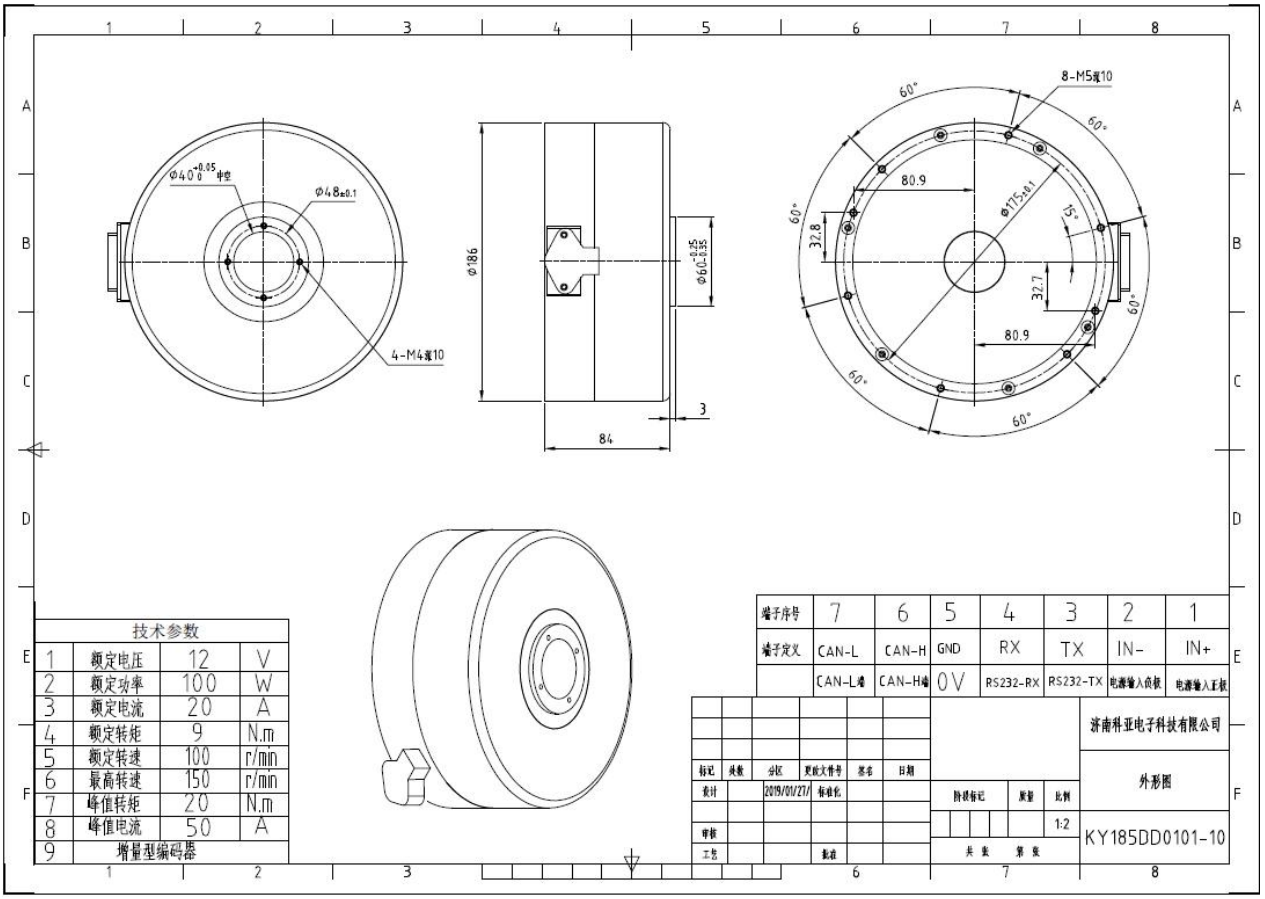
ELECTRIC STEERING MOTOR
User manual(V1.4.5)

JINANKEYA ELECTRON SCIENCE AND TECHNOLOGY CO.,LTD.
(2020-3-20)

I. Overview

1.1 Motor Parameter



1.2 Specification

- **Motor: DC12V/24V 100W**
- Continuous current 25A, maximum peak current 50A (1s)
- DC working power +9V~30VDC
- Working mode: speed mode, torque mode

1.3 Operation Condition

1.3.1 Power supply:

- Rated working power: 12V/24VDC
- Limit power supply range: 7--32VDC
- It can provide instantaneous current overload capability of 2 times continuous current

1.3.2 Feedback component:

- Incremental encoder (standard product), absolute encoder(custom made).

1.3.3 Working environment:

- Operating temperature: -25~55°C (based on environment temperature);

  Storage temperature: -35~65°C (based on environment temperature);
- Humidity: 5%--90%RH, condensation (25°C)
- Protection level: IP65,
- Insulation performance: input to the chassis DC600V, leakage current 0.07mA.The insulation resistance is 20 MΩ or more.
- Three-proof requirements: meet the requirements of three defenses (dust, moisture, salt spray).
- Vibration requirements: Frequency 5Hz ~ 25Hz, amplitude 3mm, 0.09g.

  Frequency 25Hz~200Hz, amplitude 1.47mm, 116g.

  Horizontal, vertical, and longitudinal directions are 30 min in each direction.
- Cooling method: natural cooling

II. Functional Technical Indicators

2.1 Main Function

- Working mode: Speed mode, torque mode
- Feedback: Incremental encoder (standard product), absolute encoder(custom made).
- Control: RS232, CAN, 0-5V analog signal(custom made)
- External control start/stop
- Regenerative
- Fault LED indicator
- Can be controlled by CANopen bus networking
- Realize motor speed control and data reading through RS232;
- Internal temperature monitoring of drive
- Overcurrent and overload protection
- Overvoltage and undervoltage protection
- Temperature protection
- Locked-rotor and over speed protection
- Motor short circuit protection

2.2. Working Mode Configuration Table

Operating mode	Control instruction		Feedback component
Speed mode	RS232	CAN	Incremental encoder absolute encoder
	Analog voltage 0-5v		
Position mode	RS232	CAN	Incremental encoder absolute encoder
	Analog voltage 0-5v		

2.3. Technical Parameters

Parameter	Label	Parameter value	unit
Voltage	U	9-30	VDC
Max continuous current	I_c	25	A
Max peak current	I_{max}	50 (1s overcurrent protection)	A
PWM switching frequency	f_{pwm}	10	kHz
Output encoder power supply	$+5V_{out}$	5	VDC
	I_{cc}	100	mA
Digital input	EN、DIR	Cutoff (high level): less than 1mA Brakeover (low level): 3~7mA	mA
Analog input impedance	Single-ended input	20	KΩ
Analog signal voltage	Single-ended input	0~5V	V
Under voltage	Vu	9 (adjustable)	V
Over voltage	Vo	30(adjustable)	V
Operating temperature	Industrial grade (standard product)	-25 ~ +55	℃
	Military grade	-40 ~ +65	
Storage temperature	Industrial grade (standard product)	-35 ~ +65	℃
	Military grade	-55 ~ +85	

III. Port Description

3.1. Interface Definition

7- core Definition (standard)	1	IN+	Power input +
	2	IN-	Power input -
	3	TX	RS232—TX
	4	RX	RS232—RX
	5	GND	0V
	6	CAN-H	CAN-H
	7	CAN-L	CAN-L

12- core Definition	1	IN+	Power input +
	2	IN+	Power input +
	3	IN-	Power input -
	4	IN-	Power input -
	5	D/A	Analog signal input（0-5V）
	6	TX	RS232—TX
	7	RX	RS232—RX
	8	GND	0V
	9	CAN-H	CAN-H
	10	CAN-L	CAN-L
	11	K1	Control switch, (powered when connect to K2)
	12	K2	Control switch, (powered when connect to K1)

DB17W2	1	5V	5Vout
	2	TX	RS232—TX
	3	RX	RS232—RX
	4	485+	RS485+
	5	485-	RS485-
	6	GND	0V
	7		
	8		
	9	GND	0V
	10	SIN	Analog signal（0-5V）
	11	EN	Enable
	12	F/R	Reversible
	13	5V	5Vout
	14	CAN-L	CAN-L
	15	CAN-H	CAN-H
	A1	IN-	Power input -
	A2	IN+	Power input +

3.2. Interface Description

3.2.1 TX, RX, GND: RS232 interface, to achieve command control, as well as parameter settings, operating state commissioning, etc.

3.2.2 CAN-H, CAN-L: CANopen interface

Drive internal already provided 120 Ω terminal resistance, do not need to add other terminal when use.

3.2.3 D/A: Analog input interface

It can be used for D/A Angle sensor signal input port to query the current real-time signal value through instructions;

D/A, 0V: analog control interface can be formed to realize speed and position mode control (±5 turns);

3.2.4 **IN+ 、IN- ：**

- Power input port: When the aviation plug is 12 cores, in order to ensure the reliability of the terminal power, two sets of power supplies are required to be used in parallel.
- Because the vehicle requires long power cable, when the current is large, the voltage drop more due to the line loss. We recommend below cable specification:

Cable length (m)	Cabel diameter (mm^2)	Allowable Continue Amps
1-3	2.5	＜17A
3-4.5	4	＜25A

Note: When the motor “undervoltage alarm”, there may be the following reasons:

(1) The battery is aging, and the internal resistance of the battery will increase after a long time of use, thereby reducing the battery's discharge capacity.

(2) The steering hydraulic pump is aging, the flow valve is blocked, etc., which causes the steering resistance to increase and the motor current to increase.

(3) The cable diameter is too thin, the voltage drop more, and when the torque is large, the voltage is pulled down, causing the driver to detect undervoltage.

3.3. Serial Port Connection

Using high-speed standard serial cable, DB9 plug meets the label definition:

Driver label	RS232 cable
TX	2
RX	3
GND	5

IV. Operating Instructions

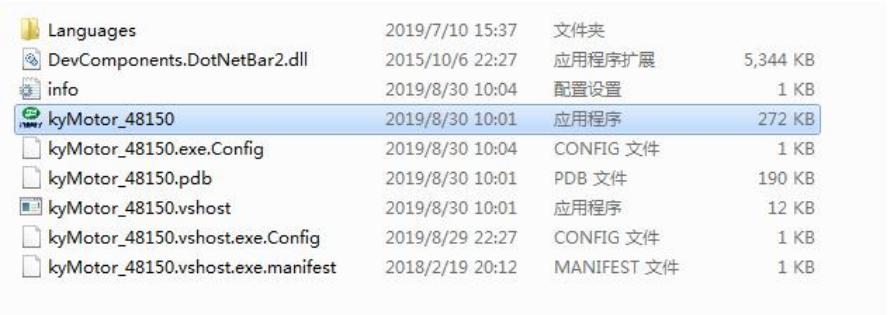
4.1. The Software Description

4.1.1 Configuration instructions

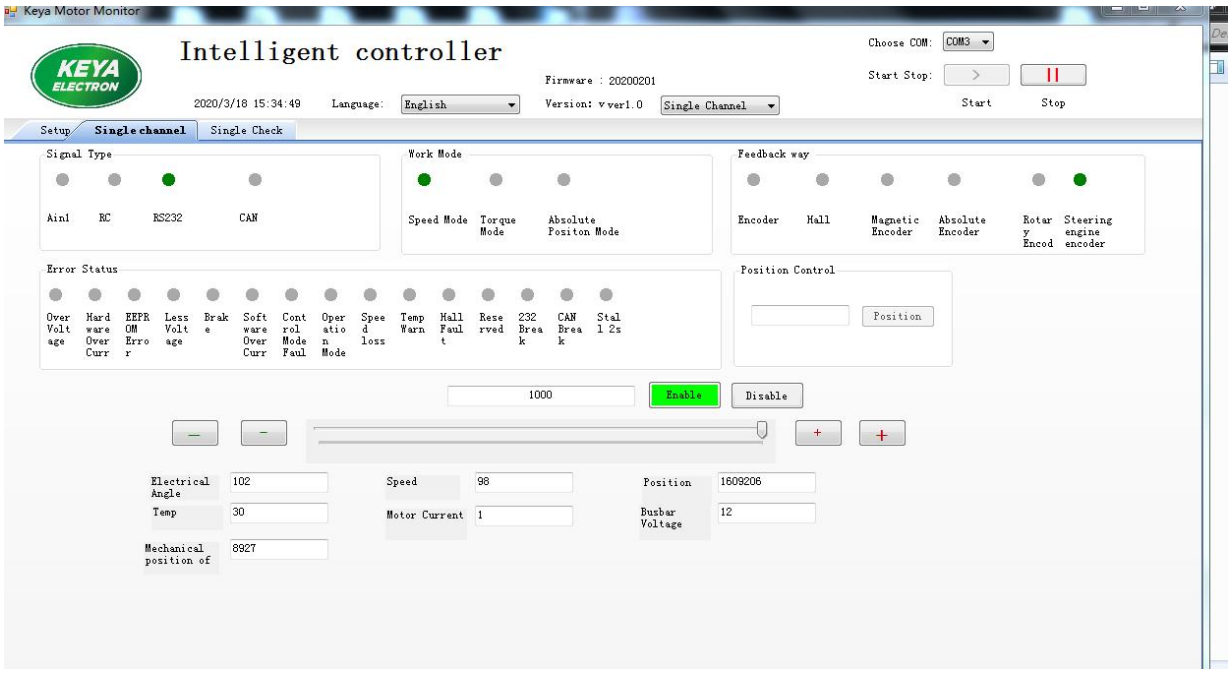
- Servo controller parameters can be set by the software.
- The software communicates with the control through RS232, and the baud rate is 115200bit.
- The software is developed under the .NET environment. XP systems need to have .NET 4.0 installed.

4.1.2 The software instructions

4.1.2.1 Double click the icon



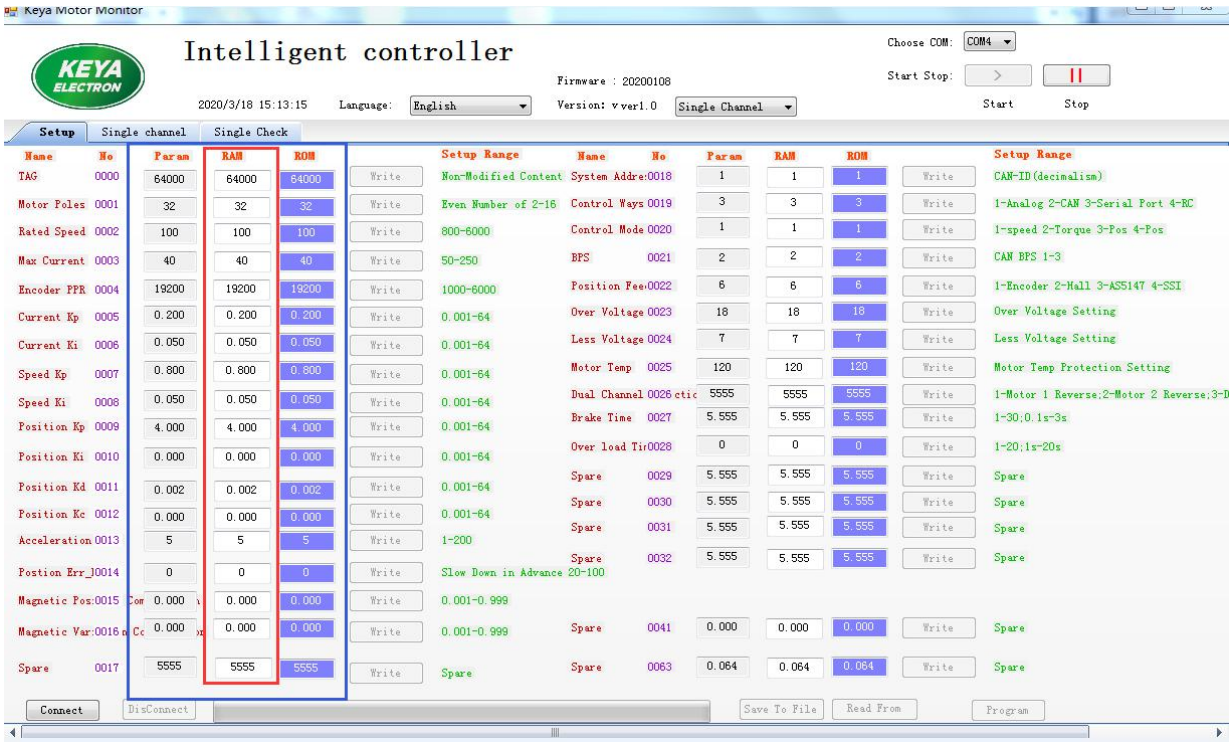
4.1.2.2 If the communication is successful, the interface will read the control parameters, while the LED on the upper left will flash green, indicating that the parameters are communicating successfully.



4.1.2.3 Open the configuration interface and click the connection button in the lower left corner to establish a connection between the software and the controller.

4.1.2.4 The RAM in the red box can be modified. The left side of the red box is the controller parameter, and the right side of the red box is the data in the E²ROM. In the correct case, the three data are consistent (equal).

As the software data is continuously scanned, when modifying the data, modify it quickly and click the “Write” button.



4.1.2.5 For example, if you want modify the number of encoder lines from 2500ppr to 1024ppr, the E^2 ROM data is 2500, please write 1024 in the RAM, and click “Write” button quickly. Confirm that 1024 is no longer changing. Same steps for other parameters, multiple parameters can be modified at the same time.

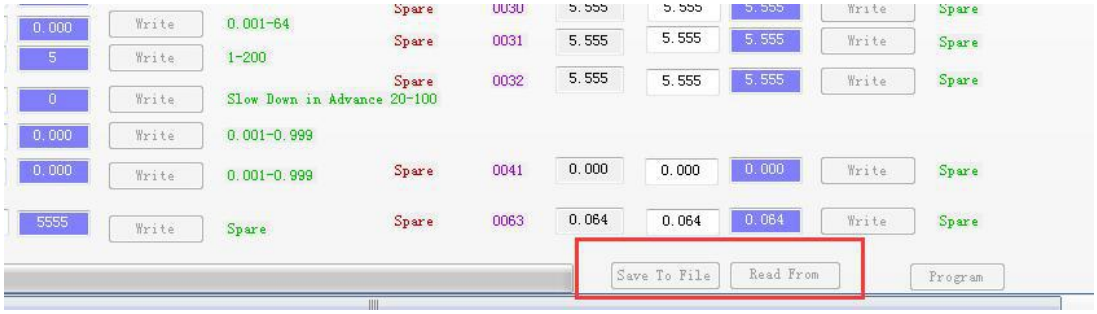
4.1.2.6 Click the "Program" button at the bottom right. Program the data in RAM to E2ROM. Note: The programming process takes 3 seconds.

4.1.2.7 The “Program” button turns red, indicating that data is being programmed. Please wait and observe the data. Untill it remind “Programed successfully”. Then the three datas(RAM, ROM, Param) in blue block are consistent, indicating that the ROM data is programed into the controller.

4.1.2.8 At this point, the modification of the control parameters is completed. Click the "Disconnect" button and click the "Exit" button.

4.1.2.9 Re-power the controller. (Note: No matter whether the configuration is modified or not, it must be powered off and reset to start normally)

4.1.2.10 When programming the configuration for multiple motors, you can “save a modified configuration to a file” and then “read from a file” to download to another motor.



4.1.3 Parameter function description

0000 Identifier. when the system is connected, identify the software communication or serial port control. (Don’t need modify)

0001 The number of motor poles (this motor is 32 poles)

0002 Rated motor speed (set to 100)

0003 Motor maximum current (Don’t need modify)

0004 The number of encoder lines, according to the encoder specification, the standard is 19200

0005 Kp parameter of controller current loop PI control (typical value 0.2)

Can be modified appropriately.

0006 Ki parameter of controller current loop PI control (typical value 0.05)

Can be modified appropriately.

0007 Kp parameter of controller speed loop PI control (typical value 0.8)

Can be modified appropriately.

0008 Ki parameter of controller speed loop PI control (typical value 0.05)

Can be modified appropriately.

0009-0012 Position loop PID control parameters

0013 Acceleration time. "10" means the acceleration time from 0rpm to rated speed is 1s.

0015 Zero position compensation of magnetic encoder

0016 Zero position compensation of rotary encoder

0018 Controller system address, or node number of control.

This parameter is used in the CAN, CANOpen, and EtherCAT buses.

For example: set the data to 1, then the ID in CAN bus: 0x0600000 + controller system address, it will be (0x06000001)

0019 Control signal selection

2. CAN open control; 3. RS232

0020 Control mode selection, including speed control, position control

1. Speed control,
2. Torque control,
3. Absolute position control, refer to the CAN open protocol
4. Relative postion control, refer to the CAN open protocol

0021 CAN bus baud rate selection (we recommend 250K)

1.125k 2.250k 3. 500k 4. 1M

0022 Position sensor selection

1. Incremental encoder
2. Hall
3. Magnetic encoder
4. SSI absolute position encoder
5. Rotary enocder
6. Steering motor incremental encoder

Other parameters: spare

4.2 Control Signal Selection

The control signal is selected by the software, this system support CAN, RS232, Analog + Switch(custom-made).

4.3 Indicator Description

4.3.1 Status indicator (blue light): Observe the status of the controller according to the blinking frequency of the indicator.

Number of flashes	Definition	Cause of issue
1	Working normally	Disability state
2	Over voltage	Supply voltage is over 30V(adjustable)
3	Hardware overcurrent protection 60A	Overcurrent protection caused by motor short circuit and field tube damage
4	EEPROM error	Data saving error
5	Less voltage	Supply voltage is lower than 9V(adjustable)
6	brake	Turn on the brake signal
7	Software overcurrent protection (software set protection value)	The phase current reaches the software setting protection value for 1 second to stop output.
8	Control mode failure	Control mode selection error
9	Working mode failure	Speed, torque working mode not selected or wrong
10	Speed loss protection	Actual speed exceeds 25% of rated value
11	Temperature alarm	The temperature above 80 °C
12	Hall error	Motor Hall is off or malfunctioning
13	Reserved	Reserved
14	232 break	232 mode, no 232 signal input
15	CAN break	CAN mode, no CAN signal input
16	Blocking for 2 seconds	Motor stalled 2s protection

4.3.2 Enable indicator (Red)

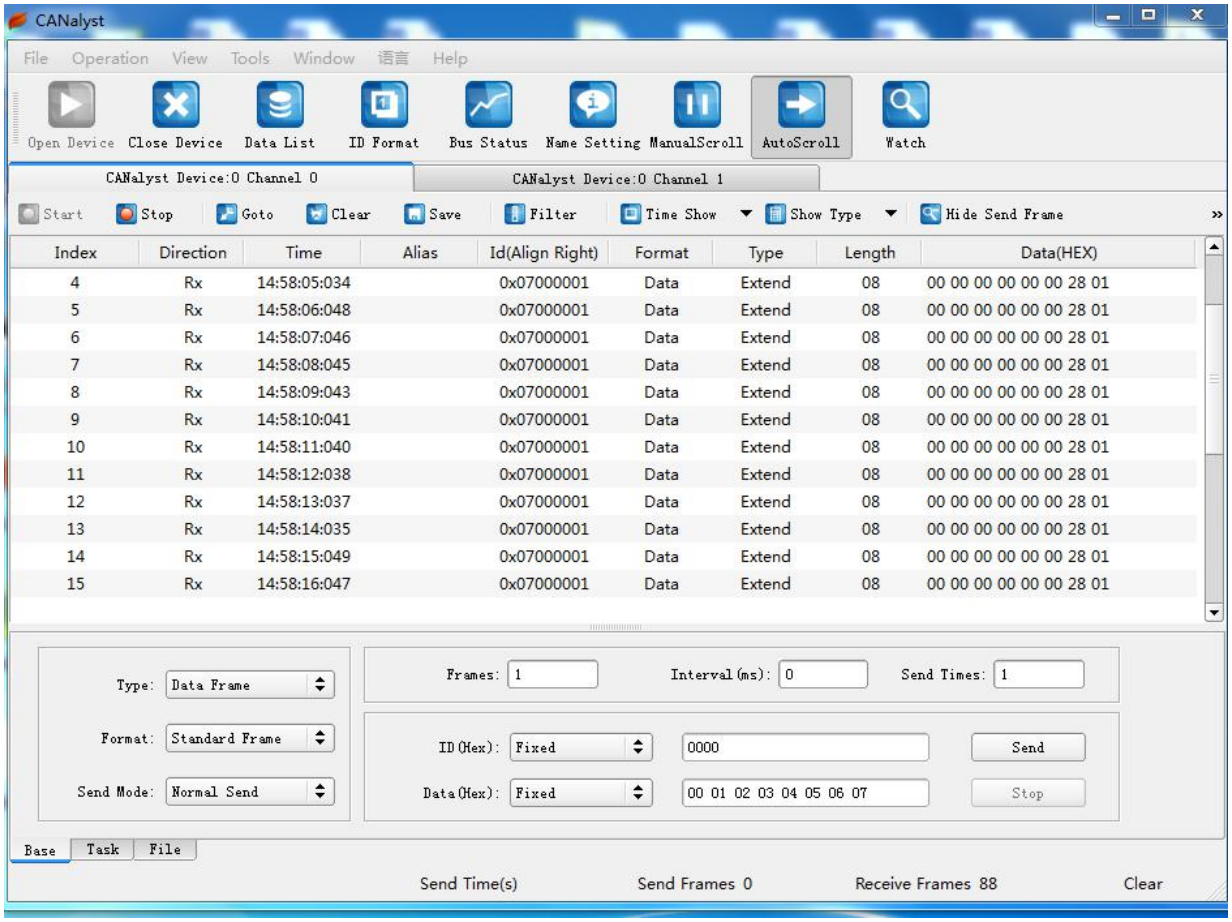
In any control mode, the red indicator light will go out after the drive is enabled.

The indicator light is always on when the controller disabled.

4.4 CANOpen Instruction

4.4.1 General Configuration

- CAN bus protocol baud rate 250Kb
- CAN bus ID with extended ID
- Sending data format: low before, high later
- According to the CANOpen format, the data adopts the query mode.
- According to the CANOpen format, there is a fixed heartbeat and send related data.
- The watchdog detects the line-off period of 1000ms (speed command is sent continuously, the interval must not exceed 1000ms)
- Query data returns are hexadecimal data, which need to be converted into decimal data.



4.4.2 CAN bus instruction

Note 1: The controller ID is a decimal number in the configuration software, and the CAN software is a hexadecimal number.

Example: 1. The configuration software sets the controller ID to 1, and the CAN software ID is 06000001 (extension ID).

2. The configuration software sets the ID to 112, and the CAN software ID is 06000070.

Note 2: ID of sending data: 0x0600000 + controller ID (hexadecimal)

ID of returned data: 0x0580000 + controller ID (hex)

ID of heartbeat data: 0x0700000 + controller ID (hex)

Enable: 23 0d 20 01 00 00 00 00

Return ID: 0x0580000 + controller ID (hexadecimal)

Data 60 0d 20 00 00 00 00 00

Disable: 23 0c 20 01 00 00 00 00

Return ID: 0x0580000 + controller ID (hexadecimal)

Data 60 0c 20 00 00 00 00 00

Speed control: 23 00 20 01 DATA_L(h) DATA_L(l) DATA_H(h) DATA_H(l)

Return ID: 0x0580000 + controller ID (hexadecimal)

Data 60 00 20 00 00 00 00 00

Motor current query: 40 00 21 01 00 00 00 00

Return ID: 0x0580000 + controller ID (hexadecimal)

Data 60 00 21 01 DAT1 DAT2 DAT3 DAT4

DAT1 =((unsigned char*)(&send_float))[3]

DAT2 =((unsigned char*)(&send_float))[4]

DAT3 =((unsigned char*)(&send_float))[1]

DAT4 =((unsigned char*)(&send_float))[2]

Fault query: 40 12 21 01 00 00 00 00

Return ID: 0x0580000 + controller ID (hexadecimal)

Data 60 12 21 01 DAT1 DAT2 00 00

DAT1 =((unsigned char*)(&TYPE_RunData.err)) [L]

DAT2 =((unsigned char*)(&TYPE_RunData.err)) [H]

TYPE_RunData.err is the fault code

Encoder speed query: 40 03 21 01 00 00 00 00

Return ID: 0x0580000 + controller ID (hexadecimal)

Data 60 03 21 01 DAT1 DAT2 00 00

DAT1 =((unsigned char*)(&send_float))[L]

DAT2 =((unsigned char*)(&send_float))[H]

Power supply voltage query: 40 0D 21 02 00 00 00 00

Return ID: 0x0580000 + controller ID (hexadecimal)

Data 60 0D 21 02 DATE 00 00 00

DATE =((unsigned char*)(&send_float))

Radiator temperature query: 40 0F 21 01 00 00 00 00

Return ID: 0x0580000 + controller ID (hexadecimal)

Data 60 0F 21 01 DATE 00 00 00

DATE = ((unsigned char*)(&send_short))

Encoder count value query: 40 04 21 01 00 00 00 00

Return ID: 0x0580000 + controller ID (hexadecimal)

Data 60 04 21 01 DAT1 DAT2 DAT3 DAT4

DAT1 = ((unsigned char*)(&send_int))[4];

DAT2 = ((unsigned char*)(&send_int))[3];

DAT3 = ((unsigned char*)(&send_int))[2];

DAT4 = ((unsigned char*)(&send_int))[1];

AD input query: 40 05 21 01 00 00 00 00

Return ID: 0x0580000 + controller ID (hexadecimal)

Data 60 05 21 01 DAT1 DAT2 00 00

DAT1 =((unsigned char*)(&send_float))[L]

DAT2 =((unsigned char*)(&send_float))[H]

Program version query: 40 01 11 11 00 00 00 00

Return ID: 0x0580000 + controller ID (hexadecimal)

Data 60 01 11 11 DAT1 DAT2 DAT3 DAT4

DAT1 = ((unsigned char*)(&send_int))[1];

DAT2 = ((unsigned char*)(&send_int))[2];

DAT3 = ((unsigned char*)(&send_int))[3];

DAT4 = ((unsigned char*)(&send_int))[4];

Heartbeat return command:

Return ID: 0x0700000 + controller ID (hexadecimal)

Return instruction: Data0 Data1 Data2 Data3 Data4 Data5 Data6 Data7

Data0 Data1, Electrical Angle: 0——1000

Data2 Data3, motor speed: – speed —— + speed

Data4 Data5, speed command: 0-1000 (rated speed) command value

Data6 Data7, Control_Close (fault code)

(Notice: the high data in the front and the low data is later)

4.4.3 CAN bus data description

4.4.3.1 Control mode

DataBox_MDL=0x230D2001 Enable

DataBox_MDL=0x230C2001 Disable

DataBox_MDL=0x23002001 Speed control

DataBox_MDL=0x23012001 Torque control

DataBox_MDL=0x23022001 Position control

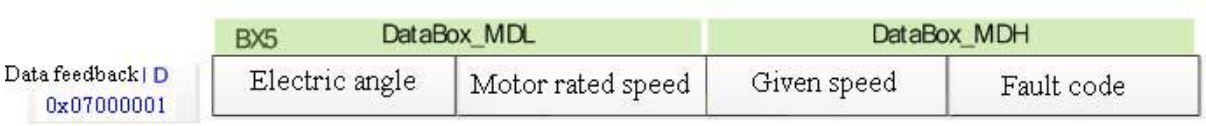
Note：need to make up 8 digit，Eg：23 0D 20 01 00 00 00 00

Speed: -1000 -- 1000, negative rated speed -- rated speed

Torque: -1000 -- 1000, negative rated torque x2 -- rated torque x2

Position: -25000 -- 25000, 2.5 circles clockwise -- 2.5 circles anticlockwise

4.4.3.2 Heartbeat data



4.4.4 CAN bus control example

4.4.4.1 Speed Control:
(Speed command value ‰) * (The setted max speed in the software) = the real speed.
If the setting max speed is 100rpm, then the Speed command setpoint -1000 - +1000 means -100rpm - +100rpm （0xFC18）(0x03E8）

The software setting control mode is CAN control (0019 is set to 2)

The software setting control mode is set to speed control (0020 is set to 1)

The software sets the system address to 1 (0018 is set to 1)

- If you want set the speed +50 (rated speed 100)

Control command ID: 0x06000001 (extended ID)

Enable: 23 0d 20 01 00 00 00 00

Speed given: 23 00 20 01 01 F4 00 00 (0x01F4 = 500)

- If you want set the speed -50 (rated speed 100)

Control command ID: 0x0600 0001 (extended ID)

Enable: 23 0d 20 01 00 00 00 00

Speed given: 23 00 20 01 FE 0C FF FF

4.4.2 Position control:
Position given value -50000 - 50000 means 5 circles clockwise - 5 circles anticlockwise (0x3CB0 FFFF) (0XC350 0000)

The software setting control mode is CAN control (0019 is set to 2)

The software setting control mode is absolute position control (0020 is set to 3)

Or the software setting control mode is set to relative position control (0020 is set to 4)

The software sets the system address to 1 (0018 is set to 1)

Control command ID: 0x0600 0001 (extended ID)

- Data transmission order:
  - （a） Disability 23 0C 20 01 00 00 00 00
  - （b） Enable 23 0D 20 01 00 00 00 00
  - （c） Position control: 23 02 20 01 DATA_L(h) DATA_L(l) DATA_H(h) DATA_H(l)

For example, you need the motor rotate 1.8 circles clockwise

- （a） Make sure the position control has been switched
- （b） Enable 23 0D 20 01 00 00 00 00
- （c） Position control command: 23 02 20 01 B9 B0 FF FF

For example, you need the motor to rotate the mechanical angle 76 degrees

counterclockwise (76 * (10000 / 360) = 2052 = 0x0804)

- （a） Make sure the position control has been switched
- （b） Enable 23 0D 20 01 00 00 00 00
- （c） Position control command: 23 02 20 08 04 00 00

4.5 Serial port instructions

4.5.1 General Configuration

Serial port configuration

The serial port communication port of the controller is set as follows:

115200bits/s
8-bit data
1 start bit
1 stop bit
No parity

- **Send control command format**

4.5.1.1 Speed control mode: 11 00 00 00 data1 data2 data3 data4

[data] is given speed value, hexadecimal, high in front, low in back.

1) Eg: speed mode , system disable

10 00 00 00 00 00 00 00

2) Eg: Speed mode, Speed 0rpm, system enable

11 00 00 00 00 00 00 00

3) Eg: Speed mode, Speed 100rpm, system enable + forward

11 00 00 00 00 00 03 E8

4) Eg: Speed mode, Speed -100rpm, System anable + reverse

11 00 00 00 FF FF FC 18

4.5.1.2 Position control mode: 21 00 00 00 data1 data2 data3 data4

[data] is given position value, hexadecimal, high in front, low in back.

1) Eg: position mode, anticlockwise 0.5 circles(machanical), enable

21 00 00 00 00 00 13 88

2) Eg: Position mod, clockwise 1.5 circles, enable

21 00 00 00 FF FF C5 67

3) Eg: Position mode, disable

20 00 00 00 00 00 00 00

- **Notice：Time interval between two instructions when sending instructions continuously.**

  20ms < Time interval < 500ms

  Command value range: int16 (signed)

  Speed control: -1000 ----+1000 (reverse and forward rated speed)

  Torque control: -1000 ----+1000 (reverse and forward rated torque)

  Position control: -25000 ----+25000 (reverse and forward 2.5 circles)

- **Query data format**

Marker (Data0): ED means query data command.

Marker (Data1): 00 means control states

01 -- Electric angle

02 -- Speed

03 -- Current

04 -- Rotor machanical position

05 -- Voltage

06 -- Temperature

07 -- Error code

08 -- Position

09 -- Program version

Such as:

Read controller motor speed

The software send: ED 02 00 00 00 00 00 00

Controller feedback: ED 02 00 64 00 00

Then, the motor speed is 100rpm

Read controller voltage

The software send: ED 05 00 00 00 00 00 00

Controller feedback: ED 05 0B 00 00 00

Then, the controller voltage is 12V

Read controller temperature (℃)

The software send: ED 06 00 00 00 00 00 00

Controller feedback: ED 06 00 1A 00 00

Then, the controller temperature is 26℃

Read controller control status

The software send: ED 00 00 00 00 00 00 00

Controller feedback: ED 00 64 10 00 00

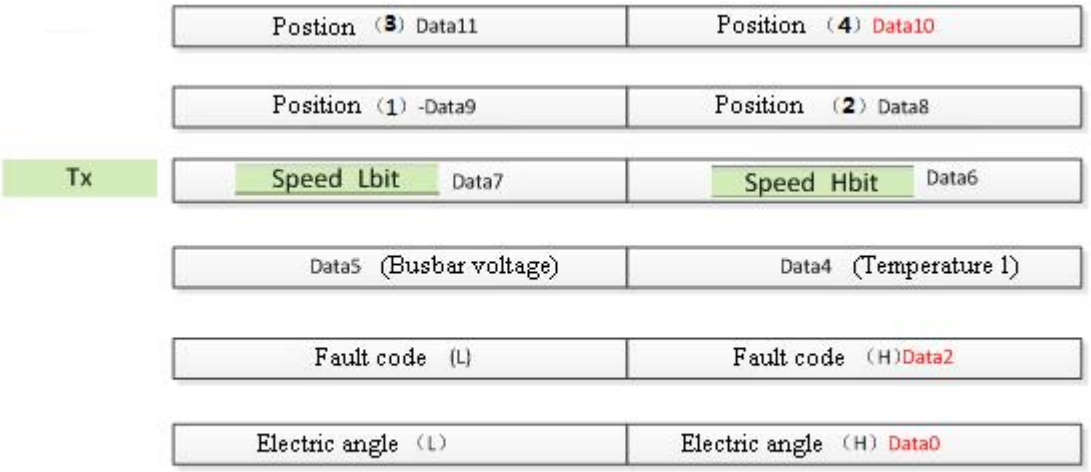
Then, the controller control status: Electric steering motor encoder, RS232 control, speed mode

6	4	1	0
1. Incremental encoder 2. Hall 3. Magnetic encoder 4. Absolute position encoder 5. Rotary enocder 6. Steering motor incremental encoder	1. AIN 2. CAN 4. RS232 8. RC	1. Speed mode 2. Position mode 3. Torque mode	

Note: The data returned by the query are all in hexadecimal and need to be converted to decimal to read.

Receive data: After successful communication, the drive status will be uploaded automatically. The data is defined as follows: (hexadecimal)

Easy to accept identification, starting with the first byte: 0xEE



- Fault code failure resolution:

At first, convert the hexadecimal to binary, then check the “1” position from the right to the left, which corresponds to the indicator blinking frequency.

Eg: The feedback data is 03 01

0 3 0 1

0000 0011 0000 0001 ; means 1, 9, 10 three faults.

The faults refer to the table in Page 10.

V. Fault protection and reset

5.1 Security level

There is only one level of security in the protection mechanism: state latching.The protection mechanism of fault information at each level is as follows:

- State latching: After the fault occurs, the system turns off the PWM, set, output the FAULT signal.

5.2 Failure protection basis

5.2.1 Temperature alarm

When the temperature of the drive exceeds 85 °C, a temperature alarm is generated; when it is restored to 80 °C, the alarm flag is cleared automatically.

5.2.2 Overcurrent protection

When the phase current reaches 50A for 1 second, it stops. Re-enable reset.

5.2.3 Overvoltage and undervoltage protection

The system will make undervoltage protection when the power supply voltage is lower than 9V;

And the system will make overvoltage protection when the power supply voltage is higher than 18V.

5.3 Fault information table

Protection category	Security Level	Turn off the PWM output	FAULT output
Temperature	Status latch	Yes	Yes
Overcurrent	Status latch	Yes	Yes
Undervoltage	Status latch	Yes	Yes
Overvoltage	Status latch	Yes	Yes
EEPROM error	Status latch	Yes	Yes

Note: When the fault status is locked, the drive will stop the power output; you can use the DIS command or make the external enable to low to clear all fault flags.

Jinan Keya Electron Science And Technology Co.,ltd

Email: sales@jnky.com

Contact: Nancy Ni

www.dcmotorkeya.com