

---

# **ELECTRIC STEERING MOTOR**

## **User manual (V2.4)**

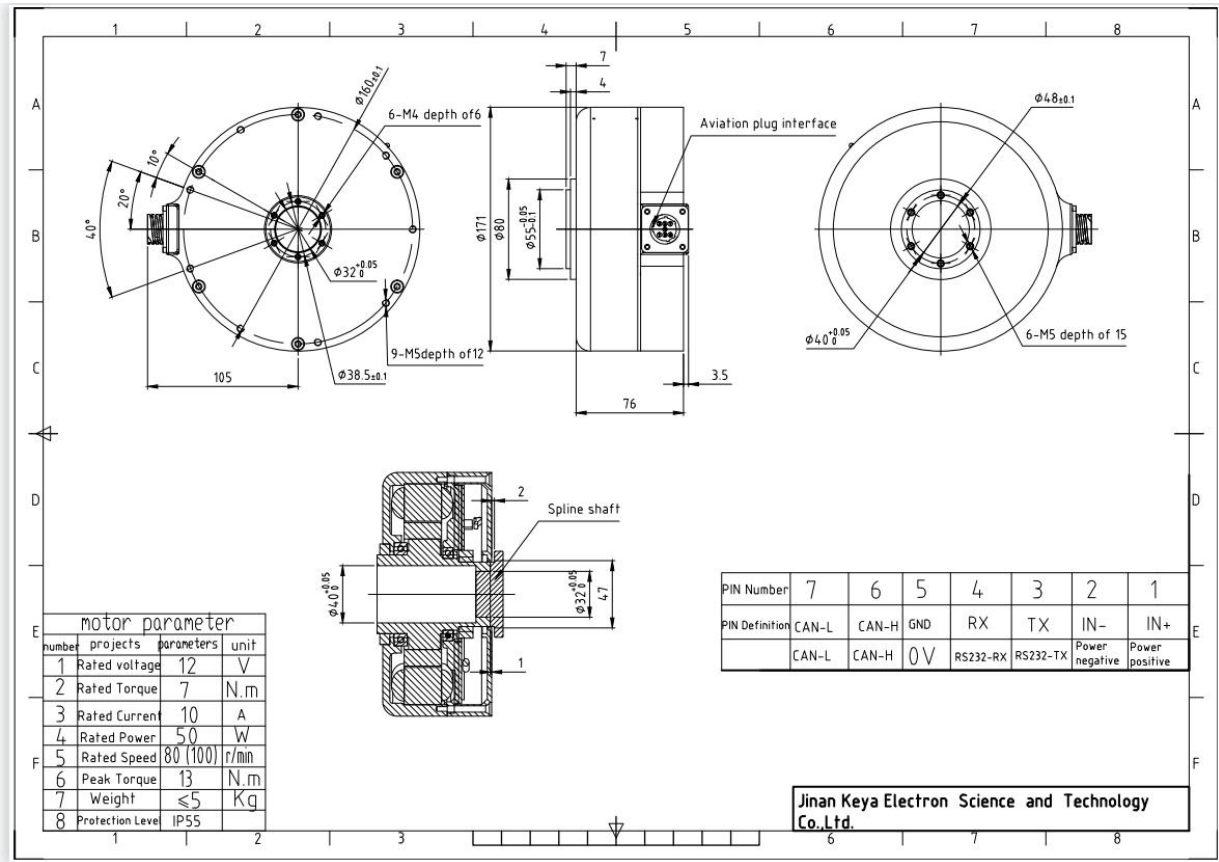


JINAN KEYA ELECTRON SCIENCE AND TECHNOLOGY CO., LTD.

---

# I. Overview

## 1.1 Motor Parameter (Model: KY170DD01005-08G)



Note: For more drawing details, please contact our sales team via [info@jnky.com](mailto:info@jnky.com)

---

## 1.2 Scope of Application

- Continuous current 10A, maximum peak current 20A;
- DC rated voltage: 12V (power supply range 9 ~ 18VDC); rated voltage: 24VDC (power supply range 9-28V);
- Speed mode, position mode (CAN);

## 1.3 Working Conditions

### 1.3.1 Power supply:

- Rated working power: 12VDC (power supply range 9 ~ 18VDC), rated voltage: 24VDC (power supply range 9-28V); It's recommended to use a battery as the power supply.
- Should provide instantaneous current overload capability of 2 times continuous current.

### 1.3.2 Requirements for operation

- Operators must wear electrostatic wristbands;
- The workbench must have anti-static measures;
- It is strictly forbidden to plug and unplug the aviation plug with power on.

### 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: IP55.
- 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. Technical Specifications

### 2.1 Main Functions

- Working mode: speed mode, position mode (CAN)
- Feedback part: Linear encoder.
- Control port: RS232, CAN
- LED indicator for faults
- Realize motor speed controlling and data reading through RS232
- Internal temperature monitoring and protection of built-in driver
- 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 signal		Feedback part
Speed mode	RS232	CAN	Linear encoder
Position mode	CAN		Linear encoder

### 2.3. Technical Parameters

Parameter	Label	Parameter value	unit
Voltage	U	9-30	VDC
Max continuous current	$I_c$	10	A
Max peak current	$I_{max}$	20	A
PWM switching frequency	$f_{pwm}$	10	kHz
Output encoder power supply	+5V <sub>out</sub>	5	VDC
	$I_{cc}$	100	mA
Under voltage protection	V <sub>u</sub>	7 (adjustable)	V

Over voltage protection	Vo	32(adjustable)	V
Operating temperature	Industrial grade (standard product)	-25 ~ +55	°C
	Military grade	-40 ~ +65	
Storage temperature	Industrial grade (standard product)	-35 ~ +65	°C
	Military grade	-55 ~ +85	

### III. Port Description

#### 3.1. Interface Definition

Port	Definition	Description	Recommended Wires
1	IN+	Power input +	2.5-4mm <sup>2</sup>
2	IN-	Power input -	2.5-4mm <sup>2</sup>
3	TX	RS232—TX	0.5 mm <sup>2</sup>
4	RX	RS232—RX	0.5 mm <sup>2</sup>
5	GND	0 V	0.5 mm <sup>2</sup>
6	CAN-H	CAN-H	0.5 mm <sup>2</sup>
7	CAN-L	CAN-L	0.5 mm <sup>2</sup>

(1) TX, RX, GND: RS232 interface, to achieve command controlling, parameter settings and operating state commissioning, etc.

(2) CAN-H, CAN-L: CAN interface, to achieve command controlling.

(3) **IN+ IN-** : Because the power line required by the vehicle is long, and the voltage drop will increase due to line loss when the current is large, it is recommended to select the wire specification according to this table.

Cable length (m)	Cable diameter (mm <sup>2</sup> )	Allowable continuous current (A)
1-3	2.5	<17A
3-4.5	4	<25A

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

- 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.
- 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 as well.
- The cable diameter is too small or the voltage decreases too much, and when the torque is high, the voltage is pulled down, causing the driver undervoltage while detecting.

---

### 3.2. Serial Port Connection

Adopt high-speed standard serial cable, DB9 plug conforms to the label definition:

Marks on driver	RS232 Pin No.
TX	2
RX	3
GND	5

## IV. Operating Instructions

### 4.1. Software configuration

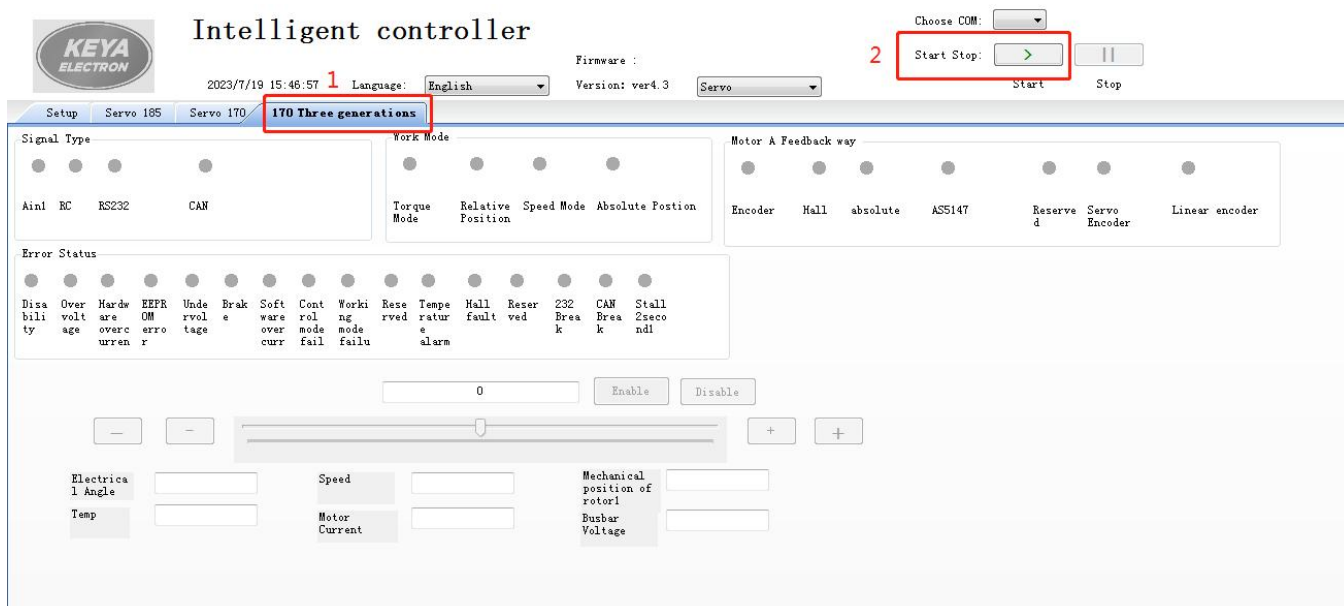
- The motor parameters can be set by the software.
- The software communicates with the motor through RS232, and the baud rate is 115200bit.
- The software is developed under the .NET environment. WINDOWNS XP system needs to have .NET 4.0 installed.

### 4.2 The software instructions

#### 4.2.1 Double click the program

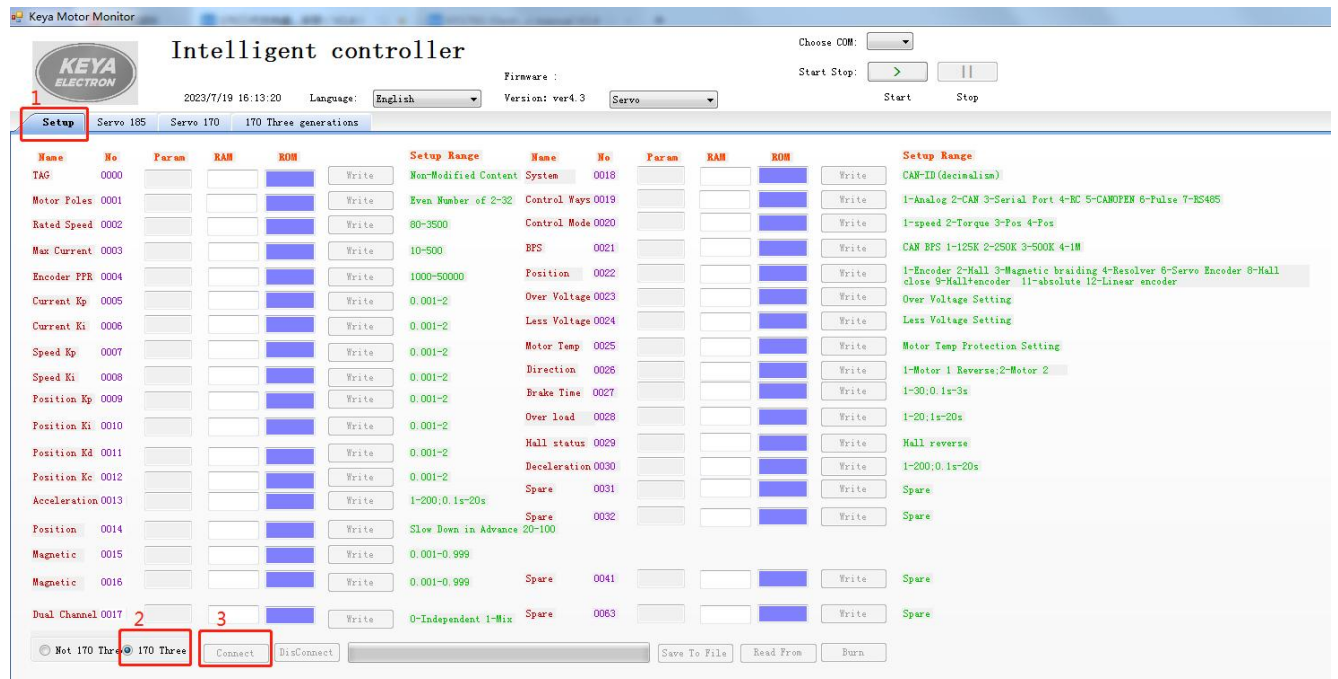
Languages	2022/9/16 9:38	文件夹	
datalog	2022/1/17 14:02	文本文档	53 KB
DevComponents.DotNetBar2.dll	2015/10/6 22:27	应用程序扩展	5,344 KB
info	2023/7/19 9:37	配置设置	1 KB
<b>kyMotor Control UtilityV4.3</b>	2022/9/17 10:13	应用程序	500 KB
kyMotor Control UtilityV4.3.exe.Config	2023/7/19 9:37	CONFIG 文件	1 KB
kyMotor Control UtilityV4.3.pdb	2021/9/7 17:09	PDB 文件	378 KB
kyMotor Control UtilityV4.3.vshost	2021/9/7 17:09	应用程序	12 KB
kyMotor Control UtilityV4.3.vshost.exe.Config	2021/9/6 10:19	CONFIG 文件	1 KB
kyMotor Control UtilityV4.3.vshost.exe.manifest	2012/6/6 2:06	MANIFEST 文件	1 KB
修改履历	2022/9/17 10:55	文本文档	1 KB

4.2.2 Click “170 three generation” icon, then click “Start” button. If the communication is successful, the interface will read the control parameters, while the LED on the upper left will turn green.

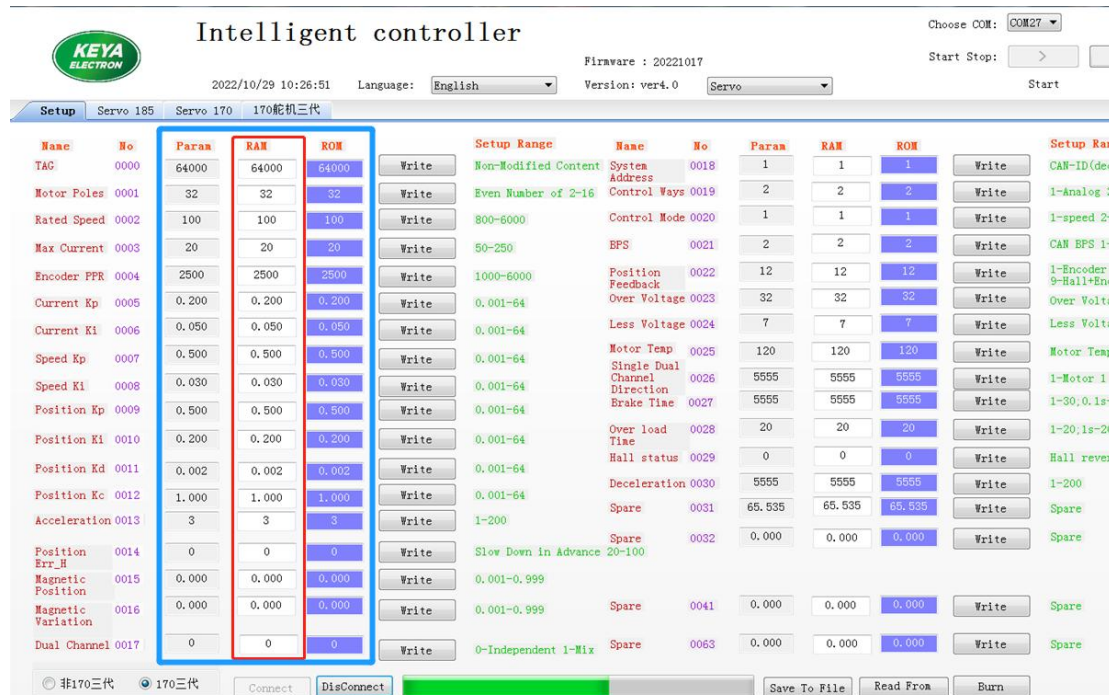




4.2.3 Go to “Setup” screen , select “170 three generation” and click the “Connect” button at the bottom to set up a connection between the software and the motor.



4.2.4 You can input data in the RAM column. The “Param” column means the motor parameters and ROM column shows the data reading from the E<sup>2</sup>ROM. In normal cases, the data in the three columns should be same. Due to the fast changes of software data, if data modification is needed, modify it quickly and click the “Write” button correspondingly.



---

4.2.5 For example, if you want to modify the number of encoder lines, the E<sup>2</sup> ROM data is 2500 and need to change to 1024. Change the parameter in 0004 (Encoder PPR) to 1024. At the same time, quickly click "Write" button beside it and the new data is written in RAM. Same steps for other parameters, multiple parameters can be modified at the same time.

4.2.6 Click the "Burn" button at the bottom to program the data in RAM to E<sup>2</sup>ROM.

Note: The "Burn" process takes 3-5 seconds.

4.2.7 After the programming is completed, it will prompt "programming complete". Check if the three data you modify are consistent. If they are, the modification is successful.

4.2.8 Until now, the modification of the control parameters is completed. Click the "Disconnect" button and then click "Exit" button.

4.2.9 Switch on the motor again. **(Attention: No matter whether the configuration was modified or not, it must switch it off and reset the motor)**

4.2.10 When programming the configuration for multiple motors, you can use "Save To File" function to save the configuration. And then click "Read From" to download to another motor.



### 4.3 Parameters instruction

**0000** Identifier. when the system is connected, identify the software communication or serial port control. (No need to modify)

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

**0002** Rated speed of motor (set to 80)

**0003** Maximum current of motor

**0004** The number of encoder lines. It's set according to the encoder, default value is 48000.

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

Can be modified appropriately.

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

---

Can be modified appropriately.

**0007** Kp parameter of motor speed loop PI control (typical value 0.5)

Can be modified appropriately.

**0008** Ki parameter of motor speed loop PI control (typical value 0.03)

Can be modified appropriately.

**0009-0012** Position loop PID control parameters

**0013** Acceleration time while rotation speed control. "3" means the acceleration time from 0 rpm to rated speed is 0.3s.

**0015** Zero position compensation of magnetic encoder

**0016** Zero position compensation of rotary encoder

**0018** Motor system address, or node number of control.

This parameter is used in the slave of CAN, CANOpen, and EtherCAT BUS.

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

**0019** Control signal selection

2-CAN control; 3-RS232

**0020** Control mode selection, including speed control and position control

1-Speed control,

3-Absolute position control

4-Relative position control

**0021** CAN bus baud rate selection (Factory setting is 250K)

1-125k

2-250k

3-500k

**0022** Position sensor selection

12- Linear encoder

0028 1- 200 corresponding to 0.1s-20s

0033 Zero point of motor, modification is not allowed.

0034 CAN heartbeat upload time (default 20ms)

0-2000 corresponding to 0-2000ms

---

Other parameters: Reserved

#### 4.4 Indicator Description

4.4.1 Status indicator (RED light): Observe the status of the motor according to the blinking frequency of the indicator.

Number of flashes	Definition	Cause of issue
The light is always on	Disabled status	Motor failure
1	Normal situation	Enabled status, work properly
2	Overvoltage	Supply voltage is higher than upper limit of voltage preset in the software
3	Hardware overcurrent protection 22A	Overcurrent protection caused by motor short circuit and field tube damage
4	EEPROM error	Data saving error
5	Undervoltage	Supply voltage is lower than the lower limit of voltage preset in the software
6	Brake	The brake signal is connected
7	Software overcurrent protection (Set protection value through software)	The phase current reaches the software setting protection value for 1 second and stop output.
8	Control mode failure	Wrong control mode selection
9	Working mode failure	Speed, torque working mode not selected or wrong selection
10	Speed loss protection	Actual speed exceeds 25% of rated value
11	Temperature alarm	The temperature is above 85 °C
12	Hall error	Motor Hall is off or failure
13	Reserved	Not activated
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

#### 4.4.2 Enable indicator (BLUE)

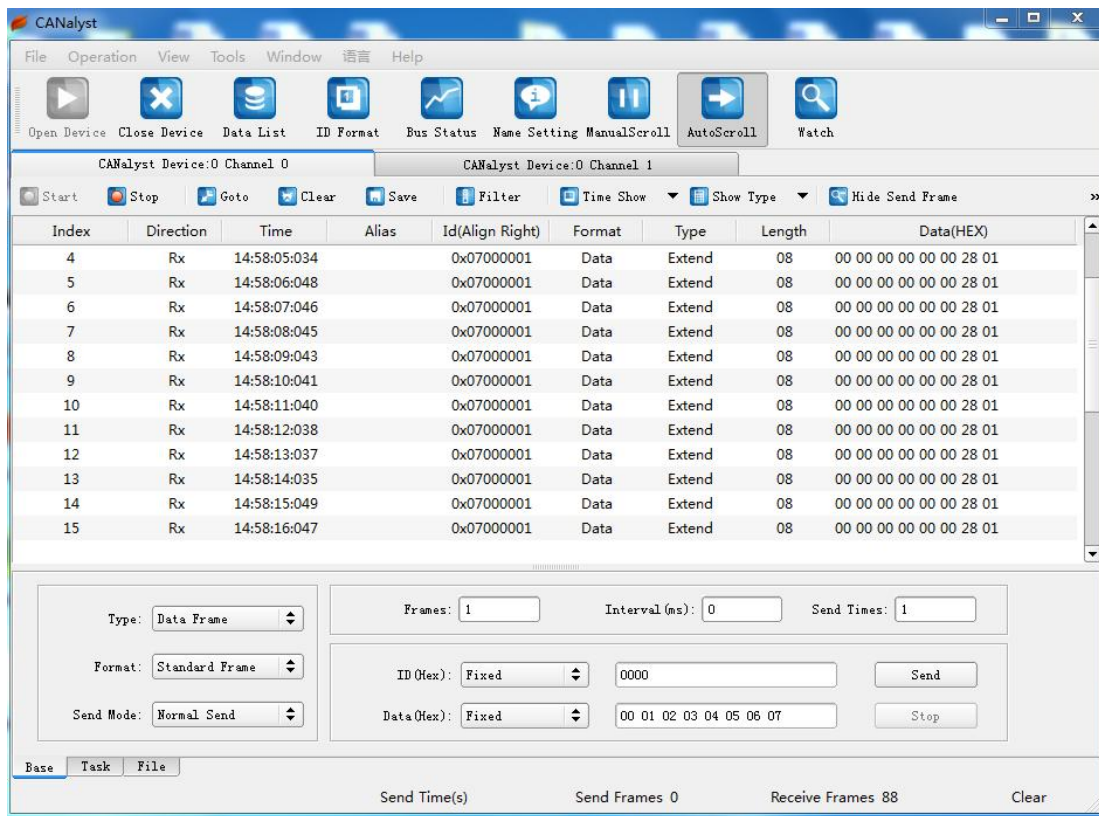
In any control mode, the blue indicator light will be on when the motor is enabled.

The indicator light is off when the motor is disabled.

### 4.5 CAN Command Instruction

#### 4.5.1 General Configuration

- CAN bus protocol baud rate 250Kb
- CAN bus ID with extended ID
- Sending data format: low digit first, high digit after (hexadecimal)
- According to 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 returning are hexadecimal data, which needs to be converted into decimal data.



#### 4.5.2 CAN bus command

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

---

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

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

ID of returned data: 0x0580000 + motor ID (hexadecimal)

ID of heartbeat data: 0x0700000 + motor ID (hexadecimal)

Speed: -1000~ +1000 corresponds to negative rated speed~ rated speed

Position: - 2147483648 ~ +2147483647 (10000/circle)

Enable = 0x23 0D 20 01 00 00 00 00

Disable = 0x23 0C 20 01 00 00 00 00

Speed = 0x23 00 20 01 00 00 00 00

Position = 0x23 02 20 01 00 00 00 00

Example: Default sending ID: 0x0600000+ motor ID (hexadecimal)

**Enable:** 23 0D 20 01 00 00 00 00

Returned ID: 0x0580000+ motor ID (hexadecimal)

Data: 60 0D 20 00 00 00 00 00

**Disable:** 23 0C 20 01 00 00 00 00

Returned ID: 0x0580000 + motor 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)

Speed control: 23 00 20 01 03 E8 00 00 Speed 100RPM (0x03E8=1000)

Returned ID: 0x0580000 + motor ID (hexadecimal)

Data: 60 00 20 00 00 00 00 00

**Position control:** 23 02 20 01 DATA\_L(H) DATA\_L(L) DATA\_H(H) DATA\_H(L)

Position control: 23 02 20 01 27 10 00 00 Rotate 360°counterclockwise

Returned ID: 0x0580000+ motor ID (hexadecimal)

---

Data: 60 02 20 00 00 00 00 00

### Control Status Query

Default sending ID: 0x0600000+motor ID (hexadecimal)

Sending: 40 01 21 01 00 00 00 00

Returned ID: 0x0580000+ motor ID (hexadecimal)

Data: 60 01 21 01+ 【control mode】 4bit+ 【position sensor】 4bit+ 【control mode】 4bit

Example:

Send: 40 01 21 01 00 00 00 00

Feedback: 60 01 21 01 10 C2 ff ff Speed mode+ linear encoder + CAN

Feedback: 60 01 21 01 30 C2 ff ff Position mode + linear + CAN

Feedback data	1	2	3	4	12(C)
Control mode	Speed control		Absolute position	Relative position	
Position sensor					
Control mode		CAN	RS232		Linear encoder

### Motor current query:

Send: 40 00 21 01 00 00 00 00

Returned ID: 0x0580000 + motor ID (hexadecimal)

Data: 60 00 21 01 DATA 00 00 00

DATA = ((unsigned char\*)&send\_float))

### Fault query:

Send: 40 12 21 01 00 00 00 00

Returned ID: 0x0580000 + motor 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. (Failure analysis is same as heartbeat and serial port.)

---

**Encoder speed query:**

Send: 40 03 21 01 00 00 00 00

Returned ID: 0x0580000 + motor 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:**

Send: 40 0D 21 02 00 00 00 00

Returned ID: 0x0580000 + motor ID (hexadecimal)

Data 60 0D 21 02 DATA 00 00 00

DATA = ((unsigned char\*)&send\_float)

**Motor temperature query:**

Send: 40 0F 21 01 00 00 00 00

Returned ID: 0x0580000 + motor ID (hexadecimal)

Data: 60 0F 21 01 DATA 00 00 00

DATA = ((unsigned char\*)&send\_short)

**Encoder count value query: (10000/circle)**

Send: 40 04 21 01 00 00 00 00

Returned ID: 0x0580000 + motor 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];

**External sensor AD query:**

Send: 40 05 21 01 00 00 00 00



---

Returned ID: 0x0580000 + motor ID (hexadecimal)

Data: 60 05 21 01 DAT1 DAT2 00 00

DAT1 = ((unsigned char\*)&send\_float)[L]

DAT2 = ((unsigned char\*)&send\_float)[H]

Note: Input 0-5V, corresponding to 0~5000

#### Analog input query (0-5V):

Send: 40 0A 21 01 00 00 00 00

Returned ID: 0x05800000+ motor ID (hexadecimal)

Data: 60 0A 21 01 F3 01 00 00 (actual range 0x02~0x01F3)

0xF3 = ((unsigned char\*)&send\_float)[L]

0x01 = ((unsigned char\*)&send\_float)[H]

Note: Input 0-5V, corresponding to 002-499

#### Program version query:

Send: 40 01 11 11 00 00 00 00

Returned ID: 0x0580000 + motor 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: 0x07000000 + motor ID (hexadecimal)

Return command: Data0 Data1 Data2 Data3 Data4 Data5 Data6 Data7

Data0 Data1, Cumulative value of angle (360°/circle)

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

Data4 Data5, motor current: with symbol

Data6 Data7, Control\_Close (error code)

---

(It means the high digit goes first and low digit goes after for heartbeat data.)

**Example:**

Upload ID: 0x07000001

Returned command: 01 68 00 14 00 00 40 01

Cumulative value of angle: 0x0168 360° /circle, when angle value reaches 0Xffff=65535, automatically clear and recount.

Error code: 0x4001 (Binary 100000000000001)

Reporting faults: CAN disconnected or disabled

Failure analysis is same as serial port error.

### 4.5.3 CAN bus control example

#### 4.5.3.1 Speed Control:

(Speed command value %) \* (The maximum speed set in the software) = actual speed.

#### 4.5.3.2 If rated speed is 100rpm,

then the given value of speed command -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 system address in the software is 1 (0018 is set to 1)

- If the given speed +50 (rated speed 100)

Control command ID: 0x06000001 (extended ID)

Enable: 23 0D 20 01 00 00 00 00

Given speed: 23 00 20 01 01 F4 00 00 (0x01F4 = 500)

- If the given speed -50 (rated speed 100)

Control command ID: 0x06000001 (extended ID)

Enable: 23 0D 20 01 00 00 00 00

Given speed: 23 00 20 01 FE 0C FF FF

---

#### 4.5.3.3 Position control:

The given position value - 50000~ + 50000 means 5 circles clockwise ~ 5 circles counterclockwise

(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 system address set in the software is 1 (0018 is set to 1)

Control command ID: 0x06000001 (extended ID)

The order to sending data:

- (a) Disable: 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, make the motor rotate 1.8 circles clockwise

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

For example: Make the motor rotate 72 degrees at mechanical angle counterclockwise ( $72 * (10000 / 360) = 2000 = 0x07D0$ )

- (a) Make sure the position control has been switched on.
- (b) Enable: 23 0D 20 01 00 00 00 00
- (c) Position control command: 23 02 20 01 07 D0 00 00

#### 4.5.3.4 CAN command switch control mode

0x230C2009 = Position reset;

0x030D2011 = Speed mode;

0x030D2031 = Absolute position mode;

0x030D2041 = Relative position mode and position reset.

Example: Default sending ID: 0x06000000+ motor ID (hexadecimal)

Speed mode:

Send: 03 0D 20 11 00 00 00 00 (Switch to speed mode)

Feedback: 7F BA 0D 0B 6C FF 13 01

Send: 23 0d 20 01 00 00 00 00    Enable

Send: 23 00 20 01 03 E8 00 00    Speed command: 0x03E8    (Forward 100RPM)

Send: 23 00 20 01 FC 18 FF FF    Speed command: 0xFFFFC18    (Reverse 100RPM)

Position mode:

Send: 03 0D 20 31 00 00 00 00    (Switch to absolute position mode)

Feedback: 7F BA 0D 0B 6C FF 13 01

Send: 23 0d 20 01 00 00 00 00    Enable

Send: 23 02 20 01 86 A0 00 01    0x186A0    Forward 10 circles (low digit first, high digit after)

Send: 23 02 20 01 27 40 FF 58    0xFF582740    Reverse 1100 circles (low digit first, high digit after)

## 4.6 Serial port command instruction

### 4.6.1 General Configuration

#### Serial port configuration

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

115200bits/s

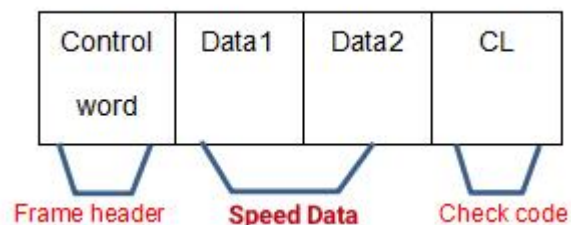
8-digit data

1 starting digit

Sum check

Hexadecimal receiving and sending

### 4.6.2 Control command format



#### Frame Header

AD: Enable

AC: Disable

---

### Speed Data:

[ Data ] is the given speed/position command data, high digit goes first and low digit goes after.

The speed data range is -1000 ~ +1000, means - 100rpm ~ 100rpm (factory setting)

### Sum check:

The check uses the sum check method, that is, starting from the frame header, accumulating all the bytes, and the low digit of the final result obtained is the check code.

### Speed control mode

Example: motor enable, speed 0rpm

Send: AD 00 00 AD

Feedback data: AC xx xx xx

Example: motor disable, speed 0rpm

Send: AC 00 00 AC

Feedback data: AC xx xx xx

Example: motor enable, speed 100rpm (Rated speed 100)

Send: AD 03 E8 98

Feedback data: AC xx xx xx

Example: motor disable, speed 100rpm (Rated speed 100)

Send: AC 03 E8 97

Feedback data: AC xx xx xx

Example: motor enable, the given speed -100rpm (Rated speed 100)

Send: AD FC 18 C1

Feedback data: AC xx xx xx

Example: motor enable, the given speed -100rpm (Rated speed 100)

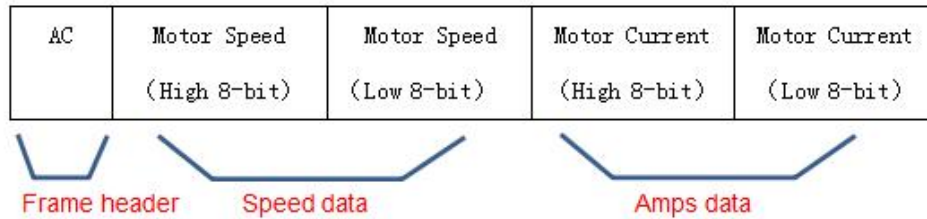
Send: AD FC 18 54 (Check code error)

Verification feedback data: A8 xx xx xx

- **Note: Time interval between two commands when sending commands continuously is 20ms < Time interval < 500ms**

### 4.6.3 Return information format

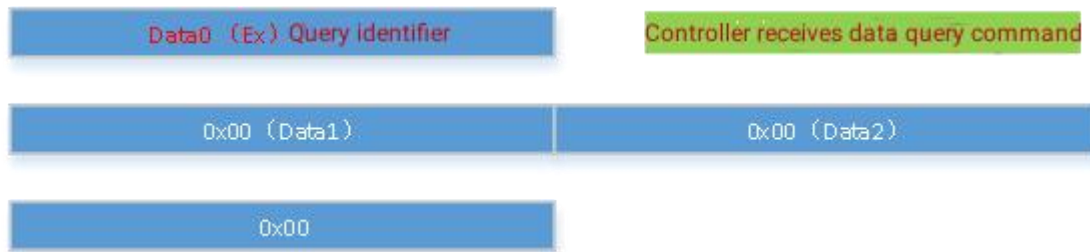
Every time the motor controller receives a control command, the controller will return a response data defined as follows: (hexadecimal)



AC: Check passed.

A8: Check failed

### 4.6.4 Query data format



Query Identifier (Data0): Ex means query data command.

Query data identifier description:

- 0xE0 System control status
- 0xE1 A channel motor rotor position electric angle (0-1000)
- 0xE2 A channel motor speed (rpm)
- 0xE3 A channel motor winding Amps (positive number)
- 0xE4 A channel motor rotor mechanical angle (0-10000)
- 0xE5 System controller voltage
- 0xE6 Motor/controller temperature
- 0xE7 A channel motor error code
- 0xE8 A channel motor rotor mechanical position (position mode)
- 0xEF System software version

Examples:

- Read motor speed (rpm)

The software send: E2 02 00 00 00

Controller feedback: E2 00 64 00 00

---

That is, the motor current speed is 100rpm

- Read controller current (A)

The software send: E3 00 00 00

Controller feedback: E3 0A 00 00 00

That is, the phase current of controller is 10A

- Read controller voltage (V)

The software send: E5 00 00 00

Controller feedback: E5 00 0B 00 00

That is, the supply voltage of controller is 12V

- Read controller temperature (°C)

The software send: E6 00 00 00

Controller feedback: E6 1A 00 00 00

That is, the current temperature of controller is 26°C

- Read motor mechanical position (10000/Circle)

The software send: E8 00 00 00

Controller feedback: E8 00 09 2D EE

That is, the current position of controller is 0x092DEE→601528

- Read analog input collection:

The software send: E9 00 00 00

Controller feedback: E9 01 F3 00 00 (actual range 0x02~0x01F3)

That is, the input voltage of current controller is 5V

Note: 0-5V corresponds to 002~499.

**Caution: The data returned by the query is all in hexadecimal, which needs to be converted to decimal for reading.**

- Read controller error code

---

The software send: E7 00 00 00

Controller feedback: E7 Data0 Data1 00 00

Example: Data0: 08

Data1: 01

That means the current controller error is hall failure and disabled.

### Error codes analysis:

#### Data0:

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Motor stalled	CAN disconnected	232 disconnected	Current sensing	Hall failure	Reserved	Motor stall	Less phase

#### Data1:

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Mode failure	Overcurrent	N/A	Undervoltage	E2PROM	Hardware protection	Overvoltage	Disabled

## V. Failure Protection and Reset

### 5.1 Fault protection basis

#### 5.1.1 Temperature alarm

When the temperature of motor exceeds 85 °C, a temperature alarm is activated. When it is restored to 75 °C, the alarm is cleared automatically.

#### 5.1.2 Overcurrent protection

When the phase current reaches the setting value with setting time, it stops. Reset the motor by enabling it again.

#### 5.1.3 Overvoltage and undervoltage protection

Undervoltage protection occurs when the power supply voltage is lower than the lower limit of voltage set in the software.

Overvoltage protection occurs when the power supply voltage is higher than the upper limit of voltage set in the software.



---

## 5.2 Failures information table

Protection category	Security level	Turn off 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 controller stops the power output. You can re-enable the motor to clear all failures information.