CNS-BL30HB

DC Brushless Driver

User Manual (Communication)

Sincerely thank you for using our company's products

This manual explains the usage and safety of the product

*Familiar with this manual, and pay attention to safety during use.

*Keep this manual and put it in a suitable place for reference at any time.

Foshan Hinson Robot Technology Co.Ltd

Contents

File version	Applicable firmware version	Modify content
V1.02	V1.1.0	Added CAN OPEN part PDO default mapping description

1 FOREWORD

The use of communication control can greatly increase the difficulty of stable operation of the drive and subsequent maintenance. The driver allows control through various communication methods. This article will describe the communication control in detail.

1.1 AUXILIARY EQUIPMENT

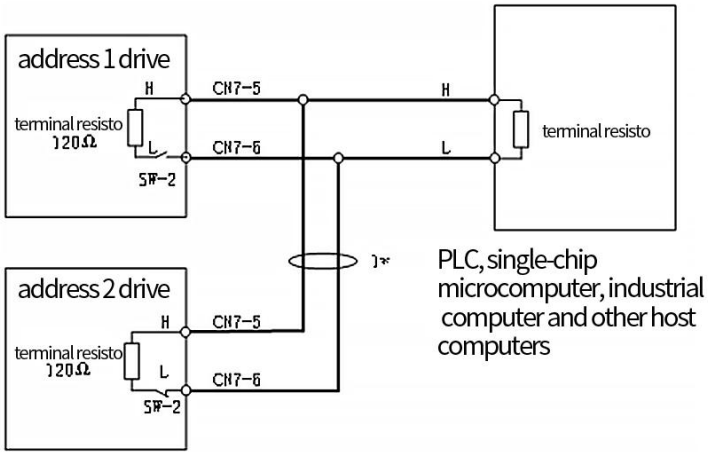
Auxiliary debugging methods are needed in communication debugging to speed up the communication debugging process. Auxiliary debugging requires the following software and hardware:

Control method	Requires software and hardware	Number	Effect
RS232 free protocol control	USB to RS232 serial cable, Recommended brands and models: Superior--Y105 Serial debugging assistant software	1	Drive parameter configuration; function test; RS232 control debugging;
RS485 free protocol control **RS485 MODBUS protocol control**	USB to RS485 serial cable Serial debugging assistant software	1	function test; RS485 control debugging;
CAN OPEN protocol control **CAN BUS free protocol control**	USB to CAN Analyzer	1	function test; CAN BUS control debugging

2 RS-485 MODUBS COMMUNICATION CONTROL

2.1 CONNECTION

The driver uses 2-wire 485 communication RS485_A(TR+) and RS485_B(TR-), and can be connected and operated with multiple RS485 devices. The driver RS485 port is located at CN7-17/18;



RS485 is a single-master multi-slave communication mode, and the devices at both ends of the communication need to be equipped with terminal resistors.

1*connection cable

The cable is recommended to use twisted pair and shielded;

2*connection line sequence

Usually, all A and A, B and B terminals are connected according to the RS485 standard, and reverse connection will cause communication failure.

2.2 CONTROLLED BY MOUDBUS PROTOCOL

Basic information

The host computer needs to determine the parameters to set the MODBUS communication:

Slave address (slave): The communication address range of the drive device is 1-127;

Read and write mode (mode): 04 (read status parameters) and 16 (write control parameters);

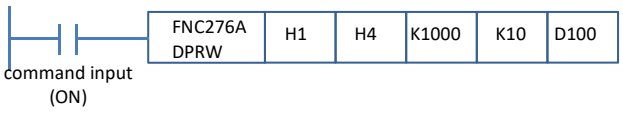
Start address (add): 1000 (read start register address) and 2000 (write start register address);

Number of reads (count): the number of registers read at a time;

MODBUS read data and write data are performed separately, so when writing the host computer program, they need to be processed separately.

Take Mitsubishi PLC FX3u series MODBUS communication as an example, after completing the serial port initialization parameter configuration, use the MODBUS communication instruction to explain:

read data



H1: slave driver address 1;

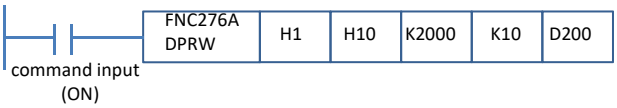
H4: read mode 04 (read input register);

K1000: register start address 1000;

K10: read data from 10 registers of slave station 1000-1009;

D100: Map the PLC register address, store the data of 10 registers from 1000 to 1009 to the D109 position from D100 to;

data input



H1: slave driver address 1;

H10: Write mode 16 (write holding register);

K1000: The start address of the holding register is 2000;

K10: Write a total of 10 register data of the slave station 2000-2009;

D200: Map the PLC register address, write the data of 10 registers from D200 to D209 to the slave station 2000 to 2009;

After completing the above settings, search the driver register function table, and directly operate the D__ register data according to the requirements to realize the control of the driver. To communicate with Siemens S7-200 series MODBUS, Siemens PLC internally redefined the MODBUS register address as follows;

Modbus Master functions supported by the protocol library

In order to support the reading and writing of the above Modbus addresses, the Modbus Master protocol library requires the slave station to support the following functions:

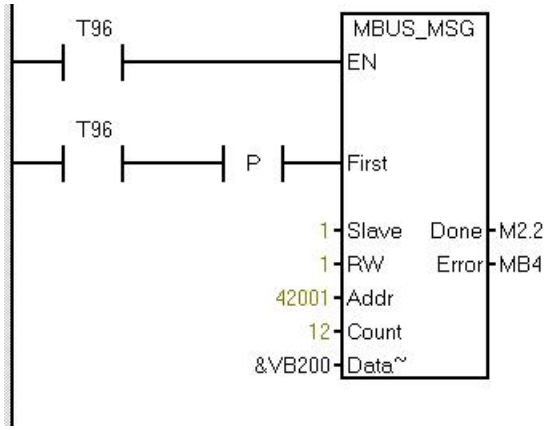
Modbus address	read/write	Modbus functions to be supported by the slave station
00001-09999 digital output	read	function 1
	write	function 5: write single output point function 15: write multiple output points
10001-19999 digital input	read	function 2
	write	–
30001-39999 digital input	read	function 4
	write	–
40001-49999 digital input	read	function 3
	write	function 6: write single register unit function 16: write multiple register unit

According to Siemens' address mapping:

The start address of the input register read from the driver 04 function code is 31001 corresponding to 1000;

16 The starting address of the function code writing holding register is 42001 corresponding to 2000;

A register in MODBUS is 2Btye in length, and when it is generally stored in the VB__ register in the S7-200, it will occupy two VB registers.



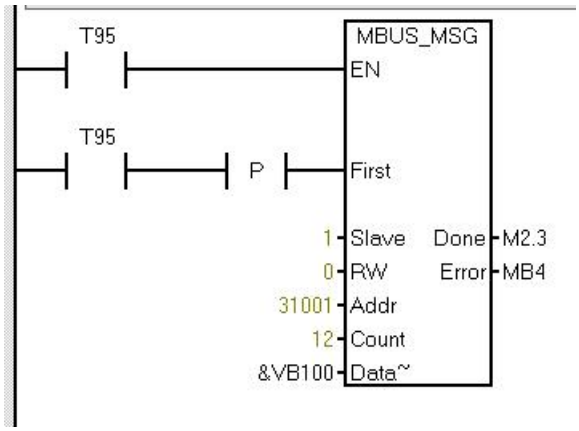
Slave-1：Slave driver address 1;

RW-1: write mode;

Addr-42001: function code 16 (write to holding register) from register start address 2000;

Count-11: write a total of 11 register data of the slave station 2000-2010;

DataPtr-&VB200: read a total of 22Byte data from VB200-VB222;



Slave-1: slave driver address 1;

RW-0: read mode;

Addr-31001: Function code 04 (write to holding register) starts from register address 1000;

Count-12: read data from 12 registers of slave station 1000-1010;

DataPtr-&VB100: read a total of 22Byte data from VB100-VB122;

2.2.1 Communication Specifications

Before communication, it is necessary to set the basic parameters of the serial port. The following parameters are the basis for establishing communication.

	Parameter	Default setting
way of communication	Single master/multiple slave mode	
Operating mode	MODBUS RTU mode	RTU mode
Receive/send method	half-duplex communication mode	
Communication ID	1-127 configurable	1/127
communication rate	9600bps, 19200bps, 38400bps, 576000bps, 115200bps	115200bps

data bit	8 bits	8 bit
Check Digit	no parity; odd parity; even parity	No parity
stop bit	1 digit; 2 digits	1 bit

2.2.2 Communication settings

To use the RS485 communication MODBUS protocol, it is necessary to set the driver through Micro-USB or RS232 before using it. Parameter configuration is performed through the Hinson configuration software on the computer side. The configuration parameters are As follows:

Parameter name	Parameter list	Setting parameters
Device interface and protocol	1. IO analog voltage control 2. RS232-HS protocol 3. RS485-HS protocol 4. CAN bus-HS protocol 5. RS485-MODBUS protocol 6. CAN OPEN protocol	RS485-MODBUS protocol
Device address	1-127	127
RS485 communication baud rate	1：9600bps 2：19200bps 3：38400bps 4：57600bps 5：115200bps	115200bps
RS485 parity check	1: no parity 2: Odd parity 3: even parity	no parity
Communication interruption protection time	0.05s-1s	0.5s

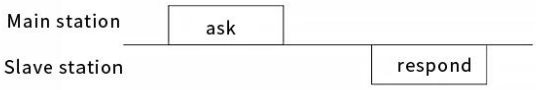
2.2.3 Master-slave station communication sequence

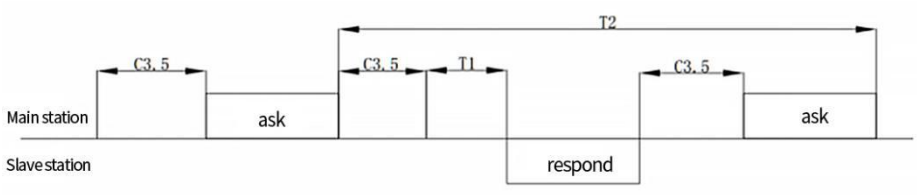
The communication mode of MODBUS protocol is single master station/multi-slave station mode,

unicast mode

The master station sends an inquiry to one slave station.

The slave executes the processing and returns a response.





NO	Name	content
C3.5	static interval	As the transmission waiting time, be sure to leave an interval of 3.5 characters or more. 3.5 If the number of words is not enough, the drive cannot respond. When the communication speed exceeds 19200 bps, please leave an interval of 1.75 ms or more.
T1	Send waiting time	The time from when the slave station switches the communication line to the sending state after receiving an inquiry from the master station to when it starts to reply. To send the waiting time of 10 ms. Actual send wait time becomes static interval (C3.5) + command processing time + send wait time (T1)
T2	communication timeout	Configurable from 0.05s – 1s

Important Hint

Since RS485 communication cannot be sent and received at the same time, it is recommended that the communication interval be controlled at 30-50ms in the communication sequence. And after confirming that the last sending is completed, continue the next sending.

2.2.4 Register address function

Write Holding Register Function (16)

register address	write value	content
2000	00 - 01	Drive Fault Clear When the drive is in the state of reporting errors, it is necessary to reset the fault of the drive. Send by default: 00, Reset operation: 01 → 00.
2001	00	Reserved
2002	00 - 03	A motor running status: 00: The driver is in a disabled state, and the motor is in a free state; 01: The driver is in the enabled state, the motor is in the control state, and it runs according to the input speed; 02: The driver decelerates to stop, and stops according to the deceleration time; 03: The driver is in braking stop state. After receiving this information, the motor will stop according to the maximum braking force; 05: Control the brake to open, and the motor is in a free state;
2003	00 - 03	B motor running status: 00: The driver is in a disabled state, and the motor is in a free state; 01: The driver is in the enabled state, the motor is in the control state, and it runs according to the input speed; 02: The driver decelerates to stop, and stops according to the deceleration time; 03: The driver is in braking stop state. After receiving this information, the motor will stop according to the maximum braking force;

<table>
<tr><td></td><td></td><td>05: Control the brake to open, and the motor is in a free state;</td></tr>
<tr><td>2004</td><td>0 - 1</td><td>A motor running direction
0: The motor runs forward;
1: The motor runs in reverse;</td></tr>
<tr><td>2005</td><td>0 - 1</td><td>B motor running direction
0: The motor runs forward;
1: The motor runs in reverse;</td></tr>
<tr><td>2006</td><td>0-16000</td><td>A motor running speed/angle
Under the speed closed-loop control, the motor speed, the range of speed value is 100-the maximum speed r/min.
Under angle closed-loop control, the motor controls the rotation angle, and the range of angle value 0-16000 is 0.00-160.00°</td></tr>
<tr><td>2007</td><td>0-16000</td><td>B motor running speed/angle
Under the speed closed-loop control, the motor speed, the range of speed value is 100-the maximum speed r/min.
Under angle closed-loop control, the motor controls the rotation angle, and the range of angle value 0-16000 is 0.00-160.00°</td></tr>
<tr><td>2008</td><td>0-1</td><td>output port control
Output port Y0-Y3 state, when the value is 1, the port is open;
<table>
<tr><td>1bit
(high bit)</td><td>1bit</td><td>1bit</td><td>1bit</td><td>1bit</td><td>1bit</td><td>1bit</td><td>1bit(low bit)</td></tr>
<tr><td>reserve</td><td>reserve</td><td>reserve</td><td>reserve</td><td>Y3</td><td>Y2</td><td>Y1</td><td>Y0</td></tr>
</table></td></tr>
</table>

Read input register function (04)

<table>
<tr><th>register address</th><th>read value</th><th>content</th></tr>
<tr><td>1000</td><td>0.0 -100.0</td><td>A motor running current
A Motor real-time current, the motor current resolution is 0.1A, and the maximum return range is 100.0A.</td></tr>
<tr><td>1001</td><td>0.0 -100.0</td><td>B motor running current
B Motor real-time current, the motor current resolution is 0.1A, and the maximum return range is 100.0A.</td></tr>
<tr><td>1002</td><td>0 - 1</td><td>A Motor running direction/direction
A The rotation direction of the motor speed under closed-loop control, and the rotation direction under the closed-loop state of steering;
<table>
<tr><td></td><td>Speed closed loop</td><td>Steering closed loop</td></tr>
<tr><td>00</td><td>The motor runs forward</td><td>positive angle adjustment</td></tr>
<tr><td>01</td><td>The motor runs in reverse</td><td>reverse angle adjustment</td></tr>
</table></td></tr>
<tr><td>1003</td><td>0 - 1</td><td>B Motor running direction/direction
B The rotation direction of the motor speed under closed-loop control, and the rotation direction under the closed-loop state of steering;
<table>
<tr><td></td><td>Speed closed loop</td><td>Steering closed loop</td></tr>
<tr><td>00</td><td>The motor runs forward</td><td>positive angle adjustment</td></tr>
<tr><td>01</td><td>The motor runs in reverse</td><td>reverse angle adjustment</td></tr>
</table></td></tr>
<tr><td>1004</td><td>0 - 8000</td><td>A motor running speed/angle
Under speed closed-loop control, the real-time speed value of motor A is 0-set rated speed, and the accuracy is 1r/min;
Under the steering closed-loop control, the current operating angle value range of A motor is 0-16000 (160.00°), and the angle accuracy is 0.01°.</td></tr>
<tr><td>1005</td><td>0 - 8000</td><td>B motor running speed/angle
Under speed closed-loop control, the real-time speed value of motor A is 0-set rated speed, and the accuracy is 1r/min;
Under the steering closed-loop control, the current operating angle value range of A motor is 0-16000 (160.00°), and the angle accuracy is 0.01°.</td></tr>
<tr><td>1006</td><td>0 - 22</td><td>Current state of the drive
When the drive is in normal operation, return: 00; when the drive is in an error state, it will output a fault code;</td><td>For details, please refer to the fault code table in Chapter 7.</td></tr>
<tr><td>1007</td><td>0 - 65535</td><td>voltage
The voltage value of the drive power input. The resolution is 0.01V
Example: the input voltage is 24.43V, the value is 2443 in decimal to hexadecimal, 09 (high) 8B (low)</td></tr>
</table>

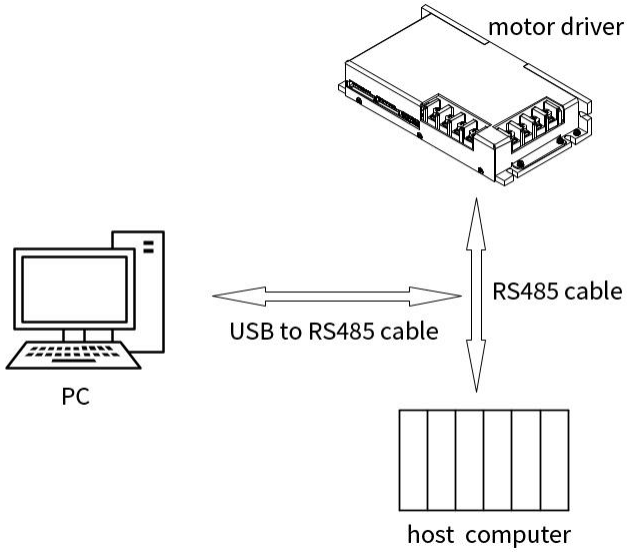
<table>
<tr><td>1008</td><td>0 - 01</td><td>A motor running direction<table><tr><td></td><td>Speed closed loop</td><td>Steering closed loop</td></tr><tr><td>00</td><td>The motor runs forward</td><td>positive angle adjustment</td></tr><tr><td>01</td><td>The motor runs in reverse</td><td>reverse angle adjustment</td></tr></table></td></tr>
<tr><td>1009</td><td>0 - 01</td><td>B motor running direction<table><tr><td></td><td>Speed closed loop</td><td>Steering closed loop</td></tr><tr><td>00</td><td>The motor runs forward</td><td>positive angle adjustment</td></tr><tr><td>01</td><td>The motor runs in reverse</td><td>reverse angle adjustment</td></tr></table></td></tr>
<tr><td>1010</td><td>0-8000</td><td>A motor running speed
Under speed closed-loop control, the real-time speed value of motor A is 0-set rated speed, and the accuracy is 1r/min;</td></tr>
<tr><td>1011</td><td>0-8000</td><td>B motor running speed
Under speed closed-loop control, the real-time speed value of motor A is 0-set rated speed, and the accuracy is 1r/min;</td></tr>
<tr><td>1012</td><td>0.00-110.00</td><td>A motor steering angle
Only when the A motor is set in the steering closed-loop mode, the feedback is the current angle of the A motor.
The angle value range of A motor is currently running is 0-12000 (120.00°), and the angle accuracy is 0.01°.</td></tr>
<tr><td>1013</td><td>0.00-110.00</td><td>B motor steering angle
Only when the A motor is set in the steering closed-loop mode, the feedback is the current angle of the A motor.
The angle value range of A motor is currently running is 0-12000 (120.00°), and the angle accuracy is 0.01°.</td></tr>
<tr><td>1014</td><td>0-65535</td><td>Store the number of pulses input by the hall sensor of motor A, the number of pulses ranges from 0 to 65535, and starts from 0 after exceeding, the cumulative counting of the motor forward rotation, the cumulative counting down of the motor reverse rotation, and the data is cleared after restarting the power;</td></tr>
<tr><td>1015</td><td>0-65535</td><td>Store the number of pulses input by the hall sensor of motor B, the number of pulses ranges from 0 to 65535, and starts from 0 after exceeding, the cumulative counting of the motor forward rotation, the cumulative counting down of the motor reverse rotation, and the data is cleared after restarting the power;</td></tr>
<tr><td>1016</td><td>00</td><td>Reserved</td></tr>
<tr><td>1017</td><td>00</td><td>Reserved</td></tr>
<tr><td>1018</td><td>00</td><td>Reserved</td></tr>
<tr><td>1019</td><td>00</td><td>Reserved</td></tr>
<tr><td>1020</td><td>00</td><td>Reserved</td></tr>
<tr><td>1021</td><td>00</td><td>Reserved</td></tr>
</table>

<table>
<tr><td>1022</td><td>-16000-16000</td><td>A Motor Hall Angle Value
Signed data, when A is a brushless DC motor and uses the Hall sensor as the steering closed-loop feedback, the current rotation angle of the A motor is measured, the value range is -16000-16000 (160.00°), and the angle accuracy is 0.01°;</td></tr>
<tr><td>1023</td><td>-16000-16000</td><td>B Motor Hall Angle Value
Signed data, when B is a brushless DC motor and uses the Hall sensor as the steering closed-loop feedback, the current rotation angle of the A motor is measured, the value range is -16000-16000 (160.00°), and the angle accuracy is 0.01°;</td></tr>
<tr><td>1024</td><td>00</td><td>Reserved</td></tr>
<tr><td>1025</td><td>00</td><td>Reserved</td></tr>
<tr><td>1026</td><td>00</td><td>Reserved</td></tr>
<tr><td>1027</td><td>00</td><td>Reserved</td></tr>
<tr><td>1028</td><td>0000-500</td><td>AVI1 input voltage value
When the driver is in the IO analog voltage control state, control the running speed of motor A, the data range is 0-500, the corresponding voltage value range is 0.00-5.00V, and the voltage accuracy is 0.01V;</td></tr>
<tr><td>1029</td><td>0000-500</td><td>AVI2 input voltage value
When the driver is in the IO analog voltage control state, control the running speed of motor B, the data range is 0-500, the corresponding voltage value range is 0.00-5.00V, and the voltage accuracy is 0.01V;</td></tr>
<tr><td>1030</td><td>0-127</td><td>Enter port code value
Input port X0-X5 state;
<table>
<tr><td>1bit(high bit)</td><td>1bit</td><td>1bit</td><td>1bit</td><td>1bit</td><td>1bit</td><td>1bit</td><td>1bit(low bit)</td></tr>
<tr><td>Reserved</td><td>Reserved</td><td>X5</td><td>X4</td><td>X3</td><td>X2</td><td>X1</td><td>X0</td></tr>
</table></td></tr>
<tr><td>1031</td><td>0-127</td><td>Enter port code value
Input port Y0-Y3 state;
<table>
<tr><td>1bit(high bit)</td><td>1bit</td><td>1bit</td><td>1bit</td><td>1bit</td><td>1bit</td><td>1bit</td><td>1bit(low bit)</td></tr>
<tr><td>Reserved</td><td>Reserved</td><td>Reserved</td><td>Reserved</td><td>Y3</td><td>Y2</td><td>Y1</td><td>Y0</td></tr>
</table></td></tr>
<tr><td>1032</td><td>-40-125</td><td>drive temperature
Signed data, the current drive temperature measured by the drive, and the measurement range is -40-125°C.</td></tr>
<tr><td>1033</td><td>00 - 01</td><td>A motor finds the origin mark
00: Steering closed-loop control did not find the origin or did not complete the search for the origin;
01: The steering closed-loop control completes finding the origin;
*The flag is only valid when the A motor is under steering closed-loop control.</td></tr>
</table>

1034	00 - 01	B motor finds the origin mark 00: Steering closed-loop control did not find the origin or did not complete the search for the origin; 01: The steering closed-loop control completes finding the origin; *The flag is only valid when the B motor is under steering closed-loop control.

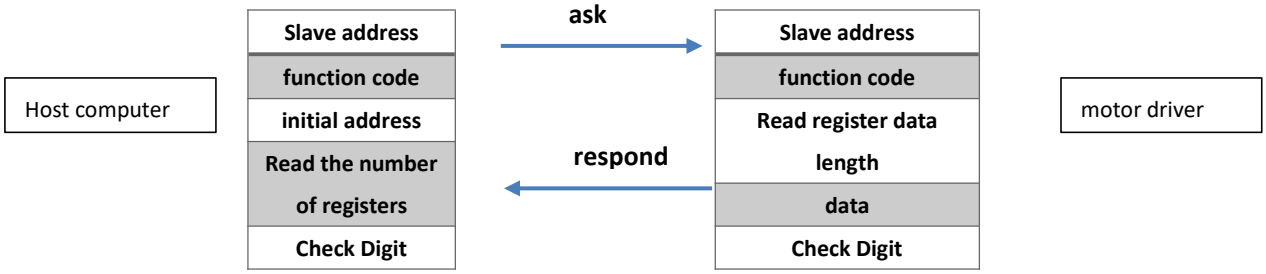
2.2.5 communication test

When the host computer communicates with the driver for MODBUS debugging, it is recommended to connect the RS485 port of the host computer to the computer through the USB to RS485 cable in advance. Obtain MODBUS raw data directly from the serial port of the computer for analysis.



According to the MODBUS protocol, writing and reading correspond to different function codes and data formats. On the computer side, the following format data needs to be detected, and then it can be determined that the MODBUS setting on the host computer side is correct.

Read input register data format



Important Hint

The main content of this chapter is used when modbus cannot be debugged successfully, and the specific data structure does not need to be dealt with in the actual programming process of PLC.

◆ **Query information structure**

slave address code	function code	initial address	read the number of registers	check digit
1 Bye	1 Bye	2Bye	1Byte	2Byte

●Slave address

The specified slave station address, the address of the slave station driver set after configuring the software;

●Function code

The read function code information supported by the drive is as follows:

function code	function
04	Read data from input register

●Initial address

The starting address of the register read from the driver, starting from 1000;

● Read the number of registers

The number of registers to read from the start address;

●Check Digit

Communication check digit.

◆ **Response information structure**

After the slave station receives the query information, it responds, and the response data is as follows:

slave address code	function code	read the number of registers	data	check digit
1 Byte	1 Byte	1 Byte	（Read the number of registers）*2 Byte	2 Byte

●slave address

The specified slave station address, the slave station drive address set by the configuration software;

●function code

The read function code information supported by the drive is as follows:

function code	function
04H	Read data from input register

●Read register data length

The register data length that the host computer needs to read;

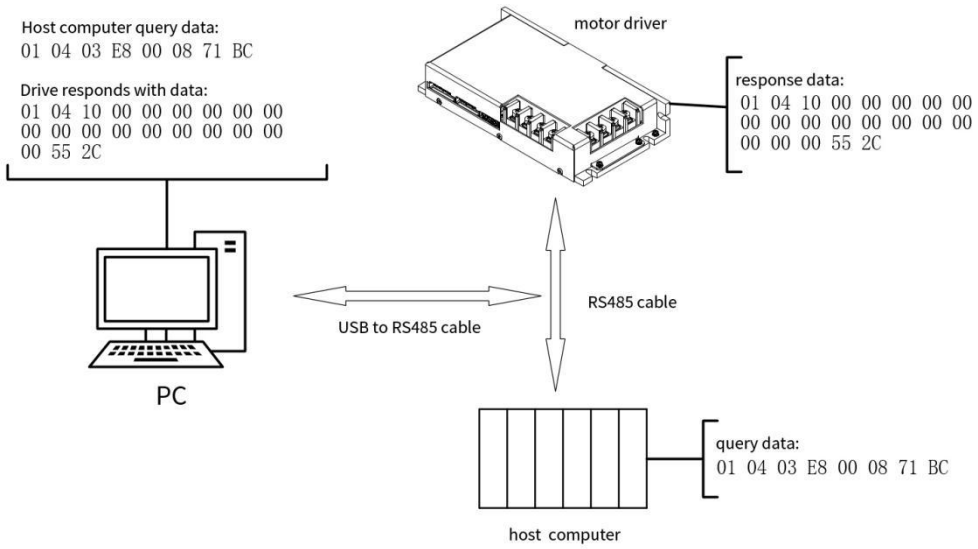
●Data

The register data that the host computer needs to read, since a register is 2Byte, the data length of this segment is the number of read registers * 2 Byte

●Check Digit

Communication check digit.

Example: The upper computer reads the driver input register parameters, the following is the analysis of the hexadecimal raw data sent once:



Connect the drive, host computer and computer via RS485. If the driver and the host computer can establish MODBUS communication, the data sent by the host computer and the driver can be received at the serial port of the computer.

The host computer asks for hexadecimal data:

01	**04**	**03 E8(1000)**	**00 08**	**71 BC**
Slave address code	function code	**initial address**	Read the number of registers	**Check Digit**
1Byte	1Btye	2Byte	2Byte	2Byte

The drive responds with hexadecimal data:

initial address

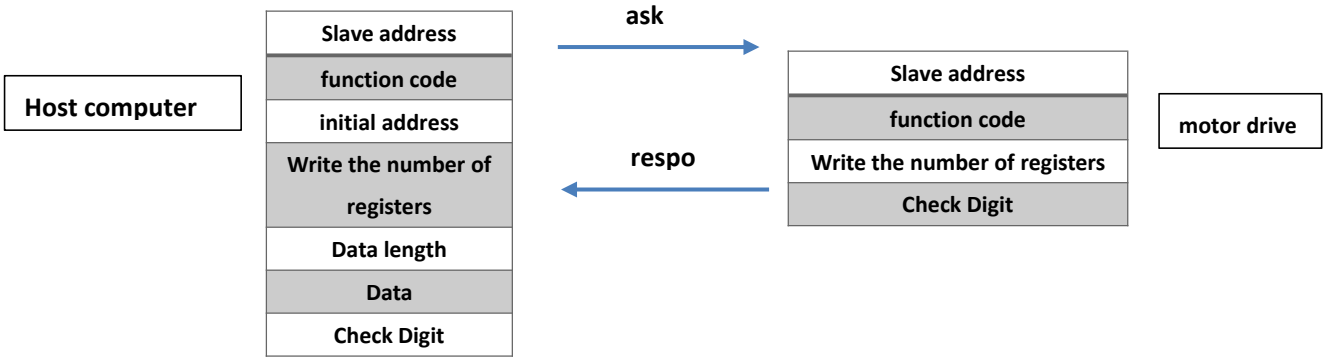
01 04 10 **00 00** 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 55 2C

01	**04**	**10**	**8 register data**	**71 BC**
Slave address code	function code	Data length	Data	Check Digit
1Byte	1Byte	1Byte	16Byte	2Byte

Data

	Data（16 Byte）							
Data	**00 00**	00 00	00 00	00 00	00 00	00 00	00 00	00 00
Corresponding register	1000	1001	1002	1003	1004	1005	1006	1007

Write holding register data format



◆ **Query information structure**

Slave address code	function code	initial address	Write the number of registers	Data length	Date	Check Digit
1Byte	1Byte	2Byte	2Byte	1Byte	(Number of registers to be written)*2 Byte	2Byte

●Slave address

The specified slave address, the address of the slave driver set by the configuration software;

●Function code

The write function code information supported by the drive is as follows:

function code	function
16	Write data to holding register

●initial address

The starting address of the holding register written from the driver starts from 2000;

●Write the number of registers

The number of registers that need to write data starting from the start address;

●Data length

The data length of the write register = the number of write registers * 2 numbers;

●Date

The data to be written starting from the start address, the data volume = the number of write registers * 2Byte;

●Check Digit

Communication check digit.

◆ **Response information structure**

After the slave station receives the query information, it responds, and the response data is as follows:

Slave address code	function code	initial address	Write the number of registers	Check Digit
1 Byte	1 Byte	2 Byte	2Byte	2 Byte

●Slave address

The specified slave address, the address of the slave driver set by the configuration software;

●function code

The write function code information supported by the drive is as follows:

function code	function
16	Write data to holding register

●initial address

The start address of the drive to write the holding register, starting from 2000;

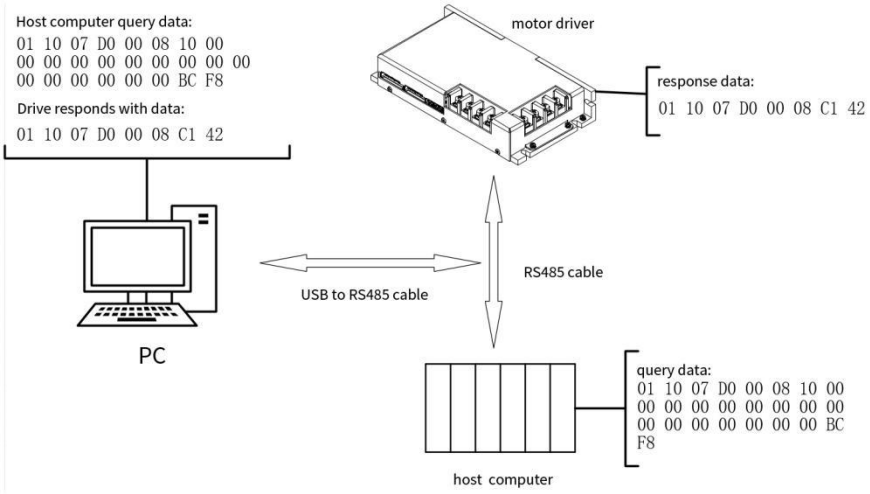
●Write the number of registers

The number of registers to write data from the start address;

●Check Digit

Communication check digit

Example: The upper computer writes the drive holding register parameters, the following is the analysis of the hexadecimal raw data sent once:



Connect the driver, host computer and computer via RS485. If the driver and host computer can establish MODBUS communication, the computer can receive the data sent by the host computer and the driver.

The host computer asks for hexadecimal data:

start address

01 10 07 D0 00 08 10 **00** **00** 00 00 00 00 00 00 00 00 00 00 00 00 00 00 BC F8

01	10	07 D0	00 08	10	8 register data	BC F8
Slave address code	function code	initial address	Read the number of registers	Data length	data	Check Digit
1Byte	1Btye	2Byte	2Byte	1Byte	16Byte	2Byte

8 register data

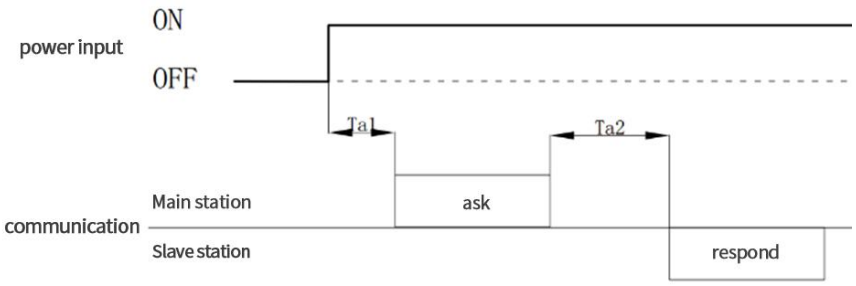
	Data（16Byte）							
Data	**00 00**	00 00	00 00	00 00	00 00	00 00	00 00	00 00
Corresponding register	2000	2001	2002	2003	2004	2005	2006	2007

The drive responds with hexadecimal data:

01	**10（16）**	**07 D0(200)**	**00 08**	**C1 42**
slave address code	function code	initial address	data length	check Digit
1Byte	1Byte	2Byte	2Byte	2Byte

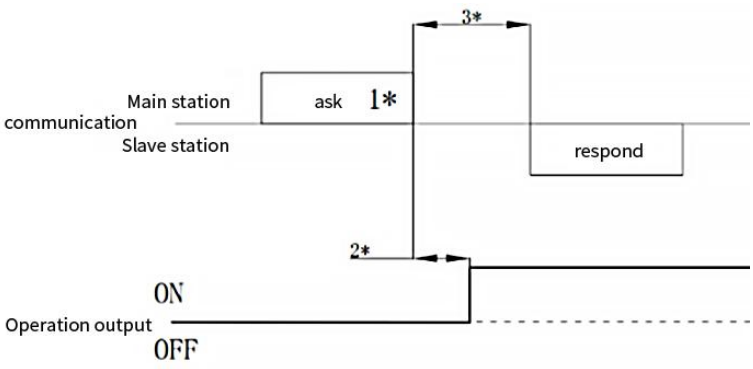
2.2.6 Control Sequence Diagram

■ **Communication start**



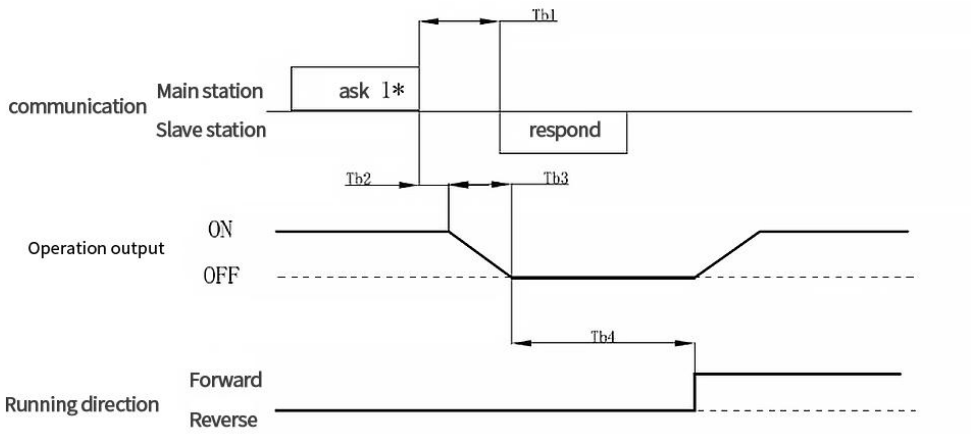
NO	Name	Content
Ta1	Power-on waiting time	1s or more
Ta2	Response Interval	T1 (send wait time) + C3.5 (quiet interval) + command processing time

■ **Start of operation**



NO	Name	Content
1*	ask for information	Contains information to get the drive started
2*	operating response time	C3.5 (quiet interval) + 4ms or less
3*	response interval	T1 (send waiting time) + C3.5 (quiet interval) + command processing time

■ **Stop, direction switch, start**



NO	Name	Content
1*	ask for information	Contains information to stop the drive
Tb1	response interval	T1 (send waiting time) + C3.5 (quiet interval) + command processing time
Tb2	command response interval	C3.5 (quiet interval) + time for command processing
Tb3	Motor deceleration time	Determined by the deceleration time configured by the drive
Tb4	Direction switching time	Motor deceleration stop to commutation time

3 RS485 AND RS232 HS PROTOCOL

3.1 RS485-HS PROTOCOL COMMUNICATION

3.1.1 Communication Specifications

Function	Parameter information	default setting
way of communication	Single master/multiple slave mode	
Receive/send method	half-duplex communication mode	
Communication ID	1-127 configurable	1/127
communication rate	9600bps, 19200bps, 38400bps, 576000bps, 115200bps	115200bps
data bit	8 bits	8 bit
Check Digit	no parity; odd parity; even parity	no parity
stop bit	1 bit; 2 bit	1 bit

3.1.2 communication settings

To use the RS485 communication HS protocol, you need to set the driver through Micr-USB or RS232 before using it. Parameter setting is performed through the Hinson configuration software on the computer side. Set the parameters as follows:

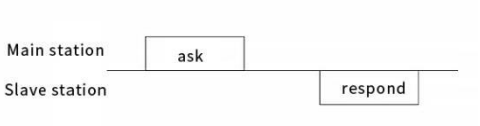
Parameter name	Parameter list	Parameter setting
Device interface and protocol	1. IO analog voltage control 2. RS232-HS protocol 3. RS485-HS protocol 4. CAN bus-HS protocol 5. 5RS485-MODBUS protocol 6. CAN OPEN protocol	RS485-HS protocol
device address	1-127	127
RS485 communication baud rate	1. 9600bps 2. 19200bps 3. 38400bps 4. 57600bps 5. 115200bps	115200bps
RS485 parity check	1. No parity 2. Odd parity 3. even parity	no parity
Communication interruption protection time	0.05s-1s	0.5s

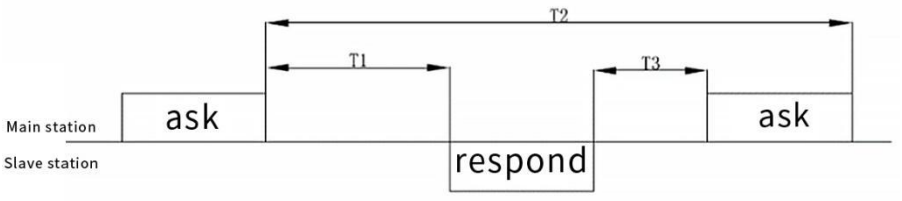
3.1.3 Master-slave station communication sequence

unicast mode

The master station sends an inquiry to one slave station.

The slave executes the processing and returns a response.





NO	Name	Content
T1	response wait interval	within 1ms
T2	communication timeout	Configurable from 0.05s – 2s
T3	Send waiting time	Recommend 5ms -30ms

3.2 RS232-HS PROTOCOL COMMUNICATION

3.2.1 Communication specifications

Paremeter	Parameter information	Default setting
way of communication	peer to peer	
Receive/send method	Full duplex communication mode	
Communication ID	1 - 127	127
communication rate	19200bps，115200bps	115200bps
data bit	8 bit	8 bit
Check Digit	no parity	no parity
stop bit	1 bit	1 bit

3.2.2 communication settings

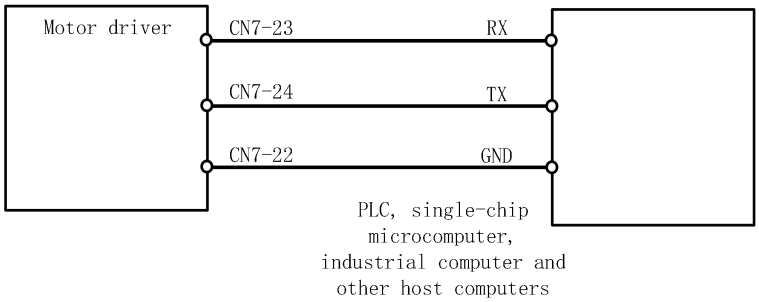
To use the RS232-HS protocol, it is necessary to set the driver through RS232 first before using it. Parameter setting is performed through the Hinson configuration software on the computer side.
Set the parameters as follows:

Parameter name	Parameter List	Setting parameters
Device interface and protocol	1.IO analog voltage control 2. RS232-HS protocol 3. RS485-HS protocol 4. CAN bus-HS protocol 5.RS485-MODBUS protocol 6.CAN OPENprotocol	RS232-HS protocol
Device address	1-127	127
RS232 communication baud rate	1. 19200bps 2. 115200bps	115200bps
Communication interruption protection time	0.05s-1s	0.5s

3.2.3 connection

RS232 wiring

The RS232 communication is located on the CN6 port, and it is recommended to use a shielded cable for connection. Driver parameter configuration can only be done through RS232 and USB port, and only 1-to-1 connection is allowed under RS232 connection.

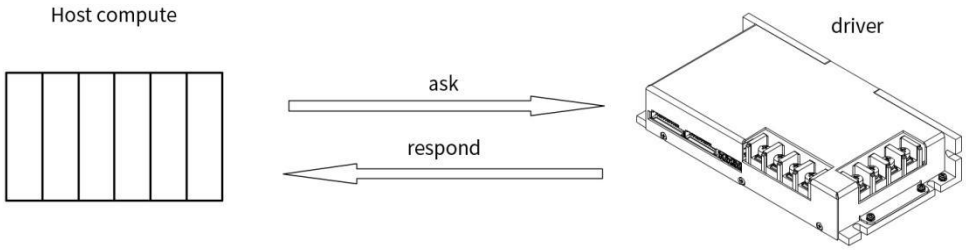


The RS232 communication line adopts the cross-connection mode, that is, the driver sends to the upper computer to receive, and the driver receives to the upper computer to send.

Special attention, the GND line must be connected under RS232 communication.

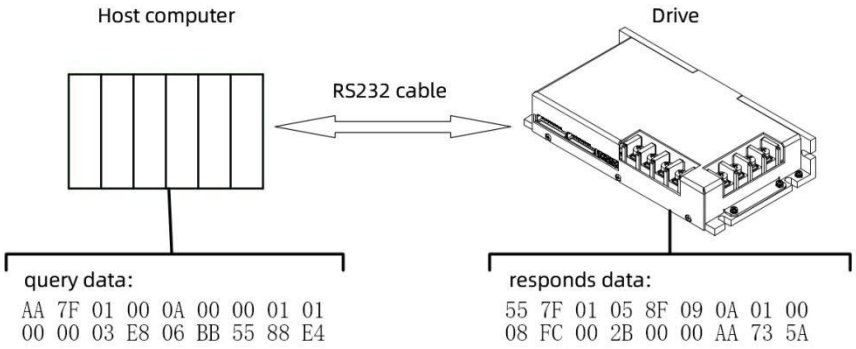
3.3 COMMUNICATION PROTOCOL

Under the HS protocol, a question-and-answer method is used for communication. The upper computer sends inquiry data. After receiving the data, the driver can respond or not respond to the data according to the received data. And the motor control is performed according to the query data. Moreover, the host computer needs to continuously send query data within the set communication timeout protection time.



3.3.1 Example of communication control

When the driver A motor is set to the speed closed-loop mode, the B motor is set to the angle closed-loop mode. Example of sending data format in RS232 control mode



Analysis of the hexadecimal data sent by the host computer:

AA	**7F**	**01**	**00**	**0A**	**00 00**	**01**	**01**	**00**	**00**
start code	Driver communication address	return data type	fault reset	output port designation	reserved bit	A motor running status	B motor running status	A motor running direction	B motor running direction
03 E8	06 BB	55	88 E4						
A motor running speed	B motor rotation target angle	end code	check code						

The drive responds with hexadecimal data parsing:

55	**7F**	**01**	**05 8F**	**09 0A**	**01**	**00**	**08 FC**	**00 2B**	**00 00**
start code	Driver communicat	return data	A motor running	B motor running	A motor running	B motor running	A motor running	B motor running	drive status

	ion address	type	current	current	direction	direction	speed	speed	code
AA	**73 5A**								
end code	check code								

3.3.2 Asking for information

1	Start code	1Byte	18 Byte
2	Driver communication address	1Byte	
3	return data type	1Byte	
4	Drive Fault Clear	1Byte	
5	Driver output port designation	1Byte	
6	Driver Mode Control	1Byte	
7	Reserved	1Byte	
8	A motor running status	1Byte	
9	B motor running status	1Byte	
10	A motor running direction	1Byte	
11	B motor running direction	1Byte	
12	A motor running speed/angle	1Byte (high bit)	
13		1Byte (low)	
14	B motor running speed/angle	1Byte (high bit)	
15		1Byte (low)	
16	end code	1Byte	
17	check code	1Byte (low)	
18		1Byte (high bit)	

●Start code (1Byte)

Communication start code: AA, communication start flag;

●Driver communication address (1Byte)

Target slave driver address, set by configuration software;

●Return data type (1Byte) 55

The host computer can access data in various drives according to requirements:

Send data	The data content of the response from the slave station
00	no response
01：	Motor current, speed angle, driver error status
02：	Power supply voltage, motor running direction, speed and angle
03：	Encoder feedback data
04：	Analog Feedback Data
05：	Analog input voltage, input, output port status.

●Drive fault clearing (1Byte)

When the drive is in the state of reporting errors, it is necessary to reset the fault of the drive. Default sending: 00, reset operation: 01 → 00;

●Driver output port designation (1Byte)

In the communication control mode, control the driver output port switch signal:

1bit (high bit)	1bit	1bit	1bit	1bit	1bit	1bit	1bit（low bit）
Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0

If sending 0A:

Then control the output of Y1 and Y3.

●Mode control (1Byte)

0x00: control mode The drive is in control mode;

0x80: In monitoring mode, the drive is in monitoring mode and not controlled;

●Reserved bit (1Byte)

invalid data: 00;

●A motor running status (1Byte)

Control the state of motor A:

Send data	Motor action
00	The driver is disabled and the motor is in a free state;
01：	When the driver is enabled, the motor is in the control state and runs according to the input speed;
02：	The drive decelerates to stop, and stops according to the deceleration time;
03：	The driver is in braking stop state. After receiving this information, the motor will decelerate and stop according to the maximum braking force.

●B motor running status (1Byte)

Control the state of motor B:

Send data	Motor action
00	The driver is disabled and the motor is in a free state;
01：	When the driver is enabled, the motor is in the control state and runs according to the input speed;
02：	The drive decelerates to stop, and stops according to the deceleration time;
03：	The driver is in braking stop state. After receiving this information, the motor will decelerate and stop according to the maximum braking force.

●A running direction (1Byte)

Control the running direction of the motor:

Send data	Motor action
00	The motor runs forward
01：	The motor runs in reverse

The direction of forward rotation and reverse rotation of the motor is relative, and the actual running direction of the motor shall prevail.

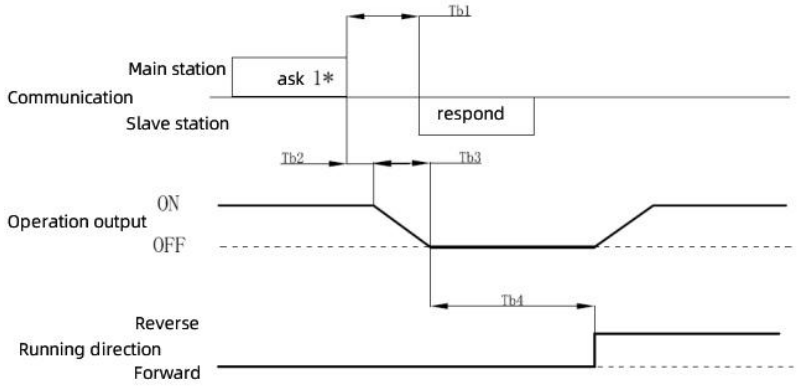
●B running direction (1Byte)

Control the running direction of the motor:

Send data	Motor action
00	The motor runs forward
01：	The motor runs in reverse

The direction of forward rotation and reverse rotation of the motor is relative, and the actual running direction of the motor shall prevail.

Direction Switch Timing



NO	Name	Content
1*	ask for information	Contains information to reverse the motor
Tb1	response interval	Command processing time + sending waiting time
Tb2	command response interval	command processing time
Tb3	Motor deceleration time	It is determined by the deceleration time configured by the drive and the current motor inertia,
Tb4	Direction switching time	The time from motor deceleration to stop to switching direction running,

Hint

When running in the forward direction, the driver will immediately execute the reverse running action when it receives the reverse switching data. If it receives the forward command again during the forward deceleration process, it will terminate the deceleration and direction switching, turn to the forward running and accelerate to target speed.

●A Motor running speed/angle (2Byte)

In the speed closed-loop mode, control the A motor to run at the target speed. In angle closed-loop mode, control the target angle of A motor rotation:

Operating mode	Motor action
speed closed loop	Motor speed, the range of speed value is 100 - the maximum speed r/min. The first Byte: the high 8 bits of the speed value; The second Byte: the lower 8 bits of the speed value;
Angle closed loop	The motor controls the rotation angle, and the angle value range is 0.00-180.00 The first Byte: the high 8 bits of the speed value; The second Byte: the lower 8 bits of the speed value;

●B Motor running speed/angle (2Byte)

In the speed closed-loop mode, control the B motor to run at the target speed. In angle closed-loop mode, control the target angle of B motor rotation:

Operating mode	Motor action
speed closed loop	Motor speed, the range of speed value is 100 - the maximum speed r/min. The first Byte: the high 8 bits of the speed value; The second Byte: the lower 8 bits of the speed value;
Angle closed loop	The motor controls the rotation angle, and the angle value range is 0.00-180.00 The first Byte: the high 8 bits of the speed value; The second Byte: the lower 8 bits of the speed value;

Hint

In the speed closed-loop mode, when the target speed is lower than the set minimum speed, the motor will not run when it is in the stopped state, and it will run and stop according to the deceleration time parameter in the current running state. When the target speed is higher than the maximum speed set by parameters, it will run according to the maximum speed set by parameters.

In angle closed-loop mode, when the target angle is greater than the parameter limit angle, it will only rotate to the parameter limit angle.

●End code (1Byte)

Communication end code: 55;

●Check code (2Byte)

For the result of CRC16 calculation, please refer to the check code calculation formula for details.

3.3.3 response message

According to the different acquisition requirements of the host computer, the slave driver can provide different data feedback, and the following instructions are made for different data feedback.

Return data 01

1	Start code	1Byte	18 Byte
2	Driver communication address	1Byte	
3	return data type	1Byte	
4	A motor running current	1Byte (high bit)	
5		1Byte (low)	
6	B motor running current	1Byte (high bit)	
7		1Byte (low)	
8	A Motor running direction/direction	1Byte	
9	B motor running direction/direction	1Byte	
10	A motor running speed/angle	1Byte (high bit)	
11		1Byte (low)	

12	B motor running speed/angle	1Byte (high bit)	
13		1Byte (low)	
14	Current state of the drive	1Byte (high bit)	
15		1Byte (low)	
16	end code	1Byte	
17	check code	1Byte (low)	
18		1Byte (high bit)	

●Start code (1Byte)

Response communication start code: 55, communication start flag;

●Driver communication address (1Byte)

Response driver communication address;

●Return data type (1Byte)

The data code of the current response: 01, this code is the same as the feedback data type of the upper computer query information;

●A motor running current (2Byte)

A motor real-time current, motor current resolution is 0.01A, maximum return range is 100.00A;

●B motor running current (2Byte)

B motor real-time current, motor current resolution is 0.01A, maximum return range is 100.00A;

●A motor running direction/direction (1Byte)

Hint

Current feedback is the current running current of the motor, but due to PWM control, when the input current of the driver is below the rated power, it is not equal to the sum of A and B currents.

A The rotation direction of the motor speed under closed-loop control, and the rotation direction under the closed-loop state of steering;

	Speed closed loop	Steering closed loop
00	The motor runs forward	positive angle adjustment
01	The motor runs in reverse	reverse angle adjustment

●B motor running direction/direction (1Byte)

B The direction of rotation under the closed-loop control of the motor speed, and the direction of rotation under the closed-loop state;

	Speed closed loop	Steering closed loop
00	The motor runs forward	positive angle adjustment
01	The motor runs in reverse	reverse angle adjustment

●A motor running speed/angle (2Byte)

A The real-time speed of the motor or the current running angle;

Under speed closed-loop control, the real-time speed value of motor A is 0-set rated speed, and the accuracy is 1r/min;

Under the steering closed-loop control, the angle value range of the current operation of the A motor is 0-12000 (120.00°), and the angle accuracy is 0.01°.

●B motor running speed/angle (2Byte)

B The real-time speed of the motor or the current running angle;

Under speed closed-loop control, the real-time speed value of motor A is 0-set rated speed, and the accuracy is 1r/min;

Under the steering closed-loop control, the angle value range of the current operation of the A motor is 0-12000 (120.00°), and the angle accuracy is 0.01°.

●Current state of the drive (2Byte)

The drive returns 00 00 under normal operation; when the drive is in error reporting state, it outputs a fault code, please refer to the fault code table for details;

●End code (1Byte)

Communication end code: AA;

●Check code (2Byte)

For the result of CRC16 calculation, please refer to the check code calculation formula for details.

return data 02

1	start code	1Byte	18 Byte
2	Driver communication address	1Byte	
3	return data type	1Byte	
4	A motor running current	1Byte (high bit)	
5		1Byte (low)	
6	B motor running current	1Byte	
7		1Byte	
8	A Motor running direction/direction	1Byte (high bit)	
9	B motor running direction/direction	1Byte (low)	
10	A motor running speed/angle	1Byte (high bit)	
11		1Byte (low)	
12	B motor running speed/angle	1Byte (high bit)	
13		1Byte (low)	
14	Current state of the drive	1Byte (high bit)	
15		1Byte (low)	
16	end code	1Byte	
17	check code	1Byte (low)	
18		1Byte (high bit)	

●Start code (1Byte)

Response communication start code: 55, communication start flag;

●Driver communication address (1Byte)

Response driver communication address;

●Return data type (1Byte)

The data code of the current response: 02, this code is the same as the data type of the upper computer query information feedback;

●Power supply voltage (2Byte)

The voltage value of the drive power supply input. The resolution is 0.01V;

Example: if the input voltage is 24.43V, then the value is 2443 in decimal to hexadecimal, 09 (high order) and 8B (low order).

●A motor running direction (1Byte)

	Speed closed loop	Steering closed loop
00	The motor runs forward	positive angle adjustment
01：	The motor runs in reverse	reverse angle adjustment

●B motor running direction (1Byte)

Hint

The forward and reverse running directions of the motor are relative directions, and the direction of the motor is also affected by the motor wiring and the motor reducer. The drive can be rotated in this default direction.

	Speed closed loop	Steering closed loop
00	The motor runs forward	positive angle adjustment
01：	The motor runs in reverse	reverse angle adjustment

●A motor running speed (2Byte)

Under speed closed-loop control, the real-time speed value of A motor is 0-the highest set speed, and the accuracy is 1r/min;

●B motor running speed (2Byte)

Under speed closed-loop control, the real-time speed value of motor B is 0-the highest set speed, and the accuracy is 1r/min;

●A motor steering angle (2Byte)

Only when the A motor is set in the steering closed-loop mode, the feedback is the current angle of the A motor;

The angle value range of A motor is currently running is 0-12000 (120.00°), and the angle accuracy is 0.01°.

●B motor steering angle (2Byte)

Only when the B motor is set in the steering closed-loop mode, the feedback is the current angle of the B motor;

The angle value range of the B motor is currently running is 0-12000 (120.00°), and the angle accuracy is 0.01°.

●End code (1Byte)

Communication end code: AA;

●Check code (2Byte)

For the result of CRC16 calculation, please refer to the check code calculation formula for details.

Return data 03

1	Start code	1Byte	18 Byte
2	driver communication address	1Byte	
3	return data type	1Byte	
4	A motor hall sensor count value	1Byte（high bit）	
5		1Byte（low bit）	
6	B motor hall sensor count value	1Byte（high bit）	
7		1Byte（low bit）	
8	reserved	1Byte	
9		1Byte	
10		1Byte	
11		1Byte	
12		1Byte	
13		1Byte	
14		1Byte	
15		1Byte	
16	end code	1Byte	
17	check code	1Byte（low bit）	
18		1Byte（high bit）	

●Start code (1Byte)

Response communication start code: 55, communication start flag;

●Driver communication address (1Byte)

Response driver communication address;

●Return data type (1Byte)

The data code of the current response: 03, this code is the same as the data type of the upper computer query information feedback;

● A motor Hall sensor 1 count value (2Byte)

Store the number of pulses input by the hall sensor of motor A, the number of pulses ranges from 0 to 65535, and starts from 0 after exceeding, the cumulative counting of the motor forward rotation, the cumulative counting down of the motor reverse rotation, and the data is cleared after restarting the power;

●B motor hall sensor count value (2Byte)

Store the number of pulses input by the hall sensor of motor B, the number of pulses ranges from 0 to 65535, and starts from 0 after exceeding, the cumulative counting of the motor forward rotation, the cumulative counting of the reverse rotation of the motor, and the data is cleared after restarting the power;

●Reserved bit (8Byte)

invalid data: 00;

●End code (1Byte)

Communication end code: AA;

●Check code (2Byte)

For the result of CRC16 calculation, please refer to the check code calculation formula for details.

Return data 04

1	Start code	1Byte	18 Byte
2	driver communication address	1Byte	
3	return data type	1Byte	
4	Reserved	1Byte	
5		1Byte	
6		1Byte	
7		1Byte	
8	A Motor Hall Angle Value	1Byte（high bit）	
9		1Byte（low bit）	
10	B Motor Hall Angle Value	1Byte（high bit）	
11		1Byte（low bit）	
12	Reserved	1Byte	
13		1Byte	
14		1Byte	
15		1Byte	
16	end code	1Byte	
17	check code	1Byte（low bit）	
18		1Byte（high bit）	

●Start code (1Byte)

Response communication start code: 55, communication start flag;

●Driver communication address (1Byte)

Response driver communication address;

●Return data type (1Byte)

The data code of the current response: 04, this code is the same as the data type of the upper computer query information feedback;

●Reserved bit (4Byte)

invalid data: 00;

●A motor Hall angle value (2Byte)

Signed data, when A is a brushless DC motor and uses the Hall sensor as the steering closed-loop feedback, the current rotation angle of the A motor is measured, the value range is -12000-12000 (120.00°), and the angle accuracy is 0.01°;

●B motor Hall angle value (2Byte)

Signed data, when B is a brushless DC motor and uses the Hall sensor as the steering closed-loop feedback, the current B motor rotation angle is measured, the value range is -12000-12000 (120.00°), and the angle accuracy is 0.01°;

●Reserved bit (4Byte)

Invalid data: 00

●End code (1Byte)

Communication end code: AA;

●Check code (2Byte)

For the result of CRC16 calculation, please refer to the check code calculation formula for details.

return data 05

1	start code	1Byte	18 Byte
2	driver communication address	1Byte	
3	return data type	1Byte	
4	A motor hall sensor count value	1Byte（high bit）	
5		1Byte（low bit）	
6	B motor hall sensor count value	1Byte（high bit）	
7		1Byte（low bit）	
8	reserved	1Byte	
9		1Byte	
10		1Byte	
11		1Byte	
12		1Byte	
13		1Byte	
14		1Byte	
15		1Byte	
16	end code	1Byte	
17	check code	1Byte（low bit）	
18		1Byte（high bit）	

●Start code (1Byte)

Communication start flag, response communication start code: 55;

●Driver communication address (1Byte)

Response driver communication address;

●Return data type (1Byte)

The data code of the current response: 05, this code is the same as the data type of the upper computer query information feedback;

●Reserved bit (4Byte)

invalid value: 00;

●AVI1 input voltage value (2Byte)

When the driver is in the IO analog voltage control state, control the running speed or angle of motor A, the value range is 0-500, and the corresponding voltage value range is 0-5.00V;

●AVI2 input voltage value (2Byte)

When the driver is in the IO analog voltage control state, control the running speed or angle of motor B, the value range is 0-500, and the corresponding voltage value range is 0-5.00V;

● Input port code value (1Byte)

Input port X0-X5 state:

0: The port has a signal;

1: No signal at the port

1bit (high bit)	**1bit**	**1bit**	**1bit**	**1bit**	**1bit**	**1bit**	**1bit(low)**
reserve	reserve	X5	X4	X3	X2	X1	X0

●Output port code value (1Byte)

Output port Y0-Y3 state:

0: The port does not output signals;

1: port output signal;

1bit (high bit)	1bit	1bit	1bit	1bit	1bit	1bit	1bit(low)
reserve	reserve	reserve	reserve	Y3	Y2	Y1	Y0

●Drive temperature (2Byte)

Signed data, the current driving temperature measured by the driver, the measuring range is -40-125°C;

●End code (1Byte)

Communication end code: AA;

●Check code (2Byte)

For the result of CRC16 calculation, please refer to the check code calculation formula for details.

return data 06

1	Start code	1Byte	18 Byte
2	driver communication address	1Byte	
3	return data type	1Byte	
4	A motor finds the origin mark	1Byte	
5	B motor finds the origin mark	1Btye	
6	Reserved	1Byte	
7		1Byte	
8		1Byte	
9		1Byte	
10		1Byte	
11		1Byte	
12		1Byte	
13		1Byte	
14		1Byte	
15		1Byte	
16	end code	1Byte	
17	check code	1Byte（low bit）	
18		1Byte（high bit）	

●Start code (1Byte)

Communication start flag, response communication start code: 55;

●Driver communication address (1Byte)

Response driver communication address;

●Return data type (1Byte)

The data code of the current response: 06, this code is the same as the data type of the upper computer query information feedback;

●A motor finds the origin flag

00: Steering closed-loop control did not find the origin or did not complete the search for the origin;

01: Finding the origin has been completed under the steering closed-loop control;

*The flag is only valid when the A motor is under steering closed-loop control.

●B motor finds the origin flag

00: Steering closed-loop control did not find the origin or did not complete the search for the origin;
01: Finding the origin has been completed under the steering closed-loop control;
*The flag is only valid when the B motor is under steering closed-loop control.
●Reserved bit (10Byte)
Invalid data is 00;
●End code (1Byte)
Communication end code: AA;
●Check code (2Byte)
The result of the CRC16 operation. For details, please refer to the check code calculation formula.

4 CAN BUS COMMUNICATION CONTROL

4.1 BY HS PROTOCOL

4.1.1 Communication Specifications

Function	Parameter information	Default parameters
Communication Specifications	CAN BUS 2.0A	
Frame ID	1-127 configurable	127
Communication rate	125 Kbps，250Kbps，500Kbps，1000Kbps	1000Kbps
Frame type	standard frame	
Frame format	Data Frame	

4.1.2 Communication settings

Using the CAN bus-HS protocol requires the driver to be set before use. Parameter setting is performed through the Hinson configuration software on the computer side.

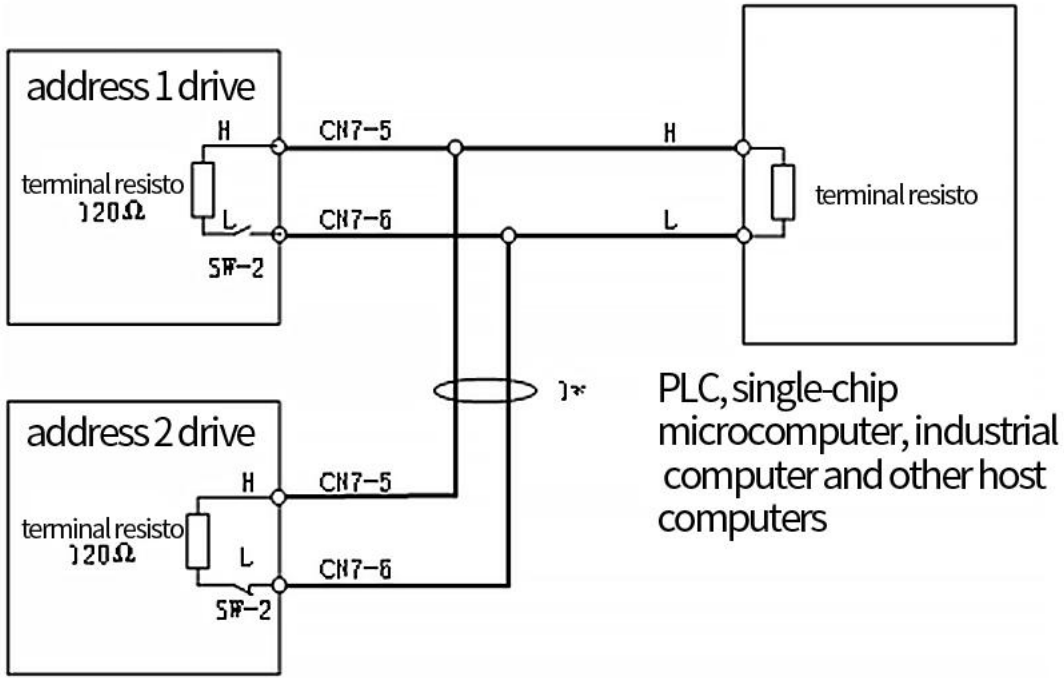
Set the parameters as follows:

parameter name	Parameter list	Setting parameters
Device interface and protocol	1.IO analog voltage control 2. RS232-HS protocol 3. RS485-HS protocol 4. CAN bus-HS protocol 5. RS485-MODBUS protocol 6. CAN OPEN protocol	CAN bus-HS protocol
device address	1-127	127

CAN bus communication baud rate	125Kbps 250Kbps 500Kbps 1000Kbps	1000Kbps
Communication interruption protection time	0.05s-1s	0.5s

4.1.3 Connection

The drive has 1 CAN BUS interface, through which multiple drives can be connected and controlled, and can also be connected to multiple CAN BUS devices.

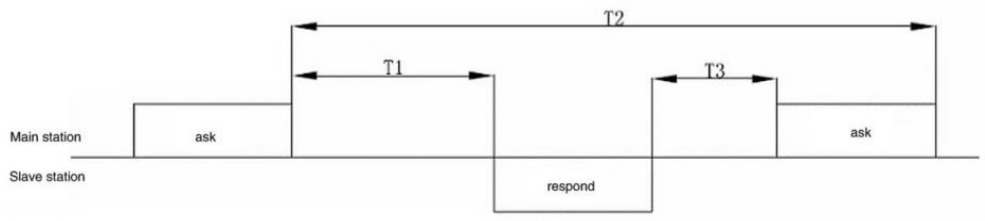


Make the CAN BUS a multi-master and multi-slave communication mode, and the devices at both ends of the communication need to be equipped with terminal resistors.

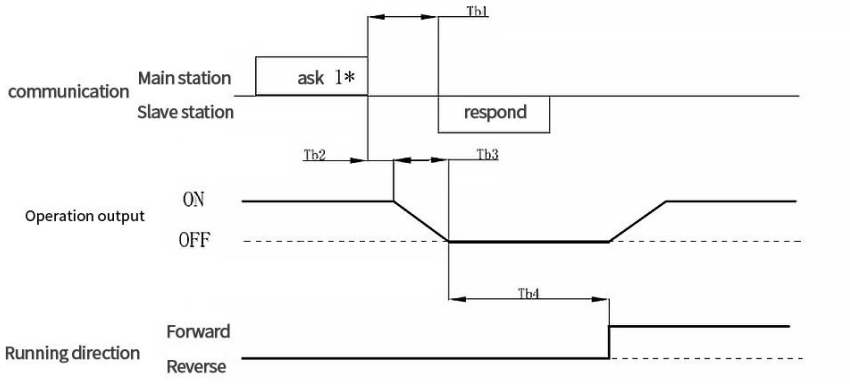
1*connection cable

The cable is recommended to use twisted pair and shielded;

4.1.4 Master-slave communication sequence



NO	Name	Content
T1	response wait interval	within 1ms
T2	communication timeout	Configurable from 0.05s – 2s
T3	Send waiting time	Recommend 5ms -30ms

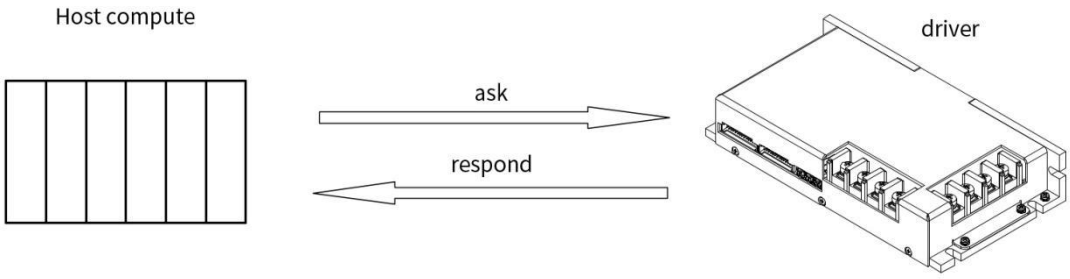


4.1.5 Control Sequence Diagram

NO	Name	Content
1*	ask for information	Contains information to stop the drive
Tb1	response interval	T1 (send waiting time) + C3.5 (quiet interval) + command processing time
Tb2	command response interval	C3.5 (quiet interval) + time for command processing
Tb3	motor deceleration time	Determined by the deceleration time configured by the drive
Tb4	direction switching time	Motor deceleration stop to commutation time

4.1.6 Communication protocol

Under the HS protocol, a question-and-answer method is used for communication. The upper computer sends inquiry data. After receiving the data, the driver can respond or not respond to the data according to the received data. And the motor control is performed according to the query data. Moreover, the host computer needs to continuously send query data within the set communication timeout protection time.



ask for information

CAN BUS communication characteristics, only 8Byte data can be sent at a time, and the driver control data is sent in 3 times.

Inquiry Information 1

Inquiry information serial number	**1Byte**	**8 Byte**
Response Information Choices	1Byte	
Drive mode	1Byte	
Drive Fault Clear	1Byte	
A motor running status	1Byte	
A motor running direction	1Byte	
A motor running speed/angle	1Byte (high bit)	
	1Byte (low)	

●Inquiry information serial number (1Byte)

The query information serial number is: 01;

●Response information selection (1Byte)

Control the slave driver to respond to data;

●Drive mode (1Byte)

default mode: 00;

●Drive fault clearing (1Byte)

When the drive is in the state of reporting errors, it is necessary to reset the fault of the drive. Default sending: 00, reset operation: 01 00;

●A motor running status (1Byte)

Control the state of motor A:

Send data	Motor action
00	The driver is disabled and the motor is in a free state;
01	When the driver is enabled, the motor is in the control state and runs according to the input speed;
02	The drive decelerates to stop, and stops according to the deceleration time;
03	The driver is in braking stop state. After receiving this information, the motor will decelerate and stop according to the maximum braking force.
05	Control the brake to open, and the motor is in a free state;

●A running direction (1Byte)

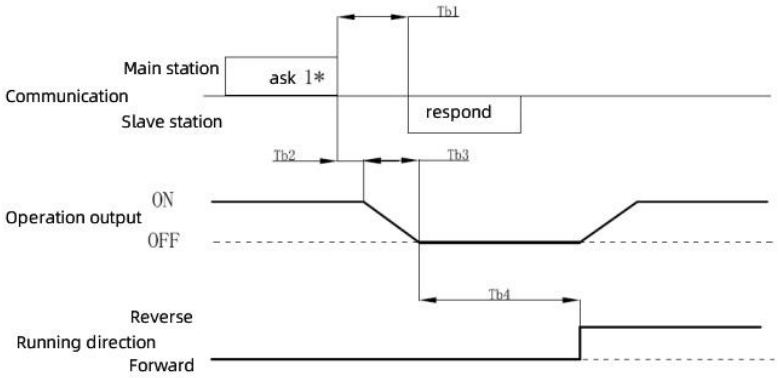
Control the running direction of the motor:

Send data	Motor action
00	The motor runs forward

01：	The motor runs in reverse

※The direction of forward rotation and reverse rotation of the motor is relative, and the actual running direction of the motor shall prevail.

Direction Switch Timing



NO	Name	Content
1*	ask for information	Contains information to reverse the motor
Tb1	response interval	Command processing time + sending waiting time
Tb2	command response interval	command processing time
Tb3	motor deceleration time	It is determined by the deceleration time configured by the drive and the current motor inertia
Tb4	direction switching time	The time from motor deceleration to stop to switching direction running

Hint

When running in the forward direction, the driver will immediately execute the reverse running action when it receives the reverse switching data. If it receives the forward command again during the forward deceleration process, it will terminate the deceleration and direction switching, turn to the forward running and accelerate to target speed.

●A motor running speed/angle (2Byte)

In the speed closed-loop mode, control the A motor to run at the target speed. In angle closed-loop mode, control the target angle of A motor rotation:

Operating mode	Motor action
speed closed loop	Motor speed, the range of speed value is set minimum speed - set maximum speed r/min, the unit is 1r/min. The first Byte: the high 8 bits of the speed value; The second Byte: the lower 8 bits of the speed value;
Angle closed loop	The motor controls the rotation angle, the angle value range is 0-12000°, and the steering wheel rotation angle is set, the unit is 0.01°; The first Byte: the high 8 bits of the speed value; The second Byte: the lower 8 bits of the speed value;

Inquiry Information 2

Inquiry information serial number	1Byte	8Byte
Response Information Choices	1Byte	
drive mode	1Byte	
Drive Fault Clear	1Byte	
B motor running status	1Byte	
B motor running direction	1Byte	
B motor running speed/angle	1Byte（high bit）	
	1Byte（low bit）	

●Inquiry information serial number (1Byte)

The query information serial number is: 02;

●Response information selection (1Byte)

Control the slave driver to respond to data;

●Drive mode (1Byte)

default mode: 00;

●Drive fault clearing (1Byte)

When the drive is in the state of reporting errors, it is necessary to reset the fault of the drive. Default sending: 00, reset operation: 01 00;

●B motor running status (1Byte)

Control the state of motor B:

Send data	Motor action
00	The driver is disabled and the motor is in a free state;
01	When the driver is enabled, the motor is in the control state and runs according to the input speed;
02	The drive decelerates to stop, and stops according to the deceleration time;
03	The driver is in braking stop state. After receiving this information, the motor will decelerate and stop according to the maximum braking force.
05	Control the brake to open, and the motor is in a free state;

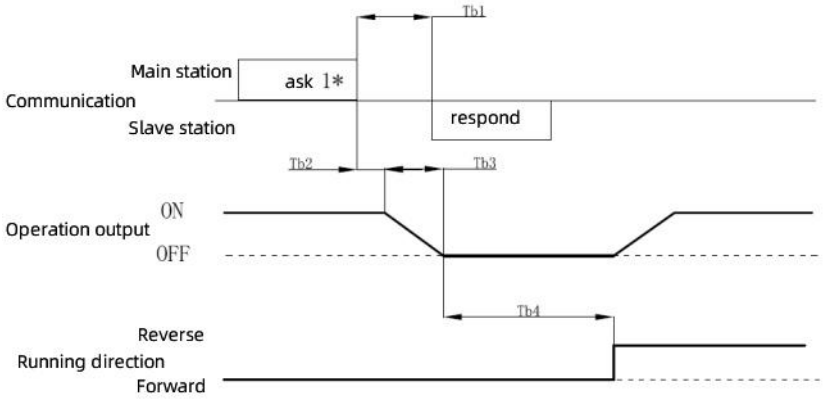
●B running direction (1Byte)

Control motor running direction:

Send data	motor action
00	The motor runs forward
01：	The motor runs in reverse

※The direction of forward rotation and reverse rotation of the motor is relative, and the actual running direction of the motor shall prevail.

Direction Switch Timing



NO.	Name	Content
1*	ask for information	Contains information to reverse the motor
Tb1	response interval	Command processing time + sending waiting time
Tb2	command response interval	command processing time
Tb3	Motor deceleration time	It is determined by the deceleration time configured by the drive and the current motor inertia,
Tb4	Direction switching time	The time from motor deceleration to stop to switching direction running,

Hint

When running in the forward direction, the driver will immediately execute the reverse running action when it receives the reverse switching data. If it receives the forward command again during the forward deceleration process, it will terminate the deceleration and direction switching, turn to the forward running and accelerate to target speed.

●B motor running speed/angle (2Byte)

In the speed closed-loop mode, control the B motor to run at the target speed. In angle closed-loop mode, control the target angle of B motor rotation;

Operating mode	motor action
speed closed loop	Motor speed, the range of speed value is set minimum speed - set maximum speed r/min, the unit is 1r/min. The first Byte: the high 8 bits of the speed value; The second Byte: the lower 8 bits of the speed value;
Angle closed loop	The motor controls the rotation angle, the angle value range is 0-12000°, and the steering wheel rotation angle is set, the unit is 0.01°; The first Byte: the high 8 bits of the speed value; The second Byte: the lower 8 bits of the speed value;

Inquiry Information 3

Inquiry information serial number	1Byte	**3Byte**
Response Information Choices	1Byte	
Driver output port designation	1Byte	

●Inquiry information serial number (1Byte)

The query information serial number is: 03;

●Response information selection (1Byte)

Control the slave driver to respond to data;

●Drive output port designation (1Byte)

1bit (high bit)	1bit	1bit	1bit	1bit	1bit	1bit	1bit (low bit)
invalid	invalid	invalid	invalid	Y3	Y2	Y1	Y0

4.1.7 Response information

CAN BUS communication characteristics, only 8Byte data can be sent at a time, and the driver responds to the data by sending

multiple times.

response message 1

Response message serial number	1Byte	**8 Byte**
A motor running current	1Byte（high bit）	
	1Byte（low bit）	
A Motor running direction/direction	1Byte	
A motor running speed/angle	1Byte（high bit）	
	1Byte（low bit）	
Current state of the drive	1Byte（high bit）	
	1Byte（low bit）	

●Response information serial number (1Byte)

The serial number is: 01;

●A motor running current (2Byte)

A motor real-time current, motor current resolution is 0.1A, maximum return range is 100.0A;

●A motor running direction/direction (1Byte)

A The rotation direction of the motor speed under closed-loop control, and the rotation direction under the closed-loop state of steering;

	Speed closed loop	Steering closed loop
00	The motor runs forward	positive angle adjustment
01	The motor runs in reverse	reverse angle adjustment

●A motor running speed/angle (2Byte)

A The real-time speed of the motor or the current running angle;

Under speed closed-loop control, the real-time speed value of motor A is 0-set rated speed, and the accuracy is 1r/min;

Under the steering closed-loop control, the angle value range of the current operation of the A motor is 0-12000 (120.00°), and the angle accuracy is 0.01°.

●Current state of the drive (2Byte)

When the drive is in normal operation, return: 00; when the drive is in an error state, it will output a fault code. For details, please refer to the fault code table in Chapter 7;

response message 2

Response message serial number	**1Byte**	**8 Byte**
B motor running current	1Byte（high bit）	
	1Byte（low bit）	
B Motor running direction/direction	1Byte	
B motor running speed/angle	1Byte（high bit）	
	1Byte（low bit）	
Current state of the drive	1Byte（high bit）	
	1Byte（low bit）	

●Response information serial number (1Byte)

The serial number is: 02;

●B motor running current (2Byte)

B motor real-time current, motor current resolution is 0.1A, maximum return range is 100.0A;

●B motor running direction/direction (1Byte)

A The rotation direction of the motor speed under closed-loop control, and the rotation direction under the closed-loop state of steering;

	Speed closed loop	Steering closed loop
00	the motor runs forward	positive angle adjustment
01	the motor runs in reverse	reverse angle adjustment

●B motor running speed/angle (2Byte)

B The real-time speed of the motor or the current running angle;

Under speed closed-loop control, the real-time speed value of motor B is 0-set rated speed, and the speed accuracy is 1r/min;

Under the steering closed-loop control, the current operating angle value range of B motor is 0-12000 (120.00°), and the angle accuracy is 0.01°.

●Power bus voltage (2Byte)

The voltage value of the drive power supply input. The resolution is 0.01V;

Example: if the input voltage is 24.43V, then the value is 2443 in decimal to hexadecimal, 09 (high order) and 8B (low order).

Response message 3

Response message serial number	1Byte	**7Byte**
A motor running direction	1Byte	
B motor running direction	1Byte	
A motor running speed	1Byte（high bit）	
	1Byte（low bit）	
B motor running speed	1Byte（high bit）	
	1Byte（low bit）	

●Response information serial number (1Byte)

The serial number is: 03;

●A motor running direction (1Byte)

	speed closed loop	Steering closed loop
00	The motor runs forward	positive angle adjustment
01：	The motor runs in reverse	reverse angle adjustment

●B motor running direction (1Byte)

	speed closed loop	Steering closed loop
00	The motor runs forward	positive angle adjustment
01：	The motor runs in reverse	reverse angle adjustment

Hint

The forward and reverse running directions of the motor are relative directions, and the direction of the motor is also affected by the motor wiring and the motor reducer. The drive can be rotated in this default direction.

●A motor running speed (2Byte)

Under speed closed-loop control, the real-time speed value of motor A is 0-set rated speed, and the speed accuracy is 1r/min;

●B motor running speed (2Byte)

Under speed closed-loop control, the real-time speed value of motor B is 0-set rated speed, and the speed accuracy is 1r/min;

Response message 4

Response message serial number	1Byte	7Byte
A motor steering angle	1Byte（high bit）	
	1Byte（low bit）	
B motor steering angle	1Byte（high bit）	
	1Byte（low bit）	
A motor hall sensor count value	1Byte（high bit）	
	1Byte（low bit）	

●Response information serial number (1Byte)

The serial number is: 04;

●A motor steering angle (2Byte)

Only when the A motor is set in the angle closed-loop mode, the feedback is the current angle of the A motor;

●B motor steering angle (2Byte)

Only when the B motor is set in the angle closed-loop mode, the feedback is the current angle of the B motor;

●A motor hall sensor count value (2Byte)

Store the number of pulses input by the hall sensor of motor A, the number of pulses ranges from 0 to 65535, and starts from 0 after exceeding, the cumulative counting of the motor forward rotation, the cumulative counting down of the motor reverse rotation, and the data is cleared after restarting the power;

Response message 5

Response message serial number	1Byte	7Byte
B motor hall sensor count value	1Byte（high bit）	
	1Byte（low bit）	
Reserved	1Byte	
	1Byte	
Reserved	1Byte	
	1Byte	

●Response information serial number (1Byte)

The serial number is: 05;

●B motor hall sensor count value (2Byte)

Store the number of pulses input by the hall sensor of motor B, the number of pulses ranges from 0 to 65535, and starts from 0 after exceeding, the cumulative counting of the motor forward rotation, the cumulative counting of the reverse rotation of the motor, and the data is cleared after restarting the power;

●Reserved bit (4Byte)

Invalid data bit: 00;

Response message 7

Response message serial number	1Byte	7Byte
Reserved	1Byte	

	1Byte	
A Motor Hall Angle Value	1Byte (high bit)	
	1Byte (low)	
B Motor Hall Angle Value	1Byte (high bit)	
	1Byte (low)	

●Response information serial number (1Byte)

The serial number is: 07;

●Reserved bit (2Byte)

invalid data bits;

●A motor Hall angle value (2Byte)

Signed data, when A is a brushless DC motor and uses the Hall sensor as the steering closed-loop feedback, the current rotation angle of the A motor is measured, the value range is -16000-16000 (160.00°), and the angle accuracy is 0.01°;

●B motor Hall angle value (2Byte)

Signed data, when B is a brushless DC motor and uses the Hall sensor as the steering closed-loop feedback, the current B motor rotation angle is measured, the value range is -16000-16000 (160.00°), and the angle accuracy is 0.01°;

Response message 9

Response message serial number	1Byte	7Byte
Reserved	1Byte	
	1Byte	
AVI1 input voltage value	1Byte (high bit)	
	1Byte (low)	
AVI2 input voltage value	1Byte (high bit)	
	1Byte (low)	

●Response information serial number (1Byte)

The serial number is: 09;

●Reserved bit (2Byte)

invalid data: 00;

●AVI1 input voltage value (2Byte)

When the driver is in the IO analog voltage control state, control the running speed of motor A, the voltage range is 0.00-5.00V, and the voltage accuracy is 0.01V;

●AVI2 input voltage value (2Byte)

When the driver is in the IO analog voltage control state, control the running angle of motor B, the voltage range is 0-5.00V, and the voltage accuracy is 0.01V;

Response message 10

Response message serial number	**1Byte**	**7Byte**
Enter port code value	1Byte	
Output port code value	1Byte	
Driver temperature value	1Byte (high bit)	
	1Byte (low)	

A motor finds the origin mark	1Byte	
B motor finds the origin mark	1Byte	

●Response information serial number (1Byte)

The serial number is: 10;

● Input port code value (1Byte)

Input port X0-X5 state:

1bit (high bit)	**1bit**	**1bit**	**1bit**	**1bit**	**1bit**	**1bit**	1Byte (low bit)
reserve	reserve	X5	X4	X3	X2	X1	X0

●Output Port Encoding Value (1Byte)

Output port Y0-Y3 status:

1bit (high bit)	**1bit**	**1bit**	**1bit**	**1bit**	**1bit**	**1bit**	1Byte (low bit)
reserve	reserve	reserve	reserve	Y3	Y2	Y1	Y0

●Drive temperature value (2Byte)

Signed data, the current drive temperature measured by the drive, and the measurement range is -40-125°C.

●A motor finds the origin flag

00: Steering closed-loop control did not find the origin or did not complete the search for the origin;

01: The steering closed-loop control completes finding the origin;

*The flag is only valid when the A motor is under steering closed-loop control.

●B motor finds the origin flag

00: Steering closed-loop control did not find the origin or did not complete the search for the origin;

01: The steering closed-loop control completes finding the origin;

*The flag is only valid when t

4.2 BY CANOPEN

4.2.1 basic concept

Electronic Data File (EDS): A directory used to record and describe all parameters of CAN open nodes; all CAN open parameters such as node type, manufacturer, serial number, version information, baud rate, etc. can be obtained through the EDS file. When using the CAN OPEN function in a PLC, etc., it is necessary to import the EDS file first, and configure it through the object dictionary recorded in the EDS.

Service Data Object (SDO): SDO is mainly used for parameter configuration of CAN OPEN master station to slave station nodes. After each SDO message is sent, there will be a response message to ensure that the message is delivered accurately. In a CAN OPEN communication system, usually there is only one master station as the client, and the slave station node responds to the SDO message as the server of the SDO. The client of the master station accesses and modifies the corresponding index parameters in the EDS of any node through the SOD message.

Process data object (PDO): PDO is used to transmit real-time data, one-way transmission, no need to receive slave station node response, a PDO message can transmit up to 8 byte data, mainly used to transmit real-time data. The master station node and the

slave station node pre-define the sending data and receiving data corresponding to the PDO, so there is no need to confirm again during the transmission process, and there is no response time and response data, thereby improving the bus utilization.

4.2.2 Communication Specifications

communication mode	bus mode
Communication rate	1000k (default), 500k, 250k, 125k
Correspondence address	1-127 (default)
TPDO	4
RPDO	4
Online message	support
Heartbeat message	support
Heartbeat consumption	support
SDO	support

4.2.3 Control message

The node supports the control message (NMT Module Control message) sent by the master station, and the control message information does not respond

message format

COB-ID	DLC	Byte 0	Byte 1
0x 000	1	CS control function code	Node-ID

CS：control function code

Command code	NMT service	Effect
0x01	enter operation state	Node starts PDO send and receive status
0x02	enter stop state	Node enters stopped state
0x80	enter pre-operational state	Node enters pre-operational state for configuring PDO mapping and parameter setting
0x81	reset node	Reset the node application layer, and the communication parameters are restored to the initial state value;
0x82	reset communication	Reset node communication, re-initialize communication when communication is disturbed, node bus error, bus off

Node-ID：Node address, when Node-ID=0, all nodes are controlled

4.2.4 Node online message

After the node is powered on and initialized, it sends a frame of online message.

message format

The frame ID of the node online message is 0x700+Node-ID, the data length is 1, and the data is 00;

COB-ID	**DLC**	**Byte 0**

0x 700+Node-ID	1	00

4.2.5 heartbeat consumption

The node supports the consumption of heartbeat messages, and the monitoring of heartbeat producers starts from receiving the first heartbeat.

Note: The consumer heartbeat timeout should be greater than the corresponding producer heartbeat period.

<table>
<tr><th>index</th><th>sub-index</th><th>Data Type</th><th>Default Value</th><th>Description</th></tr>
<tr><td rowspan="2">0x1016</td><td>0x00</td><td>Uns 8</td><td>0x01</td><td>Number of parameter entries</td></tr>
<tr><td>0x01</td><td>Uns 32</td><td>0x00</td><td>Consumer heartbeat over time
<table>
<tr><td>31-24bit（MSB）</td><td>23-16bit</td><td>15-0bit（LSB）</td></tr>
<tr><td>reserve</td><td>Node-ID</td><td>heartbeat overtime</td></tr>
</table>
If the heartbeat time is 0 or the Node-ID is 0 or greater than 127, the corresponding object entry is invalid. Heartbeat timeout in ms.</td></tr>
</table>

4.2.6 heartbeat message

The node supports sending heartbeat messages, configure the heartbeat time according to the index 0x1017, and send heartbeat messages;

Index	Sub-index	Data Type	Default Value	Description
0x1017	None	Uns 16	0x00	Configure the heartbeat packet sending cycle, in ms;

Message format

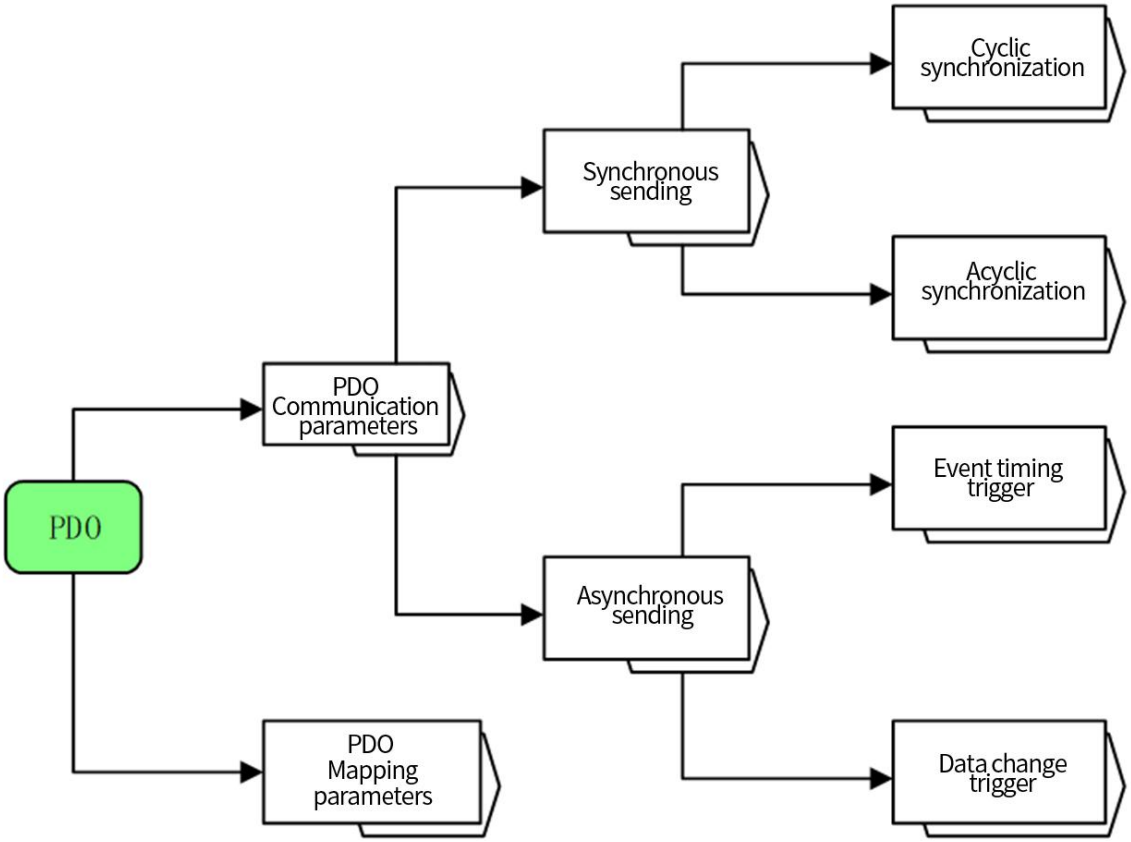
COB-ID	DLC	Byte 0
0x 700+Node-ID	1	status code

The frame ID of the node online message is 0x700+Node-ID, the data length is 1, and the data represents the current node status.

status code:

Status code	Status
0x05	enter operation state
0x04	enter stop state
0x7F	enter pre-operational state

4.2.7 process data object（PDO）



PDO transmission/reception is observed by the (slave) CAN node side.

The node defines 4 TPDOs and 4 RPDOs;

PDO message format

COB-ID	DLC	Data
	1-8	1-8Byte

COB-ID

Object	COB-ID
TPDO1 (send)	0x180+Node ID
RPDO1 (receive)	0x200+Node ID
TPDO2 (send)	0x280+Node ID
RPDO2 (receive)	0x300+Node ID
TPDO3 (send)	0x380+Node ID
RPDO3 (receive)	0x400+Node ID
TPDO4 (send)	0x480+Node ID
RPDO4 (receive)	0x500+Node ID

Data (1-8Byte): The length is determined according to the mapping data in the mapping parameter

PDO communication parameters

Index	Sub-index	Data Type	Default Value	Description
RPDO	0x00	Uns 8	0x08	Number of parameter entries
0x1400--0x15FF	0x01	Uns 32	0x 00	COB-ID
TPDO	0x02	Uns 8	0x 00	Synchronization type
0x1800--0x19FF	0x03	Uns 16	0x 00	Production prohibition constraint time (unit: 100us)
	0x05	Uns 16	0x 00	Event timer trigger time (unit: ms)
	0x06	Uns 8	0x 00	SYNC synchronization initial value

Number of parameter entries: the number of sub-indexes contained in this index;

COB-ID: Receive and send the frame ID of the current PDO

Synchronization type:

Synchronization type	Trigger condition (B=two conditions met, O=one condition met)			PDO transmission
	SYNC(Synchronization message received)	RTR(Remote frame received)	Event(data change or timer)	
00	B	-	B	acyclic synchronization
01-240	O	-	-	Cycle, synchronization, (1-240 value represents the number of received SYNC messages between two PDO messages)
241-251	-	-	-	reserve
252	B	B	-	Sync, after RTR
253	-	O	-	Asynchronous, after RTR
254	-	O	O	Asynchronous, manufacturer-specific events
255	-	O	O	Asynchronous, device subprotocol specific events

Production prohibition constraint time: the minimum interval between two PDO transmissions, to avoid excessive PDO transmissions, resulting in increased bus load;

Event timer trigger time: PDO timing sending cycle time;

SYNC synchronization initial value: Set the number of received synchronization messages, PDO for synchronous transmission, and reply PDO data only after receiving the set number of times of synchronization messages. For example, if it is set to 5, it will respond to PDO data after receiving 5 synchronization messages;

PDO mapping parameters

Set the data in the corresponding index that the current PDO needs to transmit, and complete the custom transmission of data by setting the PDO to map to the corresponding index. The maximum length of a PDO is 8Byte, and the length of the mapped data cannot exceed 8Byte. The sensor has a default mapping, and the mapping can be changed according to requirements;

RPDO default mapping

Only RPDO1, RPDO2, and RPDO3 are used in the default mapping;

Index	Sub-index	Data Type	Default Value	Description
RPDO1	0x00	Uns 8	8	Number of parameter entries
0x1600	0x01	Uns 32	0x20010008	Mapped as 8-bit length data of index 0x2001 sub-index 0x00;
	0x02	Uns 32	0x20030008	Mapped as 8-bit length data of index 0x2003 sub-index 0x00;
	0x03	Uns 32	0x20050010	Mapped as 16-bit length data of index 0x2005 sub-index 0x00;

	0x04	Uns 32	0x20080010	Mapped as 16-bit length data of index 0x2008 sub-index 0x00;
	0x05-0x08	Uns 32	0x00	not mapped;
RPDO2	0x00	Uns 8	8	Number of parameter entries
0x1601	0x01	Uns 32	0x20020008	Mapped as 8-bit length data of index 0x2002 sub-index 0x00;
	0x02	Uns 32	0x20040008	Mapped as 8-bit length data of index 0x2004 sub-index 0x00;
	0x03	Uns 32	0x20060010	Mapped as 16-bit length data of index 0x2006 sub-index 0x00;
	0x04	Uns 32	0x20090010	Mapped as 16-bit length data of index 0x2009 sub-index 0x00;
	0x05-0x08	Uns 32	0x00	not mapped;
RPDO3	0x00	Uns 8	8	Number of parameter entries
0x1602	0x01	Uns 32	0x20000008	Mapped as 8-bit length data of index 0x2000 sub-index 0x00;
	0x02-0x08	Uns 32	0x00	not mapped;

TPDO default mapping

Index	Sub-index	Data Type	Default Value	Description
TPDO1	0x00	Uns 8	8	Number of parameter entries
0x1A00	0x01	Uns 32	0x200A0010	Mapped as 16-bit length data of index 0x200A sub-index 0x00;
	0x02	Uns 32	0x200C0008	Mapped as 8-bit length data of index 0x200C sub-index 0x00;
	0x03	Uns 32	0x200E0010	Mapped as 16-bit length data of index 0x200E sub-index 0x00;
	0x04	Uns 32	0x20100010	Mapped as 16-bit length data of index 0x2010 sub-index 0x00;
	0x05-0x08	Uns 32	0x00	not mapped;
TPDO2	0x00	Uns 8	8	Number of parameter entries
0x1A01	0x01	Uns 32	0x200B0010	Mapped to the 16-bit length data of index 0x200B sub-index 0x00;
	0x02	Uns 32	0x200D0008	8-bit length data mapped to index 0x200D sub-index 0x00;
	0x03	Uns 32	0x200F0010	Mapped to the 16-bit length data of index 0x200F sub-index 0x00;
	0x04	Uns 32	0x20110010	Mapped as 16-bit length data of index 0x2011 sub-index 0x00;
	0x05-0x08	Uns 32	0x00	not mapped;
TPDO3	0x00	Uns 8	8	Number of parameter entries
0x1A02	0x01	Uns 32	0x20160010	Mapped as 16-bit length data of index 0x2016 sub-index 0x00;
	0x02	Uns 32	0x20230010	Mapped as 16-bit length data of index 0x2023 sub-index 0x00;
	0x03	Uns 32	0x20220008	8-bit length data mapped to index 0x200F sub-index 0x00;
	0x04-0x08	Uns 32	0x00	not mapped;
TPDO4	0x00	Uns 8	8	Number of parameter entries
0x1A03	0x01	Uns 32	0x20240008	Mapped as 8-bit length data of index 0x2024 sub-index 0x00;
	0x02	Uns 32	0x20250008	Mapped as 8-bit length data of index 0x2025 sub-index 0x00;
	0x03-0x08	Uns 32	0x00	not mapped;

4.2.8 service data object（SDO）

Send message format

COB-ID	DLC	LBS				MBS
0x600+Node-ID	8	control function code	object index	object subindex	up to 4 bytes	

Control function code

Command code	Description
0x40	Command word when sending message
0x2F	Indicates that the sending data is 1 byte;
0x2B	Indicates that the sent data is 2 bytes;
0x23	Indicates that the sending data is 4 bytes;

Node response message format

COB-ID	DLC	LBS			
0x580+Node-ID	8	Receive command code	object index	object subindex	up to 4 bytes

control function code

Command code	Describe
0x60	SDO message sent successfully
0x80	SDO message error
0x4F	Indicates that the received data is 1 byte;
0x4B	Indicates that the received data is 2 bytes;
0x43	Indicates that the received data is 4 bytes;

4.2.9 EDS file

Please visit the website to obtain the EDS file;

4.2.10 Index definition

Object index	Object subindex	Object description	Data type	Attributes	Defaults
0x2000	0x00	Troubleshooting	Uns 8	read/write	When the drive is in the state of reporting errors, it is necessary to reset the fault of the drive. Send by default: 00, Reset operation: 01 00, 01 data keep at least one data sending cycle.
0x2001	0x00	A motor running	Uns 8	read/write	00: The driver is in a disabled state, and the motor is in a free state; 01: The driver is in the enabled state, the motor is in the control state, and it runs according to the input speed; 02: The driver decelerates to stop, and stops according to the deceleration time; 03: The drive brakes and stops. After receiving this information, the motor will decelerate and stop according to the maximum braking force.
0x2002	0x00	B motor running	Uns 8	read/write	00: The driver is in a disabled state, and the motor is in a free state;

<table>
<tr><td></td><td></td><td></td><td></td><td></td><td>01: The driver is in the enabled state, the motor is in the control state, and it runs according to the input speed;
02: The driver decelerates to stop, and stops according to the deceleration time;
03: The drive brakes and stops. After receiving this information, the motor will decelerate and stop according to the maximum braking force.</td></tr>
<tr><td>0x2003</td><td>0x00</td><td>A motor running direction</td><td>Uns 8</td><td>read/write</td><td>0: A motor rotates forward;
1: The motor rotates in reverse;</td></tr>
<tr><td>0x2004</td><td>0x00</td><td>B motor running direction</td><td>Uns 8</td><td>read/write</td><td>0: B motor rotates forward;
1: The motor rotates in reverse;</td></tr>
<tr><td>0x2005</td><td>0x00</td><td>A motor speed</td><td>Uns 16</td><td>read/write</td><td>0-8000: Under speed closed-loop control, the speed of motor A, the range of speed value is 10-8000r/min.</td></tr>
<tr><td>0x2006</td><td>0x00</td><td>B motor speed</td><td>Uns 16</td><td>read/write</td><td>0-8000: Under speed closed-loop control, the speed of motor B, the range of speed value is 10-8000r/min.</td></tr>
<tr><td>0x2007</td><td>0x00</td><td>output port control</td><td>Uns 8</td><td>read/write</td><td>0-15: Control the driver output port Y0-Y3 switch output in the communication control mode; the relationship between the output value and the port action;
<table>
<tr><th>Y0</th><th>Y1</th><th>Y2</th><th>Y3</th></tr>
<tr><td>1</td><td>2</td><td>4</td><td>8</td></tr>
</table></td></tr>
<tr><td>0x2008</td><td>0x00</td><td>A motor angle</td><td>Uns 16</td><td>read/write</td><td>0-12000: Under angle closed-loop control, the motor controls the rotation angle, the value range is 0-12000, and the corresponding angle range is 0-120.00°</td></tr>
<tr><td>0x2009</td><td>0x00</td><td>B motor angle</td><td>Uns 16</td><td>read/write</td><td>0-12000: Under angle closed-loop control, the motor controls the rotation angle, the value range is 0-12000, and the corresponding angle range is 0-120.00°</td></tr>
<tr><td>0x200A</td><td>0x00</td><td>A motor current current</td><td>Uns 16</td><td>read only</td><td>0-1000: Motor real-time current, motor current resolution is 0.1A, maximum return range is 100.0A.</td></tr>
<tr><td>0x200B</td><td>0x00</td><td>B motor current current</td><td>Uns 16</td><td>read only</td><td>0-1000: Motor real-time current, motor current resolution is 0.1A, maximum return range is 100.0A.</td></tr>
<tr><td>0x200C</td><td>0x00</td><td>A The current direction of the motor</td><td>Uns 8</td><td>read only</td><td>0: The motor rotates forward;
1: The motor rotates in reverse;</td></tr>
<tr><td>0x200D</td><td>0x00</td><td>B motor current direction</td><td>Uns 8</td><td>read only</td><td>0: The motor rotates forward;
1: The motor rotates in reverse;</td></tr>
<tr><td>0x200E</td><td>0x00</td><td>A The current speed of the motor</td><td>Uns 16</td><td>read only</td><td>0-8000: Under speed closed-loop control, the motor speed, the range of speed value is 10–8000r/min, the actual maximum speed is determined by the characteristics of the motor.</td></tr>
<tr><td>0x200F</td><td>0x00</td><td>B motor current speed</td><td>Uns 16</td><td>read only</td><td>0-8000: Under speed closed-loop control, the motor speed, the range of speed value is 10-8000r/min. , the actual maximum speed is determined by the motor characteristics.</td></tr>
<tr><td>0x2010</td><td>0x00</td><td>A motor current angle</td><td>INT 16</td><td>read only</td><td>-16000--16000: Under angle closed-loop control, motor A controls the rotation angle, corresponding to the angle range</td></tr>
</table>

<table>
<tr><td></td><td></td><td></td><td></td><td></td><td>-160.00°-160.00°.</td></tr>
<tr><td>0x2011</td><td>0x00</td><td>B motor current angle</td><td>INT 16</td><td>read only</td><td>-16000--16000: Under angle closed-loop control, motor B controls the rotation angle, corresponding to the angle range -160.00°-160.00°</td></tr>
<tr><td>0x2012</td><td>0x00</td><td>A Motor Hall count value</td><td>Uns 16</td><td>read only</td><td>0-65535: The number of pulses input by Hall sensor of A motor.</td></tr>
<tr><td>0x2013</td><td>0x00</td><td>B Motor Hall count value</td><td>Uns 16</td><td>read only</td><td>0-65535: The number of input pulses of Hall sensor of B motor.</td></tr>
<tr><td>0x2014</td><td>0x00</td><td>invalid</td><td></td><td></td><td></td></tr>
<tr><td>0x2015</td><td>0x00</td><td>invalid</td><td></td><td></td><td></td></tr>
<tr><td>0x2016</td><td>0x00</td><td>voltage</td><td>Uns 16</td><td>rite</td><td>0-8000: The voltage value of the drive power input. The resolution is 0.01V
Example: the input voltage is 24.43V, the value is 2443 in decimal to hexadecimal, 09 (high) 8B (low)</td></tr>
<tr><td>0x2017</td><td>0x00</td><td>invalid</td><td></td><td></td><td></td></tr>
<tr><td>0x2018</td><td>0x00</td><td>invalid</td><td></td><td></td><td></td></tr>
<tr><td>0x2019</td><td>0x00</td><td>invalid</td><td></td><td></td><td></td></tr>
<tr><td>0x201A</td><td>0x00</td><td>invalid</td><td></td><td></td><td></td></tr>
<tr><td>0x201B</td><td>0x00</td><td>invalid</td><td></td><td></td><td></td></tr>
<tr><td>0x201C</td><td>0x00</td><td>invalid</td><td></td><td></td><td></td></tr>
<tr><td>0x201D</td><td>0x00</td><td>invalid</td><td></td><td></td><td></td></tr>
<tr><td>0x201E</td><td>0x00</td><td></td><td></td><td></td><td></td></tr>
<tr><td>0x201F</td><td>0x00</td><td>V1 voltage</td><td>Uns 16</td><td>只读</td><td>0-1000: The voltage value of the A motor input to the V1 port to adjust the speed or angle, and the detected voltage range is 0-10.00V.</td></tr>
<tr><td>0x2020</td><td>0x00</td><td>V2 voltage</td><td>Uns 16</td><td>read only</td><td>0-1000: The voltage value of motor B input to adjust the speed or angle of the V1 port, and the detected voltage range is 0-10.00V.</td></tr>
<tr><td>0x2021</td><td>0x00</td><td>input port status</td><td>Uns 8</td><td>read only</td><td>0-63: Control the driver output port X0-X5 switch input in the communication control mode; the relationship between the input value and the port action;<table><tr><th>X0</th><th>X1</th><th>X2</th><th>X3</th><th>X4</th><th>X5</th></tr><tr><td>1</td><td>2</td><td>4</td><td>8</td><td>16</td><td>32</td></tr></table></td></tr>
<tr><td>0x2022</td><td>0x00</td><td>Drive current status code</td><td>Uns 8</td><td>read only</td><td>00-15: Display the current fault status code of the drive. For detailed fault code analysis, refer to Chapter 7.</td></tr>
<tr><td>0x2023</td><td>0x00</td><td>drive current temperature</td><td>INT 16</td><td>read only</td><td>-40-125: Current drive temperature measured by the drive, measuring range -40-125°C.</td></tr>
<tr><td>0x2024</td><td>0x00</td><td>A motor finds the origin state</td><td>Uns 8</td><td>read only</td><td>00: Steering closed-loop control did not find the origin or did not complete the search for the origin;
01: The steering closed-loop control completes finding the origin;
*The flag is only valid when the A motor is under steering closed-loop control.</td></tr>
</table>

0x2025	0x00	B motor finds the origin state	Uns 8	read only	00: Steering closed-loop control did not find the origin or did not complete the search for the origin; 01: The steering closed-loop control completes finding the origin; *The flag is only valid when the B motor is under steering closed-loop control.
0x2026	0x00	A motor Hall angle	INT 16	read only	-16000--16000: When the Hall signal of motor A is directly used as angle feedback, the current angle of motor A, the angle range is -160.00–160.00°. It is only valid when the A motor is in the steering closed-loop mode;
0x2027	0x00	B Motor Hall Angle	INT 16	read only	-16000--16000: When driving a brushless DC motor, when directly using the Hall signal of motor B as angle feedback, the current angle of motor B, the angle range is -160.00–160.00°. It is only valid when the B motor is in the steering closed-loop mode;

5 FAULT STATUS CODE

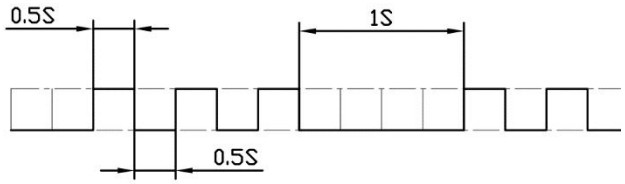
Fault code of the drive in case of abnormal state

Fault code	LED flashing	Fault description	Handling measures
1	1	A motor short circuit protection	1) Fault reset or restart of the drive; 2Is the acceleration/deceleration time parameter too small or the PID parameter too large; 3) Is the power supply voltage too low than the motor operating voltage; 4) Test after replacing the battery for power supply; 5) Check if there is a short circuit in the motor circuit; 6) Swap motor A and B for testing to determine if the motor end issue is still a driver issue; 7) Replace the driver, motor, or cable after determining the cause
2	2	Short circuit protection for motor B	
3	3	A motor overload protection	1) Fault reset or restart of the drive; 2) Is the acceleration/deceleration time parameter too small or the PID parameter too large; 3) If the motor is equipped with a band brake, check if the band brake opens normally; 4) Whether there is a malfunction in the external mechanical structure; 5) Check if the wiring sequence of the drive end motor U, V, and W is correct; 6) Swap motor A and B for testing to determine if the motor end issue is still a driver issue; 7)Determine the cause and replace the driver, motor, or cable;
4	4	B motor overload protection	
5	5	A motor stall protection	1) Fault reset or restart of the drive; 2) No or abnormal motor speed feedback signal; 3) Motor feedback signal parameter setting error; 4) Motor stuck or overspeed; 5) Inappropriate parameters for motor stall protection; 6)Is the acceleration and deceleration time too small, or the PID parameters too large;
6	6	B motor stall protection	
7	7	A motor limit protection	1) Fault reset or restart of the drive; 2) Check if the wiring of the corresponding motor limit port is a normally closed signal; 3) Check if the corresponding motor limit switch is triggered; 4)The corresponding motor parameters are configured in angle closed-loop mode;
8	8	B motor limit protection	
9	9	A motor angle adjustment reverse	1) Fault reset or restart of the drive; 2)Check the installation direction parameters of the corresponding encoder or the default rotation direction parameters of the motor;
10	10	B motor angle adjustment reverse	

11	11	Communication connection failure of upper computer	1) Fault reset or restart of the drive; 2) Check the connection between the driver and the upper computer; 3) Check whether the upper computer sends data; Check the data format and transmission interval sent by the upper computer;
12	12	Undervoltage protection	1) Fault reset or restart of the drive; 2) The power supply voltage is too low; The undervoltage protection threshold voltage is set too high;
13	13	Overvoltage protection	1) Fault reset or restart of the drive; 2) The power supply voltage is too high; The overvoltage protection threshold voltage is set too low;
14	14	Drive high temperature	The operating temperature of the driver exceeds 80 ℃;
15	15	A motor Hall signal error	1) Fault reset or restart of the drive; 2) Check if the signal wire connection of the Hall sensor at the motor end is intact, and if the sequence of HU, HV, and HW is correct; 3) Swap motor A and B for testing to determine if the motor end issue is still a driver issue; Determine the cause and replace the driver, motor, or cable;
16	16	B motor Hall signal error	
0	Light	Drive normal state	Normal state

※LED alarm example

The display status of the LED when the driver is reporting a fault number 3



If an error occurs, please check the cause first before restoring the drive.

If the possible fault is eliminated and the drive fault still occurs, it is possible that the drive has been damaged.

After troubleshooting, follow the drive startup steps to restart the drive

Hint

The above faults are detected by the driver, and the causes of the faults include but are not limited to the above.

6 AFTER SALES AND TECHNICAL SUPPORT

6.1 WARRANTY PERIOD

The driver provides a limited warranty period. During the warranty period, if the product cannot be used normally due to quality issues or design defects, Xingsong Technology will provide free after-sales maintenance.

■ Warranty period: 2 years from sale.

6.2 WARRANTY SCOPE

If the product is within the warranty period and falls within the warranty conditions, Hinson's after-sales service will provide free repair or replacement

■The warranty and after-sales service of this product are only limited in Chinese Mainland;

■The product cannot be used normally due to unpacking during transportation;

■The inability to function properly due to damage to the components of the product itself;

■Product design defects that prevent normal use;

6.3 SCOPE OF EXEMPTION

During the use of the product, please note that within the following conditions, Xingsong Technology after-sales service will not provide free after-sales service or warranty.

Failure to install the product correctly according to the instructions, resulting in product damage;

Using this product in unsuitable environments and conditions, resulting in product damage;

Damage to the product due to non standardized operation in the product manual;

Disassembling or repairing products without permission from our company;

Product damage caused by irresistible external forces such as natural disasters and fires.

6.4 TECHNICAL SUPPORT

Website：www.hinson-xs.com

QQ：2636178756

■Appendix I

C Language CRC Check Function

cBuffer：Calculate an array of CRC checksums.

iBufLen：The length of the array.

```
unsigned int CRC_Verify(unsigned char *cBuffer, unsigned int iBufLen)
{
    unsigned int i, j;                          //#define wPolynom 0xA001
    unsigned int wCrc = 0xffff;
    unsigned int wPolynom = 0xA001;
/*-----------------------------------------------------------------------------*/
    for (i = 0; i < iBufLen; i++)
    {
        wCrc ^= cBuffer[i];
for (j = 0; j < 8; j++)
        {
            if (wCrc &0x0001)
            {    wCrc = (wCrc >> 1) ^ wPolynom;    }
            else
            {   wCrc = wCrc >> 1;    }
         }
     }
return wCrc;
}
```

PLC writing CRC verification function

The CRC-16 code consists of two bytes. At the beginning, each bit of the CRC register is pre-set to 1 (0xffff). Then, the CRC register is XOR with 8-bit data, and the CRC register is shifted from high to low. Zero is filled in the highest bit (MSB) position, while the lowest bit (LSB) is already shifted out of the CRC register. If it is 1, the register is XOR with the predefined polynomial code (16 # A001), Otherwise, if LSB is zero, there is no need to perform XOR. Repeat the above shift from high to low 8 times. After processing the first 8-bit data, XOR the CRC register value with the next 8-bit data and perform 8 shifts similar to the previous data. After all character processing is completed, the value in the CRC register is the final CRC value.

The following is the calculation process of CRC:

Set the CRC register and assign it the value FFFF (hex).

XOR the first 8-bit character of the data with the lower 8 bits of the 16 bit CRC register and store the result in the CRC register.

Move the CRC register one bit to the right, fill the MSB with zero, move it out, and check the LSB.

If LSB is 0, repeat step three; If LSB is 1, the CRC register is different from the polynomial code.

Repeat steps 3 and 4 until all 8 shifts are completed. At this point, an 8-bit data processing is completed.

Repeat steps 2 to 5 until all data is processed.

The content of the final CRC register is the CRC value.

Input parameters:

```
Pointer to the data area to be verified, with the first byte being the data length
LD0          DataBuff          IN          DWORD
Network 1
 LD          SM0.0
 MOVW        16#FFFF, AC0                  //Initialize CRC register
 BTI         *LD0, LW4                     //The first byte of the data buffer is the data length
 MOVD        LD0, LD6
 INCD        LD6                           //Pointer to the first pending byte
 Network 2
 LD          SM0.0
 FOR         AC2, +1, LW4                  //Start looping through each byte
 Network 3
 LD          SM0.0
 XORB        *LD6, AC0                     //Bytes are XOR first with the low bits of the CRC register
 Network 4
 LD          SM0.0
 FOR         AC1, +1, +8                   //Shift processing loop, processing 8 bits of a byte
 Network 5
 LD          SM0.0
 SRW         AC0, 1                        //CRC Register Shift Right One Bit
 Network 6
 LD          SM1.1                         //If the displacement bit is 0, it enters the next cycle
 XORW        16#A001, AC0                  //If the shift bit is 1, the CRC register is XOR with polynomial 16 # A001
 Network 7
 NEXT
 Network 8
 LD          SM0.0
 INCD        LD6                           //Pointer plus 1 points to the next byte
 Network 9
 NEXT
 Network 10
 LD          SM0.0
 SWAP        AC0                           //Swap CRC register high and low bytes
 MOVW        AC0, *LD6                     //CRC verification value written to the end of the data area
```

Note: After the PLC writes the CRC verification function, use it to calculate the CRC verification code. Compare the verification code with the CRC verification code calculated by the SinloCRC16 verification tool. If it is not the same, then the CRC verification function written by the PLC must be incorrect.