User manual

Precautions

1. Please use according to the working parameters specified in this article, otherwise it may cause serious damage to this product!

2. During joint operation, the control mode cannot be switched. To switch, a stop running command needs to be sent before switching.

3. Before use, please check if all components are intact. If any parts are missing or damaged, please contact technical support in a timely manner.

4. Please do not disassemble the motor randomly to avoid irreparable faults.

5. Ensure that there is no short circuit when connecting the motor and that the interface is correctly connected as required.

Legal Statement

Before using this product, users must carefully read this manual and operate the product according to its contents. If the user violates the content of this manual and uses this product, resulting in any property damage or personal injury accidents, our company shall not be held responsible. Due to the numerous components of this product, do not let children come into contact with it to avoid accidents. To extend the service life of the product, please do not use it in high temperature and high pressure environments. This manual has included various functional introductions and usage instructions as much as possible during printing. However, due to the continuous improvement of product functions and design changes, there may still be discrepancies with the products purchased by users.

After sales policy

The after-sales service of this product is strictly in accordance with the Consumer Rights Protection Law of the People's Republic of China and the Product Quality Law of the People's Republic of China. The service content is as follows:

1. Warranty period and content

(1) Users who place orders for this product through online channels can enjoy a no reason return service within seven days from the day after signing for it. When returning goods, the user must present a valid purchase voucher and return the invoice. Users must ensure that the returned goods maintain their original quality and function, have an intact appearance, and that the trademarks and various markings of the goods and accessories are complete. If there are any gifts, they must be returned together. If the product is artificially damaged, dismantled, the packaging box is missing, or the spare parts are missing, no return will be processed. The logistics costs incurred during the return process are borne by the user (refer to the

"After sales Service Fee Standards" for charging standards). If the user has not settled the logistics fees, they will be deducted from the refund amount based on the actual amount incurred. Within seven days from the date of receiving the returned goods, the paid payment shall be returned to the user. The refund method is the same as the payment method. The specific arrival date may be influenced by factors such as banks and payment institutions.

(2) The warranty period for this product is 1 year.

(3) Within 7 days from the day after the user signs for the item, if there is any non-human damage or performance malfunction, the customer must be confirmed by the FULLING MOTOR after-sales service center and return the item. When returning the item, the customer must present a valid purchase voucher and return the invoice. If there are any gifts, they should be returned together.

(4) Within 7 to 15 days from the day after the user signs for it, if there is any non-human damage or performance failure, the FULLING MOTOR after-sales service center will conduct a replacement service for the user and replace the entire set of goods after testing and confirmation. After the exchange, the warranty period of the product itself is recalculated.

(5) Within 15 to 365 days from the day after the user signs for it, if it is confirmed by the FULLING MOTOR after-sales service center that the product itself has a quality fault, free repair services can be provided. The faulty products replaced belong to FULLING MOTOR Company. No faulty products, return as is. This product has undergone strict testing before leaving the factory. If there is any quality failure other than the product itself, we have the right to refuse the user's return or exchange request.

2. The following situations are not covered by warranty regulations:

1. Exceeding the warranty period specified in the warranty terms.

2. Product damage caused by incorrect use without following the instructions.

3. Damage caused by improper operation, maintenance, installation, modification, testing, and other improper use.

4. Conventional mechanical losses and wear caused by non quality faults.

5. Damage caused by abnormal working conditions, including but not limited to falls, impacts, liquid immersion, severe impacts, etc.

6. Damage caused by natural disasters (such as floods, fires, lightning strikes, earthquakes, etc.) or uncontrollable forces.

7. Damage caused by exceeding peak torque usage.

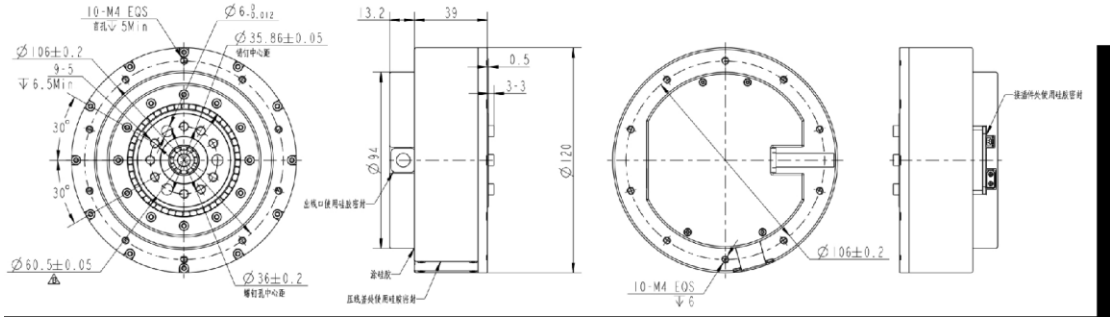
8. Non genuine spiritual feet or inability to provide legal purchase vouchers.

9. Other malfunctions or damages caused by non product design, technology, manufacturing, quality, and other issues.

If the above situation occurs, users need to pay the fees themselves.

1. Motor specification parameters

1. **Appearance and installation dimensions**



When fixing, the depth of the screw should not exceed the thread depth of the enclosure.

1.2 Standard Usage Status

1.2.1 Rated voltage: 48 VDC

1.2.2 Voltage range: 24V -60 VDC

1.2.3 Rated load (CW): 40 N.m

1.2.4 Operating direction: CW/CCW viewed from the output shaft direction

1.2.5 Usage posture: The axis direction is horizontal or vertical

1.2.6 Standard operating temperature: 25 ± 5 □

1.2.7 Temperature range for use: -20~50 □

1.2.8 Standard operating humidity: 65%

1.2.9 Humidity range: 5-85%, no condensation

1.2.10 Storage temperature range: -30~70 □

1.2.11 Insulation level: Class B

1.3 Electrical characteristics

1.3.1 No load speed: 200 rpm ± 10%

1.3.2 No load current: 2Arms

1.3.3 Rated load: 40 N.m

1.3.4 Rated load speed: 167rpm ± 10%

1.3.5 Rated load phase current (peak): 27Apk ± 10%

1.3.6 Peak load: 120 N.m

1.3.7 Maximum load phase current (peak): 90apk ± 10%

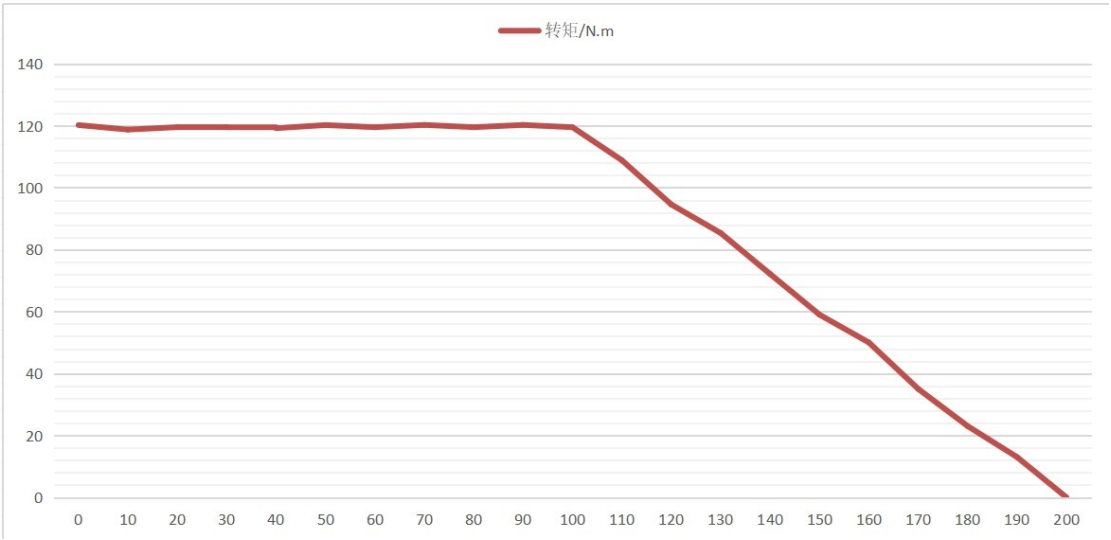
1.3.8 Insulation resistance/stator winding: DC 500VAC, 100M Ohms

1.3.9 High voltage resistance/stator and cover: 600 VAC, 1s, 2mA

1.3.10 Motor back electromotive force: 16.9Vrms/krpm±10%

1.3.11 Torque constant: 2.1N.m/Arms

1.3.12 T-N curve



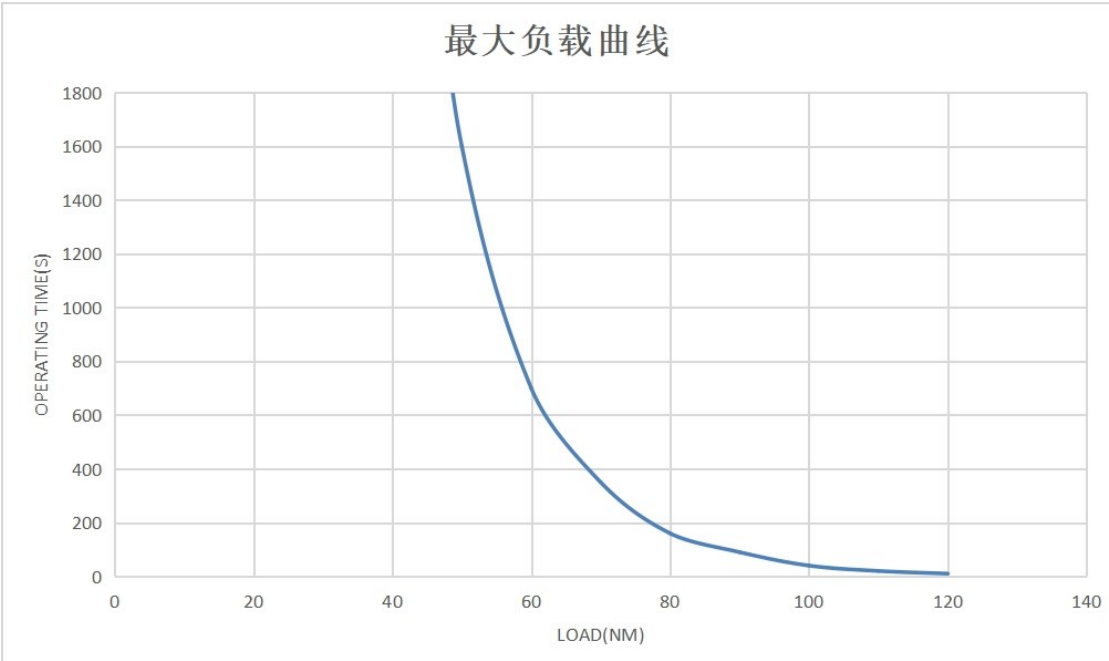
1.3.13 Maximum overload curve

Test conditions:

Environmental temperature: 25 □

Maximum temperature for winding resistance: 130 □ (this is the constraint temperature, actually 180 degrees)

Speed: 24rpm



test data

Load	Operating time(s)
120	10
110	20
100	40
90	90
80	160
70	350
60	700
50	1600
46	2500
40	rated

1.4 Mechanical characteristics

1.4.1 Weight: 1420g ± 20g

1.4.2 Number of Poles: 42 Poles

1.4.3 Number of phases: 3 phases

1.4.4 Drive mode: FOC

1.4.5 Reduction ratio: 9:1

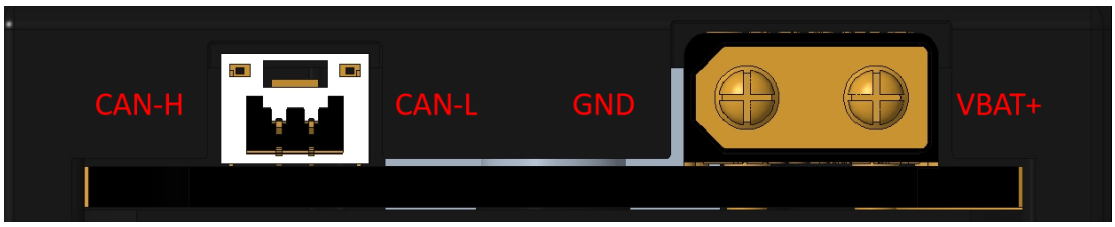
2 Drive Product Information

2.1 Driver Product Specifications

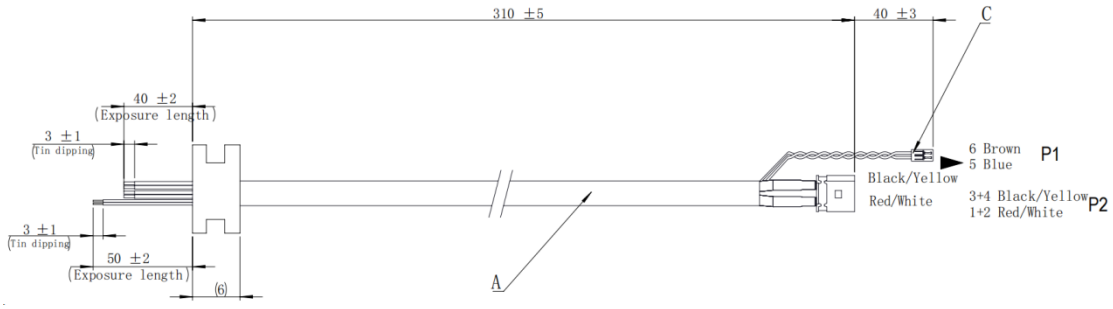
Product specifications	
Rated working voltage	48VDC
Maximum allowable voltage	60VDC
Rated working phase current	27Apk
Maximum allowable phase current	90Apk
Standby power consumption	≤ 40mA
CAN bus bit rate	1Mbps
Size	Φ 84mm
Working environment temperature	-20 □ to 50 □
The maximum temperature allowed by the control board	80 □
Encoder resolution	14 bit (absolute value of a single lap)

2.2 Driver Interface Definition

2.2.1 Driver Interface Definition



2.2.2 Definition of drive harness sequence



Definition of drive harness sequence	
Blue	CAN_H
Brown	CAN_L
Black	GND
Red	VBAT+

2.2.3 Recommended brand and model of driver interface

Board end model	Brand manufacturer	Line end model	Brand manufacturer
XT30APW-M	AMASS	XT30UW-F	AMASS
GH1.25-2PWT	Any	GH1.25-T	Any

2.3 Main components and specifications

Number	project	Specifications	quantity
1	MCU chip	GD32F103RET6	1 PCS
2	Driver chip	DRV8353SRTAT	1 PCS
3	Magnetic encoder chip	AS5047P	1 PCS
4	Thermistor	LTS00-104J395T19E010/ NCP18XH103F03RB	2 PCS
5	Power MOS	ISC030N12NM6	12 PCS

Instructions for using the Studio

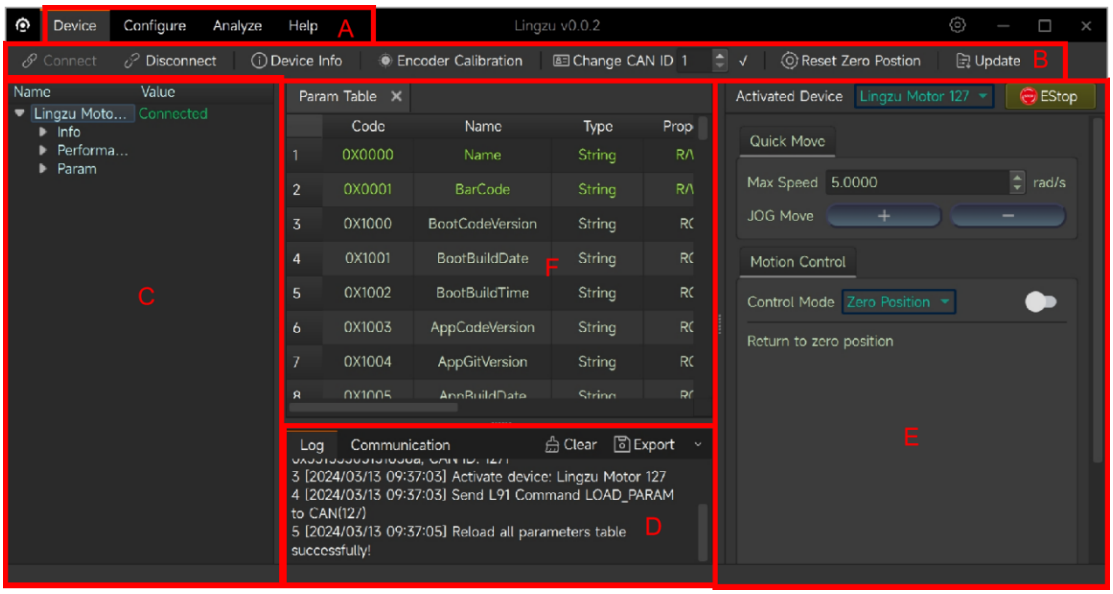
3.1 Hardware Configuration

The joint motor adopts CAN communication method, with two communication lines connected to the debugger through CAN to USB tool. The debugger needs to install ch340 driver in advance and works in AT mode by default.

It should be noted that we developed the debugger based on a specific CAN to USB tool, so we need to use the recommended serial port tool for debugging. If you want to port to other debugger platforms, you can refer to Chapter 3 of the manual for development.

It is recommended to use USB-CAN module for the CAN to USB tool, with a frame header of 41 54 and a frame tail of 0D 0A corresponding to the serial protocol.

3.2 Studio interface and instructions



Mainly including:

A. Module selection

- Equipment module
- Configuration module
- Analysis module
- Help module

B. Sub module selection

Equipment modules include

- Connecting or disconnecting motor equipment

- Motor equipment information
- Motor encoder calibration
- Modify the motor CAN ID
- Set the mechanical zero position of the motor
- Motor program upgrade

The configuration module includes:

- Parameter table, can view and modify motor parameters
- Upload parameters, you can upload the parameters in the motor to the parameter table
- Download parameters, you can download the data from the parameter table to the motor
- Export parameters and download the data from the parameter table to the local location
- Restore to factory settings by resetting the data in the parameter table to factory settings
- Clear warning to clear motor errors, such as excessive temperature

The analysis module includes:

- Oscilloscope, which can view the curve of parameter changes over time
- Frequency, can adjust the frequency of viewing data
- Channel, can be configured to view data
- Start and stop drawing
- Output waveform data to local

The help module includes:

- Instructions for use, which can be opened
- Regarding, you can view software information

C. Motor information inquiry

- Device information
- Parameter Table Information

D. Data bar

- log information
- Communication information

E. Run debugging area

- Select device
- Convenient operation area, which can quickly control the forward and reverse

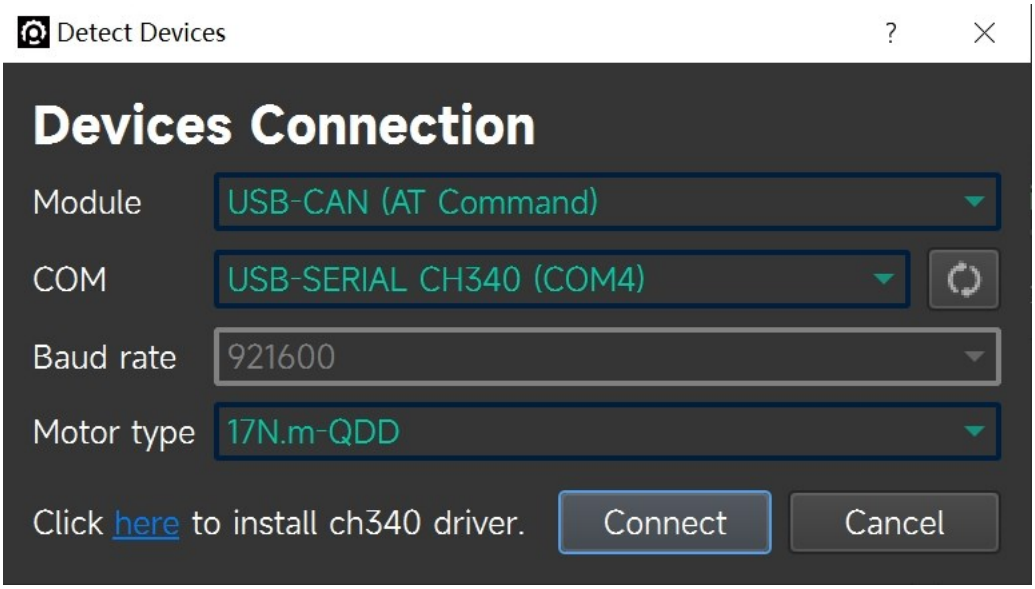
rotation of the motor

- Motion control area, which can control the motor to operate in various modes

F. Sub module display area

3.3 Motor settings

3.3.1 Motor connection settings



Connect the CAN to USB tool (install ch340 driver, default working in AT mode), click on the connection submodule in the device module, select the corresponding serial port connection and motor type, and click connect.

3.3.2 Basic Settings

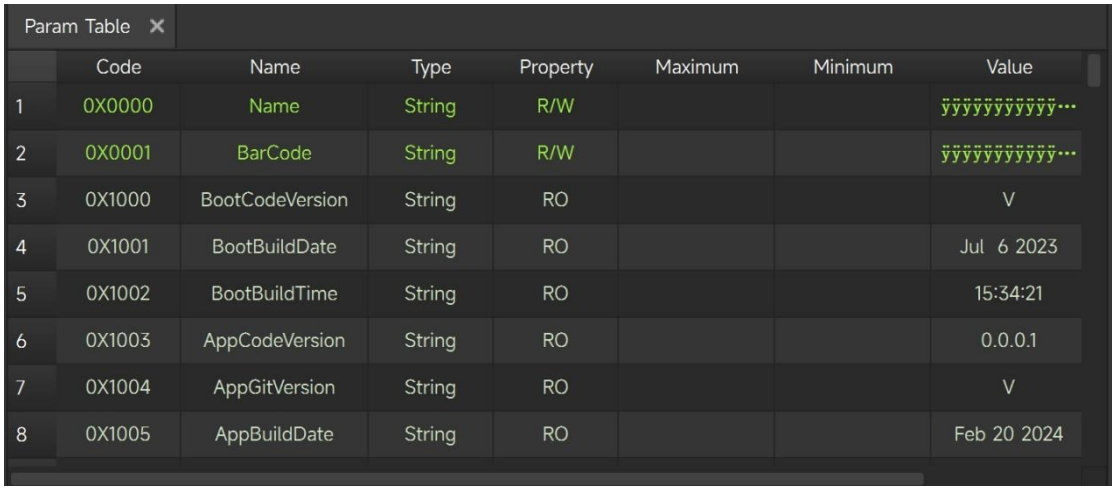
(1) Modify the motor ID number.

(2) Motor magnetic encoder calibration, reinstallation of the motor board and motor, or replacement of the motor's three-phase line sequence connection, etc., requires re magnetic encoder calibration.

(3) Set the zero position (power loss) and set the current position to 0.

(4) Motor program upgrade. When there is an update to the motor program, click the upgrade button and select the upgrade file to proceed with the upgrade.

3.3.3 Parameter Table



After successfully connecting the motor, click on the parameter table module in the configuration module. The log will display all parameters loaded successfully, indicating that the motor related parameters have been successfully read (note: the parameter table needs to be configured when the motor is in standby mode, and cannot be refreshed if the motor is in running mode). The interface will display the motor related parameters, and the blue parameters are the stored parameters inside the motor, You can make modifications in the current value column after the corresponding parameters. Clicking on the download parameter will download the parameters from the debugger to the motor. Clicking on the upload parameter will upload the parameters from the motor to the debugger. The green parameter of the motor is the observation parameter, and the collected parameter can be observed in real time.

Note: The torque limit, protection temperature, and over temperature time of the motor should not be changed arbitrarily. Our company will not bear any legal responsibility for any harm caused to the human body or irreversible damage to joints due to improper operation of this product.

Parameter table							
Function code	Name	Parameter type	Properties	Maximum value	Minimum value	Current value (for reference)	Remarks
0X0000	Name	String	Read/Write			ÿ ÿ ÿ	
0X0001	BarCode	String	Read/Write			ÿ ÿ ÿ	
0X1000	BootCo	String	Read			0.1.5	

	deVersion		Only				
0X1001	BootBuildDate	String	Read Only			Mar 16 2022	
0X1002	BootBuildTime	String	Read Only			20:22:09	
0X1003	AppCodeVersion	String	Read Only			0.0.0.1	Motor program version number
0X1004	AppGitVersion	String	Read Only			7b844b0fM	
0X1005	AppBuildDate	String	Read Only			Apr 14 2022	
0X1006	AppBuildTime	String	Read Only			20:30:22	
0X1007	AppCodeName	String	Read Only			Lingzu_motor	
0X2000	EchoPara1	Uint16	Configuration	74	5	5	
0X2001	EchoPara2		Configuration	74	5	5	
0X2002	EchoPara3	Uint16	Configuration	74	5	5	
0X2003	EchoPara4	Uint16	Configuration	74	5	5	
0X2004	EchoFreHz	Uint32	Read/Write	10000	1	500	
0X2005	MechOf	Float	Read/W	7	-7	4.619583	Low-

	fset		rite				speed end position offset value
0X2006	chasu_offset	Float	Read/Write	6.28	0	4.52	Differential offset value
0X2007	ElecOffset	Float	Read/Write	6.28	0	3.27	Electrical Angle Offset Value
0X2008	I_FW_MAX	float	Read/Write	33	0	0	Weak magnetic current value, default to 0
0X2009	CAN_ID	Uint8	Setting	127	0	1	This node ID
0X200a	CAN_MASTER	Uint8	Setting	127	0	0	Can host ID
0X200b	CAN_TIMEOUT	Uint32	Read/Write	100000	0	0	Can timeout threshold, default to 0
0X200c	MotorOverTemp	Int16	Read/Write	1500	0	800	Motor protection temperature value,

							temp (degrees) * 10
0X200d	OverTempTime	Uint32	Read/Write	1000000	1000	20000	Over temperature time
0X200e	GearRatio	Float	Read/Write	64	1	9	Transmission ratio
0X200f	Kt_Nm/Amp	float	Read/Write	1	1	0	0
0X2010	Tq_CaliType	Uint8	Read/Write	1	0	1	Torque calibration method setting
0X2011	Cur_Fit_Gain	Float	Read/Write	1	0	0.6	Current filtering parameters
0X2012	Cur_Kp	Float	Read/Write	200	0	0.05	Current kp
0X2013	Cur_ Ki	Float	Read/Write	200	0	0.05	Current ki
0X2014	Spd_Kp	Float	Read/Write	200	0	2	Speed kp
0X2015	Spd_ Ki	Float	Read/Write	200	0	0.021	Speed ki
0X2016	Loc_Kp	Float	Read/Write	200	0	30	Location kp
0X2017	Spd_Fit_	Float	Read/Write	1	0	0.1	Speed filtering

	Gain						parameters
0X2018	Limit_Spd	Float	Read/Write	200	0	2	Position mode speed limit
0X2019	Limit_Cur	Float	Read/Write	90	0	90	Position and speed modes Current limitation
0X3000	TimeUse0	Uint16	Read Only			5	
0X3001	TimeUse1	Uint16	Read Only			0	
0X3002	TimeUse2	Uint16	Read Only			10	
0X3003	TimeUse3	Uint16	Read Only			0	
0X3004	EncoderRaw	Int16	Read Only			11396	Magnetic encoder sampling value
0X3005	McuTemps	Int16	Read Only			337	Internal temperature of MCU, *10
0X3006	MotorTemps	Int16	Read Only			333	Motor NTC temper

							ature, * 10
0X3007	VBus (mv)	Uint16	Read Only			24195	Bus voltage
0X3008	Adc1Offset	Int32	Read Only			2084	ADC sampling channel 1 zero current bias
0X3009	Adc2Offset	Int32	Read Only			2084	ADC sampling channel 2 zero current bias
0X300a	Adc1Raw	Uint16	Read Only			1232	ADC sampling value 1
0X300b	Adc2Raw	Uint16	Read Only			1212	ADC sampling value 2
0X300c	VBUS	Float	Read Only			24.195	Bus voltage V
0X300d	CmdId	Float	Read Only			0	ID Ring Instruction, A
0X300e	CmdIq	Float	Read Only			0	Iq ring instruction, A

0X300f	Cmdloc ref	Float	Read Only			0	Position loop comma nd, rad
0X3010	Cmdsp dref	Float	Read Only			0	Speed loop comma nd, rad/s
0X3011	CmdTor que	Float	Read Only			0	Torque comma nd, nm
0X3012	CmdPo s	Float	Read Only			0	Mit protocol angle instructi on
0X3013	CmdVel	Float	Read Only			0	Mit protocol speed instructi on
0X3014	Rotatio n	Int16	Read Only			1	Number of laps
0X3015	ModPo s	Float	Read Only			4.3634 09	Motor uncount ed mechan ical angle, rad
0X3016	MechP os	Float	Read Only			0.7776 79	Mechan ical angle of load end coil, rad

0X3017	MechVel	Float	Read Only			0.036618	Load end speed, rad/s
0X3018	ElecPos	Float	Read Only			4.714761	Electrical perspective
0X3019	Ia	Float	Read Only			0	U line current, A
0X301a	Ib	Float	Read Only			0	V-line current, A
0X301b	IC	Float	Read Only			0	W-line current, A
0X301c	Tick	Uint32	Read Only			31600	
0X301d	PhaseOrder	Uint8	Read Only			0	Calibration direction markings
0X301e	IQF	Float	Read Only			0	
0X301f	BoardTemps	Int16	Read Only			359	On board temperature, * 10
0X3020	Iq	Float	Read Only			0	Iq original value, A

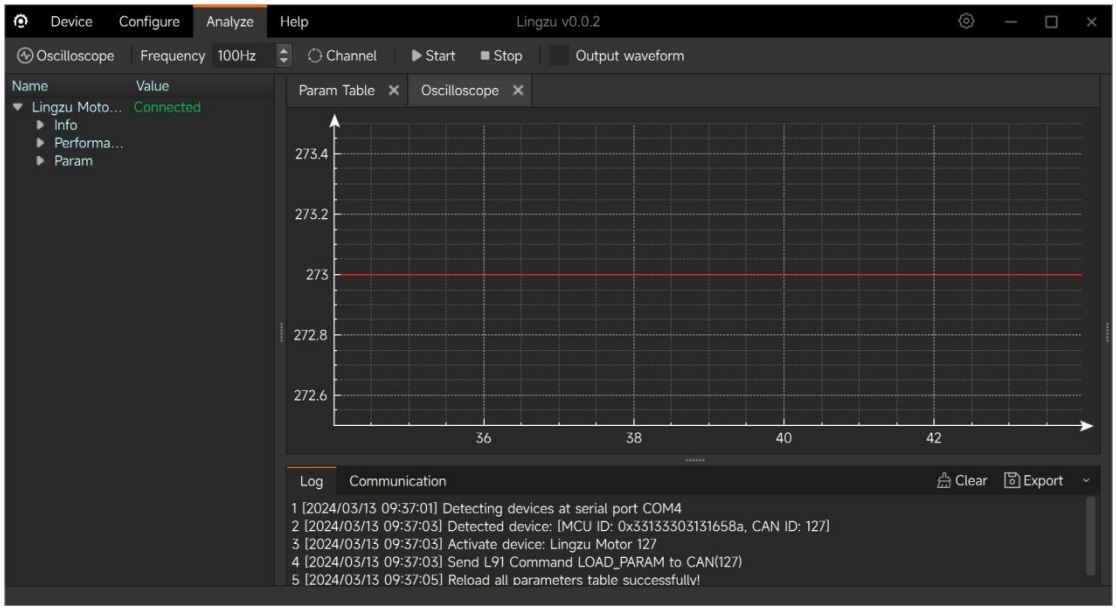
0X3021	ID	Float	Read Only			0	ID original value, A
0X3022	FaultSta	Uint32	Read Only			0	Fault state value
0X3023	WarnSta	Uint32	Read Only			0	Warning status value
0X3024	DRV_ Fault	Uint16	Read Only			0	Driver chip fault value
0X3025	DRV_ Temp	Int16	Read Only			48	Driver chip temperature value, degrees
0X3026	Uq	Float	Read Only			0	Q-axis voltage
0X3027	Ud	Float	Read Only			0	D-axis voltage
0X3028	DTC_ U	Float	Read Only			0	U-phase output duty cycle
0X3029	DTC_ V	Float	Read Only			0	V-phase output duty cycle
0X302a	DTC_	Float	Read			0	W-

	W		Only				phase output duty cycle
0X302b	V_ Bus	Float	Read Only			24.195	Vbus in closed-loop
0X302c	V_ Ref	Float	Read Only			0	Closed loop vq, vd synthesis voltage
0X302d	Torque _ FDB	Float	Read Only			0	Torque feedback value, nm
0X302e	Rated_ I	Float	Read Only			8	Motor rated current
0X302f	Limit_ I	Float	Read Only			27	Motor limit maximum current

3.3.4 Oscilloscope

This interface supports viewing the graph generated by observing real-time data, including motor Id/Iq current, temperature, real-time output speed, rotor (encoder) position, output position, etc.

Click on the oscilloscope module in the analysis module, select the appropriate parameters in the channel (parameter meanings can refer to 3.3.3), set the output frequency, and click start drawing to observe the data graph. Stop drawing to stop observing the graph.



Example of communication box command:

41 54 90 07 e8 0c 08 05 70 00 00 01 00 00 00 0d 0a

The meaning is as follows

41 54	90 07 e8 0c	08	05 70 00 00 01 00 00 00 00	0d 0a
Frame header	Extended frame	Number of data bits	Data frame	End of frame

Translating the extended CAN ID to the real CAN ID requires the following conversion:

90 07 e8 0c is converted to binary as 1001 0000 0000 0111 1110 1000 0000 1100. If the 100 on the right is removed, it becomes 1 0010 0000 1111 1101 0000 0001. Convert it to hexadecimal and it becomes 12 00 FD 01. Refer to the communication protocol description and the meaning is as follows:

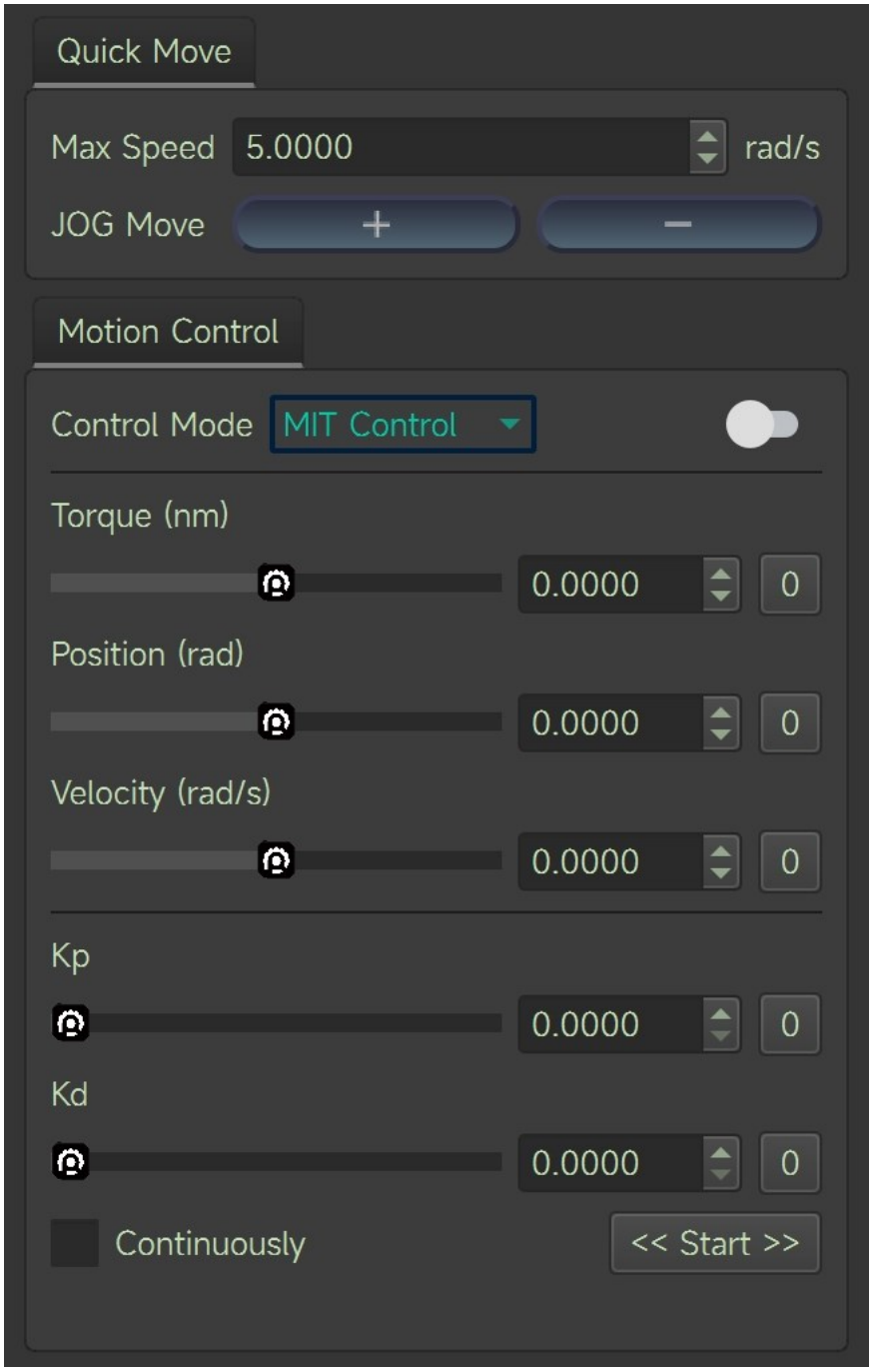
12 (hexadecimal)	00	FD	01
Communication type 18 (base 10)	Meaningless	Host ID	Motor CAN ID

3.3.6 CAN communication fault protection

When CAN_ When the TIMEOUT value is 0, this function is not enabled

When CAN_ When the TIMEOUT value is non-0, when the motor does not receive a can command within a certain period of time, the motor enters reset mode, and 12000 is 1 second

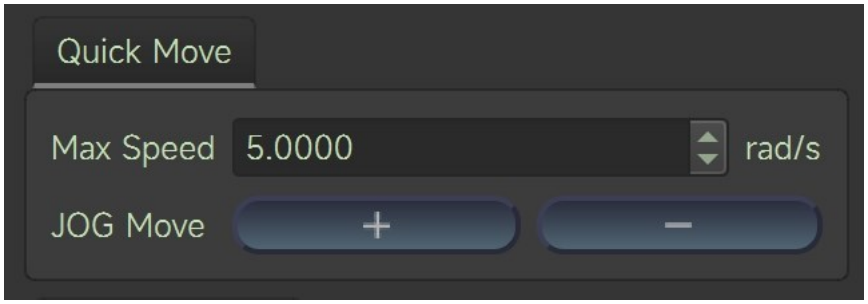
3.4 Control demonstration



Jog Run:

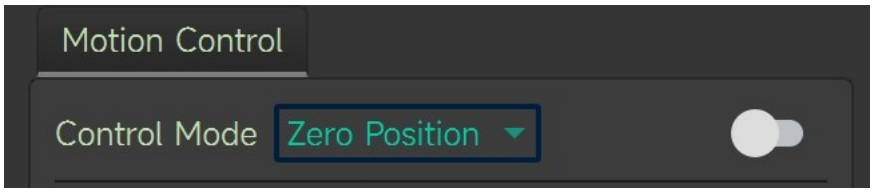
Set the maximum speed, click run, then click JOG to run the motor in both forward

and reverse directions

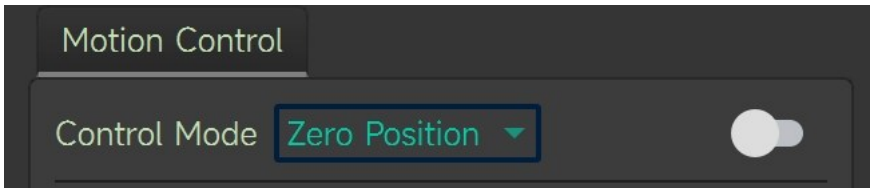


Control mode switching:

The motor control mode can be switched on the sports mode interface

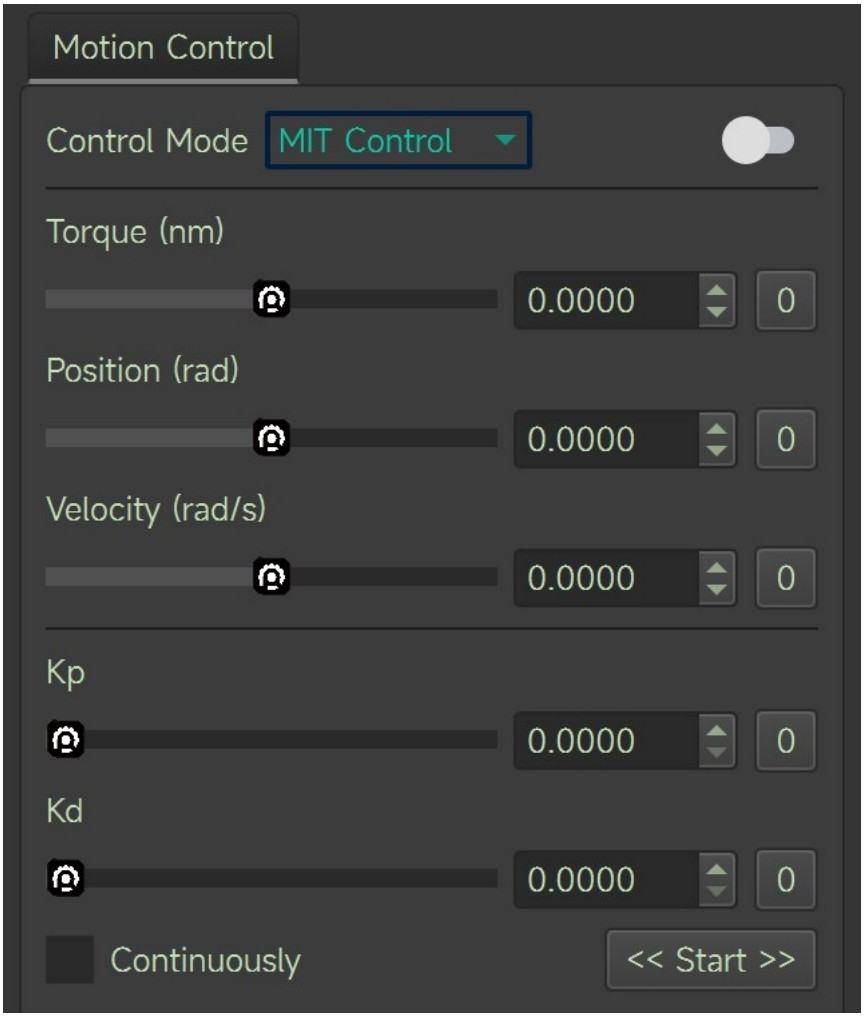


3.4.1 Zero point mode



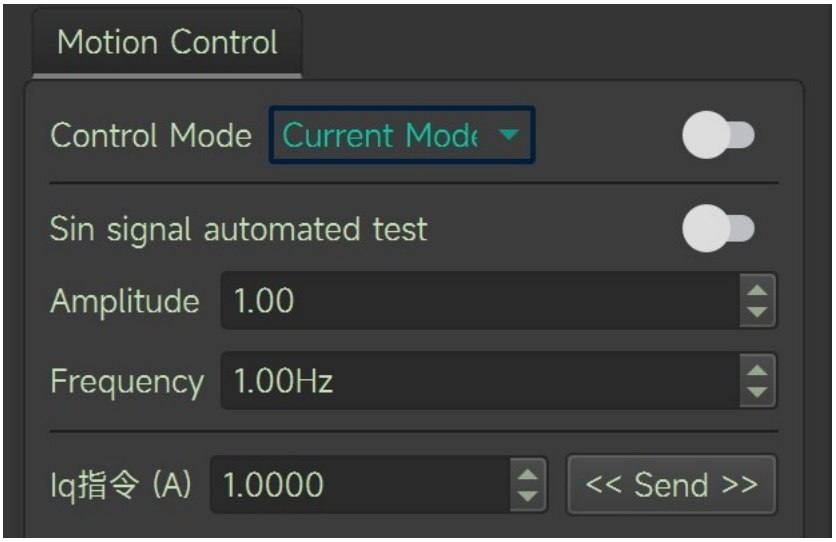
Click the switch button on the right side, and the motor will slowly return to the mechanical zero position

3.4.2 Operation and control mode



Click the switch button on the right side, then set five parameter values, click start or continue sending, the motor will return to the feedback frame and run according to the target command; Click the switch button on the right again, and the motor will stop.

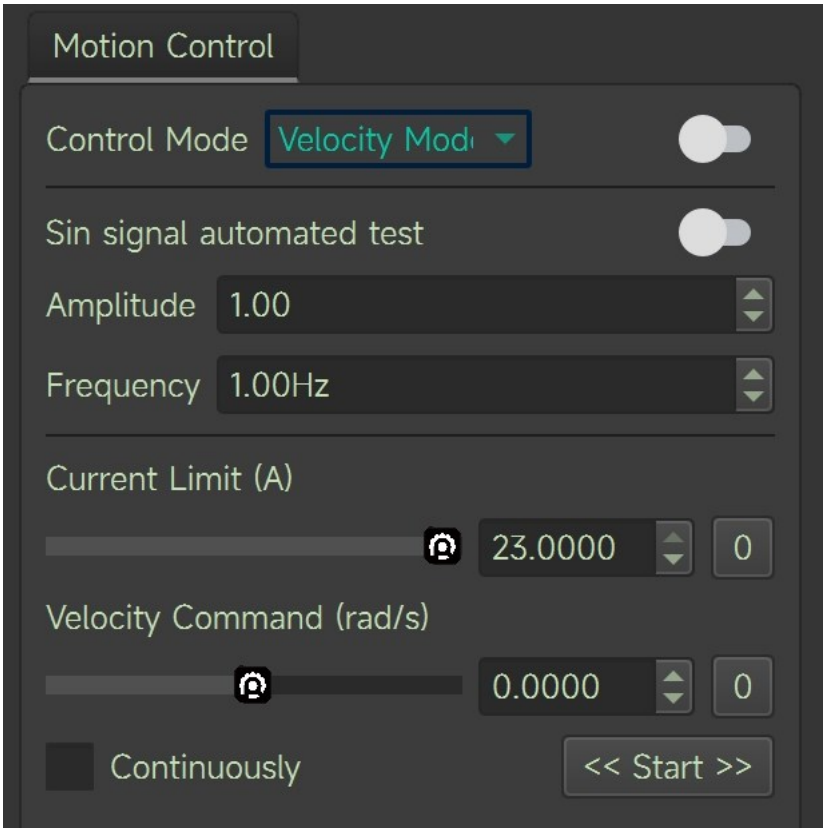
3.4.2 Current mode



Manually switch the current mode, click the right switch button, and then set the Iq current command value. Start or continue sending, and the motor will follow the current command. Click the right switch button again, and the motor will stop.

Click the switch button on the right side of the control mode, input the amplitude and frequency of the sine automatic test, and then click the switch button on the right side of the sine automatic test. The iq (A) of the motor will run according to the set amplitude and frequency.

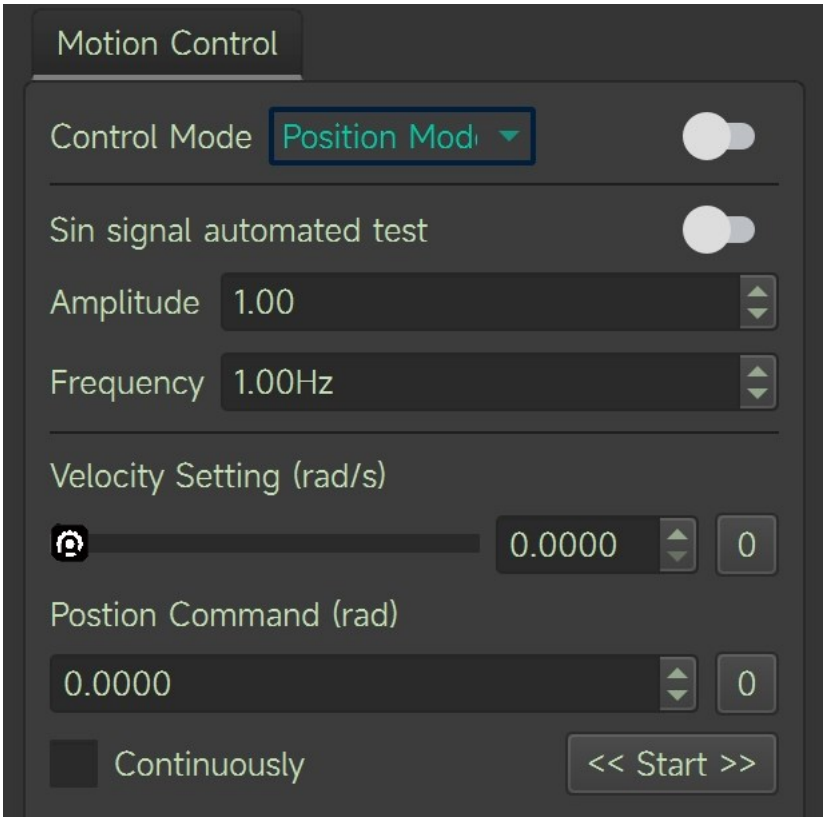
3.4.3 Speed mode



Manually switch speed mode, click the right switch button, then set the speed command value, start or continuously send, the motor will follow the speed command to run, click the right switch button again, the motor will stop.

Click the switch button on the right side of the control mode, input the amplitude and frequency of the sine automatic test, and then click the switch button on the right side of the sine automatic test. The motor speed (rad/s) will run according to the set amplitude and frequency.

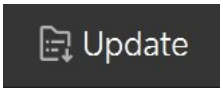
3.4.4 Position mode



Manually switch position mode, click the right switch button, then set the position command value (rad), start or continuously send, the motor will follow the target position command to run, click the right switch button again, the motor will stop. The maximum speed followed by the position can be modified by setting the speed.

Click the switch button on the right side of the control mode, input the amplitude and frequency of the sine automatic test, and then click the switch button on the right side of the sine automatic test. The position (rad) of the motor will run according to the set amplitude and frequency.

3.5 Firmware updates



The first step is to click on the upgrade of the device module and select the bin file to be burned; The second step is to confirm the upgrade and start updating the firmware of the motor. After the progress is completed, the motor update is completed and automatically restarts.

4. Driver Communication Protocol and Instructions for Use

The motor communication uses CAN 2.0 communication interface, with a baud rate

of 1Mbps and an extended frame format, as shown below:

Data Domain	29 digit ID			8Byte data area
size	Bit28~bit24	Bit23-8	Bit7~0	Byte0~Byte7
describe	Communication type	Data Area 2	Target address	Data Area 1

The control modes supported by the motor include:

Operation control mode: Given 5 parameters for motor operation control;

Current mode: Given the specified Iq current of the motor;

Speed mode: Given the specified operating speed of the motor;

Position mode: Given the specified position of the motor, the motor will run to that specified position;

4.1 Description of Communication Protocol Types

4.1.1 Obtain device ID (communication type 0); Obtain the device ID and 64 bit MCU unique identifier

Data Domain	29 digit ID			8Byte data area
size	Bit28~bit24	Bit23-8	Bit7~0	Byte0~Byte7
describe	0	Bit15~8: Used to identify the host CAN_ ID	Target motor CAN_ ID	0

Response frame:

Data Domain	29 digit ID			8Byte data area
size	Bit28~bit24	Bit23-8	Bit7~0	Byte0~Byte7
describe	0	Target motor CAN_ ID	0XFE	64 bit MCU unique identifier

4.1.2 Operation Control Mode Motor Control Command (Communication Type 1) is used to send control commands to the motor

Data Domain	29 digit ID			8Byte data area
Size	Bit28~bit24	Bit23-8	Bit7~0	Byte0~Byte7
Description	1	Byte2: Moment (0~65535) Corresponding (-120Nm~120Nm)	Target motor CAN_ ID	Byte0~1: Target angle [0~65535] corresponds to (-4 π~4 π) Byte2~3: Target angular velocity [0~65535] corresponds to (-15rad/s~15rad/s) Byte4~5: Kp [0~65535] corresponds to (0.0~5000.0) Byte6~7: Kd [0~65535] corresponds to (0.0~100.0) After converting the above data, high bytes come first and low bytes come last

Response frame: Response motor feedback frame (see communication type 2)

4.1.3 Motor feedback data (communication type 2) is used to provide feedback on the motor operation status to the host

Data Domain	29 digit ID			8Byte data area
Size	Bit28~bit24	Bit23-8	Bit7~0	Byte0~Byte7
Description	2	Bit8~Bit15: Current motor CAN ID Bit21~16: Fault information (0 none 1 available) Bit21: Uncalibrated Bit20: HALL encoding fault Bit19: Magnetic encoding fault Bit18: Over temperature Bit17: Overcurrent Bit16: Undervoltage fault Bit22~23: Mode status 0: Reset mode [reset] 1: Cali mode [calibration] 2: Motor	Host CAN_ ID	Byte0~1: Current angle [0~65535] corresponds to (-4 π~4 π) Byte2~3: Current angular velocity [0~65535] corresponds to (-15rad/s~15rad/s) Byte4~5: Current torque [0~65535] corresponds to (-120Nm~120Nm) Byte6~7: Current temperature: Temperature (Celsius) * 10 The above data has high

		mode [Run]		bytes first and low bytes last

4.1.4 Motor Enable Operation (Communication Type 3)

Data Domain	29 digit ID			8Byte data area
Size	Bit28~bit24	Bit23-8	Bit7~0	Byte0~Byte7
Description	3	Bit15~8: Used to identify the main CAN_ ID	Target motor CAN_ ID	

Response frame: Response motor feedback frame (see communication type 2)

4.1.5 Motor stop running (communication type 4)

Data Domain	29 digit ID			8Byte data area
Size	Bit28~bit24	Bit23-8	Bit7~0	Byte0~Byte7
Description	4	Bit15~8: Used to identify the main CAN_ ID	Target motor CAN_ ID	During normal operation, the data area needs to be cleared to 0; When Byte [0]=1: Clear fault;

Response frame: Response motor feedback frame (see communication type 2)

4.1.6 Setting the mechanical zero position of the motor (communication type 6) will set the current motor position to the mechanical zero position (power loss)

Data Domain	29 digit ID			8Byte data area
Size	Bit28~bit24	Bit23-8	Bit7~0	Byte0~Byte7

Description	6	Bit15~8: Used to identify the main CAN_ ID	Target motor CAN_ ID	Byte [0]=1

Response frame: Response motor feedback frame (see communication type 2)

4.1.7 Setting Motor CAN_ ID (Communication Type 7)

Change the current motor CAN_ ID, effective immediately.

Data Domain	29 digit ID			8Byte data area
Size	Bit28~bit24	Bit23-8	Bit7~0	Byte0~Byte7
Description	7	Bit15~8: Used to identify the main CAN_ ID Bit16~23: Pre set CAN_ ID	Target motor CAN_ ID	

Response frame: Response motor broadcast frame (see communication type 0)

4.1.8 Single parameter reading (communication type 17)

Data Domain	29 digit ID			8Byte data area
Size	Bit28~bit24	Bit23-8	Bit7~0	Byte0~Byte7
Description	17	Bit15~8: Used to identify the main CAN_ ID	Target motor CAN_ ID	Byte0~1: index, parameter list detailed in

		Bit23-16:00 indicates successful read 01 indicates read failure		4.1.11 Byte2~3:00 Byte4~7:00 The above data has low bytes first and high bytes last

Response frame:

Data Domain	29 digit ID			8Byte data area
Size	Bit28~bit24	Bit23-8	Bit7~0	Byte0~Byte7
Description	17	ID	Host CAN_ ID	Byte0~1: index, parameter list detailed in 4.1.11 Byte2~3:00 Byte4~7: Parameter data, 1 byte of data in Byte4 The above data has low bytes first and high bytes last

4.1.9 Single parameter writing (communication type 18) (power loss)

Data Domain	29 digit ID			8Byte data area
Size	Bit28~bit24	Bit23-8	Bit7~0	Byte0~Byte7

Description	18	Bit15~8: Used to identify the main CAN_ ID	Target motor CAN_ ID	Byte0~1: index, parameter list detailed in 4.1.11 Byte2~3:00 Byte4~7: Parameter data The above data has low bytes first and high bytes last

Response frame: Response motor feedback frame (see communication type 2)

4.1.10 Fault Feedback Frame (Communication Type 21)

Data Domain	29 digit ID			8Byte data area
Size	Bit28~bit24	Bit23-8	Bit7~0	Byte0~Byte7
Description	21	Bit15~8: Used to identify the main CAN_ ID	Motor CAN_ ID	Byte0~3: fault value (non 0: faulty, 0: normal) Bit16: A-phase current sampling overcurrent Bit15~bit8: Overload fault Bit7: Encoder not calibrated Bit5: C-phase current sampling overcurrent Bit4: B-phase

				current sampling overcurrent Bit3: Overvoltage fault Bit2: Undervoltage fault Bit1: Driver chip failure Bit0: Motor over temperature fault, default to 80 degrees Byte4~7: warning value Bit0: Motor over temperature warning, default to 75 degrees

4.1.11 Read and write a single parameter list

Parameter index	Parameter Name	Description	Type	Byte count	Unit/Description	R/W read and write permissions
0X7005	Run_Mode	0: Operation control mode 1: Position mode	Uint8	1		W/R

		2: Speed mode 3: Current mode				
0X7006	Iq_ Ref	Current mode Iq command	Float	4	-90-90A	W/R
0X700A	Spd_ Ref	Speed mode speed command	Float	4	-15-15 rad/s	W/R
0X700B	Limit_ Torque	Torque limitation	Float	4	0-120Nm	W/R
0X7010	Cur_ Kp	Kp of current	Float	4	Default value 0.05	W/R
0X7011	Cur_ Ki	Ki of current	Float	4	Default value 0.05	W/R
0X7014	Cur_ Fit_ Gain	Current filtering coefficient filter_ Gain	Float	4	0~1.0, default value 0.06	W/R
0X7016	Ref	Position mode angle command	Float	4	Rad	W/R
0X7017	Limit_ Spd	Position mode speed limit	Float	4	0-15 rad/s	W/R

0X7018	Limit_ Cur	Speed position mode current limit	Float	4	0-90A	W/R
0x7019	mechPos	Load end-ring mechanical angle	float	4	rad	R
0x701A	iqf	The iq filter values	float	4	-90~90A	R
0x701B	mechVel	Load end speed	float	4	-15~15rad/s	R
0x701C	VBUS	busbar voltage	float	4	V	R
0x701E	loc_kp	positional kp	float	4	Default value 30	W/R
0x701F	spd_kp	Speed kp	float	4	Default value 5	W/R
0x7020	spd_ki	Speedki	float	4	Default value 0.005	W/R
0x7021	spd_filt_gain	Speed filter value	float	4	Default value 0.1	W/R

Reading example:

To read loc_ For example, kp:

The read instruction is

Size	Bit28~bit24	Bit23-8	Bit7~0	Byte0~Byte7
	11	00FD	7F	1E 70 00 00 00 00 00 00 00 00
Description	Type 17 Hexadecimal 0x11	Host ID 0xFD	Target motor CAN_ ID 7F	Byte0~1: index, corresponding to loc_ Kp

The feedback command is

Size	Bit28~bit24	Bit23-8	Bit7~0	Byte0~Byte7
	11	007F	FD	1E 70 00 00 00 00 00 F0

				41
Description	Type 17 Hexadecimal 0x11	Bit 15~8: Target motor CAN_ ID 7F	Host ID 0xFD	Byte0~1: index, corresponding to loc_ Kp Byte4~7: loc_ Kp value is 30, right high byte, (32-bit single precision) hexadecimal IEEE-754 standard floating-point number

4.2 Instructions for using control mode

4.2.1 Program Examples

The following provides examples of various mode control motors (taking gd32f303 as an example)

Below are various instance call libraries, function and macro definitions

#Define P_ MIN -12.5f

#Define P_ MAX 12.5f

#Define V_ MIN -15.0f

#Define V_ MAX 15.0f

#Define KP_ MIN 0.0f

#Define KP_ MAX 500.0f

#Define KD_ MIN 0.0f

#Define KD_ MAX 5.0f

#Define T_ MIN -120.0f

#Define T_ MAX 120.0f

Struct exCanIdInfo{

Uint32_ T id: 8;

```
Uint32_ T data: 16;
Uint32_ T mode: 5;
Uint32_ T res: 3;
};
Can_ Receive_ Message_ Struct rxMsg;
Can_ Trasnmit_ Message_ Struct txMsg={
. tx_ Sfid=0,
. tx_ Efid=0xff,
. tx_ Ft=CAN_ FT_ DATA,
. tx_ Ff=CAN_ FF_ Extended,
. tx_ Dlen=8,
};
#Define txCanIdEx (((struct exCanIdInfo)&(txMsg. tx_efid)))
#Define rxCanIdEx (((struct exCanIdInfo)&(rxMsg. rx_efid)))//Parse the extended frame ID to a custom data structure
Int float_ To_ Uint (float x, float x_min, float x_max, int bits){
Float span=x_ Max - x_ Min;
Float offset=x_ Min;
If (x>x_max) x=x_ Max;
Else if (x<x_min) x=x_ Min;
Return (int) (x-offset) * (float) ((1<<bits) -1))/span);
}
#Define can_ Txd() can_ Message_ Transmit (CAN0,&txMsg)
#Define can_ Rxd() can_ Message_ Receive (CAN0, CAN-FIFO1,&rxMsg)
```

Here are some common types of communication to send:

1. Motor Enable Run Frame (Communication Type 3)

```
    Void motor_ Enable (uint8_t id, uint16_t master_id)
    {
    TxCanIdEx. mode=3;
    TxCanIdEx. id=id;
    TxCanIdEx. res=0;
    TxCanIdEx. data=master_ ID;
    TxMsg. tx_ Dlen=8;
```

```
TxCanIdEx. data=0;
Can_ Txd();
}
```

2. Operation control mode motor control command (communication type 1)

```
Void motor_ Control mode (uint8_t id, float torque, float MechPosition, float speed, float kp, float kd)
{
TxCanIdEx. mode=1;
TxCanIdEx. id=id;
TxCanIdEx. res=0;
TxCanIdEx. data=float_ To_ Uint (torque, T.MIN, T.MAX, 16);
TxMsg. tx_ Dlen=8;
TxMsg. tx_ Data [0]=float_ To_ Uint (MechPosition, P_MIN, P_MAX, 16)>>8;
TxMsg. tx_ Data [1]=float_ To_ Uint (MechPosition, P_MIN, P_MAX, 16);
TxMsg. tx_ Data [2]=float_ To_ Uint (speed, V-MIN, V-MAX, 16)>>8;
TxMsg. tx_ Data [3]=float_ To_ Uint (speed, V-MIN, V-MAX, 16);
TxMsg. tx_ Data [4]=float_ To_ Uint (kp, KP-MIN, KP-MAX, 16)>>8;
TxMsg. tx_ Data [5]=float_ To_ Uint (kp, KP-MIN, KP-MAX, 16);
TxMsg. tx_ Data [6]=float_ To_ Uint (kd, KD-MIN, KD-MAX, 16)>>8;
TxMsg. tx_ Data [7]=float_ To_ Uint (kd, KD-MIN, KD-MAX, 16);
Can_ Txd();
}
```

3. Motor stop running frame (communication type 4)

```
Void motor_ Reset (uint8_t id, uint16_t master_id)
{
TxCanIdEx. mode=4;
TxCanIdEx. id=id;
TxCanIdEx. res=0;
TxCanIdEx. data=master_ ID;
TxMsg. tx_ Dlen=8;
For (uint8_t i=0; i<8; i++)
{
TxMsg. tx_ Data [i]=0;
```

```
}
Can_ Txd();
}
```

4. Motor mode parameter writing command (communication type 18, operation mode switching)

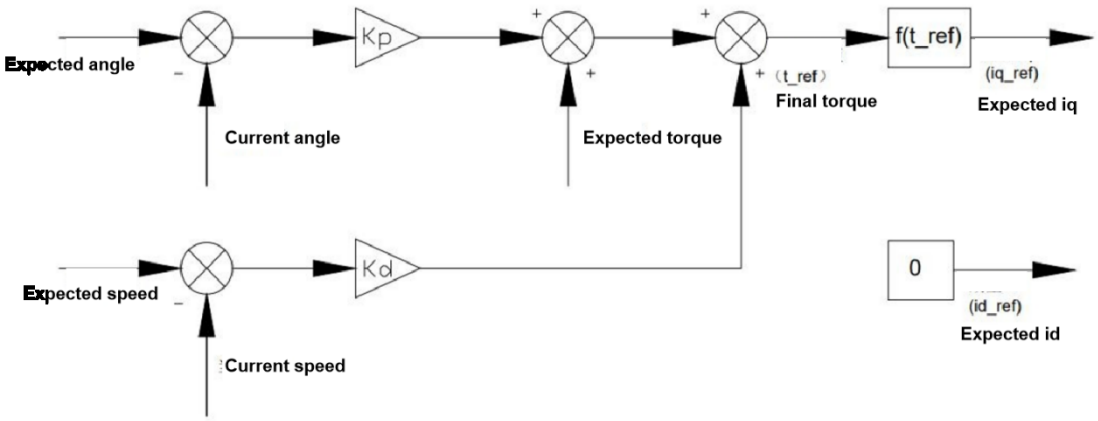
```
Uint8_ T runmode;
Uint16_ T index;
Void motor_ Modechange (uint8_t id, uint16_t master_id)
{
TxCanIdEx. mode=0x12;
TxCanIdEx. id=id;
TxCanIdEx. res=0;
TxCanIdEx. data=master_ ID;
TxMsg. tx_ Dlen=8;
For (uint8_t i=0; i<8; i++)
{
TxMsg. tx_ Data [i]=0;
}
Memcpy (&txMsg. tx_data [0],&index, 2);
Memcpy (&txMsg. tx_data [4],&runmode, 1);
Can_ Txd();
}
```

5. Motor mode parameter writing command (communication type 18, control parameter writing)

```
Uint16_ T index;
Float ref;
Void motor_ Write (uint8_t id, uint16_t master_id)
{
TxCanIdEx. mode=0x12;
TxCanIdEx. id=id;
TxCanIdEx. res=0;
TxCanIdEx. data=master_ ID;
TxMsg. tx_ Dlen=8;
```

```
For (uint8_t i=0; i<8; i++)
{
TxMsg. tx_ Data [i]=0;
}
Memcpy (&txMsg. tx_data [0],&index, 2);
Memcpy (&txMsg. tx_data [4],&ref, 4);
Can_ Txd();
}
```

4.2.2 Operation and control mode



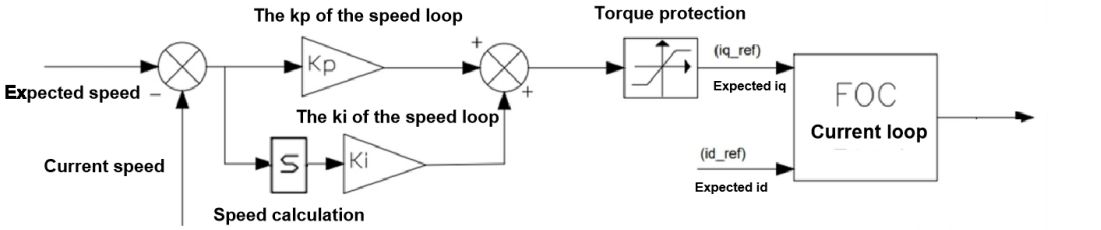
After the motor is powered on, it defaults to operation control mode;

Send motor enable running frame (communication type 3) -->Send operation control mode motor control command (communication type 1) -->Receive motor feedback frame (communication type 2)

4.2.3 Current mode

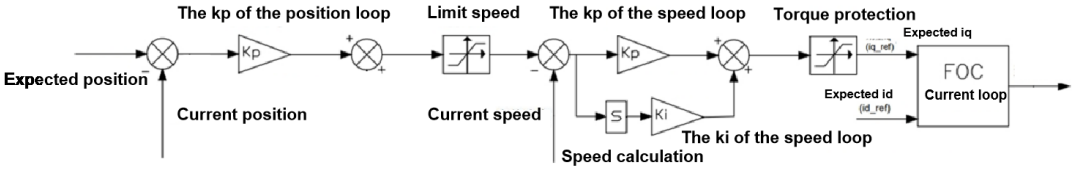
Send motor mode parameter writing command (communication type 18) Set runmode parameter to 3-->Send motor enable running frame (communication type 3) -->Send motor mode parameter writing command (communication type 18) Set **iq_ The ref** parameter is a preset current command

4.2.4 Speed mode



Send motor mode parameter writing command (communication type 18) Set runmode parameter to 2-->Send motor enable running frame (communication type 3) -->Send motor mode parameter writing command (communication type 18) Set **limit_ The cur** parameter is the preset maximum current command -->Send motor mode parameter writing command (communication type 18) Set **SPD_ The ref** parameter is a preset speed command

4.2.5 Position mode



Send motor mode parameter write command (communication type 18) to set the runmode parameter to 1 --> Send motor enable run frame (communication type 3) --> Send motor mode parameter write command (communication type 18) to set the limit_spd parameter to the preset maximum speed instruction --> Send motor mode parameter write command (communication type 18) to set the loc_ref parameter to the preset position instruction.

4.2.6 Stopping operation

Send motor stop running frame (communication type 4)