSB was 0, no exclusive OR takes place.

This process is repeated until eight shifts have been performed. After the last (eighth) shift, the next 8-bit byte is exclusive OR with the register's current value, and the process repeats for eight more shifts as described above. The final contents of the register, after all the bytes of the message have been applied, is the value of CRC code.

CRC-16 calculation procedure:

1) Load a 16-bit register with FFFF hex (all 1's). Call this the CRC register;

2) Exclusive OR the first 8-bit byte of the message with the LSB of the CRC register, putting the result in the CRC register;

3) Shift the CRC register one bit to the right (toward the LSB), zero−filling the MSB. Extract and examine the LSB;

4) If the LSB was 0: Repeat Step 3 (another shift);
   If the LSB was 1: Exclusive OR the CRC register with the A001 hex;

5) Repeat Step 3 and Step 4 until 8 shifts have been performed. When this is done, a complete 8−bit byte has been processed;

6) Repeat Step 2 to Step 5 for the next 8−bit byte of the message. Continue repeating this procedure until all bytes have been processed;

7) The final contents of the CRC register are the CRC value. When the 16−bit CRC (two 8−bit bytes) is transmitted in the message, the low order byte will be transmitted first, followed by the high order byte.

NOTE: The calculating of CRC code starts from ＜slave address＞ and except for all bytes of <CRC code>.

SmartGen
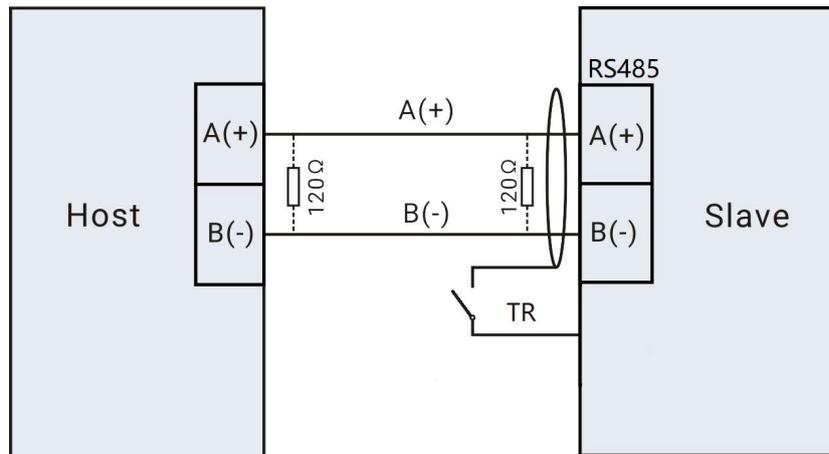
Fig. 1 Single Device Communication Wiring Diagram

NOTE 1: Two 120-ohm resistors can be connected on both ends of the cable according to the site's requirement.

NOTE 2: The controller has 120-ohm resistors inside it, and after the "RS485 TR" parameter is enabled, the RS485 bus will connect with the 120-ohm resistors in parallel.

3   CONTROLLER INTERNAL REGISTER ADDRESS AND DATA

3.1   FUNCTION CODE 03H MAPPING ALARM, COIL STATUS AND DATA FIELD

Table 3 Alarm, Coil Status and Data Field

Address	Item	Description	Bytes
0000	Common Alarm	1 as active (low order)	1 bit
	Common Shutdown	1 as active	1 bit
	Common Warning	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	In Local Mode	1 as active	1 bit
	In Remote Mode	1 as active	1 bit
	Override Mode	1 as active	1 bit
	Self-check Mode	1 as active	1 bit
	Auto Mode	1 as active	1 bit
	Reserved	1 as active	1 bit
	Normal Running	1 as active	1 bit
	Reserved	1 as active (high order)	1 bit
0001	Emergency Stop	1 as active	1 bit
	Overspeed Shutdown	1 as active	1 bit
	Underspeed Shutdown	1 as active	1 bit
	Loss of Speed Signal Shutdown	1 as active	1 bit
	ECU Shutdown	1 as active	1 bit
	ECU Comm. Failure Shutdown	1 as active	1 bit
	Remote Emergency Stop	1 as active	1 bit
	Reserved	1 as active	1 bit
	Water Temp. Shutdown Input	1 as active	1 bit
	Oil Temp. Shutdown Input	1 as active	1 bit
	Oil Pressure Shutdown Input	1 as active	1 bit
	ECU High Water Temp. Shutdown	1 as active	1 bit
	ECU High Oil Temp. Shutdown	1 as active	1 bit
	ECU Low Oil Pressure Shutdown	1 as active	1 bit
	High RPU 1 Shutdown	1 as active	1 bit
	Low RPU 1 Shutdown	1 as active	1 bit
0002	Sensor 1 High Shutdown	1 as active	1 bit
	Sensor 1 Low Shutdown	1 as active	1 bit
	Sensor 2 High Shutdown	1 as active	1 bit
	Sensor 2 Low Shutdown	1 as active	1 bit
	Sensor 3 High Shutdown	1 as active	1 bit
	Sensor 3 Low Shutdown	1 as active	1 bit

Address	Item	Description	Bytes
	Sensor 4 High Shutdown	1 as active	1 bit
	Sensor 4 Low Shutdown	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
0003	Input 1 Shutdown	1 as active	1 bit
	Input 2 Shutdown	1 as active	1 bit
	Input 3 Shutdown	1 as active	1 bit
	Input 4 Shutdown	1 as active	1 bit
	Input 5 Shutdown	1 as active	1 bit
	Input 6 Shutdown	1 as active	1 bit
	High RPU 2 Shutdown	1 as active	1 bit
	Low RPU 2 Shutdown	1 as active	1 bit
	Emergency Stop (RPU)	1 as active	1 bit
	Input 1 Shutdown (RPU)	1 as active	1 bit
	Input 2 Shutdown (RPU)	1 as active	1 bit
	Input 3 Shutdown (RPU)	1 as active	1 bit
	Input 4 Shutdown (RPU)	1 as active	1 bit
	Input 5 Shutdown (RPU)	1 as active	1 bit
	RPU Overspeed Shutdown	1 as active	1 bit
	Emergency Stop BW Warning (RPU)	1 as active	1 bit
0004	Overspeed Warning	1 as active	1 bit
	Underspeed Warning	1 as active	1 bit
	Loss of Speed Signal Warning	1 as active	1 bit
	Charging Failure Warning	1 as active	1 bit
	Start Failure Warning	1 as active	1 bit
	Stop Failure Warning	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	RPU560 Comm. Failure	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	Fuel Leakage Warning Input	1 as active	1 bit
0005	Sensor 1 Open Warning		
	Sensor 1 High Warning		

Address	Item	Description	Bytes
	Sensor 1 Low Warning		
	Reserved		
	Sensor 2 Open Warning		
	Sensor 2 High Warning		
	Sensor 2 Low Warning		
	Reserved		
	Sensor 3 Open Warning		
	Sensor 3 High Warning		
	Sensor 3 Low Warning		
	Reserved		
	Sensor 4 Open Warning		
	Sensor 4 High Warning		
	Sensor 4 Low Warning		
	Reserved		
0006	High Water Temp. Warning		
	High Oil Temp. Warning		
	Low Oil Pressure Warning		
	Reserved		
	Reserved		
	Reserved		
	Input 1 Warning		
	Input 2 Warning		
	Input 3 Warning		
	Input 4 Warning		
	Input 5 Warning		
	Input 6 Warning		
	ECU Warning		
	RPU Speed Sensor BW Warning		
	Power Overvoltage Warning		
	Power Undervoltage Warning		
0007	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	RPU Input 1 BW Warning		
	RPU Input 2 BW Warning		
	RPU Input 3 BW Warning		
	RPU Input 4 BW Warning		
	RPU Input 5 BW Warning		
	RPU Fuel Relay BW Warning		
	RPU Shutdown BW Warning		
	RPU Speed Sensor BW Warning		

Address	Item	Description	Bytes
	RPU Battery 1 Fault		
	RPU Battery 2 Fault		
0008	Input 1 Indicator	1 as active	1 bit
	Input 2 Indicator	1 as active	1 bit
	Input 3 Indicator	1 as active	1 bit
	Input 4 Indicator	1 as active	1 bit
	Input 5 Indicator	1 as active	1 bit
	Input 6 Indicator	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
0009	Input 1 Active		2Bytes
	Input 2 Active		
	Input 3 Active		
	Input 4 Active		
	Input 5 Active		
	Input 6 Active		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
0010	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		

Address	Item	Description	Bytes
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
0011	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
0012	RPU Emergency Stop Status	1 as active	1 bit
	RPU Input 1 Status	1 as active	1 bit
	RPU Input 2 Status	1 as active	1 bit
	RPU Input 3 Status	1 as active	1 bit
	RPU Input 4 Status	1 as active	1 bit
	RPU Input 5 Status	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	RPU Output 1 Status	1 as active	1 bit
	RPU Output 2 Status	1 as active	1 bit
	RPU Output 3 Status	1 as active	1 bit
	RPU Output 4 Status	1 as active	1 bit
	RPU Output 5 Status	1 as active	1 bit
	RPU Power Status	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
0013	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		

Address	Item	Description	Bytes
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
0014	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
0015	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
0016	Emergency Stop Input Status	1 as active	1 bit
	Reserved	1 as active	1 bit

Address	Item	Description	Bytes
	Reserved	1 as active	1 bit
	Input 1 Status	1 as active	1 bit
	Input 2 Status	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	Input 3 Status	1 as active	1 bit
	Input 4 Status	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
0017	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Output 1 Status		
	Output 2 Status		
	Output 3 Status		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
0018	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		

Address	Item	Description	Bytes
	Reserved		
	Reserved		
0019	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
	Reserved		
0020	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
	Reserved	1 as active	1 bit
0021	Speed	16-bit Unsigned	2Bytes
0022	ECU Water Temp.	16-bit Signed	2Bytes
0023	ECU Oil Temp.	16-bit Signed	2Bytes
0024	ECU Oil Pressure	16-bit Signed	2Bytes
0025	Reserved		2Bytes
0026	Sensor 1 Data	16-bit Signed	2Bytes
0027	Sensor 2 Data	16-bit Signed	2Bytes
0028	Sensor 3 Data	16-bit Signed	2Bytes
0029	Sensor 4 Data	16-bit Signed	2Bytes
0030	Power Supply Voltage	16-bit Signed	2Bytes

Address	Item	Description	Bytes
0031	Charger Voltage	16-bit Signed	2Bytes
0032	Engine Running Status	Table 13 Engine Running Status	2Bytes
0033	Status Delay	16-bit Unsigned	2Bytes
0034	Remote Start Status	16-bit Unsigned	2Bytes
0035	Remote Start Delay	16-bit Unsigned	2Bytes
0036	Reserved		2Bytes
0037	Total Running Time	32-bit Unsigned, Unit-Hour, One decimal place	4Bytes
0038	Total Start Times	16-bit Unsigned	2Bytes
0039	Reserved		2Bytes
0040	Reserved		2Bytes
0041	Controller Model	16-bit Unsigned	2Bytes
0042	Controller Software Version	16-bit Unsigned	2Bytes
0043	Controller Hardware Version	16-bit Unsigned	2Bytes
0044	Release Year	16-bit Unsigned	2Bytes
0045	Release Month	16-bit Unsigned	2Bytes
0046	Released Day	16-bit Unsigned	2Bytes
0047	Reserved		2Bytes
0048	Total Event Logs	16-bit Unsigned	2Bytes
0049	Reserved		2Bytes
0050	Reserved		2Bytes
0051	Reserved		2Bytes
0052	Reserved		2Bytes
0053	Reserved		2Bytes
0054	Reserved		2Bytes
0055	Reserved		2Bytes
0056	Sensor 1 Resistance	16-bit Unsigned	2Bytes
0057	Sensor 2 Resistance	16-bit Unsigned	2Bytes
0058	Sensor 3 Resistance/Current/Voltage Value	16-bit Unsigned	2Bytes
0059	Sensor 4 Resistance	16-bit Unsigned	2Bytes
0060	Reserved		2Bytes
0061	Reserved		2Bytes
0062	RPU Speed Value	16-bit Unsigned	2Bytes
0063	Reserved		2Bytes
0064	Coolant Level	16-bit Signed	2Bytes
0065	Oil Temp.	16-bit Signed	2Bytes
0066	Coolant Pressure	16-bit Signed	2Bytes
0067	Fuel Pressure	16-bit Signed	2Bytes
0068	Fuel Temp.	16-bit Signed	2Bytes
0069	Air Inlet Temp.	16-bit Signed	2Bytes
0070	Exhaust Outlet Temp.	16-bit Signed	2Bytes
0071	Turbo Pressure	16-bit Signed	2Bytes

Address	Item	Description	Bytes
0072	Fuel Consumption	16-bit Signed	2Bytes
0073	Total Fuel Consumption	32-bit Unsigned	4Bytes

Example 1 - Read the alarm detail:

If "ECU High Water Temp. Shutdown" needs to be read, check the table above and find its coil address is 0001, so it needs to read one data address.

Assuming the slave (controller) address is 01, the master or host (could be PC) request command is as following:

Table 4 Master (PC) Request Frame

Slave Address	Function Code	Start Address (0001)		Request Data Length(1)		CRC 16	
		MSB	LSB	MSB	LSB	LSB	MSB
01	**03**	**00**	**01**	**00**	**01**	**D5**	**CA**

The slave response is as following:

Table 5 Slave (Controller) Response Frame

Slave Address	Function Code	Data Length (Bytes)	Data of Address 0001	CRC 16	
				LSB	MSB
01	**03**	**02**	**08 00**	**BF**	**84**

Table 6 Data Analysis

Address	Data Received (Hexadecimal)	Convert to Binary	Meaning
0001	08H 00H	0000 1000 0000 0000	Data of 12 is 1, which means "ECU High Water Temp. Shutdown" is active

SmartGen

Example 2 - Read the monitoring data:

If "ECU Water Temp.(current value is 99°C)" needs to be read, check the table above and find its coil address is 0022, so it needs to read one data address.

Assuming the slave address is 01, the master request command is as following:

Table 7 Master Request Command

Slave Address	Function Code	Start Address (0022)		Request Data Length(1)		CRC 16	
		MSB	LSB	MSB	LSB	LSB	MSB
01	03	00	16	00	01	65	CE

The slave response command is as following:

Table 8 Slave Response Command

Slave Address	Function Code	Data Length (Bytes)	Data		CRC 16	
			Data of Address 0022 MSB	Data of Address 0022 LSB	LSB	MSB
01	03	02	00	63	F8	6D

Fill the data received into the address respectively, as shown in the table below.

Table 9 Data Analysis

Address	Data Received (Hex)	ECU Water Temp. (Decimal)
0022	0063H	99

3.2 FUNCTION CODE 05H MAPPING REMOTE COIL FIELD

Table 10 Remote Coil Field

Address	Item	Description
0000H	Remote Start Key	Active only when sending FF00
0001H	Remote Stop Key	Active only when sending FF00
0002H	Reserved	Active only when sending FF00
0003H	Reserved	Active only when sending FF00
0004H	Reserved	Active only when sending FF00
0005H	Remote Reset Key	Active only when sending FF00
0006H	Reserved	Active only when sending FF00
0007H	Reserved	Active only when sending FF00
0008H	Reserved	Active only when sending FF00
0009H	Reserved	Active only when sending FF00
000AH	Reserved	Active only when sending FF00
000BH	Reserved	Active only when sending FF00
000CH	Remote Mute Key	Active only when sending FF00

NOTE 1: The remote command in the table above only needs to be sent once.

NOTE 2: When using Modbus address to communicate: it needs to send FF00H to load corresponding address as 1, and send 0000H to load corresponding address as 0.

Example:

If taking Remote Reset Key as example, check the table first and find its remote address is 0005. Assuming the slave address is 01, the master request command is as following:

Table 11 Master Request Command

Slave Address	Function Code	Remote Address(0005)		Remote Data		CRC 16	
		MSB	LSB	MSB	LSB	LSB	MSB
01	05	00	05	FF	00	9C	3B

The slave response command is as following:

Table 12 Slave Response Command

Slave Address	Function Code	Remote Address(0003)		Remote Data		CRC 16	
		MSB	LSB	MSB	LSB	LSB	MSB
01	05	00	05	FF	00	9C	3B

Checking the alarm status in the controller will confirm if the action of alarm reset is performed.

3.3 ENGINE RUNNING STATUS

Table 13 Engine Running Status

No.	Content	Description
0	Standby	No delay value in the status
1	Preheat	
2	Fuel Output	No delay value in the status
3	Crank	
4	Crank Rest	
5	Safety on Delay	
6	Start Idle	
7	Warming up	
8	Wait for On-load	No delay value in the status
9	Normal Running	No delay value in the status
10	Cooling	
11	Stop Idle	
12	ETS	
13	Wait for Stop	
14	Stop Failure	No delay value in the status

SmartGen

4 FAQ

4.1 GROUNDING OF THE CABLE SHIELD

To prevent the coupling of interference on the cable, one end of the cable shield should be grounded.

4.2 TERMINATION RESISTOR

At both ends of the linear network (between the two communication ports furthest apart), two 120-ohm termination resistors need to be installed in parallel. According to the signal transmission theory, the termination resistor can avoid the signal reflections and improve the signal integrity effectively. The value of two termination resistor in parallel is basically equal to the characteristic impedance of the transmission cable.

A standard RS-485 network will usually use the termination resistor. The resistor can be avoided while the cable is too short, or it is a temporary or lab test.

4.3 RS485 TO USB CONVERTER

It can communicate with PC via the SmartGen SG72A converter.

4.4 SOLUTIONS FOR COMMUNICATION FAILURE

1) Check the positive and negative of RS485, or network cable is connected correctly. Check the RS485 converter (if any) is normal;

2) Check the termination resistors are connected correctly or not;

3) Check the communication parameters setting is correct or not. Baud rate, data bit, parity bit and stop bit meet the requirement of controller;

4) Check the Terminal COM is connected correctly with the USB port of PC via RS485 converter;

5) Check the communication address of controller is correct, and the default address is 01;

6) When using function code 03, the maximum data length to be read is 120 addresses, and the ending address can't exceed the greatest one of Modbus communication address;

7) If there is offset address in the Modbus communication address, the actual Modbus communication address equals to the base address plus offset address;

8) Function code 05 uses Modbus address to communicate: Although 1 means active, and 0 means inactive, it needs to send FF00H to load corresponding bit as 1, and send 0000H to load corresponding bit as 0.

9) As for CRC-16, the low-order byte is checked first, the high-order byte is checked later.

10) The frequency of multiple read operations for controller data should not too high, and the recommended interval between two read operations is no less than 500ms;

11) Please configure each controller's communication module address before networking. Same module address is not allowed in one network.

12) Modbus serial protocol does not support multiple masters, so multiple software can not communicate with the controller at the same time;

13) Disconnect the RS485 cables to the controller, test the voltage difference of RS485 Terminal A and B on the controller, if the result is between -200mV and +200mV, it means the communication port is abnormal;

14) If the cable length is too long, it will result in signal attenuation. Therefore, it is recommended to use high-quality cable or add repeaters in the cable.

_____