Joint Module Communication Protocol

Simple User Manual

Applicable Driver: Integrated Communication Module

Version:V1.0

Date：2025.12.6

Disclaimer

Thank you for purchasing the Juxie motor drive system. Please read this disclaimer carefully before use. Once used, you are deemed to have acknowledged and accepted all contents of this disclaimer.

Please install and use this product in strict accordance with the manual, product instructions, and relevant laws, regulations, policies, and guidelines. During product use, you undertake to be responsible for your own actions and all consequences arising therefrom. Juxie shall not be legally liable for any losses caused by improper use, installation, or modification by the user.

Without permission, no part of this document may be copied or reproduced in any form. The final right to interpret this disclaimer is reserved by Juxie Intelligent Drive Technology Co., Ltd.

This protocol description is only applicable to versions above 1.16.07.

Directory

1. Module Power-On Program Operation Logic

Module Power-On

1. Enter the boot program, jump to App after 300 ms.
2. App initialization takes approximately 2 s.
3. After initialization, upload a data frame with ID 0x700 + ID and data 00, and the watchdog locks automatically on power-on.
4. After the motor is enabled, delay 100 ms before rotating to a new position.
5. Notes
6. Watchdog data frame interval must be less than 500 ms; otherwise, the motor locks.
7. There is a 100 ms delay for both brake release and engagement.

2. CAN Bus Wiring Specification Guide

2.1 Signal Line Definition

- CAN_H: High- level data line; dominant level ≈ 3.5 V, recessive level ≈ 2.5 V.
- CAN_L: Low- level data line; dominant level ≈ 1.5 V, recessive level ≈ 2.5 V.
- CAN_GND: Signal ground; establishes a common ground reference to improve common- mode interference immunity.

2.2 Termination Resistor

- Function: Eliminate signal reflection at the bus end and ensure signal integrity.
- Specification:
- Quantity: One termination resistor must be installed at each physical end of the bus.

- Resistance: 120 Ω.
- Location: Only at the two farthest nodes; do not install at intermediate nodes.
- Connection: Parallel between CAN_H and CAN_L.
- Check: Measure resistance at both ends of the main bus with power off. A healthy network with two 120 Ω resistors shows ≈ 60 Ω total resistance.

3. Protocol Description

The CAN communication of Juxie modules supports CAN 2.0 and CANFD, both following the CANopen 402 protocol (CiA 402 / DS402) for drives and motion control.

3.1 Communication Configuration

- CANFD default: CANFD acceleration, standard frame, 1 M arbitration segment, 5 M data segment.

3.2 Actuator ID

Range: 0x01 ~ 0x7F

3.3 Object Dictionary

All parameters, data, and functions are accessed via a 16- bit index and 8- bit subindex.

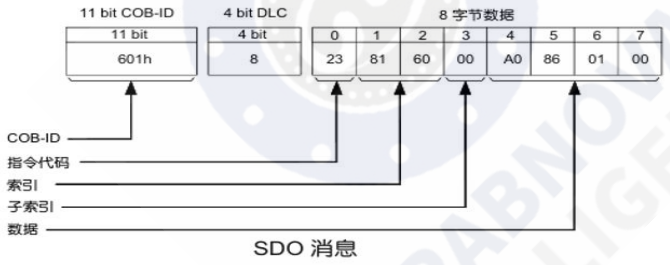
3.4 Main Control Modes

Mode	Name	Features
PP (Profile Position)	Profile Position Mode	Predefined position profile (target position, speed, acceleration); drive executes as planned.
PV (Profile Velocity)	Profile Velocity Mode	Reaches and maintains target speed with preset acceleration/deceleration ramps.
CSP (Cyclic Synchronous	Cyclic Synchronous	Host cyclically sends target position; drive tracks synchronously with high precision.

Mode	Name	Features
Position)	Position	
CSV (Cyclic Synchronous Velocity)	Cyclic Synchronous Velocity	Host cyclically sends target speed; position is derived by integration.
CST (Cyclic Synchronous Torque)	Cyclic Synchronous Torque	Directly controls motor output torque.
MIT	Precise Torque Control	Mixed control of current, speed, and position.

4. CANFD/CANopen SDO Control Examples

4.1 SDO Message Parsing



4.2 Example: Modify Motor ID (Master → Slave)

CAN Field	Length	Description
StdID	11 bit	600 + Dev_ID
DLC	—	0x08
[0]	16 bit	Command code
[1–2]	32 bit	Dictionary index
[3]	16 bit	Subindex
[4–7]	32 bit	New ID (1–127, little- endian)

Command Code Description

Code	Description	Direction
2Fh	Write 1 byte	Master → Slave
2Bh	Write 2 bytes	Master → Slave
27h	Write 3 bytes	Master → Slave
23h	Write 4 bytes	Master → Slave
60h	Write response	Slave → Master
40h	Read 1 byte	Master → Slave
4Fh	Read response (1 byte)	Slave → Master
4Bh	Read response (2 bytes)	Slave → Master
47h	Read response (3 bytes)	Slave → Master
43h	Read response (4 bytes)	Slave → Master

4.3 Dictionary Operation Example

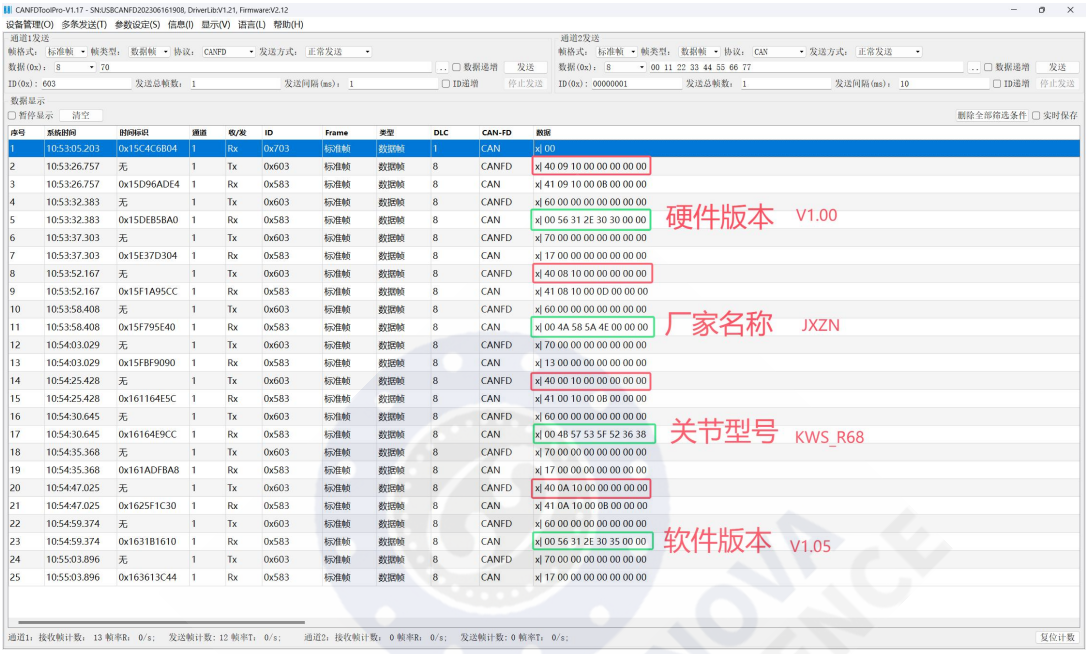
In the example, the red-texted content represents the target value, with data values listed from highest to lowest.

- **Get motor version information**

ID	DLC	Data(HEX)	Description
60X	8	40 08 10 00 00 00 00 00	Manufacturer Name
60X	8	60 00 00 00 00 00 00 00	
60X	8	70 00 00 00 00 00 00 00	
ID	DLC	Data(HEX)	Description
60X	8	40 00 10 00 00 00 00 00	Joint type
60X	8	60 00 00 00 00 00 00 00	
60X	8	70 00 00 00 00 00 00 00	
ID	DLC	Data(HEX)	Description
60X	8	40 0A 10 00 00 00 00 00	Firmware version
60X	8	60 00 00 00 00 00 00 00	
60X	8	70 00 00 00 00 00 00 00	

ID	DLC	Data(HEX)	Description
60X	8	40 09 10 00 00 00 00 00	硬件版本
60X	8	60 00 00 00 00 00 00 00	
60X	8	70 00 00 00 00 00 00 00	

The read version needs to be converted to ASCII code.



Modify Motor ID

表格 ID	DLC	Data (HEX)	Description
60x	8	23 30 25 00 05 00 00 00	Set Motor ID to 5

● **Zero Calibration Procedure** (Motor must be disabled during setup.)

1. Enable the motor and move it to the target zero position.
2. Send calibration command: 60x 23 31 25 00 01 00 00 00.
3. Read the current position feedback; confirm position = 0 for success.
4. Disable the motor and engage brake; send calibration reset command: 60x 23 31 25 00 01 00 00 00.
5. Re-enable the motor and release brake for normal control.

ID	DLC	Data (HEX)	Description
60x	8	23 31 25 00 01 00 00 00	Set Zero Point

● **Read Motor Position**

ID	DLC	Data (HEX)	Description
60x	8	40 64 60 00 00 00 00 00	Read Actual Motor Position
58x	8	43 64 60 00 00 00 00 00	Feedback Actual Motor Position

● **Position analysis of the feedbac**k: As shown in the figure below, the high-bit data comes after the low-bit data. Converting 3F FE to decimal yields 16382, which is then divided by 182 to obtain the position information 90.

ID	Frame	类型	DLC	CAN-FD	数据
0x601	标准帧	数据帧	8	CANFD	x| 43 64 60 00 00 00 00 00
0x581	标准帧	数据帧	8	CANFD-...	x| 43 64 60 00 FE 3F 00 00

● **Disable Limit & Watchdog**

Disables the 500ms watchdog protection.

表 ID	DLC	Data (HEX)	Description
60x	8	23 50 26 00 01 00 00 00	Disable Watchdog & Limit Protection

● **Set Heartbeat**

Heartbeat feedback active after setting; restores default after power cycle.

ID	DLC	Data (HEX)	Description

ID	DLC	Data (HEX)	Description
60x	8	2B 17 10 00 D0 07 00 00	Set Heartbeat to 2000ms

● **Set Positive Motor Limit**

Motor must be disabled; default: −180° ~ +180°.Save to Flash after setup.

ID	DLC	Data (HEX)	Description
60x	8	23 7D 60 02 30 7F 00 00	Set Positive Limit
60x	8	40 7D 60 01 00 00 00 00	Read Positive Limit

● **Set Negative Motor Limit**

Motor must be disabled; default: −180° ~ +180°.Save to Flash after setup.

ID	DLC	Data (HEX)	Description
60x	8	23 7D 60 01 C8 80 FF FF	Set Negative Limit
60x	8	40 7D 60 02 00 00 00 00	Read Negative Limit

● **Modify CANFD Data Segment Baud Rate**

Baud rate mapping:1=5M, 2=4M, 3=2M, 4=1MTakes effect after reboot; CAN not supported. Save to Flash.

ID	DLC	Data (HEX)	Description
60x	8	23 40 25 00 01 00 00 00	Set CANFD Data Segment Baud Rate

● **SDO Position Mode Example**

Position loop speed = reducer output speed

Motor Enable

ID	DLC	Data (HEX)	Description

ID	DLC	Data (HEX)	Description
601	8	2B 40 60 00 06 00 00 00	Ready to Start
601	8	2B 40 60 00 07 00 00 00	Release Brake
601	8	2B 40 60 00 0F 00 00 00	Motor Enable Active

Set Operation Mode

ID	DLC	Data (HEX)	Description
601	8	2F 60 60 00 01 00 00 00	Set Position Mode

Set Motion Parameters

ID	DLC	Data (HEX)	Description
601	8	23 83 60 00 D0 07 00 00	Acceleration = 2000 RPM/s (motor side, 0~10000)
601	8	23 84 60 00 D0 07 00 00	Deceleration = 2000 RPM/s (motor side, 0~10000)
601	8	23 81 60 00 0A 00 00 00	Speed = 10 RPM (reducer output, 0~30)

Absolute Motion

ID	DLC	Data (HEX)	Description
601	8	23 7A 60 00 00 40 00 00	Set Target Position (−32767 ~ 32768)
601	8	2B 40 60 00 4F 00 00 00	Start Absolute Motion

Stop Motion

ID	DLC	Data (HEX)	Description
601	8	2B 40 60 00 0F 00 00 00	Stop Motor Motion

● SDO Velocity Mode Example

Velocity loop speed/acceleration = motor side speed

Motor Enable

ID	DLC	Data (HEX)	Description

ID	DLC	Data (HEX)	Description
601	8	2B 40 60 00 06 00 00 00	Ready to Start
601	8	2B 40 60 00 07 00 00 00	Release Brake
601	8	2B 40 60 00 0F 00 00 00	Motor Enable Active

Set Operation Mode

ID	DLC	Data (HEX)	Description
601	8	2F 60 60 00 03 00 00 00	Set Velocity Mode

Set Motion Parameters

ID	DLC	Data (HEX)	Description
601	8	23 83 60 00 E8 03 00 00	Acceleration = 1000 RPM/s (0~10000)
601	8	23 84 60 00 E8 03 00 00	Deceleration = 1000 RPM/s (0~10000)
601	8	23 FF 60 00 F4 01 00 00	Speed = 500 RPM

- **Current Mode Example：**

Max current recommended default (match motor peak current)

Motor Enable

ID	DLC	Data (HEX)	Description
601	8	2B 40 60 00 06 00 00 00	Ready to Start
601	8	2B 40 60 00 07 00 00 00	Release Brake
601	8	2B 40 60 00 0F 00 00 00	Motor Enable Active

Set Operation Mode

ID	DLC	Data (HEX)	Description

ID	DLC	Data (HEX)	Description
601	8	2F 60 60 00 0A 00 00 00	Set Current Mode

Set Motion Parameters

ID	DLC	Data (HEX)	Description
601	8	2B 71 60 00 F4 01 00 00	Target Current = 500 mA (refer to motor specs)

- **Set Motor PI Parameters**

ID	DLC	Data (HEX)	Description
601	8	23 32 25 00 00 00 00 00	Set Current Loop P
601	8	23 33 25 00 00 00 00 00	Set Current Loop I
601	8	23 34 25 00 00 00 00 00	Set Velocity Loop P
601	8	23 35 25 00 00 00 00 00	Set Velocity Loop I
601	8	23 36 25 00 00 00 00 00	Set Position Loop P
601	8	23 37 25 00 00 00 00 00	Set Position Loop I

- **Read PI Parameters**

Description	Data (HEX)
Read Current Loop P	40 32 25 00 00 00 00 00
Read Current Loop I	40 33 25 00 00 00 00 00
Read Velocity Loop P	40 34 25 00 00 00 00 00
Read Velocity Loop I	40 35 25 00 00 00 00 00
Read Position Loop P	40 36 25 00 00 00 00 00
Read Position Loop I	40 37 25 00 00 00 00 00
Read Max Current	40 38 25 00 00 00 00 00

● **SDO Feedback Status Reading Examples**

Read motion status

ID	DLC	Data(HEX)	Description	Details
601	8	40 78 60 00 00 00 00 00	Read Motor Actual Current	
581	8	4B 78 60 00 45 01 00 00	Feedback Motor Actual Current	High byte rear, low byte front. 0145 converted to decimal: 325, unit: mA
601	8	40 64 60 00 00 00 00 00	Read Motor Actual Position	
581	8	43 64 60 00 FE 3F 00 00	Feedback Motor Actual Position	High byte rear, low byte front. 3FFE converted to decimal: 16382, divided by 182 = 90°. Pulses per revolution: 65536, range: -32767 ~ 32768 (angle: -180° ~ 180°)
601	8	40 6C 60 00 00 00 00 00	Read Motor Actual Speed	
581	8	43 6C 60 00 F9 01 00 00	Feedback Motor Actual Speed	High byte rear, low byte front. 01F9 converted to speed: 50.5 RPM
601	8	40 3F 60 00 00 00 00 00	Read Motor Fault	
581	8	4B 3F 60 00 00 00 00 00	Feedback Motor Fault	0x0001: Overvoltage; 0x0002: Undervoltage; 0x0004: Overtemperature alarm; 0x0008: Locked rotor; 0x0010: Overload; 0x0020: Current sampling error; 0x0040: Positive limit protection; 0x0080: Negative limit protection; 0x0100: Encoder communication timeout; 0x0200: Motor over maximum speed; 0x0400: Power- on electrical angle initialization failure; 0x1000: Excessive position error; 0x2000: Encoder fault

ID	DLC	Data(HEX)	Description	Details
601	8	40 41 60 00 00 00 00 00	Read Motor Motion Status	
581	8	4B 41 60 00 00 00 00 00	Feedback Motor Motion Status	0x0021: Ready to start; 0x0023: Brake released; 0x0027: Enabled; 0x0008: Fault state
601	8	40 62 26 00 00 00 00 00	Read MOS Temperature Sensor	
581	8	43 62 26 00 18 01 00 00	Feedback Temperature Sensor	High byte rear, low byte front. 0118 converted to decimal: 280, temperature: 28.0 °C
601	8	40 63 26 00 00 00 00 00	Read Motor Temperature Sensor	
581	8	43 62 26 00 18 01 00 00	Feedback Temperature Sensor	High byte rear, low byte front. 0118 converted to decimal: 280, temperature: 28.0 °C

5. CANFD Custom PDO Control Examples

● **Single Actuator Control Message**

The host computer sends commands to the actuator.

CAN Frame	CAN Frame	Field Name	Byte	Description (Data)
StdID	StdID	Device Address	11bit	0x100
DLC	DLC	Frame Length		7

[0]	Bit[7]	Enable Control	1bit	0x00: Disable; 0x01: Enable
	Bit[6]	Brake Control	1bit	0x00: Brake Engage; 0x01: Brake Release
	Bit[5]	Clear Error Control	1bit	0: Default; 1: Reset Error
	Bit[4~1]	Control Mode	4bit	0x01: Profile Position Mode; 0x02: Profile Velocity Mode;0x03: CSP Position Mode; 0x04: CSV Velocity Mode;0x05: Current Loop Mode;
	Bit[0]	Reserved		
[1]	Target Param 1		16bit	Byte[1] High 8-bit, Byte[2] Low 8-bit:1. Position Mode (0x01,0x03): Output Target Position [-32768~32767] → (-180°~180°);2. Velocity Mode (0x02,0x04): Motor Target Speed [-32768~32767], Unit: RPM;3. Current Mode (0x05): Motor Target Iq Current [-32768~32767], Unit: mA;
[2]				

[3]	Target Param 2	16bit	Byte[3] High 8-bit, Byte[4] Low 8-bit:1. Profile Position/Velocity Mode (0x01,0x02): Profile Acc/Dec [0~20000], Unit: RPM/s;
[4]			
[5]	Target Feedforward Param	16bit	Byte[5] High 8-bit, Byte[6] Low 8-bit:1. Feedforward Reserved;2. Position Mode (0x01): Output Profile Velocity;
[6]			

- **Example of a single-axis control message (using motor ID 1 as an example)**

CAN帧		字段名称	字节	内容说明（数据）	1、电机使能，抱闸释放，轮廓位置模式，轮廓速度5RPM，加减速2000RPM/S，运行到0度位置		4、电机使能，抱闸释放，轮廓速度模式，轮廓速度500RPM，加减速1000RPM/S		5、电机使能，抱闸释放，电流环模式，目标电流1000mA	
StdID		设备地址	11bit	0x100|Dev_ID	0x101		0x101		0x101	
DLC		帧长度		7	7		7		7	
[0]	Bit[7]	使能控制	1bit	0x00：下使能 0x01：上使能	1	电机使能	1	电机使能	1	电机使能
	Bit[6]	抱闸控制	1bit	0x00：抱闸吸合 0x01：抱闸释放	1	抱闸释放	1	抱闸释放	1	抱闸释放
	Bit[5]	清楚错误控制	1bit	0：默认 1：复位错误	0	默认	0	默认	0	默认
	Bit[4~1]	控制模式	4bit	0x01：轮廓位置模式；0x02：轮廓速度模式；0x03：CSP位置模式；0x04：CSV速度模式；0x05：电流环模式；	0001	轮廓位置模式	0010	轮廓速度模式	0101	电流环模式
	Bit[0]	预留			0		0		0	
[1]		目标参数1	16bit	Byte[1]为高8位，Byte[2]为低8位： 1、当控制模式为位置模式时（0x01，0x03），对应为输出端目标位置，范围是[-32768到32767]对应(-180°~180°)； 2、当控制模式为速度模式时（0x02，0x04），对应为电机端的目标速度，范围是[-32768到32767]，单位：RPM； 3、当控制模式为电流模式时（0x05），对应为电机目标Iq电流，范围是[-32768到32767]，单位：mA；	0x0000	位置0度（0cnt）	0x01F4	轮廓速度500RPM	0x03E8	目标电流1000mA
[2]										
[3]		目标参数2	16bit	Byte[3]为高8位，Byte[4]为低8位； 1、当控制模式为轮廓位置模式与轮廓速度模式时（0x01，0x02），对应为轮廓的加减速，范围是[0到20000]，单位：RPM/S；	0x07D0	轮廓加减速2000RMP/S	0x03E8	轮廓加减速1000RMP/S	0x0000	任意值
[4]										
[5]		目标前馈参数	16bit	Byte[5]为高8位，Byte[6]为低8位； 1、前馈参数的预留； 2、当控制模式为位置模式时（0x01），对应为输出端的轮廓速度；	0x0005	输出端速度5RPM	0x0000	任意值	0x0000	任意值
[6]										

序号	状态	ID(0x)	协议	长度	名称	数据	帧类型	每次发送帧数	发送次数
0	无	101	CANFD加	7	PP模式 转动到0cnt 加减速2000 速度5	C2 00 00 07 D0 00 05	标准帧	1	1
1	无	101	CANFD加	7	PP模式 转动到16384cnt 加减速1000 速度8	C2 40 00 03 E8 00 08	标准帧	1	1
2	无	101	CANFD加	7	PP模式 转动到-16384cnt 加减速1000 速度8	C2 C0 00 03 E8 00 08	标准帧	1	1
3	无	101	CANFD加	7	速度模式 速度500 加减速1000	C4 01 F4 03 E8 00 00	标准帧	1	1
4	无	101	CANFD加	7	速度模式 速度-500 加减速1000	C4 fe 0c 03 E8 00 00	标准帧	1	1
5	无	101	CANFD加	7	电流模式 电流1000	CA 03 E8 00 00 00 00	标准帧	1	1
6	无	101	CANFD加	7	电流模式 电流-1000	CA FC 18 00 00 00 00	标准帧	1	1
7	无	101	CANFD加	7	CSP模式 转动50cnt	C6 00 32 00 00 00 00	标准帧	1	1
8	无	101	CANFD加	7	CSP模式 转动100cnt	C6 00 64 00 00 00 00	标准帧	1	1

全选 反选 上移 下移 删除 清空 导入 导出 并行发送模式 失败停止 发送速度: 1 倍 列表发送

Simple example: 0x301 feedback, 0x081 feedback error

0x101	标准帧	数据帧	7	CANFD-...	x| C2 00 00 07 D0 00 05
0x301	标准帧	数据帧	12	CANFD-...	x| FF FE 00 00 FF E2 00 00 01 04 01 D0
0x081	标准帧	数据帧	8	CANFD-...	x| 02 00 00 00 00 00 00 00

● Multi- Actuator Control Message (Broadcast up to 8 motors)

CAN Frame	Field Name	Byte	Description (Data)
StdID	Device Address	11bit	0x200
DLC	Frame Length		64
[0]~[6]	Control Packet 1	112bit	Same as single- axis control message protocol
[7]~[13]	Control Packet 2	112bit	
[14]~[20]	Control Packet 3	112bit	
[21]~[27]	Control Packet 4	112bit	
[28]~[34]	Control Packet 5	112bit	
[35]~[41]	Control Packet 6	112bit	
[42]~[48]	Control Packet 7	112bit	
[49]~[55]	Control Packet 8	112bit	
[56]	Actuator ID for Packet 1	8bit	Corresponding actuator ID (Dev_ID)

CAN Frame	Field Name	Byte	Description (Data)
[57]	Actuator ID for Packet 2	8bit	
[58]	Actuator ID for Packet 3	8bit	
[59]	Actuator ID for Packet 4	8bit	
[60]	Actuator ID for Packet 5	8bit	
[61]	Actuator ID for Packet 6	8bit	
[62]	Actuator ID for Packet 7	8bit	
[63]	Actuator ID for Packet 8	8bit	

- **Simple Example (7 axes; unused filled with 00)**

0x200	D0 16 E5 00 00 00 00 D0 2D 15 00 00 00 00 D0 17 EA 00 00 00 00 D0 FD 65 00 00 00 00 D0 E1 07 00 00 00 00 D0 F4 E9 00 00 00 00 D0 08 C8 00 00 00 00 00 00 00 00 00 00 00 01 02 03 04 05 06 07 00
0x301	16 E3 00 00 F5 2D 01 98 01 DC 70 00
0x302	2D 17 00 00 F7 B0 01 B5 01 DE 70 00
0x303	18 41 FF A0 F4 49 01 BE 01 C8 70 00
0x304	FD 78 FF E8 04 50 01 AB 01 E0 70 00
0x305	E2 80 FE 8B FA CD 01 BE 01 F4 70 00
0x306	F5 51 FF A6 FF 2F 01 CC 01 E4 70 00
0x307	08 B2 00 01 FF C3 01 C2 01 E2 70 00

● MIT Mode Single-Action Actuator Control Message

The host computer sends commands to the actuator.

<table>
<tr><td colspan="5">The host computer sends commands to the actuator.</td></tr>
<tr><td colspan="2">CAN frame</td><td>Field Name</td><td>byte</td><td>Content Description (Data)</td></tr>
<tr><td colspan="2">StdID</td><td>Device Address</td><td>11bit</td><td>0x110|Dev_ID</td></tr>
<tr><td colspan="2">DLC</td><td>Frame Length</td><td></td><td>9</td></tr>
<tr><td rowspan="4">[0]</td><td>Bit[7]</td><td>Enable Control</td><td>1bit</td><td>0x00: Lower enable 0x01: Upper enable</td></tr>
<tr><td>Bit[6]</td><td>Brake Control</td><td>1bit</td><td>0x00: Brake engaged 0x01: Brake released</td></tr>
<tr><td>Bit[5]</td><td>Clear Error Control</td><td>1bit</td><td>0: Default 1: Reset error</td></tr>
<tr><td>Bit[4:1]</td><td>Control Mode</td><td>4bit</td><td>0x01: Contour position mode; 0x02: Contour speed mode; 0x03: CSP position mode; 0x04: CSV speed mode; 0x05: Current loop mode; 0x06: MIT mode;</td></tr>
<tr><td>[0]</td><td>Bit[0]</td><td>Reserved</td><td></td><td></td></tr>
<tr><td colspan="2">[1]</td><td rowspan="2">Target Position</td><td rowspan="2">16bit</td><td rowspan="2">Byte[0]为高 8 位，Byte[1]为低 8 位;
[0-65535]对应(-Pos Max~ Pos Max);</td></tr>
<tr><td colspan="2">[2]</td></tr>
<tr><td colspan="2">[3]</td><td rowspan="2">Target Velocity</td><td rowspan="2">12bit</td><td rowspan="3">Byte[3]-Bit[3:0] represents the high 4 bits, while Byte[4] represents the low 8 bits; the range [0~4095] corresponds to [0~500]; Byte[2] represents the high 8 bits, and Byte[3]-Bit[7:4] represents the low 4 bits; the range [0~4095] corresponds to (-Vel Max~Vel Max).</td></tr>
<tr><td rowspan="2">[4]</td><td>Bit[7:4]</td></tr>
<tr><td>Bit[3:0]</td><td rowspan="2">Position gain Kp value</td><td rowspan="2">12bit</td><td rowspan="2">Byte[3]-Bit[3:0] represents the high 4 bits, while Byte[4] represents the low 8 bits; the range [0~4095] corresponds to [0~500]; Byte[2] represents the high 8 bits, and Byte[3]-Bit[7:4] represents the low 4 bits; the range [0~4095] corresponds to (-Vel Max~Vel Max).</td></tr>
<tr><td colspan="2">[5]</td></tr>
<tr><td colspan="2">[6]</td><td rowspan="2">Speed gain Kd value</td><td rowspan="2">12bit</td><td rowspan="2">Byte[6]-Bit[3:0] represents the high 4 bits; Byte[7] corresponds to the low 8 bits (0–4095), mapping to the range (-T(Max)–T(Max). Byte[5] represents the high 8 bits, while Byte[6]-Bit[7:4] represents the low 4 bits (0–4095), mapping to the range [0, 5].</td></tr>
<tr><td>[7]</td><td>Bit[7:4]</td></tr>
</table>

	Bit[3:0]	Target Torque	12bit	Byte[6]-Bit[3:0] represents the high 4 bits; Byte[7] corresponds to the low 8 bits (0–4095), mapping to the range (-T(Max)–T(Max). Byte[5] represents the high 8 bits, while Byte[6]-Bit[7:4] represents the low 4 bits (0–4095), mapping to the range [0, 5].
[8]				

6. Actuator Feedback Message

Three Trigger Methods for Actuator Feedback Message

1. Send a synchronization frame (Host sends sync message ID: 0x80)
2. Triggered by multi- actuator control message
3. Triggered by single- axis control message

CAN Frame	Field Name	Byte	Description (Data)
StdID	Device Address	11bit	0x300
DLC	Frame Length		12
[0]	Actual Load Position	16bit	Byte[0] high 8- bit, Byte[1] low 8- bit; [-32768 ~ 32767] corresponds to (-180° ~ 180°)
[1]			
[2]	Actual Motor Speed	16bit	Byte[2] high 8- bit, Byte[3] low 8- bit; int16 [-32768 ~ 32767], Unit: RPM
[3]			
[4]	Actual Current (Iq)	16bit	Byte[4] high 8- bit, Byte[5] low 8- bit; int16 [-32768 ~ 32767], Unit: mA

CAN Frame		Field Name	Byte	Description (Data)
[5]				
[6]		Error Code	16bit	Byte[6] high 8- bit, Byte[7] low 8- bit; Refer to error code table
[7]				
[8]		Motor Winding Temperature	16bit	Byte[8] high 8- bit, Byte[9] low 8- bit; int16, Unit: 0.1 °C
[9]				
[10]		Control Mode Feedback	8bit	0x01: Profile Position Mode; 0x02: Profile Velocity Mode;0x03: CSP Position Mode; 0x04: CSV Velocity Mode;0x05: Current Loop Mode
[11]	Bit[7]	Enable State	1bit	0x00: Disabled; 0x01: Enabled
	Bit[6]	Brake State	1bit	0x00: Brake Engaged; 0x01: Brake Released
	Bit[5]	Fault State	1bit	0x00: Normal; 0x01: Fault
	Bit[4]	Position In- place State	1bit	0x00: Running; 0x01: Position Reached
	Bit[3~0]	Reserved		

Example Explanation

Tx	0x101	标准帧	数据帧	7	CANFD-...	x| C2 00 00 07 D0 00 05
Rx	0x301	标准帧	数据帧	12	CANFD-...	x| 09 E1 FD E5 FF 51 00 00 00 F0 01 C0

09 E1	13°	
FD E5	-539 RPM	
FF 51	-175 mA	
00 00	No error	
00 F0	24 ℃	
01	Profile Position Mode	
C0	11000000	Enabled, brake released, normal state, running

7. CANopen PDO Description

The PDO mapping for CANopen is not implemented; users need to perform the corresponding mapping.

Define the indexes and channels to be configured:

u8 User_pdo_1017[8] = {0x2B,0x17,0x10,0x00,0xE8,0x03,0x00,0x00}; //Slave heartbeat setting

u8 User_pdo_1801_01[8] = {0x23,0x01,0x18,0x01,0x81,0x02,0x00,0x80}; //Disable unused PDO channels

u8 User_pdo_1802_01[8] = {0x23,0x02,0x18,0x01,0x81,0x03,0x00,0x80}; //Disable unused PDO channels

u8 User_pdo_1803_01[8] = {0x23,0x03,0x18,0x01,0x81,0x04,0x00,0x80}; //Disable unused PDO channels

u8 User_pdo_1401_01[8] = {0x23,0x01,0x14,0x01,0x01,0x03,0x00,0x80}; //Disable unused PDO channels

u8 User_pdo_1402_01[8] = {0x23,0x02,0x14,0x01,0x01,0x04,0x00,0x80}; //Disable unused PDO channels

u8 User_pdo_1403_01[8] = {0x23,0x03,0x14,0x01,0x01,0x05,0x00,0x80}; //Disable unused PDO channels

u8 User_pdo_1800_02[8] = {0x2F,0x00,0x18,0x02,0x01,0x00,0x00,0x00}; //Set TPDO mode

u8 User_pdo_1A00_00_s[8] = {0x2F,0x00,0x1A,0x00,0x00,0x00,0x00,0x00}; //Clear TPDO1 mapping

u8 User_pdo_1A00_01[8] = {0x23,0x00,0x1A,0x01,0x10,0x00,0x41,0x60}; //Map 6041 Status word

u8 User_pdo_1A00_02[8] = {0x23,0x00,0x1A,0x02,0x10,0x00,0x78,0x60}; //Map 6078 Actual current

u8 User_pdo_1A00_03[8] = {0x23,0x00,0x1A,0x03,0x20,0x00,0x64,0x60}; //Map 6064 Actual position

u8 User_pdo_1A00_00_e[8] = {0x2F,0x00,0x1A,0x00,0x03,0x00,0x00,0x00}; //Save mapping

u8 User_pdo_1800_01[8] = {0x23,0x00,0x18,0x01,0x81,0x01,0x00,0x00}; //Map ID 0x181

u8 User_pdo_1400_02[8] = {0x2F,0x00,0x14,0x02,0x01,0x00,0x00,0x00}; //Set RPDO mode

u8 User_pdo_1600_00_s[8] = {0x2F,0x00,0x16,0x00,0x00,0x00,0x00,0x00}; //Clear RPDO1 mapping

u8 User_pdo_1600_01[8] = {0x23,0x00,0x16,0x01,0x10,0x00,0x40,0x60}; //Map 6040 Control word

u8 User_pdo_1600_02[8] = {0x23,0x00,0x16,0x02,0x10,0x00,0x71,0x60}; //Map 6071 Target current

u8 User_pdo_1600_03[8] = {0x23,0x00,0x16,0x03,0x20,0x00,0x7A,0x60}; //Map 607A Target position

u8 User_pdo_1600_00_e[8] = {0x2F,0x00,0x16,0x00,0x03,0x00,0x00,0x00}; //Save mapping

u8 User_pdo_1400_01[8] = {0x23,0x00,0x14,0x01,0x01,0x02,0x00,0x00}; //Map ID 0x201

u8 User_pdo_6060[8] = {0x2F,0x60,0x60,0x00,0x08,0x00,0x00,0x00}; //Set operation mode: 8 Synchronous cyclic mode

Example as follows

ID	Frame Type	Frame Format	Length	Data	Description
0x0701	Data Frame	Standard Frame	0x01	00	Power-on feedback
0x0000	Data Frame	Standard Frame	0x02	81 01	Start node
0x0601	Data Frame	Standard Frame	0x08	2B 17 10 00 C8 00 00 00	Configure heartbeat time
0x0581	Data Frame	Standard Frame	0x08	60 17 10 00 00 00 00 00	
0x0601	Data Frame	Standard Frame	0x08	23 00 18 01 81 01 00 80	Disable txPDO
0x0581	Data Frame	Standard Frame	0x08	60 00 18 01 00 00 00 00	Disable txPDO
0x0601	Data Frame	Standard Frame	0x08	23 01 18 01 81 02 00 80	Disable txPDO
0x0581	Data Frame	Standard Frame	0x08	60 01 18 01 00 00 00 00	Disable txPDO
0x0601	Data Frame	Standard Frame	0x08	23 02 18 01 81 03 00 80	Disable txPDO
0x0581	Data Frame	Standard Frame	0x08	60 02 18 01 00 00 00 00	Disable txPDO
0x0601	Data Frame	Standard Frame	0x08	23 03 18 01 81 04 00 80	Disable txPDO
0x0581	Data Frame	Standard Frame	0x08	60 03 18 01 00 00 00 00	Disable txPDO

ID	Frame Type	Frame Format	Length	Data	Description
0x0601	Data Frame	Standard Frame	0x08	23 00 14 01 01 02 00 80	Disable rxPDO
0x0581	Data Frame	Standard Frame	0x08	60 00 14 01 00 00 00 00	Disable rxPDO
0x0601	Data Frame	Standard Frame	0x08	23 01 14 01 01 03 00 80	Disable rxPDO
0x0581	Data Frame	Standard Frame	0x08	60 01 14 01 00 00 00 00	Disable rxPDO
0x0601	Data Frame	Standard Frame	0x08	23 02 14 01 01 04 00 80	Disable rxPDO
0x0581	Data Frame	Standard Frame	0x08	60 02 14 01 00 00 00 00	Disable rxPDO
0x0601	Data Frame	Standard Frame	0x08	23 03 14 01 01 05 00 80	Disable rxPDO
0x0581	Data Frame	Standard Frame	0x08	60 03 14 01 00 00 00 00	Disable rxPDO
0x0601	Data Frame	Standard Frame	0x08	2F 00 18 02 01 00 00 00	Set TPDO mode
0x0581	Data Frame	Standard Frame	0x08	60 00 18 02 00 00 00 00	
0x0601	Data Frame	Standard Frame	0x08	2F 00 1A 00 00 00 00 00	Clear txPDO1
0x0581	Data Frame	Standard Frame	0x08	60 00 1A 00 00 00 00 00	
0x0601	Data	Standard	0x08	23 00 1A 01 10	Map 6041 Status

ID	Frame Type	Frame Format	Length	Data	Description
	Frame	Frame		00 41 60	Word
0x0581	Data Frame	Standard Frame	0x08	60 00 1A 01 00 00 00 00	
0x0601	Data Frame	Standard Frame	0x08	23 00 1A 02 10 00 78 60	Map 6078 Actual Current
0x0581	Data Frame	Standard Frame	0x08	60 00 1A 02 00 00 00 00	
0x0601	Data Frame	Standard Frame	0x08	23 00 1A 03 20 00 64 60	Map 6064 Actual Position
0x0581	Data Frame	Standard Frame	0x08	60 00 1A 03 00 00 00 00	
0x0601	Data Frame	Standard Frame	0x08	2F 00 1A 00 03 00 00 00	Save mapping
0x0581	Data Frame	Standard Frame	0x08	60 00 1A 00 00 00 00 00	
0x0601	Data Frame	Standard Frame	0x08	23 00 18 01 81 01 00 00	Map ID 0x181
0x0581	Data Frame	Standard Frame	0x08	60 00 18 01 00 00 00 00	
0x0601	Data Frame	Standard Frame	0x08	2F 00 14 02 01 00 00 00	Set RPDO mode
0x0581	Data Frame	Standard Frame	0x08	60 00 14 02 00 00 00 00	
0x0601	Data Frame	Standard Frame	0x08	2F 00 16 00 00 00 00 00	Clear rxPDO1

ID	Frame Type	Frame Format	Length	Data	Description
0x0581	Data Frame	Standard Frame	0x08	60 00 16 00 00 00 00 00	
0x0601	Data Frame	Standard Frame	0x08	23 00 16 01 10 00 40 60	Map 6040 Control Word
0x0581	Data Frame	Standard Frame	0x08	60 00 16 01 00 00 00 00	
0x0601	Data Frame	Standard Frame	0x08	23 00 16 02 10 00 71 60	Map 6071 Target Current
0x0581	Data Frame	Standard Frame	0x08	60 00 16 02 00 00 00 00	
0x0601	Data Frame	Standard Frame	0x08	23 00 16 03 20 00 7A 60	Map 607A Target Position
0x0581	Data Frame	Standard Frame	0x08	60 00 16 03 00 00 00 00	
0x0601	Data Frame	Standard Frame	0x08	2F 00 16 00 03 00 00 00	Save mapping
0x0581	Data Frame	Standard Frame	0x08	60 00 16 00 00 00 00 00	
0x0601	Data Frame	Standard Frame	0x08	23 00 14 01 01 02 00 00	Map ID 0x201
0x0581	Data Frame	Standard Frame	0x08	60 00 14 01 00 00 00 00	
0x0601	Data Frame	Standard Frame	0x08	2F 60 60 00 07 00 00 00	Set operation mode to CSP
0x0581	Data	Standard	0x08	60 60 60 00 00	

ID	Frame Type	Frame Format	Length	Data	Description
	Frame	Frame		00 00 00	
0x0701	Data Frame	Standard Frame	0x01	7F	
0x0000	Data Frame	Standard Frame	0x02	01 00	Start node
0x0080	Data Frame	Standard Frame	0x00	-	Sync frame

8. Encoder Description

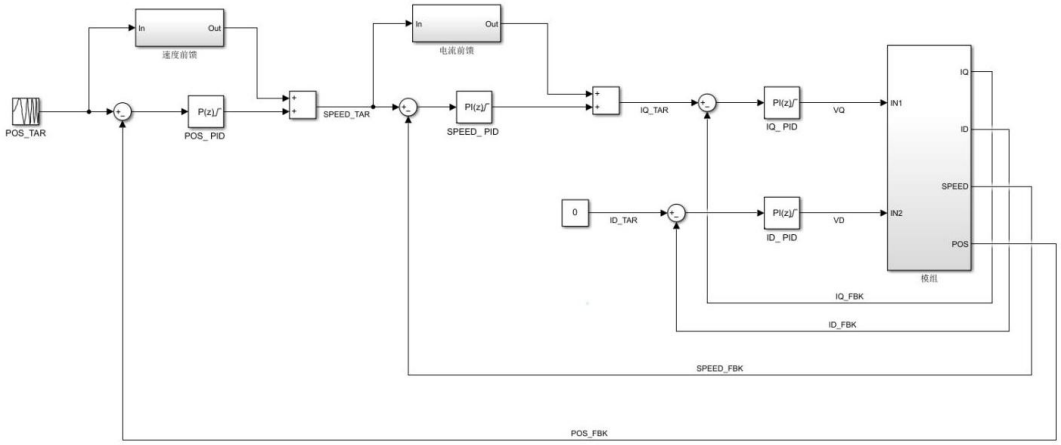
1. Encoder Resolution and Single-Cycle Position Feedback

The joint module employs a dual absolute single-turn encoder for full closed-loop control: the motor-side encoder has a 14-bit resolution (16,384 counts per turn), while the load-side encoder offers a 16-bit resolution (65,536 position values per turn). Resolution indicates the number of discrete positions output per turn—for example, the load-side 16-bit encoder provides position feedback ranging from 0 to 65,535. When the axis crosses the zero position, a numerical jump occurs (forward zero-crossing: 65,535 → 0; reverse zero-crossing: 0 → 65,535).

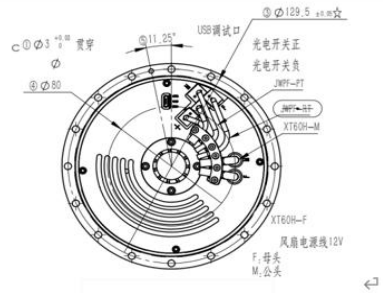
The 16-bit absolute encoder exhibits an angle error of ≤±0.5 and a repeatability accuracy of ≤±0.05. Additionally, position jump errors during zero-crossing transitions (e.g., 65,535→0) require additional compensation to prevent instantaneous disturbances in full closed-loop control.

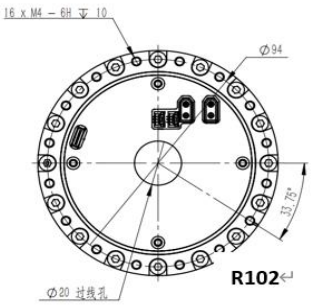
The calculation formula for the encoder position value corresponding to the single-turn angle θ is: Position value = (θ/360) × 65,536 (Example: 20 corresponds to a position value of (20/360) × 65,536 ≈ 3,641). The encoder has 4,096 line

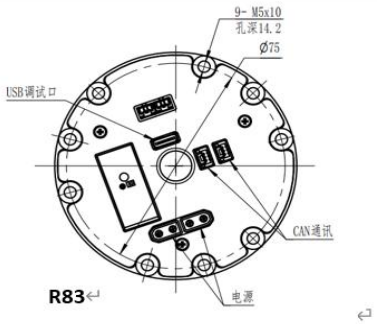
位置反馈示意图

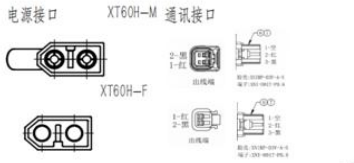
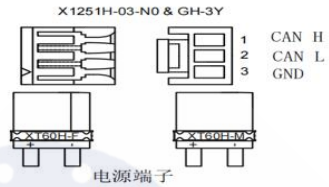
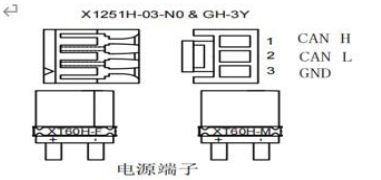
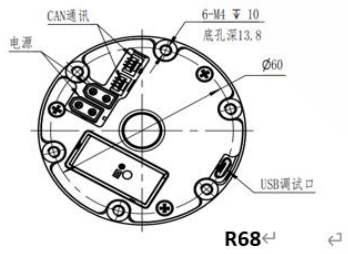

In
Out
速度前馈
In
Out
电流前馈
POS_TAR
P(z)/
POS_ PID
SPEED_TAR
PI(z)/
SPEED_ PID
IQ_TAR
PI(z)/
IQ_ PID
VQ
IN1
IQ
ID
SPEED
POS
IN2
0
ID_TAR
PI(z)/
ID_ PID
VD
模组
IQ_FBK
ID_FBK
SPEED_FBK
POS_FBK

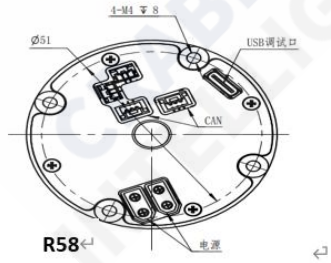
9. Module Terminal Description

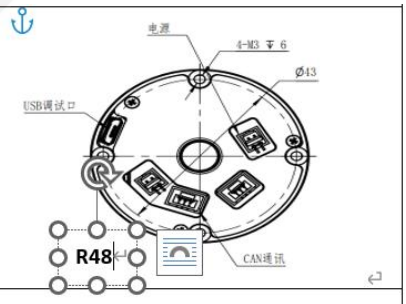
<table>
<tr>
<td> 

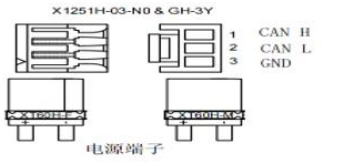
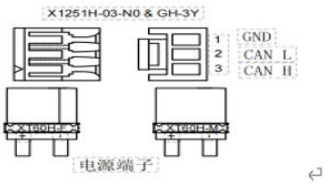
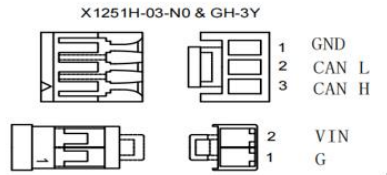
R120&R120MAX</td>
<td> 

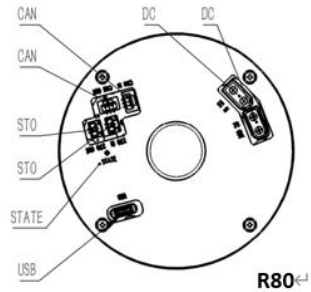
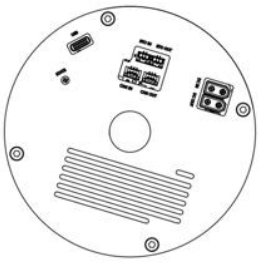
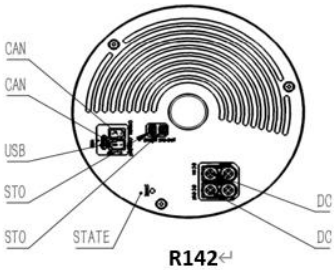
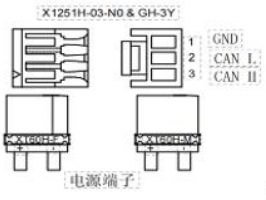
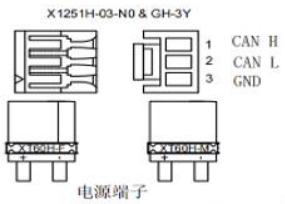
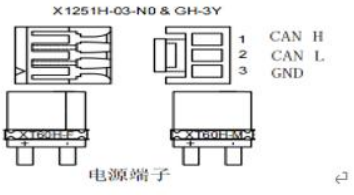
R102</td>
<td> 

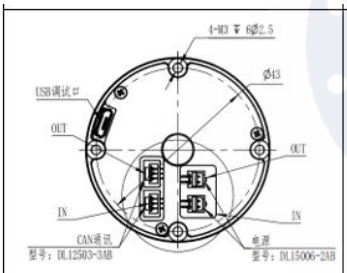
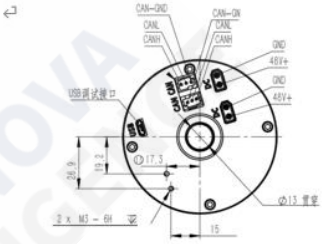
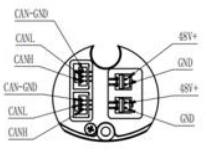
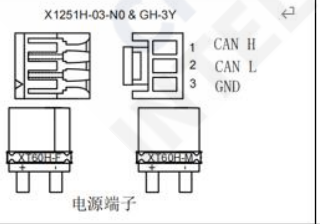
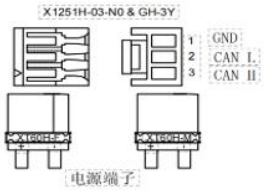
R83</td>
</tr>
<tr>
<td> 
</td>
<td> 
</td>
<td> 
</td>
</tr>
<tr>
<td>电源端子：XT30AW-M.G.Y
品牌：AMASS
pin 定义:1、G· · 2、VIN
通信端子：X1251WVS-03HF-LPSW
品牌：XKB Connection(中国星坤)
pin 定义:1、H· · 2、L· · 3、G</td>
<td>电源端子：XT30AW-M.G.Y
品牌：AMASS
pin 定义:1、G· · · 2、VIN
通信端子：X1251WVS-03HF-LPSW
品牌：XKB Connection(中国星坤)
pin 定义:1、H· · 2、L· · 3、G</td>
<td>电源端子：XT30AW-M.G.Y
品牌：AMASS
pin 定义:1、G· · · 2、VIN
通信端子：B03B-XNISK-A-1(LF)(SN)
品牌：JST
pin 定义:1、H· · 2、L· · 3、G</td>
</tr>
<tr>
<td> 

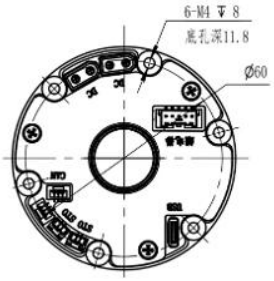
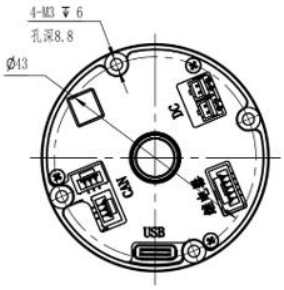
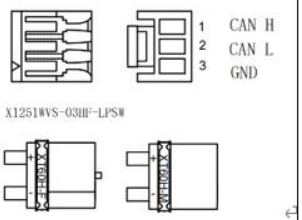
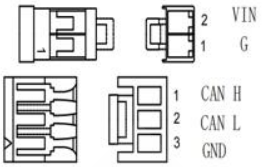
R68</td>
<td> 

R58</td>
<td> 

R48</td>
</tr>
<tr>
<td> 
</td>
<td> 
</td>
<td> 
</td>
</tr>
<tr>
<td>电源端子：XT30AW-M.G.Y
品牌：AMASS
pin 定义:1、G· · · 2、VIN
通信端子：X1251WVS-03HF-LPSW
品牌：XKB Connection(中国星坤)
pin 定义:1、H· · 2、L· · 3、G</td>
<td>电源端子：DL15006-2AB
品牌：德利来
pin 定义:1、G· · · 2、24+
通信端子：DL12503-3AB
品牌：德利来
pin 定义:1、G· · · 2、L· · 3、H</td>
<td>电源端子：B02B-XNISK-A-1(LF)(SN)
品牌：JST
pin 定义:1、G· · · 2、VIN
通信端子：B03B-ZESK-1D(LF)(SN)
品牌：JST
pin 定义:1、G· · · 2、L· · 3、H</td>
</tr>
</table>

 R80	 R110	 R142
		
电源端子：XT30U-M 母头 品牌：AMASS pin 定义:1、G···2、24+ 通信端子：DL12503-3AB 品牌：德利来 pin 定义:1、G···2、L··3、H	电源端子：XT30AW-M.G.Y 品牌：AMASS pin 定义:1、G···2、VIN 通信端：B03B-XNISK-A-1(LF)(SN) 品牌：JST pin 定义:1、H··2、L··3、G	电源端子：XT30AW-M.G.Y 品牌：AMASS pin 定义:1、G···2、VIN 通信端子：X1251WVS-03HF-LPSW 品牌：XKB Connection(中国星坤) pin 定义:1、H··2、L··3、G

 R48-101-DNF	 R83-101-FBN	 R80-50-13NSFN
		
端子：B02B-XNISK-A-1(LF)(SN) 品牌：JST pin 定义:1、G···2、VIN 通信端子：B03B-ZESK-1D(LF)(SN) 品牌：JST pin 定义:1、G···2、L··3、H	电源端子：XT30AW-M.G.Y 品牌：AMASS pin 定义:1、G···2、VIN 通信端子:B03B-XNISK-A-1(LF)(SN) 品牌：JST pin 定义:1、H··2、L··3、G	电源端子：XT30U-M 母头 品牌：AMASS pin 定义:1、G···2、24+ 通信端子：DL12503-3AB 品牌：德利来 pin 定义:1、G···2、L··3、H

 R68-101-lite	 R48-101-lite	
		
电源端子：XT30AW-M.G.Y 品牌：AMASS pin 定义:1、G 2、VIN 通信端子：X1251WVS-03HF-LPSW 品牌：XKB Connection(中国星坤) pin 定义:1、H 2、L 3、G	端子型号：电源 DL15006-2AB 通信 DL12503-3AB 调试 HC-0.8-3PLT 通信端子：HC-0.8-6PLT 品牌：HCTL(华灿天禄) pin 定义：1、G 2、L 3、H 4、G 5、L 6、H	

10. Troubleshooting Guide

For motor error or abnormality alarms, refer to the following error codes.

报错内容	触发条件	保护方式	清除错误方式	报错代码	说明
欠压	1、母线电压<40V	快速保护	重启清除	0x0002	硬件欠压保护为35V
过压	1、母线电压>72V	一般保护	重启清除	0x0001	
过温	1、电机线圈NTC温度>95°（持续100ms） 2、驱动器NCT温度 >100°（持续100ms） 电机与驱动 温度大于80°，做上升斜率保护；	一般保护	恢复到85°以下指令清除	0x0004	
堵转	1、电机Iq电流≥设置最大电流，持续2S	限幅保护	重启清除	0x0008	
过载	1、电机的Iq电流积分大于设置阈值	一般保护	重启清除	0x0010	
速度跟踪误差过大	1、期望速度与实际速度误差>320RPM 持续20ms 只针对120&120MAX，其他模组按照200RPM阈值设置；	快速保护	重启清除	0x0020	负载突变的情况会触发，负载缓慢上升会触发堵转与速度误差过大
正限位保护	1、限位的光电开关触发或者超过设置的软件角度限位 2、目标速度的方向与限位方向一致 软件正限位默认参数：32767； 软件负限位默认参数：-32767；	快速保护	指令清除，向限位反方向可以运行	0x0040	
负限位保护				0x0080	
输出端编码器异常	1、通过输出端编码器计算速度与速度端比较；当输出端编码器计算速度为0，速度端为100RPM持续20ms；反向对比；	快速保护	重启清除	0x0100	
位置跃迁过大	1、CSP模式下发送的目标位置Δ>100nct（发送周期为1ms）	伺服在当前位置不响应	指令清除	0x0200	
电流采样错误	1、初始化判断电流采样的基准是否在合理范围	快速保护	重启清除	0x0400	
电流跟踪误差超差	1、电流的目标值与反馈值误差>10A 持续20ms 阈值设置按照额定电流；	快速保护	重启清除	0x0800	缺相可能触发，电机参数设置异常；
位置跟踪误差过大	1、期望位置与实际误差误差>2000cnt 持续20ms	快速保护	重启清除	0x1000	

Protection Level Description

保护等级	保护方式说明
一般保护	1、速度减速到设置阈值 2、抱闸吸合 3、100ms等待 4、关PWM
快速保护	1、关PWM 2、抱闸吸合
限幅保护	1、限幅到最大电流的二分之一 2、抱闸吸合 3、100ms等待 4、关PWM