



iSMD low-voltage servo integrated motor User manual

Preface

Thank you for using Kinco servo products! iSMD servo drive are a new generation of small size, good performance and high stability products developed by Kinco after market research and according to the characteristics of the automation industry. iSMD support CAN, Modbus, EtherCAT, Profinet control methods, and have standard STO function, widely used in logistics storage equipment, mobile handling equipment, sorting cars, mobile service robots and other fields. For customers who have special requirements in communication mode, installation mode, protection level, etc., we also provide customized servo drives and motors for various application scenarios.

Please read the manual carefully and follow the operation requirements in the manual, which will help you correctly set up the drive to achieve the best performance.

Confirmation

- Please confirm whether the product is damaged during transportation
- Please confirm the motor model, drive model, and motor wiring model are consistent with the model you ordered according to the nameplate information on the drive and motor.
- Please confirm whether the terminal accessories of the drive is complete, and the motor oil seal and keys are complete on the drive.

If there is any problem with any of the above, please contact Kinco or your supplier to solve it.

Part list

Accessory package	Name	Model	Count
iSMD 60D/80D accessory package(IP20)	Communication connector X1 (AA/AS model)	F7GM-3.5-16P-7	1
	Communication connector X1 (EA/ES/PA/PS model)	M8-P4D-PWA05	2
	IO connector X3A	F7GM-3.5-16P-A	1
	Battery connector X3D	F7EMM-3.5-02P-A-100	1
	Power supply connector X5	F9AP-7.5-04P-A	1
	STO connector X4	F7GM-3.5-06P-4	1
	Terminal resistance	iSMD-R120 (Note: This is a free accessory. Please contact sales if you would like to receive one.)	1
iSMD 60D/80D accessory package(IP65)	Communication aviation plug X1 (EA/ES/PA/PS model)	M8-P4D-PWA05	2
	DI aviation plug X3A&X3B	M8-P4A-PLA05	2
	DO aviation plug X3C	M8-P3A-PLA05	1
	STO aviation plug X4	M8-P6A-PWA05	1
	Battery aviation plug X3D	M8-S3A-PLA05	1
	60 motor power supply aviation plug X5	M12-S5L-PLDP9	1
	80 motor power supply aviation plug X5	M923-S6-GCA15	1



Note

- Different models have different function and are equipped with different connector.

Safety attentions

Security Definition

The following safety symbols are used in the safety-related contents of this manual. Please observe them to avoid injury.



Danger

Wrong operation could lead to death or serious injury.



Warning

Risk of minor injury or serious property damage.



Note

Information marked for attention helps ensure proper product operation.



Remind

Suggestion, reminding, or reference linking

Security claims

- **Operating qualifications**

This product must be operated by trained professionals. In addition, the operator must go through professional skills training, familiar with the installation, wiring, operation and maintenance of the equipment, and correctly respond to various emergency situations in use.

- **Unpacking and acceptance**



Note

- Before unpacking, first check whether the appearance of the package is normal and intact, without obvious scratches, extrusion deformation, damage, or be affected with damp. For safety reasons, do not use products with damaged packaging or parts.
- Please verify that the model number and quantity match the order information as per the order list inside the box and the nameplate information on the product.
- Please confirm whether all terminal accessories are complete and there are no defects or rust on the surface of the product.

● Storage and transportation



Warning

- When handling, please handle with care. Do not stack the products too high to prevent product damage and personal injury caused by product fall.
- If the drive is not used for a long time, it should be confirmed whether there is water or dew inside the product before use, otherwise there is the risk of burning the drive. Please carry out closer protection and necessary inspection.
- If the driver is placed in a low temperature environment for a long time, place it at a temperature higher than 0 °C for a period of time before use to avoid damage to the driver.
- Please store the product in accordance with the specifications and avoid shock and vibration of the driver during transportation and storage, otherwise it will cause the risk of product damage.



Remind

- If the drive is not intended to be used for a long time, it is recommended that the drive be powered on every few months to prevent damage to internal components due to moisture. At the same time, the drive should be more tightly protected storage and pre-use inspection.
- When the product needs to be transported over long distances, it is necessary to pack the product into a closed box before it can be transported.

● Installation and operation



Warning

- Install in a place that can support the weight of the drive, otherwise there is a risk of injury or damage to property when dropped.
- Ensure that the driver is securely installed and screws are tightened. Pay attention to sharp edges and corners of the driver to avoid injuries or cuts caused by falling.
- The product should be installed in a place that is not easy to burn, to prevent the invasion of dust, corrosive gases, conductive objects, liquids and flammable substances, and to maintain good heat dissipation conditions;
- Ensure that all protective measures are activated before powering on the product.
- Do not touch the shell, brake resistance and other parts to test the temperature during the operation of the product, otherwise there is the risk of burning.
- In the application of dust, metal debris, oil pollution serious, should use a good sealing equipment electrical cabinet.
- The static electricity on the human body may seriously damage the sensitive components inside. Before performing related operations, observe the measures and methods specified in ESD. Otherwise, the drive may be damaged.
- It is forbidden to modify the drive without permission. If the drive is damaged due to private modification, the Department will not be responsible for it!



Danger

- If the surface of the product is damaged or the parts are incomplete, do not install and operate, otherwise there is a risk of fire and injury.
- Foreign objects such as screws, metal gaskets, and metal rods falling into the drive pose a fire hazard and property damage.
- Wiring operations must be carried out by qualified personnel, otherwise there is a risk of electric shock.
- Ensure the input power is completely disconnected before wiring ,otherwise, electric shock may occur.
- The ground terminal of the driver must be grounded reliably; otherwise, electric shock may occur.
- RB+ and RB- terminals are used to connect brake resistors. Do not short circuit them, otherwise they may cause damage or fire.
- This product controls a potentially dangerous movement. Failure to comply with the regulations or perform operations in accordance with the requirements of this manual may result in personal injury, injury, and damage to the product and associated systems.
- Before powering on the device, ensure that the power supply meets requirements. Otherwise, the device may be damaged or a fire may occur.
- When energized, do not touch the product and terminal, do not disassemble the parts of the product, otherwise there is the risk of electric shock.
- Maintenance operations should be performed 10 minutes after the power is disconnected; otherwise, there is a risk of electric shock.
- Only professional personnel can replace the parts, it is strictly prohibited to leave the wire or metal in the machine, otherwise there is the risk of fire.



Remind

- Please use the cables specified by our company whenever possible when wiring. If the cable is not specified by the kinco, please purchase the specified wiring material or equivalent after confirming the rated current or use environment of the model used.
- The product is strictly prohibited from operating beyond the set parameters, being modified or disassembled privately, being wired in violation of regulations, or being overloaded.

● Maintain



Danger

- It is strictly prohibited for non-professionals to carry out equipment installation, wiring, maintenance, inspection or component replacement.
- It is strictly prohibited to perform equipment maintenance while it is powered on. Otherwise, there is a risk of electric shock!
- When a product malfunctions or cannot be used normally, it must be sent back to the manufacturer for repair. Users are not allowed to disassemble the product without authorization.

Manual version revision record

Version	Date	Update content
V1.0	2026-09	Match the KSPRO software

Content

Preface	1
Confirmation	1
Part list	1
Safety attentions	2
Manual version revision record	5
Chapter 1 Product description	14
1.1 Product information	14
1.1.1 Naming rule	14
1.1.2 Nameplate description	14
1.1.3 Part description	15
1.1.4 Dimension description	18
1.2 Product specification parameter	26
1.3 Overload characteristic	28
1.4 Drive list used in iSMD integrated motor	28
Chapter 2 Accessory description	30
2.1 Cable description	30
2.2 External installation instruction	32
Chapter 3 Product installation	35
3.1 Product installation	35
3.1.1 Use requirement	35
3.1.2 Precaution	36
3.1.3 iSMD installation method	37
3.1.4 Safety operation and inspection	37
3.1.5 Oil seal installation instruction	38
3.2 EMC configuration description	39
Chapter 4 Wiring terminal	41
4.1 Interface terminal description	41
4.1.1 iSMD IP20 model	41
4.1.1.1 Bus communication interface (X1)	41
4.1.1.2 PC debugging interface (X2)	41
4.1.1.3 IO interface and battery interface (X3A&X3D)	42
4.1.1.4 STO interface (X4)	43

4.1.1.5 Power supply and braking resistor interface (X5)	43
4.1.2 iSMD IP65 model	43
4.1.2.1 Bus communication interface (X1)	43
4.1.2.2 PC debugging interface (X2)	43
4.1.2.3 IO interface and battery interface (X3A&X3B&X3C&X3D)	44
4.1.2.4 STO interface (X4)	45
4.1.2.5 Power supply interface (X5)	45
4.1.2.6 iSMD Rotary DIP switch	45
4.2 STO function description	48
4.2.1 Failure-safety of iSMD drive equipment	48
4.2.2 Implementation	48
4.2.3 Requirements	49
Chapter 5 System wiring	53
5.1 Electrical wiring diagram	53
5.1.1 iSMD IP20 model electrical wiring diagram	53
5.1.2 iSMD IP65 model electrical wiring diagram	54
5.1.2.1 Digital input signal	54
5.1.2.2 Digital output signal	55
5.1.3 STO electrical wiring diagram	56
5.2 External wiring diagram	57
5.3 Power supply description	60
Chapter 6 Kincoservo PRO software	61
6.1 Software installation	61
6.1.1 Software acquisition	61
6.1.2 Software installation	62
6.1.3 Version update	63
6.2 Quick start guide	64
6.2.1 Software window introduction	64
6.2.2 Connect to the computer	64
6.2.3 Save & Close project	67
6.2.4 Parameter monitoring and management	67
6.2.4.1 Parameter modification	68
6.2.4.2 Parameter monitoring	69
6.2.4.3 Export parameter file	70
6.2.4.4 Import parameter file	71

6.2.4.5 Bulk actions	72
6.2.4.6 Compare	73
6.2.4.7 Param Groups	74
6.2.5 Digital input and output setting	75
6.2.5.1 Digital input	76
6.2.5.2 Digital output	78
6.2.6 Motor configuration	79
6.2.7 Scope	83
6.2.7.1 Sampling settings and execution	84
6.2.7.2 Start sampling	87
6.2.7.3 Waveform display settings	87
6.2.7.4 Cursor	89
6.2.7.5 Waveform export and view historical waveform	90
6.2.7.6 Waveform import	91
6.2.8 Download FW	91
6.2.9 Fault manage	92
Chapter 7 Commissioning	96
7.1 Pre-operation preparation	96
7.1.1 Cable connection and environment check	96
7.2 Commissioning	96
Chapter 8 Servo basic function	98
8.1 Server status	98
8.1.1 Server status	98
8.1.2 Control word and status word description	100
8.1.2.1 Control word description 604000	100
8.1.2.2 Status word description 604100	102
8.2 Speed control mode (-3, 3)	103
8.2.1 Speed mode function block diagram	103
8.2.2 Configuration process and panel operation	104
8.2.3 KS PRO operation	106
8.2.4 IO function related to speed mode	110
8.3 Position control mode (1)	112
8.3.1 Position mode function block diagram	112
8.3.2 Configuration process and panel operation	113
8.3.3 KS PRO operation	115

8.3.4 IO function related to position mode	118
8.4 Torque control mode (4)	120
8.4.1 Torque mode function block diagram	120
8.4.2 Configuration process	120
8.4.3 KS PRO operation	122
8.4.4 IO function related to torque mode	125
8.5 Homing mode (6)	126
8.5.1 Homing mode function block diagram	126
8.5.2 Configuration process	127
8.5.3 KS PRO operation	134
8.5.4 IO function related to homing mode	135
8.6 Pulse mode (-4)	136
8.6.1 Pulse mode function block diagram	137
8.6.2 Configuration process and panel operation	138
8.6.3 KS PRO operation	140
8.6.4 Encoder pulse output	142
8.6.5 IO function related to pulse mode	143
Chapter 9 Servo application function	144
9.1 Absolute value encoder motor configuration	144
9.1.1 Multi-turn motor power-on configuration	144
9.1.2 Multi-turn data range	145
9.1.3 Multi-turn prohibition	145
9.1.4 Battery description	146
9.2 Limit function	147
9.2.1 Limit function configuration	147
9.2.2 IO function about limit	149
9.3 Motor brake control	149
9.3.1 Brake control configuration	150
9.3.2 IO function about brake control	152
9.4 Stop mode control	153
9.4.1 Brake control configuration	153
9.4.2 IO function about stop mode	156
9.5 Field weakening control	156
9.5.1 Field weakening control configuration	156
9.6 Fast capture and high-speed probe	157

9.6.1 Fast capture configuration	157
9.6.2 High-speed probe configuration	158
Chapter 10 Performance adjustment	161
10.1 Online auto-tuning	161
10.2 Manual parameter tuning	161
10.2.1 Operation process and related parameters	162
Step1: Speed loop setting method	162
Step2: Position loop setting method	168
10.3 Gain switch	171
10.3.1 Switch PI through a digital input port	172
10.3.2 Auto PI switch	172
10.3.3 Directly set the PI pointer value	173
10.4 Other factors affecting performance	173
Chapter 11 List of common object parameters	174
Object list	174
11.1 Mode and control (0x6040)	176
11.2 Metrical data	177
11.3 Target object (0x607A)	177
11.4 Multi-segment position/Multi-segment speed (0x2020)	178
11.5 Performance object (0x6065)	179
11.6 Homing control (0x6098)	180
11.7 Speed loop parameter (0x60F9)	180
11.8 Position loop parameter (0x60FB)	180
11.9 Input/output port parameter (0x2010)	181
11.10 Pulse input parameters (0x2508)	182
11.11 Parameter for storage (0x2FF0)	183
11.12 Error code (0x2601)	183
11.13 Stop mode	184
11.14 Alarm parameter	185
11.15 Other parameter	185
Chapter 12 Alarm exclusion	186
Chapter 13 Communication	190
13.1 RS485 communication	190
13.1.1 RS485 cable connection	190

13.1.2 RS485 communication parameters list	190
13.1.3 MODBUS RTU protocol	191
13.1.4 RS232 communication protocol	193
13.1.5 Troubleshooting measures for communication faults	197
13.2 CANopen communication	198
13.2.1 Hardware description	198
13.2.2 Software description	199
13.2.2.1 EDS description	199
13.2.2.2 SDO description	200
13.2.3 PDO description	204
13.2.3.1 PDO COB-ID description	204
13.2.3.2 COB-ID	205
13.2.3.3 PDO transmission type	206
13.2.3.4 Protection mode/monitoring type description	207
13.2.3.5 Startup process description	209
13.2.3.6 Emergency message description	210
13.2.4 CANopen Bus communication setting	211
13.2.5 Interpolation mode based on CANopen	212
13.3 EtherCAT Communication	214
13.3.1 EtherCAT protocol	214
13.3.1.1 Overview	214
13.3.1.2 EtherCAT specification	215
13.3.1.3 EtherCAT system	216
13.3.1.4 EtherCAT distributed clock	217
13.3.1.5 EtherCAT state machine	219
13.3.2 CoE protocol	221
13.3.2.1 CoE object dictionary	221
13.3.2.2 Process data communication	222
13.3.2.3 Mailbox data communication	224
13.3.2.4 DS402 guild regulation	224
13.3.3 Application cases	227
13.3.3.1 Communication parameter settings	227
13.3.3.2 Kinco AX500 controller application	228
13.3.3.3 TwinCAT3 application	238
13.3.3.4 OMRON NJ series controller application	249
13.4 Profinet communication	264

13.4.1 Profinet introduction	264
13.4.1.1 Supported message	265
13.4.1.2 I/O data signal	267
13.4.1.3 Control word and status word	268
13.4.2 Application description	273
13.4.3 Acyclic communication	274
13.4.3.1 Drive IP and device name setting	274
13.4.3.2 Project configuration	275
13.4.3.3 SINA_PARA_S (FB287) introduction	277
13.4.3.4 FB287 read-write parameter	277
13.4.3.5 Read-write parameters using WRREC and RDREC	278
13.4.3.6 Fault diagnosis function	280
13.4.4 Application class1	281
13.4.4.1 Overview	281
13.4.4.2 Project configuration	281
13.4.4.3 SINA_SPEED (FB285) introduction	284
13.4.5 Application class3	285
13.4.5.1 Overview	285
13.4.5.2 Project configuration	285
13.4.5.3 SINA_POS (FB284) introduction	288
13.4.5.4 FB284 function description	289
13.4.6 Application class4	295
13.4.6.1 Overview	295
13.4.6.2 Project configuration	295
13.4.6.3 MC_Home introduction	301
13.4.6.4 MC_TorqueLimiting introduction	308
13.4.6.5 DSC (Dynamic servo control) introduction	309
13.4.7 S7 200 SMART application	310
13.4.7.1 Application description	310
13.4.7.2 Acyclic communication	311
13.4.7.3 Telegram 1 application	316
13.4.7.4 Telegram 111 application	319
13.4.8 Additional telegram 750 application	328
13.4.8.1 Application description	328
13.4.8.2 Telegram 1 + additional telegram 750	328
13.4.8.3 Telegram 111+ additional telegram 750	331

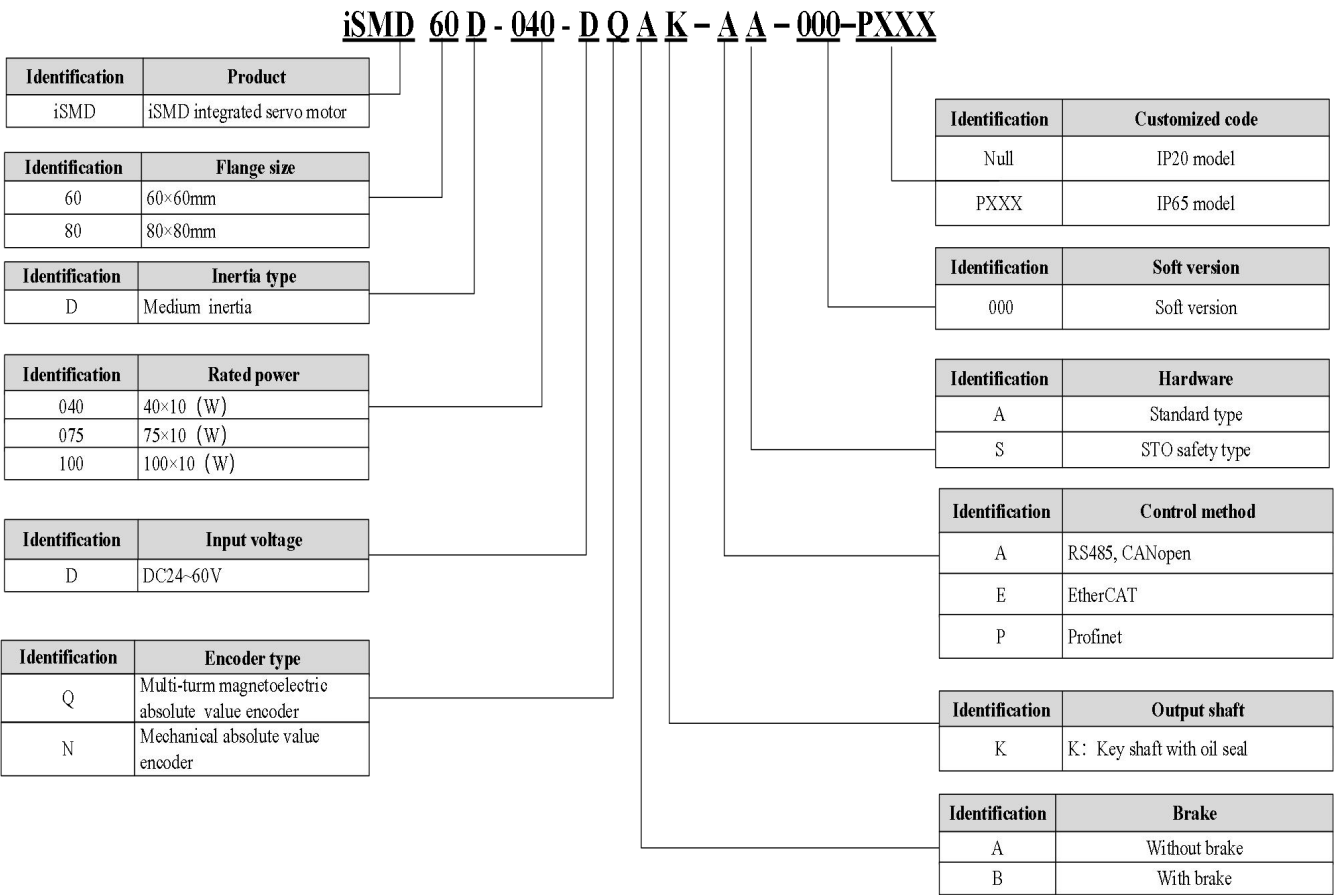
13.4.8.4 Telegram 105 + additional telegram 750 334

Appendix I Commonly used formulas 339

Chapter 1 Product description

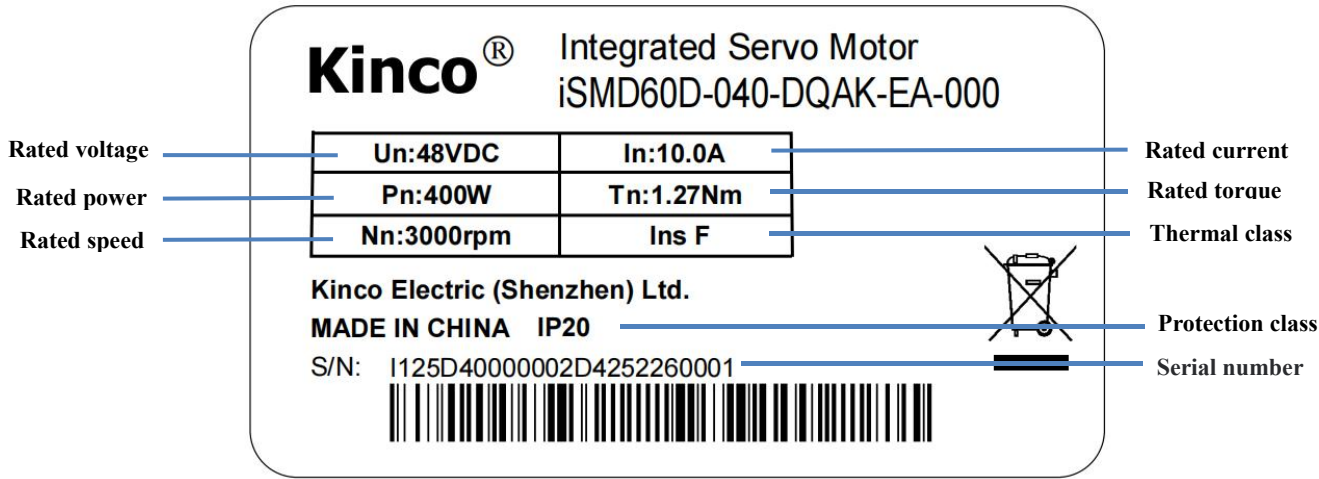
1.1 Product information

1.1.1 Naming rule



Note: The IP20 model supports pulse function, the IP65 model does not support pulse function .

1.1.2 Nameplate description

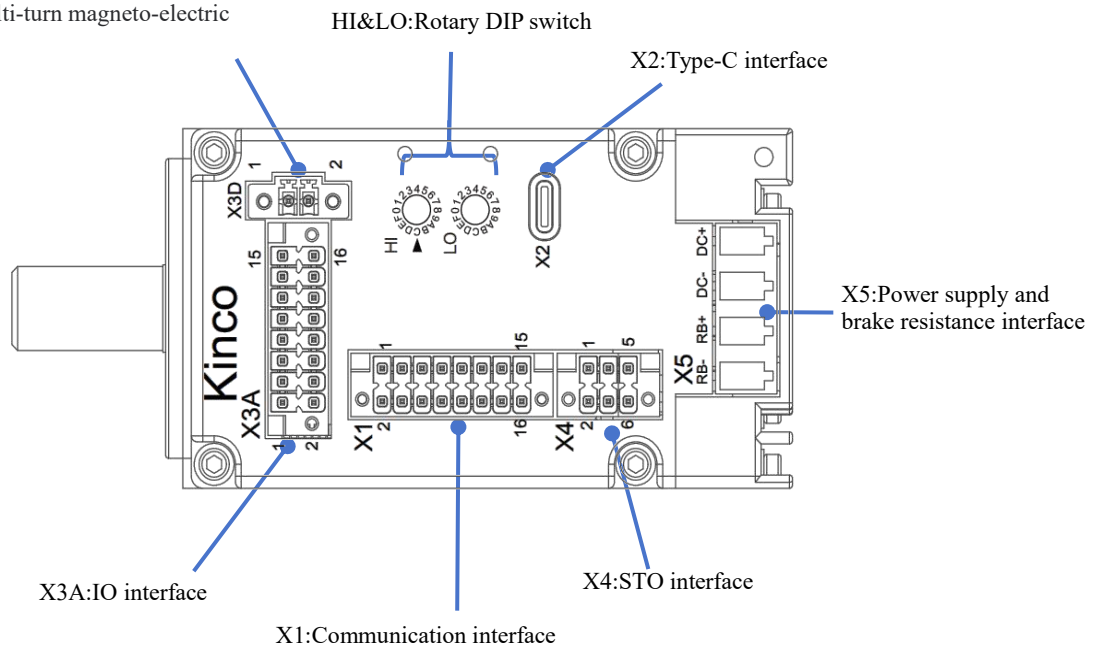


1.1.3 Part description

IP20-AS model (Only applicable to model with STO function)

X3D: Battery interface

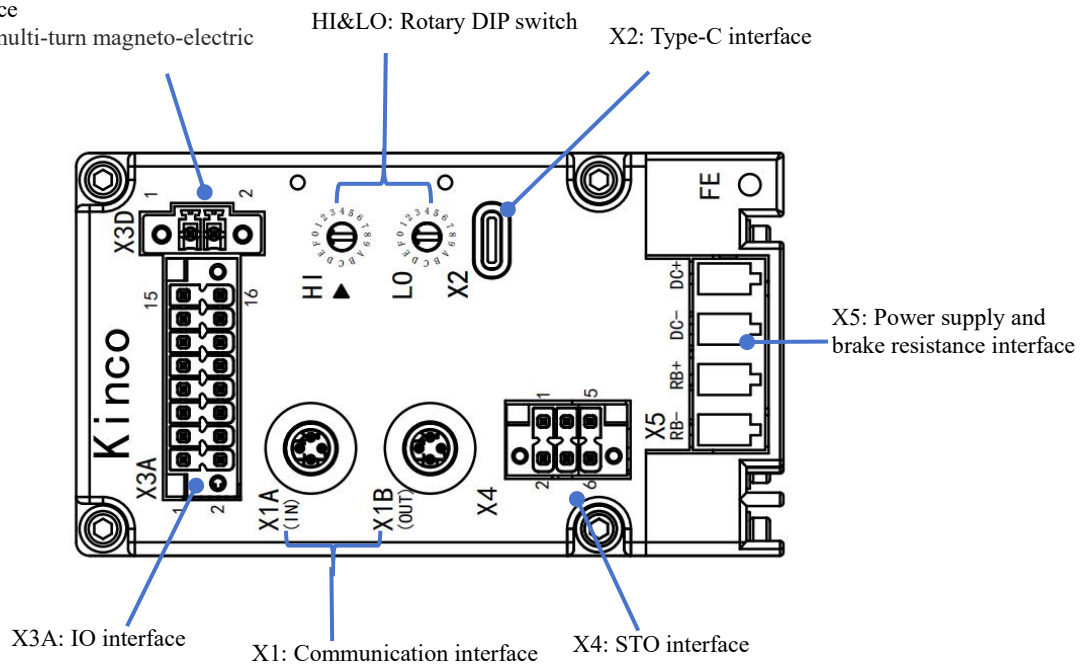
(Only applicable to multi-turn magneto-electric encoders)

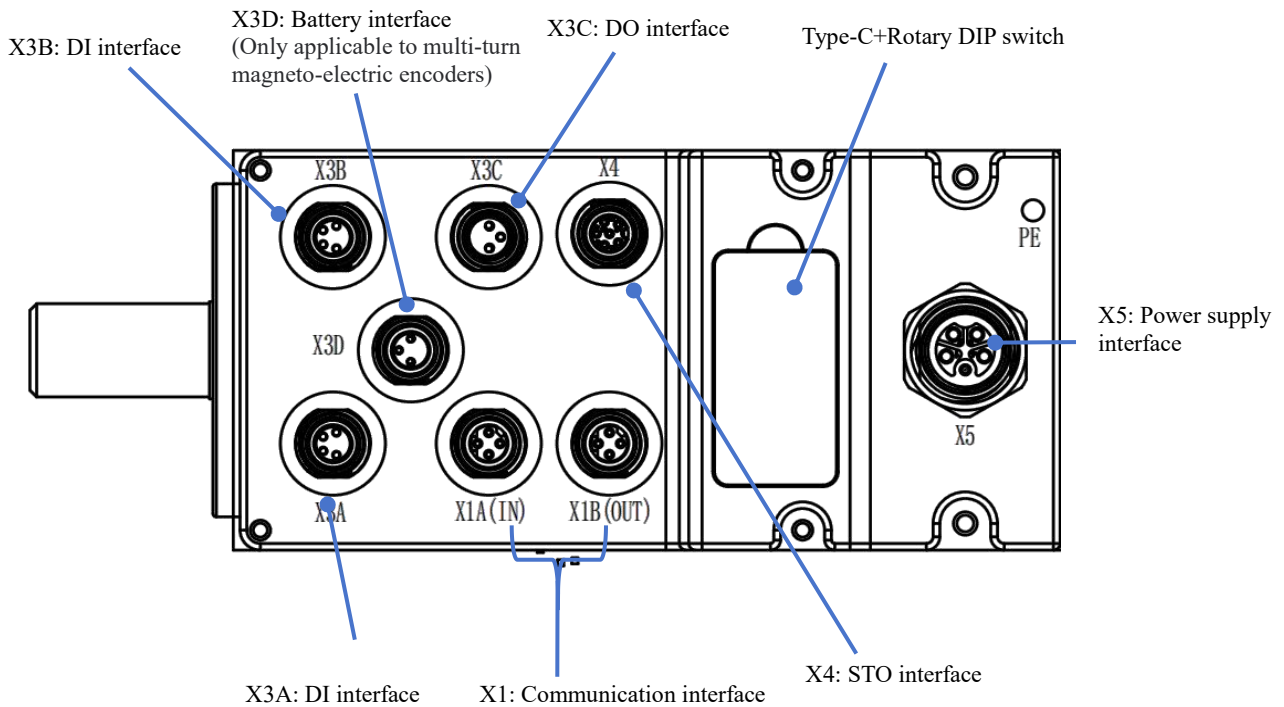


IP20-ES /PS model (Only applicable to model with STO function)

X3D: Battery interface

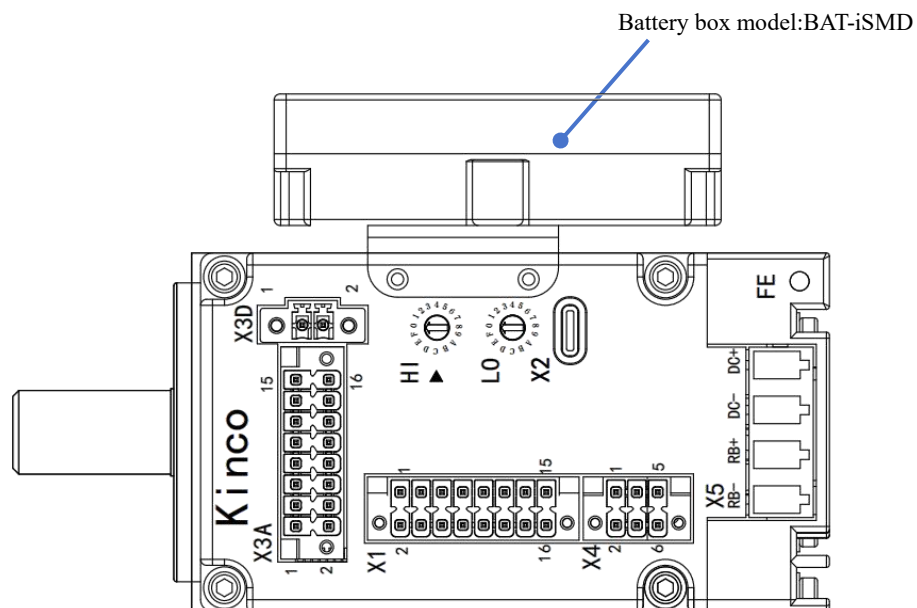
(Only applicable to multi-turn magneto-electric encoders)



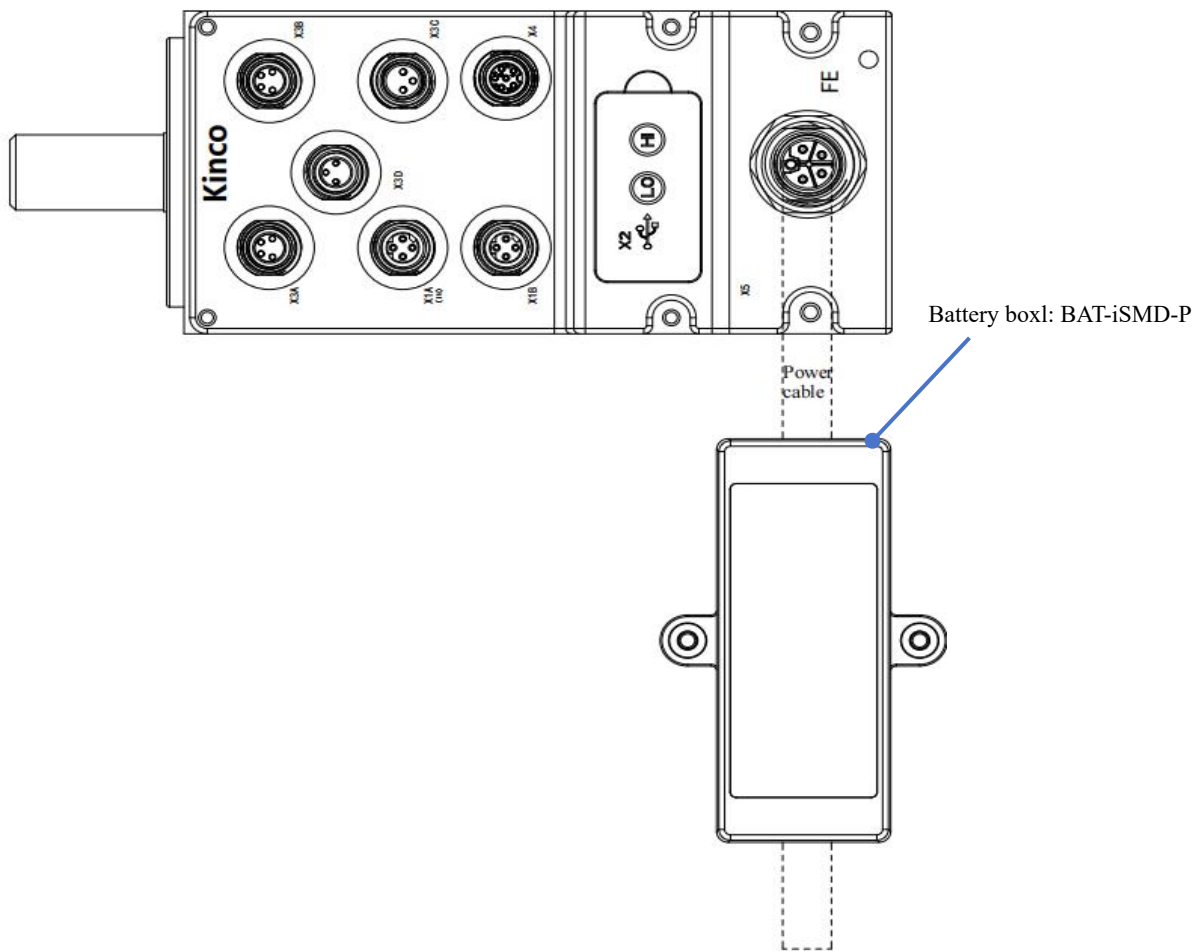
IP65-ES/PS model (Only applicable to model with STO function)**Note**

- The iSMD PA & PS model does not have a rotary dip switch.
- The AA/EA/PA model do not have X4 interface, mechanical multi-turn N encoder without X3D

The X3D interface is a battery interface for multi-turn magnetoelectric encoder. When in use, an external battery box (optional accessory) can be connected. The installation position of the battery box is as follows

IP20 model

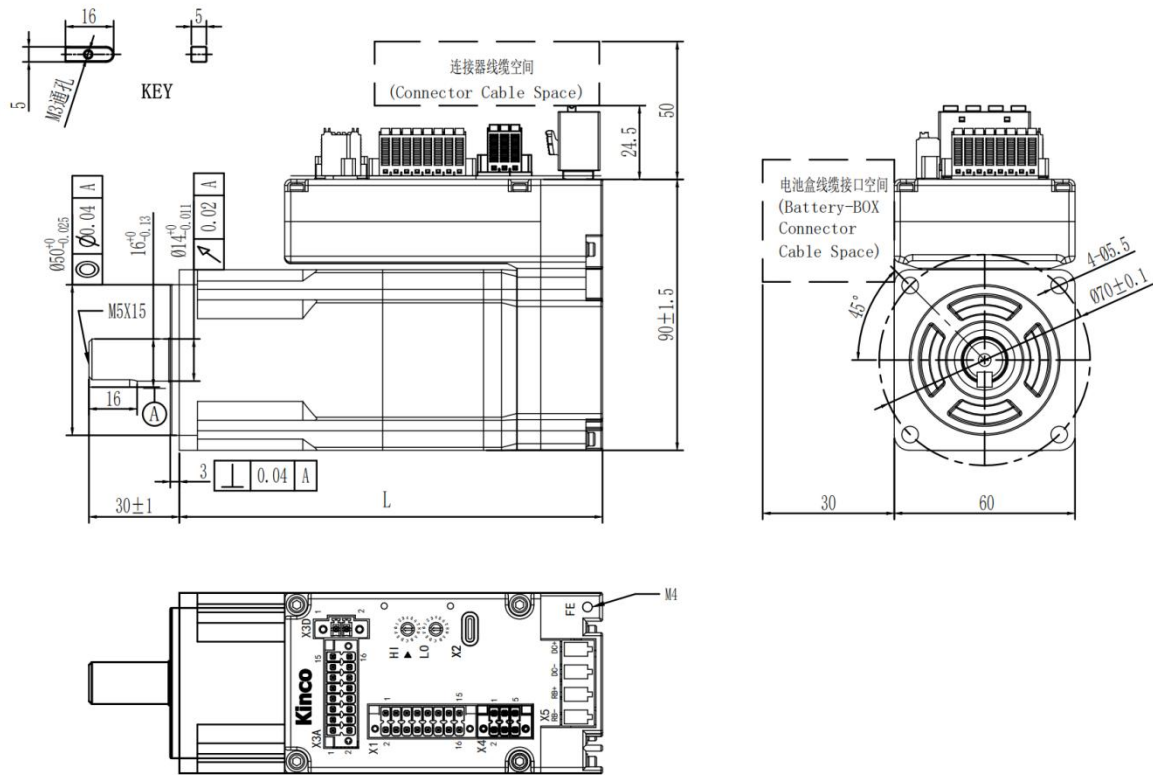
IP65 model



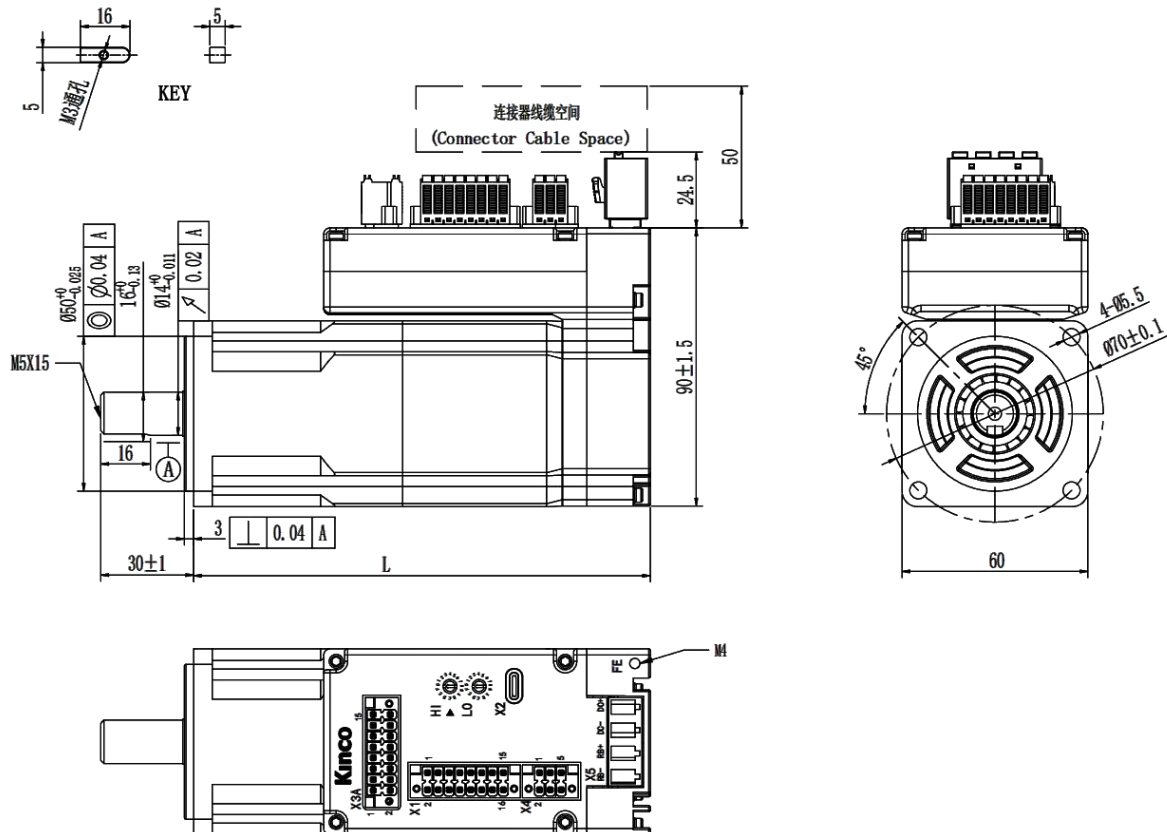
1.1.4 Dimension description

IP20 model

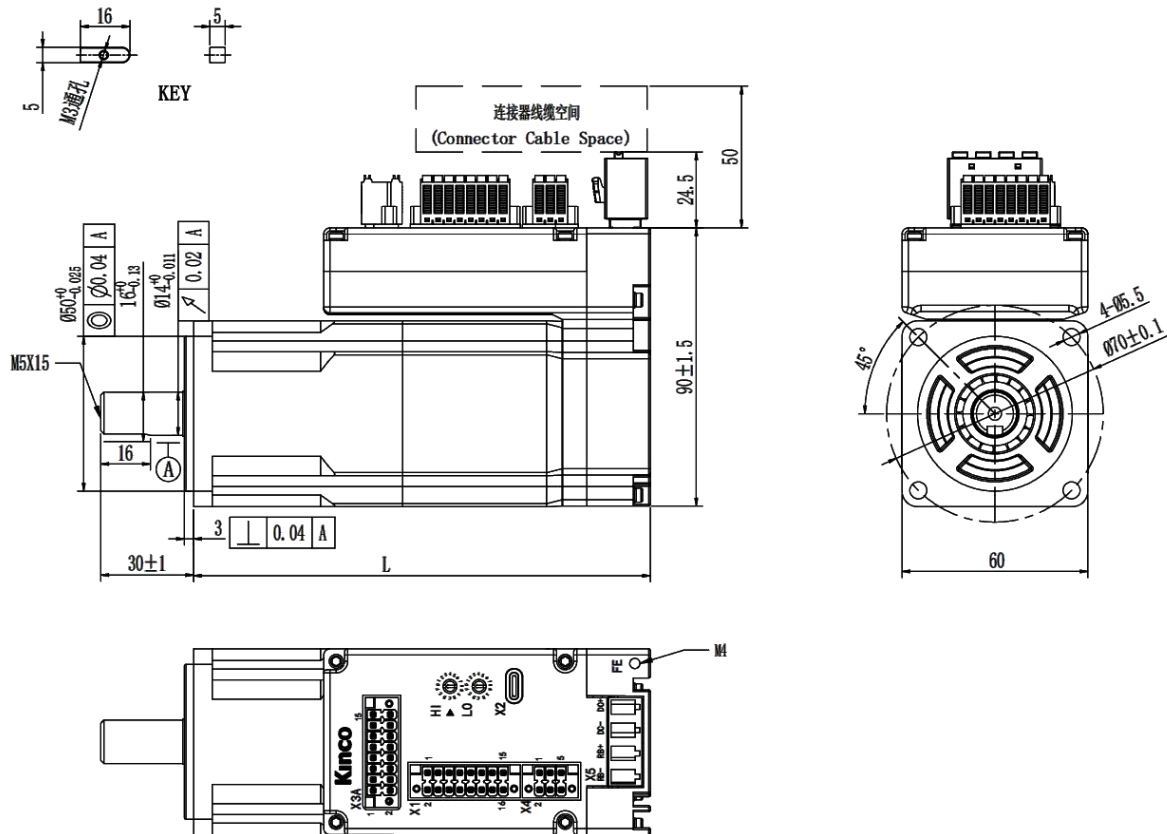
iSMD60D-040-DQ□K-A▲-000 (when ▲=A, there is no X4 interface)



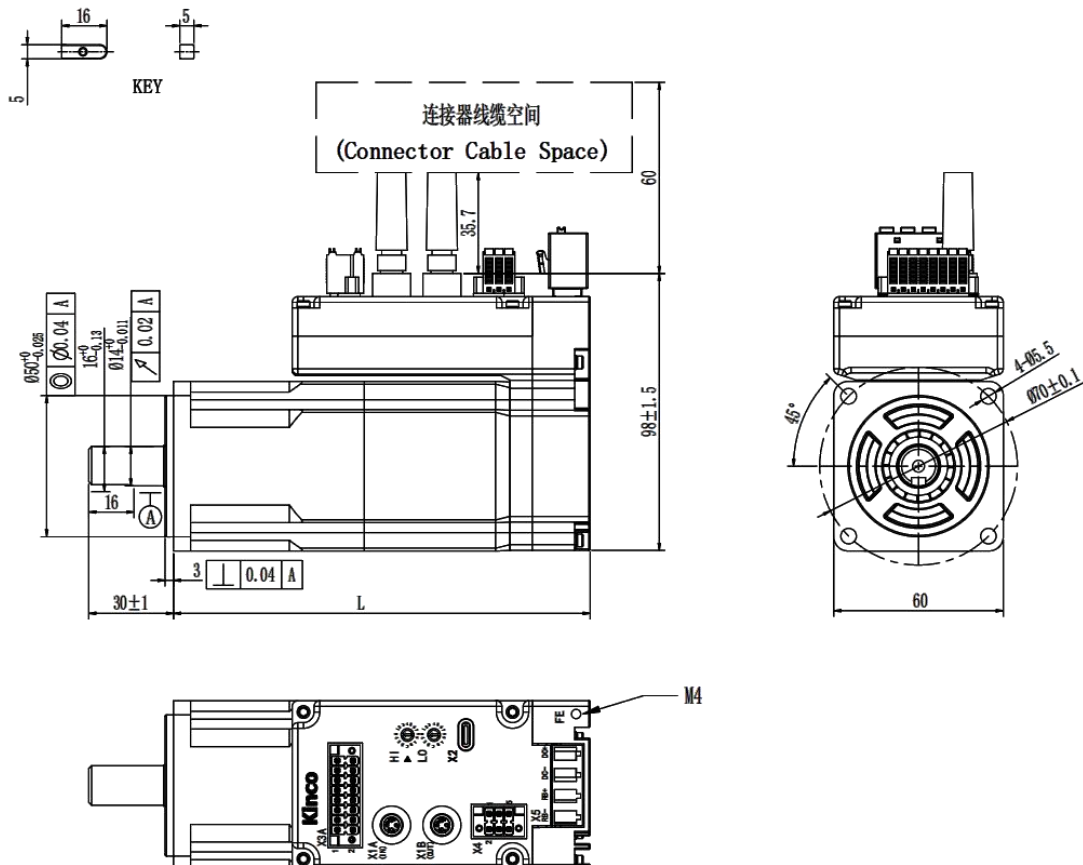
iSMD60D-040-DN□K-A▲-000 (when ▲=A, there is no X4 interface)



iSMD60D-040-DQ□K-E/P ▲-000 (when ▲=A, there is no X4 interface)

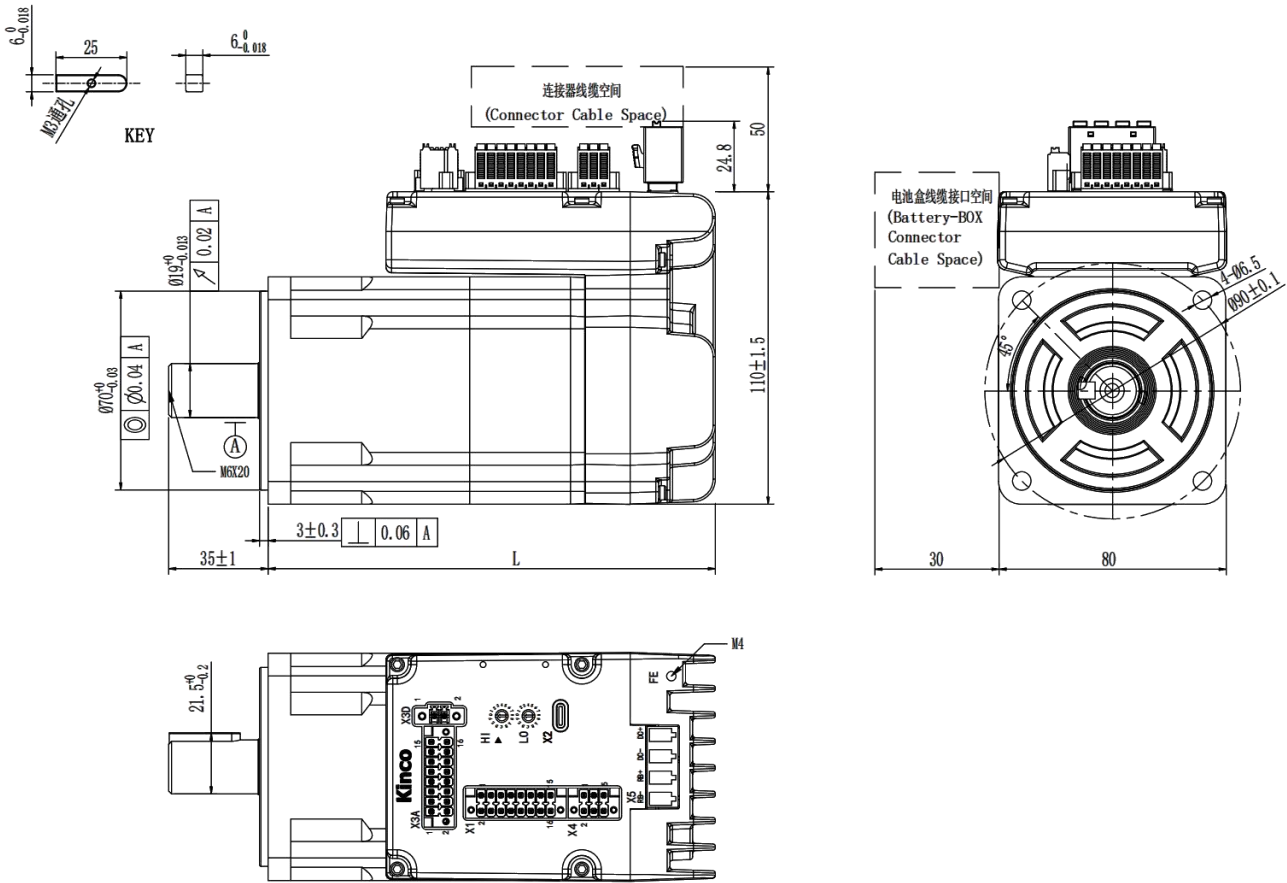


iSMD60D-040-DN□K-E/P ▲-000 (when ▲=A, there is no X4 interface)

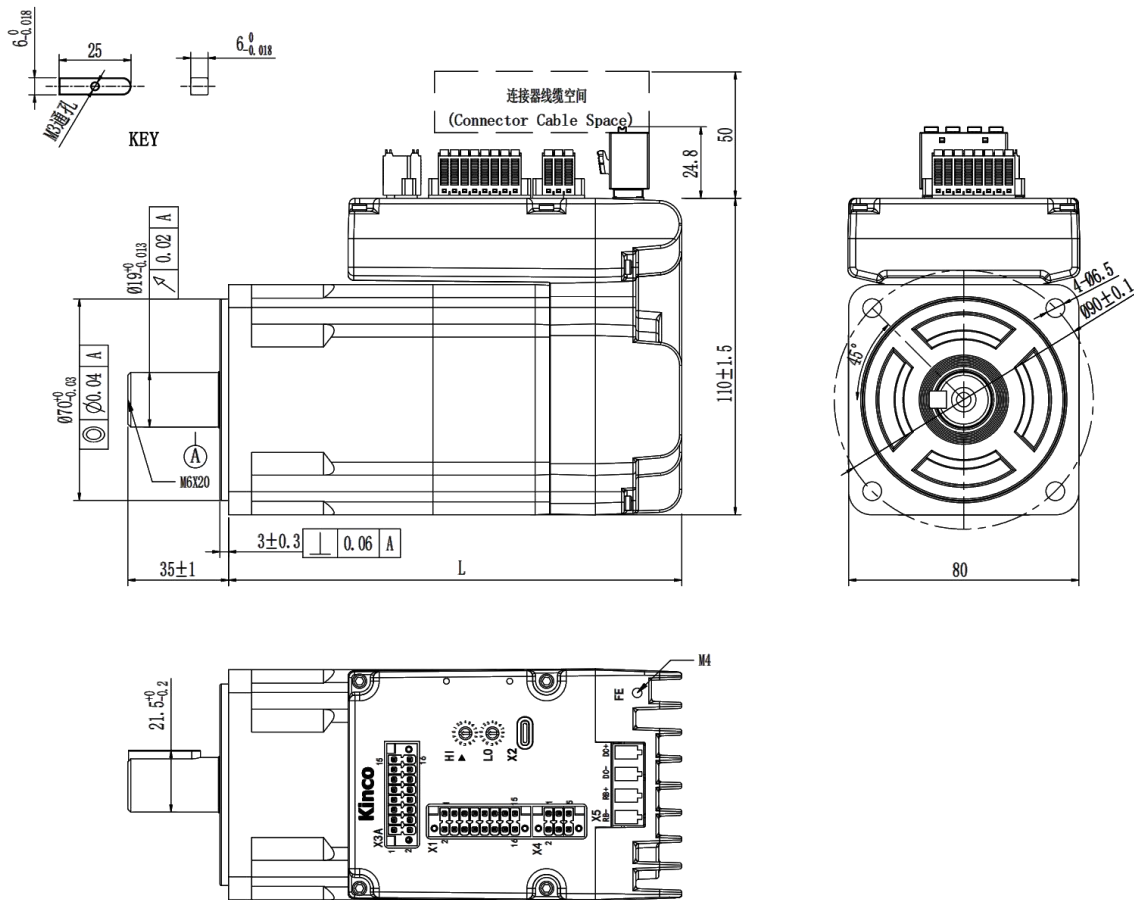


iSMD60 series model	With brake	Weight (Kg)	Overall dimension L (mm)
iSMD60D-040-DQAK-A▲-000		1.4	107±1.5
iSMD60D-040-DQBK-A▲-000	√	1.8	140.5±1.5
iSMD60D-040-DQAK-E/P▲-000		1.4	107±1.5
iSMD60D-040-DQBK-E/P▲-000	√	1.8	140.5±1.5
iSMD60D-040-DNAK-A▲-000		1.4	114±1.5
iSMD60D-040-DNBK-A▲-000	√	1.8	147.5±1.5
iSMD60D-040-DNAK-E/P▲-000		1.4	114±1.5
iSMD60D-040-DNBK-E/P▲-000	√	1.8	147.5±1.5

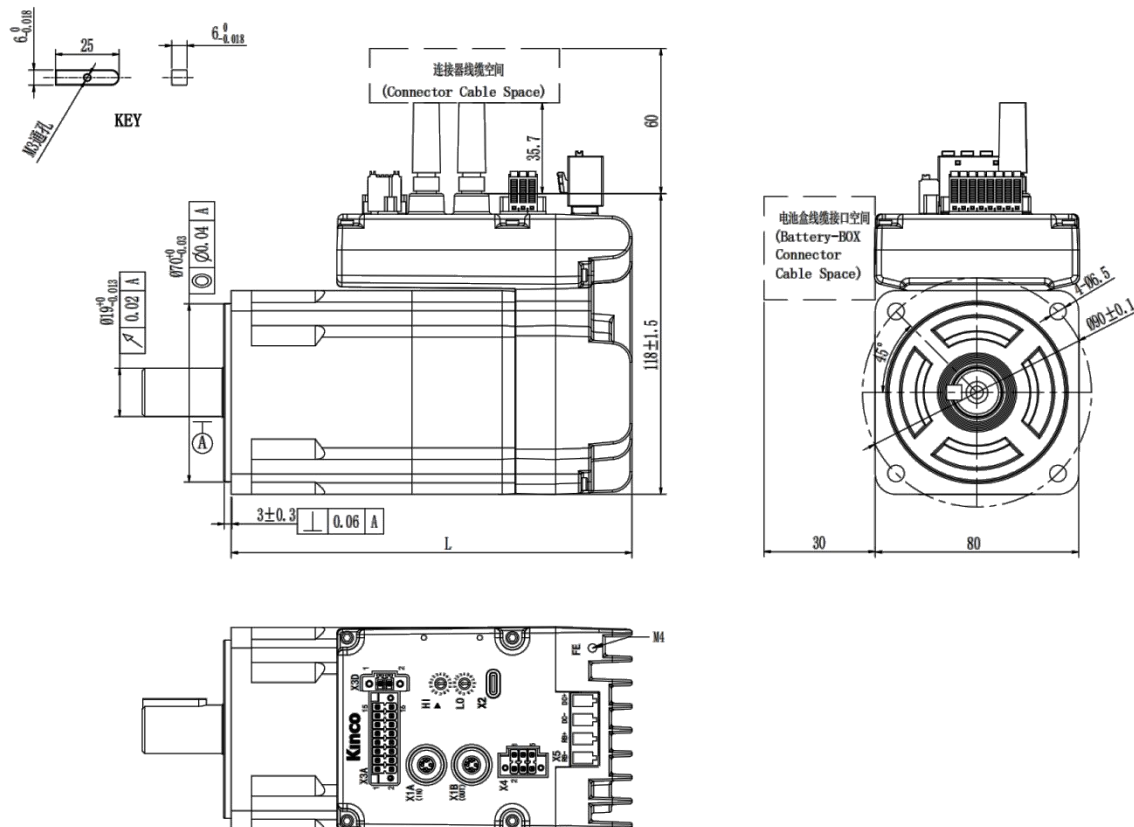
iSMD80D-075/100-DQ□K-A▲-000(when ▲=A, there is no X4 interface)



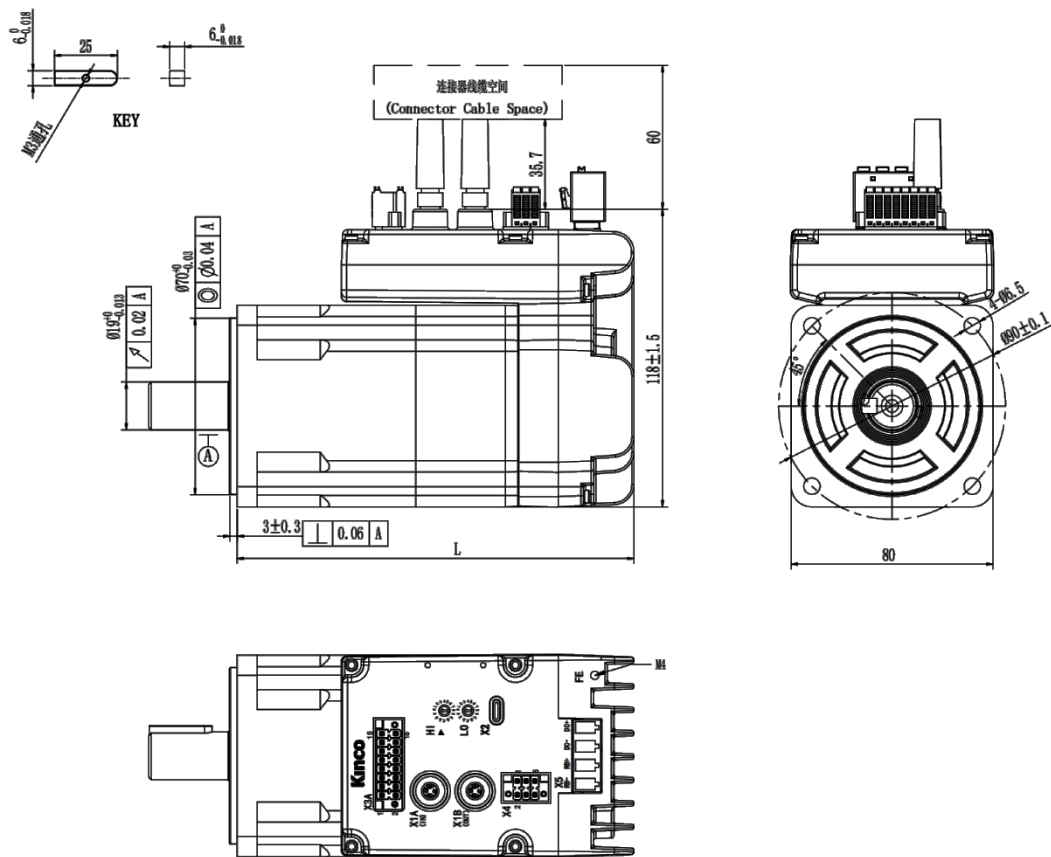
iSMD80D-075/100-DN□K-A▲-000(when ▲=A, there is no X4 interface)



iSMD80D-075/100-DQ□K-E/P▲-000(when ▲=A, there is no X4 interface)

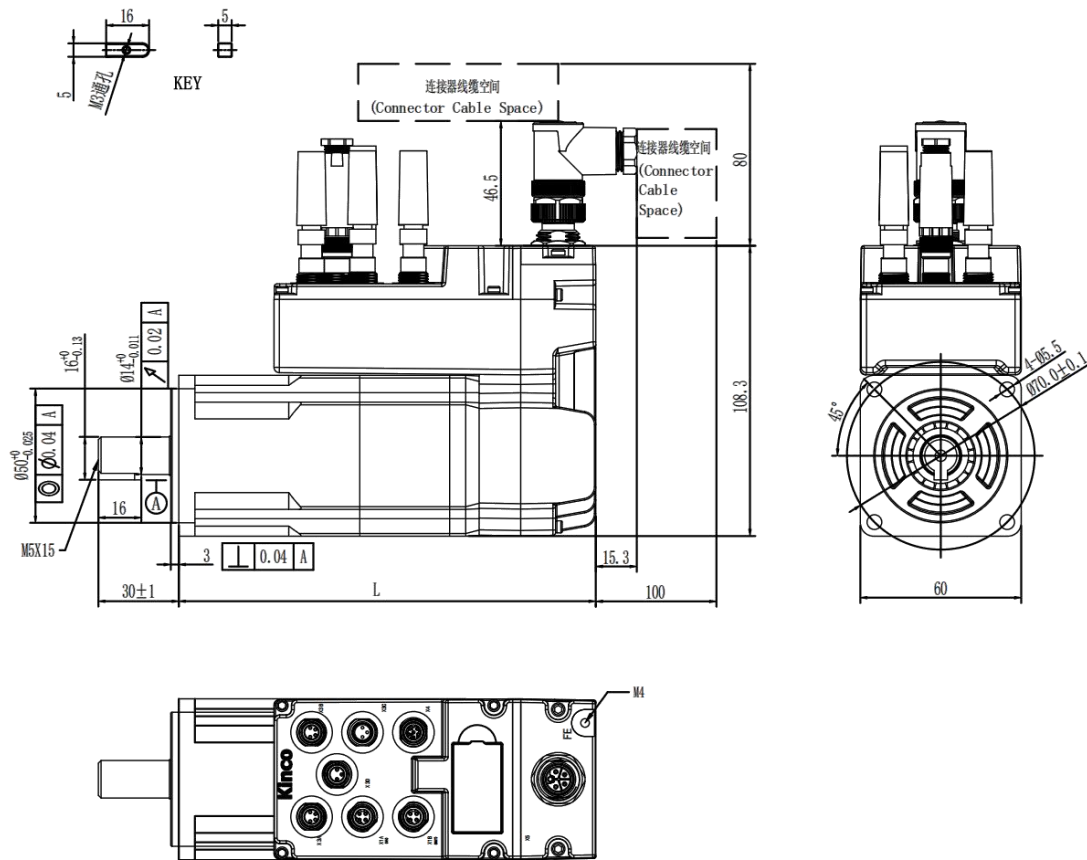
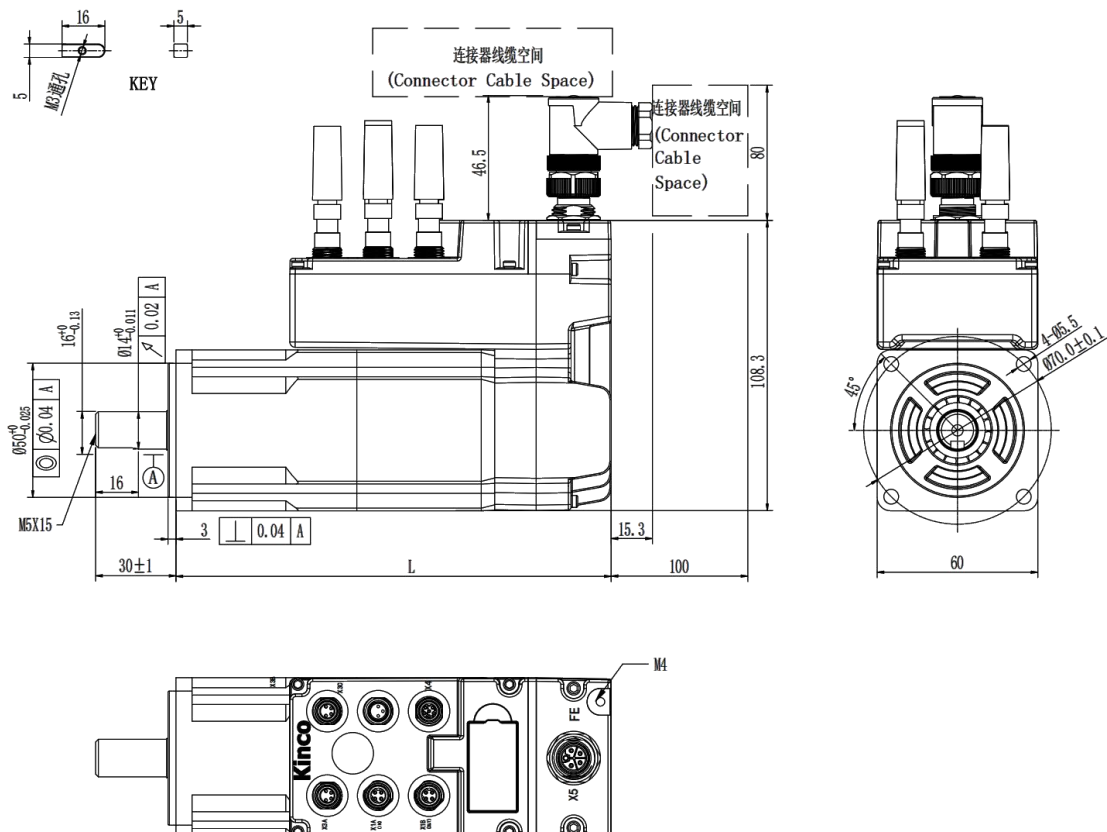


iSMD80D-075/100-DN□K-E/P ▲-000(when ▲=A, there is no X4 interface)

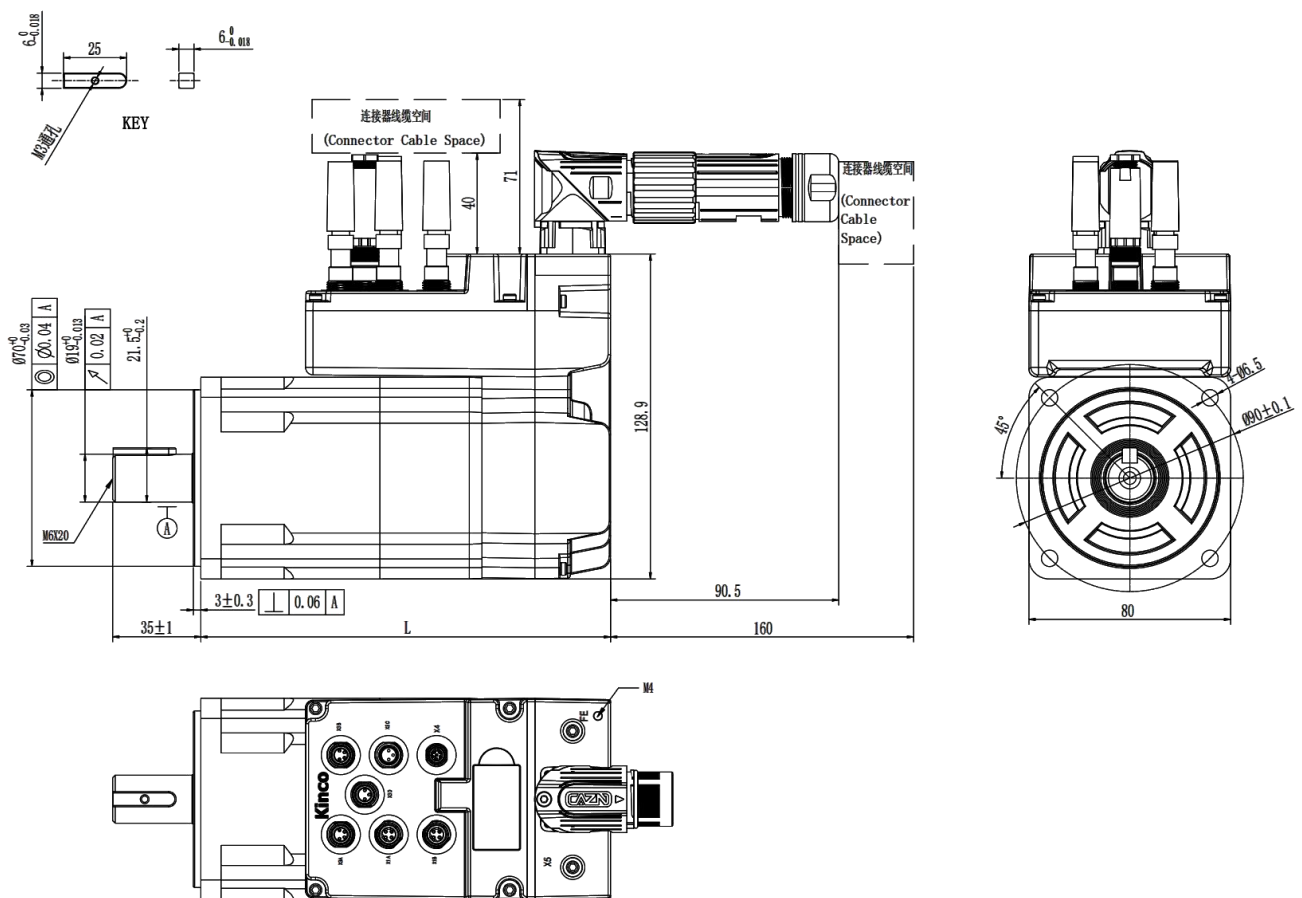


iSMD80 series model	With brake	Weight (Kg)	Overall dimension L (mm)
iSMD80D-075-D■AK-A▲-000		2.4	124.5±1.5
iSMD80D-075-D■BK-A▲-000	√	3.1	157.5±1.5
iSMD80D-075-D■AK-E/P▲-000		2.4	124.5±1.5
iSMD80D-075-D■BK-E/P▲-000	√	3.1	157.5±1.5
iSMD80D-100-D■AK-A▲-000		2.9	139.5±1.5
iSMD80D-100-D■BK-A▲-000	√	3.6	172.5±1.5
iSMD80D-100-D■AK-E/P▲-000		2.9	139.5±1.5
iSMD80D-100-D■BK-E/P▲-000	√	3.6	172.5±1.5

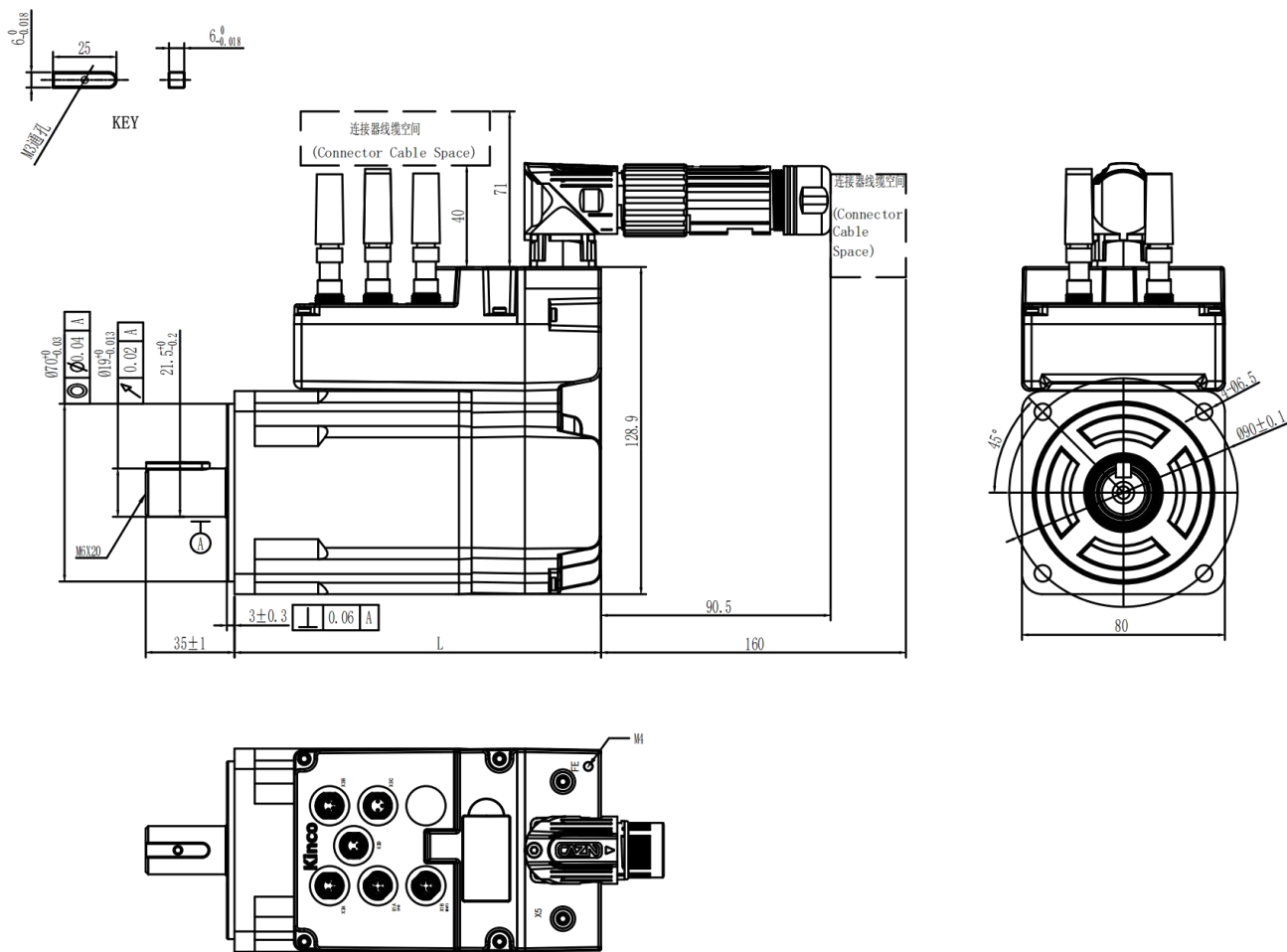
IP65 model

iSMD60D-040-DQ□K-E/P ▲-000-P000 (when ▲=A, there is no X4 interface)**iSMD60D-040-DN□K-E/P▲-000-P000 (when▲=A, there is no X4 interface)**

iSMD60 series model	With brake	Weight (Kg)	Overall dimension L (mm)
iSMD60D-040-DQAK-E/P ▲-000-P000		1.6	122±1.5
iSMD60D-040-DQBK-E/P ▲-000-P000	√	2	155.5±1.5
iSMD60D-040-DNAK-E/P ▲-000-P000		1.6	129±1.5
iSMD60D-040-DNBK-E/P ▲-000-P000	√	2	162.5±1.5

iSMD80D-075/100-DQ□K-E/P ▲-000-P000 (when ▲=A, there is no X4 interface)

iSMD80D-075/100-DN□K-E/P ▲-000-P000 (when ▲=A, there is no X4 interface)



iSMD80 series model	With brake	Weight (Kg)	Overall dimension L (mm)
iSMD80D-075-D■AK-E/P ▲-000-P000		2.8	129.5±1.5
iSMD80D-075-D■BK-E/P ▲-000-P000	√	3.7	162.5±1.5
iSMD80D-100-D■AK-E/P ▲-000-P000		3.3	144.5±1.5
iSMD80D-100-D■BK-E/P ▲-000-P000	√	4	177.5±1.5



Note

The iSMD PA & PS model does not have a rotary dip switch.

1.2 Product specification parameter

IP20 model

Model parameter			*1 iSMD60D-040-D■□K-★▲-000 *2 iSMD60D-040-D■□K-A▲-000-05V0	*1 iSMD80D-075-D■□K-A▲-000-05V0 *2 iSMD80D-075-D■□K-A▲-000-05V0	*1 iSMD80D-100-D■□K-★▲-000 *2 iSMD80D-100-D■□K-A▲-000-05V0	
Input	Power supply		48VDC, 10A Allowable fluctuation range 24V~60V		48VDC, 17A Allowable fluctuation range 24V~60V	48VDC, 24A Allowable fluctuation range 24V~60V
	Logic power		24VDC,1A (allowable fluctuation range 18V~ 30V), power supply and logic supply electrical isolation (500VDC) ▲=A: 24V, with fuse 60V/0.5A ▲=S: 24V, Logic power is an isolated power supply, Isolated voltage :500VDC			
Complete machine	Max continuous output current (Arms)		10		16.2	21.2
	Peak current (AP)		48AP		100AP	100AP
	Power output		*1 ★=A:With 24V output @200mA and is only used for power supply to sensors =E/P:None *2 With 5V output @200mA and is only used for power supply to sensors			
	Rated power Pn (W)		400		750	1000
	Rated speed nN (rpm)		3000		3000	3000
	Max speed (rpm)		4000		4000	3500
	Max torque Ts (Nm)		1.27		2.39	3.18
	Max torque Tm (Nm)		3.81		7.17	9.54
	Rotary inertia Jm(Kg·cm ²)		0.48		1.66	2.27
			0.49 (With brake)		1.72 (With brake)	2.33 (With brake)
	Brake	Brake holding torque (Nm)		2		4
Rated power (W)		7.6		11.5	11.5	
Rated voltage (VDC)		24		24	24	
Duty			S1 (Continuous duty,It was measured based on SMK60D-0040 and SMK80D-0075) , S3-80%-10 min (iSMD 1000W motor)			
Energy consumption brake			Need external brake resistance (iSMD60D:≥5Ω/100W; iSMD80D:≥3Ω/400W)			
Energy consumption brake voltage absorption point			63V±2V(default)			
Overvoltage alarm voltage			68V±2V(default)			
Undervoltage alarm voltage			16V±2V(default)			
Cooling mode			Natural cooling			
Input specification	Digital input		3-channel DI, Common COM1 port, High level :12.5V~30VDC, Low level: 0~5VDC,Max frequency : 1KHz Min turn-on voltage < 11V, Max cut-off voltage > 5V, Max input voltage:30V, complies with IEC61131-2 Type3, DI2 support probe function, Response time ≤10us, Input impedance: 5.1K			
	Pulse input		2-channel pulse input,Pulse+direction, CCW+CW,phase A+phase B (12V ~ 24V), Max frequency:500KHz It can be multiplexed as 2-channel DI (DI4 and DI5) by software setting Min turn-on voltage < 11V, Max cut-off voltage > 5V, Max input voltage:30V, complies with IEC61131-2 Type3.			
	Analog input		AA/AS model:1-channel voltage type analog input, 12bit, -10V ~ +10V, differential connection method			
Output specification			2-channel DO, open collector output, Max control voltage:30V, Max load current:100mA			
Type-C			Debug special, can use Kinco PC software connection			
STO			▲=S:2-channel STO, STO interface can be connected with the safety controller, safety switch, safety sensor, etc., to realize the safety function of the driver			
Protection function			Overvoltage protection, undervoltage protection, I ² T protection, short circuit protection, driver overheat protection.			
Bus function	Modbus/RS485		It supports a maximum 115.2K baud rate and can communicate with the controller using Modbus RTU			
	CanOpen		It supports a maximum of 1MHz baud rate and can communicate with the controller using the CANopen			
	EtherCAT		Support CoE(CiA402 protocol)and CSP/CSV/PP/PV/PT/HM mode, communication speed 100Mbps			
	Profinet		Non periodic parameter read and write; Message 1, message 3, message 5, message 9, message 102, message 105, message 111; Support IRT synchronous mode; Support absolute value encoder; Support DSC dynamic servo control			
Use environment	Vibration		Without packaging: Sinusoidal vibration 10Hz~58Hz 0.075mm 58Hz~150Hz 10m/s ² (1g) With packaging: random vibration 2Hz~3Hz 10 (m/s ²) ² /Hz 10Hz~20Hz 1 (m/s ²) ² /Hz 50Hz~2000Hz 0.5 (m/s ²) ² /Hz			
	Working temperature		-20℃ ~ 40℃			
	Storage temperature		-20 ~ 65℃ Short-term storage (Within one month)			
			-10 ~ 55℃ long term storage (More than 1 month)			
	Humidity		Less than 90%RH (no condensation)			
	Protection class		IP20			
	Installation site		Dust-free, dry, lockable			
	Installation method		Motor flange installation			
	Altitude		Less than 3000m,power derating 1.5%/100m between 1000m and 3000m			
	Atmospheric		67kpa ~ 106kpa			
Pollution class		PD2				

IP65 model

Model parameter			iSMD60D-040-D□□K-E▲-000-PXXX iSMD60D-040-D□□K-P▲-000-PXXX	iSMD80D-075-D□□K-E▲-000-PXXX iSMD80D-075-D□□K-P▲-000-PXXX	iSMD80D-100-D□□K-E▲-000-PXXX iSMD80D-100-D□□K-P▲-000-PXXX
Input	Power supply		48VDC, 10A Allowable fluctuation range 24V~60V	48VDC, 17A Allowable fluctuation range 24V~60V	48VDC, 24A Allowable fluctuation range 24V~60V
	Logic power		24VDC,1A (allowable fluctuation range 18V~ 30V), power supply and logic supply electrical isolation (500VDC) ▲=A: 24V, with fuse 60V/0.5A , ▲=S: 24V, Logic power is an isolated power supply, Isolated voltage :500VDC		
Complete machine	Max continuous output current (Arms)		10	16.2	21.2
	Peak current (Peak)		48AP	100AP	100AP
	Rated power Pn (W)		400	750	1000
	Rated speed nN (rpm)		3000	3000	3000
	Max speed (rpm)		4000	4000	3500
	Max torque Ts (Nm)		1.27	2.39	3.18
	Max torque Tm (Nm)		3.81	7.17	9.54
	Rotary inertia Jm(Kg·cm²)		0.48	1.66	2.27
			0.49 (With brake)	1.72 (With brake)	2.33 (With brake)
	Brake	Brake holding torque (Nm)		2	4
Rated power (W)		7.6	11.5	11.5	
Rated voltage (VDC)		24	24	24	
Duty			S1 (Continuous duty,It was measured based on SMK60D-0040 and SMK80D-0075) , S3-80%-10 min (iSMD 1000W motor)		
Energy consumption brake			Need external brake resistance module (Model:PRC-69-40-20 V2)		
Energy consumption brake voltage absorption point			63V±2V(default)		
Overvoltage alarm voltage			68V±2V(default)		
Undervoltage alarm voltage			16V±2V(default)		
Cooling mode			Natural cooling		
Digital input			4-channel DI, can be configured with source/sink input, High level :12.5V~30VDC, Low level: 0~5VDC, Max frequency : 1KHz Min turn-on voltage < 11V, Max cut-off voltage > 5V, Max input voltage:30V DI2 ,DI3 support probe function, Response time ≤10us, complies with IEC61131-2 Type3, Input impedance: 2.7K		
Output specification			2-channel digital output, software can set active/passive output . For active output, external 24V logic power supply needs to be connected to the power socket terminal (X5). DO1 load current : 200mA, DO2 load current : 100mA, Max control voltage : 30V		
Type-C			Debug special, can use Kinco PC software connection		
STO			▲=S:2-channel STO, STO interface can be connected with the safety controller, safety switch, safety sensor, etc., to realize the safety function of the driver		
Protection function			Overvoltage protection, undervoltage protection, I²T protection, short circuit protection, driver overheat protection.		
Bus function	EtherCAT		Support CoE(CiA402 protocol)and CSP/CSV/PP/PV/PT/HM mode, communication speed 100Mbps		
	Profinet		Non periodic parameter read and write; Message 1, message 3, message 5, message 9, message 102, message 105, message 111; Support IRT synchronous mode; Support absolute value encoder; Support DSC dynamic servo control		
Use environment	Vibration		Without packaging: Sinusoidal vibration 10Hz~58Hz 0.075mm 58Hz~150Hz 10m/s²(1g) With packaging: random vibration 2Hz~3Hz 10 (m/s²)²/Hz 10Hz~20Hz 1 (m/s²)²/Hz 50Hz~2000Hz 0.5 (m/s²)²/Hz		
	Working temperature		-20℃ ~ 40℃		
	Storage temperature		-20 ~ 65℃ Short-term storage (Within one month)		
			-10 ~ 55℃ long term storage (More than 1 month)		
	Humidity		Less than 90%RH (no condensation)		
	Protection class		IP65		
	Installation site		Dust-free, dry, lockable		
	Installation method		Motor flange installation		
	Altitude		Less than 3000m,power derating 1.5%/100m between 1000m and 3000m		
Atmospheric		67kpa ~ 106kpa			
Pollution class		PD2			

Note:■ = Q: Multi-turn communication type magnetoelectric encoder
 = N:Mechanical absolute value encoder
 □ = A:Without brake
 = B:With brake
 ▲ = A:Without STO
 = S:With STO
 ★ = A: RS485,CANopen
 = E: EtherCAT
 = P: Profinet

1.3 Overload characteristic

The drive calculates the heat by the output current, and implements the overload protection function for the driver and the motor. Overload protection means that when the driver and motor exceed the rated current, continuous use cannot be guaranteed, and only overload operation can be carried out under the overload protection characteristics.



Note

- After the driver or motor IIT failure (fault code: 080.0), driver output short circuit (fault code: 008.0), driver bus voltage is too high (fault code: 002.0) and other alarms, stop the fault first, check the cause of the fault, and then try to run. If the root cause of the fault is not identified, a high-frequency reset may cause damage to the drive hardware.

Name	CANopen	Modbus	Type	Data type	Explanation
Motor IIT actual utilization	2FF01010	2A00	RL	Unsigned 8	The actual load rate of the motor
Actual drive IIT utilization	2FF01110	2A01	RL	Unsigned 8	The actual load rate of the drive
Motor temperature alarm point	64101810	7180	RWSL	Integer 16	The default alarm point for the temperature is 120 °C

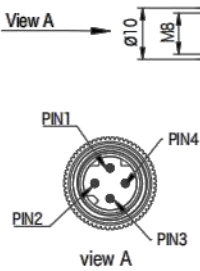
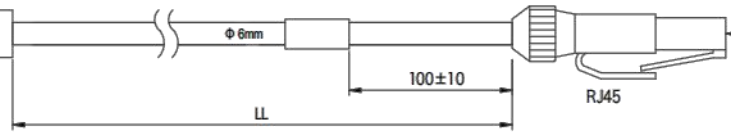
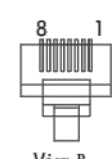
1.4 Drive list used in iSMD integrated motor

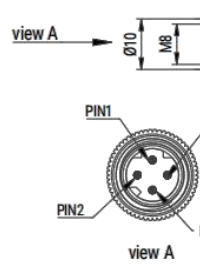
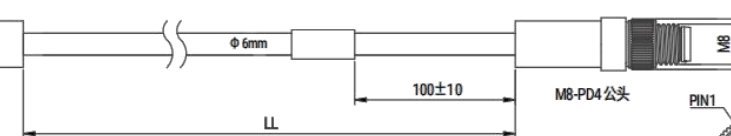
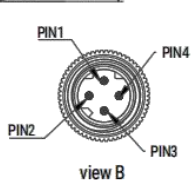
iSMD model	Compatible driver model
iSMD60D-040-D■□K-AA-000-05V0 iSMD60D-040-D■□K-AS-000-05V0	iSMD125-AA-000-0000-05V0 iSMD125-AS-000-0000-05V0
iSMD80D-075-D■□K-AA-000-05V0 iSMD80D-075-D■□K-AS-000-05V0	iSMD135-AA-000-0000-05V0 iSMD135-AS-000-0000-05V0
iSMD80D-100-D■□K-AA-000-05V0 iSMD80D-100-D■□K-AS-000-05V0	iSMD135-AA-000-0000-05V0 iSMD135-AS-000-0000-05V0
iSMD60D-040-D■□K-AA-000 iSMD60D-040-D■□K-AS-000 iSMD60D-040-D■□K-EA-000 iSMD60D-040-D■□K-ES-000 iSMD60D-040-D■□K-PA-000 iSMD60D-040-D■□K-PS-000	iSMD125-AA-000-0000-0000 iSMD125-AS-000-0000-0000 iSMD125-EA-000-0000-0000 iSMD125-ES-000-0000-0000 iSMD125-PA-000-0000-0000 iSMD125-PS-000-0000-0000
iSMD80D-075-D■□K-AA-000 iSMD80D-075-D■□K-AS-000 iSMD80D-075-D■□K-EA-000 iSMD80D-075-D■□K-ES-000 iSMD80D-075-D■□K-PA-000 iSMD80D-075-D■□K-PS-000	iSMD135-AA-000-0000-0000 iSMD135-AS-000-0000-0000 iSMD135-EA-000-0000-0000 iSMD135-ES-000-0000-0000 iSMD135-PA-000-0000-0000 iSMD135-PS-000-0000-0000

iSMD model	Compatible driver model
iSMD80D-100-D■□K-AA-000 iSMD80D-100-D■□K-AS-000 iSMD80D-100-D■□K-EA-000 iSMD80D-100-D■□K-ES-000 iSMD80D-100-D■□K-PA-000 iSMD80D-100-D■□K-PS-000	iSMD135-AA-000-0000-0000 iSMD135-AS-000-0000-0000 iSMD135-EA-000-0000-0000 iSMD135-ES-000-0000-0000 iSMD135-PA-000-0000-0000 iSMD135-PS-000-0000-0000
iSMD60D-040-D■□K-EA-000-PXXX iSMD60D-040-D■□K-ES-000-PXXX iSMD60D-040-D■□K-PA-000-PXXX iSMD60D-040-D■□K-PS-000-PXXX	iSMD125-EA-000-0P00-0000 iSMD125-ES-000-0P00-0000 iSMD125-PA-000-0P00-0000 iSMD125-PS-000-0P00-0000
iSMD80D-075-D■□K-EA-000-PXXX iSMD80D-075-D■□K-ES-000-PXXX iSMD80D-075-D■□K-PA-000-PXXX iSMD80D-075-D■□K-PS-000-PXXX	iSMD135-EA-000-0P00-0000 iSMD135-ES-000-0P00-0000 iSMD135-PA-000-0P00-0000 iSMD135-PS-000-0P00-0000
iSMD80D-100-D■□K-EA-000-PXXX iSMD80D-100-D■□K-ES-000-PXXX iSMD80D-100-D■□K-PA-000-PXXX iSMD80D-100-D■□K-PS-000-PXXX	iSMD135-EA-000-0P00-0000 iSMD135-ES-000-0P00-0000 iSMD135-PA-000-0P00-0000 iSMD135-PS-000-0P00-0000

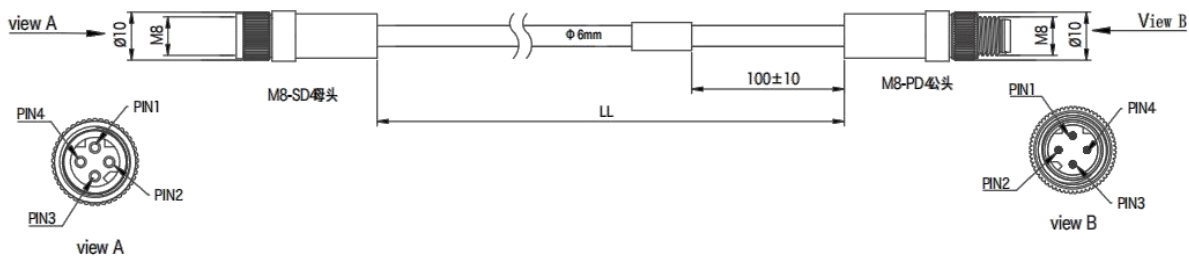
Chapter 2 Accessory description

2.1 Cable description

EtherCAT/ Profinet communication interface (X1) cable			
Model :M8PD4-RJ45-LL-F Specification:2P*26AWG, shielded, drag chain 10 million times Optional cable length:0.5m,1.5m,3m,5m View A  M8-PD4公头  View B  View B			
Signal	M8-PD4 male	Color	RJ45
TX+	1	White-orange	1
RX+	2	White-green	3
TX-	3	Orange	2
RX-	4	Green	6

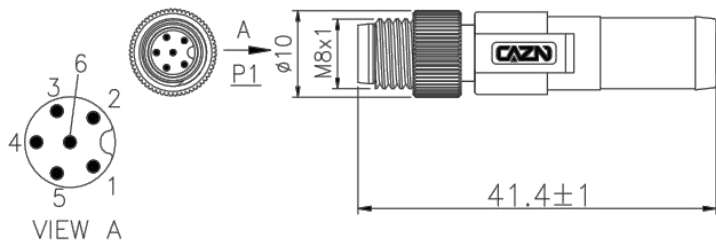
Model :M8PD4-M8PD4-LL-F Specification:2P*26AWG, shielded, drag chain 10 million times Optional cable length:0.5m,1.5m,3m,5m view A  M8-PD4公头  View B  view B			
Signal	M8-PD4 male	Color	M8-PD4 male
TX+	1	White-orange	1
RX+	2	White-green	2
TX-	3	Orange	3
RX-	4	Green	4

Model :M8SD4-M8PD4-LL-F
Specification:2P*26AWG, shielded, drag chain 10 million times
Optional cable length:0.5m,1.5m,3m,5m



Signal	M8-SD4 female	Color	M8-PD4 male
TX+	1	White-orange	1
RX+	2	White-green	2
TX-	3	Orange	3
RX-	4	Green	4

iSMD IP65 model STO interface accessory



Model	PIN	Color
M8-P6A-MWO	1	Red
	4	
	5	
	2	Black
	3	
	6	



Note
● The cable used in the iSMD model should not exceed 10 meters in length.

2.2 External installation instruction

Table 2-1 Recommended circuit breaker model

Model	Recommended circuit breaker		
	Model	Specification	Manufacture factory
iSMD60D-040-D■□K-★▲-000	FAZ-C16/1	277V/16A	Eaton
iSMD80D-075-D■□K-★▲-000	FAZ-C25/1	277V/25A	Eaton
iSMD80D-100-D■□K-★▲-000	FAZ-C16/1	277V/16A	Eaton
iSMD60D-040-D■□K-★▲-000-PXXX	FAZ-C16/1	277V/16A	Eaton
iSMD80D-075-D■□K-★▲-000-PXXX	FAZ-C25/1	277V/25A	Eaton
iSMD80D-100-D■□K-★▲-000-PXXX	FAZ-C40/1	277V/40A	Eaton

Table 2-2 Recommended noise filter model

Model	Recommended noise filter		
	Model	Specification	Manufacture factory
iSMD60D-040-D■□K-★▲-000	TY410S-10	250V/10A	TYZE
iSMD80D-075-D■□K-★▲-000	QMN30M4-L	250V/30A	QIMAI
iSMD80D-100-D■□K-★▲-000	QMN30M4-L	250V/30A	QIMAI
iSMD60D-040-D■□K-★▲-000-PXXX	TY410S-10	250V/10A	TYZE
iSMD80D-075-D■□K-★▲-000-PXXX	QMN30M4-L	250V/30A	QIMAI
iSMD80D-100-D■□K-★▲-000-PXXX	QMN30M4-L	250V/30A	QIMAI

Table 2-3 Recommended contactor model

Model	Recommended contactor		
	Model	Specification	Manufacture factory
iSMD60D-040-D■□K-★▲-000	CB-60A (20A)	20A/ 125V/220V 20A/125V/220V	Chint
iSMD80D-075-D■□K-★▲-000	CB-60A (40A)	40A/ 125V/220V	Chint
iSMD80D-100-D■□K-★▲-000	CB-60A (40A)	40A/ 125V/220V	Chint
iSMD60D-040-D■□K-★▲-000-PXXX	CB-60A (20A)	20A/ 125V/220V	Chint
iSMD80D-075-D■□K-★▲-000-PXXX	CB-60A (40A)	40A/ 125V/220V	Chint
iSMD80D-100-D■□K-★▲-000-PXXX	CB-60A (40A)	40A/ 125V/220V	Chint

Table 2-4 Recommended fuse model

Model	Recommended fuse		
	Model	Specification	Manufacture factory
iSMD60D-040-D■□K-★▲-000	ANL35A80VDCM8	35A	Vicfuse
iSMD80D-075-D■□K-★▲-000	ANL60A80VDCM8	60A	Vicfuse
iSMD80D-100-D■□K-★▲-000	ANL60A80VDCM8	60A	Vicfuse
iSMD60D-040-D■□K-★▲-000-PXXX	ANL35A80VDCM8	35A	Vicfuse
iSMD80D-075-D■□K-★▲-000-PXXX	ANL60A80VDCM8	60A	Vicfuse
iSMD80D-100-D■□K-★▲-000-PXXX	ANL60A80VDCM8	60A	Vicfuse

**Warning**

- A surge suppressor is installed on the coil of the relay and electromagnetic contactor to prevent the damage caused by the peak voltage to the equipment.
- Please make sure to connect a fuse on the power input side of iSMD to prevent electrical accidents caused by power short circuit.
- Do not connect the external brake resistor to the DC+ and DC- ends. Before powering on the driver, check whether the cable connections are correct.
- High voltage may remain in the drive after power off. Please check that the CHARGE indicator is off ten minutes after power off before disassembling and repairing the drive.
- Please do not share power with welding machines, lasers and other equipment;

Table 2-5 Recommended wire diameter

Model	iSMD60D (400W)	iSMD80D (750W&1kW)
Power supply wire diameter (AWG)	14AWG, strip length 6~7mm	12AWG, strip length 6~7mm
IO terminal wire diameter	16~26AWG, 0.2~1.5 mm ² , strip length 10mm	
STO terminal wire diameter	16~26AWG, 0.2~1.5 mm ² , strip length 10mm	
To ensure the stable and safe use of the product, the following points should be noted when wiring and connecting the drive: ◆During the assembly process, pay attention to the fixation of the cable. do not apply stress to the connector part, otherwise it may cause damage to the connector. ◆For the motor power cable and encoder cable, please use the cable provided by the manufacturer. Calculate the connection distance of the equipment accurately and select the shorter cable as much as possible. ◆The bending radius of the cable should be more than 5 to 10 times the wire diameter to prevent the inner core of the cable from breaking after long-term use. During the use of the cable, it should avoid being subjected to tension. In some drag chain applications, the use of flexible cable also needs to be considered. ◆Generally, the cable is carried out according to the output current of the drive at 8A/mm². When the power cable is longer than 30 meter, the wire diameter needs to be increased. ◆Keep a distance of more than 30cm between power cable and signal cable. Try to avoid laying them in parallel. Do not lay power cable and signal cable in the same conduit or bundle them together.		

Brake resistance

The energy generated by the servo motor in the braking state will be fed back to the DC bus of the driver. When the voltage of the DC bus exceeds the protection range, the driver will report that the bus voltage is too high and the excess energy needs to be absorbed by an external brake resistor. When the external brake resistance is used, the brake resistance resistance value and brake resistance power can be set at the driver end, and the driver will turn on the brake resistance overtemperature protection function according to the set resistance value and power. It should be noted that the resistance value of the optional external brake resistor cannot be lower than the recommended resistance value.

Table 2-6 Recommended specifications for braking resistor

Model	Minimum resistance of brake resistance [Ω]	Brake resistance power [W]	Brake resistance voltage [VDC] (minimum value)
iSMD60D-040-D $\blacksquare$ K- $\star\blacktriangle$ -000	10	100	500
iSMD80D-075-D $\blacksquare$ K- $\star\blacktriangle$ -000	5	200	500
iSMD80D-100-D $\blacksquare$ K- $\star\blacktriangle$ -000	3.5	200	500
iSMD60D-040-D $\blacksquare$ K- $\star\blacktriangle$ -000-PXXX	10	100	500
iSMD80D-075-D $\blacksquare$ K- $\star\blacktriangle$ -000-PXXX	5	200	500
iSMD80D-100-D $\blacksquare$ K- $\star\blacktriangle$ -000-PXXX	3.5	200	500

Table 2-7 Brake resistance parameters

Address	Name	Bits	Modbus address	Command Type	Unit	Description
60F70110	Brake resistance resistance value	uint 16	0x6010	RW	Ω	You can enter the resistance value and power of the external brake resistance. The default value is 0, indicating that the external brake resistance power and temperature detection are disabled
60F70210	Brake resistance power	uint 16	0x6020	RW	W	
60F70310	Brake resistance time constant	uint 16	0x6030	RW	S	External brake resistance time constant $S = DEC * 256 / 1000$

**Note**

- The chopper voltage setting must be greater than the actual input voltage of the driver, otherwise it will cause brake resistance or drive damage.

Chapter 3 Product installation

3.1 Product installation

3.1.1 Use requirement

- Ensure that this document is available to design engineers, installers, and personnel responsible for commissioning machines or systems that use the product.
- Please ensure that you always follow the requirements of this document and also consider the documentation for other components and modules.
- Please consider the legal requirements applicable to your destination and:
 - Regulations and standards
 - Test organization and insurance company regulations
 - National specifications

Transport and storage condition

- Please ensure that the product is not subjected to more than permitted burdens during transportation and storage, including:
 - Mechanical load
 - The temperature is not allowed
 - Moisture
 - Corrosive gas
- Please store and transport in original packaging, which provides adequate protection against routine problems.

Technical requirement

The general conditions for the correct and safe use of the product must always be observed:

- The connection and environmental conditions specified in the product technical data and the technical requirements of all other connected components. Products are only allowed to operate in accordance with relevant safety procedures if they meet product specifications.
- Follow the instructions and warnings in this document.

Operator requirement

- This product should only be operated by an electrical engineer who is familiar with the following provisions:
 - Installation and operation of electrical control systems
 - Applicable regulations for the operation of safe engineering systems
 - Applicable provisions for accident protection and occupational safety
 - Familiarize yourself with the documentation of the product

Operator must take mechanical protection measures to protect personal safety. Please wear necessary protective equipment, such as anti-smash shoes, safety clothing, safety glasses, protective gloves and sleeves.

3.1.2 Precaution

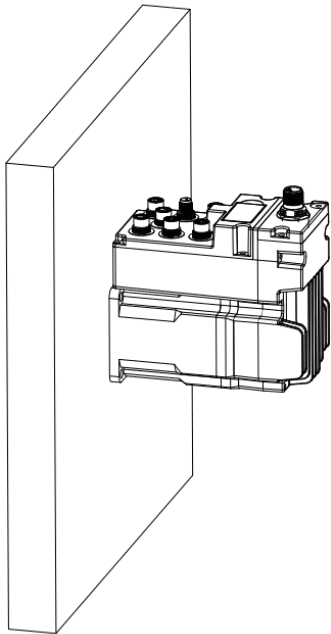


Note

- Please use the servo system in a well-ventilated, dry and dust-free place, free of vibration, grinding fluid, oil mist, iron powder, chips, and immersion of moisture, oil, and water, away from the furnace and other heat sources.
- When the servo system parts are found to be incomplete or damaged, installation and use shall not be carried out.
- Do not use the servo motor in a closed environment, and the servo system must be installed in a control box in a room without rain and direct sunlight, and the surrounding must be non-flammable.
- When making mechanical connections, use a coupling or expansion sleeve and keep the shaft of the motor in line with the shaft of the machine. When installed, it meets the centering accuracy requirements. If the centering is not sufficient, it will cause vibration, which may sometimes damage bearings and encoders.
- When fixing the drive and motor, ensure that each fixing point is locked.
- Do not block the driver vent when installing the servo driver. Reserve heat dissipation space around the driver to ensure that the servo driver is not affected when installing the brake resistance.
- Avoid any foreign matter entering the servo system, screws, metal chips and other conductive foreign matter or flammable foreign matter entering the servo drive may cause fire and electric shock, for safety reasons, please do not use damaged or damaged parts of the servo drive and servo motor.
- The shaft of the motor and the shaft of the equipment must be installed in good alignment. Please use the shaft connector or the expansion sleeve.
- Do not "bend" the cable or apply "tension" to it, and do not make it too tight when wiring (in use).
- Do not use gasoline, thinner, alcohol, acid or alkaline detergent to avoid discoloration or damage to the shell.
- Improper installation may cause damage to the motor encoder. Please note the following during installation:
- When installing pulleys on motor shafts with keyways, screw holes are used at shaft ends. In order to install the pulley, first insert the double-headed nail into the screw hole of the shaft, use a washer on the surface of the coupling end, and gradually lock the pulley with a nut.
 - For motor shafts with keyways, install them using screw holes at shaft ends. For shafts without keyways, frictional coupling or similar methods are used.
 - When the pulley is removed, the pulley removal device is used to prevent the bearing from being strongly impacted by the load.
 - To ensure safety, install a protective cover or similar device in the rotating area, such as a pulley mounted on the shaft.
- When used in places with water drops, please use it on the basis of confirming the protection level of the servo motor. (except shaft through) Specify servo-products with oil seals for use in areas where oil droplets may drip onto shaft through parts. Conditions of use of oil seal:
 - Make sure the oil level is lower than the lip of the oil seal when using.
 - Use in a condition where the oil seal can keep the oil spray level good.
 - When the servo motor is installed vertically, please be careful not to make the oil seal lip oil.

3.1.3 iSMD installation method

iSMD60D/80D is installed on a 255mm* 255mm* 8mm (L*W*H) black oxide LY12 aluminum plate for auxiliary heat dissipation; as shown in the following figure



3.1.4 Safety operation and inspection

Item	Content	Checked
Preparations before product installation	During installation, operators should wear electrostatic wristbands, and the installation hole positions should accurately correspond to structural components to avoid collisions between devices and structures, and to prevent cable from being squeezed or pulled.	☐
Input Voltage	Whether the voltage of the power supply and logic power supply are within the allowable range.	☐
Cable	Whether there is discoloration or oxidation in the cable and terminal; Whether the insulation layer is aged or cracked ;	☐
Environment	Whether there is dust, metal shavings and other foreign matter on the surface of the product; Whether the working temperature is -20~40℃, and whether the humidity is less than 90% (no condensation).	☐
Abnormal sound	Whether abnormal noise occurs during startup and operation.	☐
Indicator light	Whether the indicator light display normally	☐
Staff Training	Installation and maintenance personnel are required to complete Kinco's standard factory training.	☐
Note: 1. Before conducting the inspection items, please ensure that the product is in a power-off state 2. If the product is damaged, it needs to be returned to the factory for repair. It is not allowed to disassemble and repair it without authorization		

3.1.5 Oil seal installation instruction

The bearing of the motor has a double-sided dust proof effect, and the assembly of the oil seal will increase the loss of the motor, resulting in a decrease in the efficiency of the motor, if it is not necessary to install the oil seal, it is not recommended to install the oil seal. Before assembling the oil seal, please ensure that the installation hole and the oil seal are free of debris, oil, dust, etc. Incorrect installation of the oil seal may cause abnormal operation of the motor (such as the no-load running current increases, and the temperature rise of the motor shaft increases). Please install the oil seal correctly according to the following steps:

- 1.Before the oil seal installation, grease is evenly applied in the oil seal lip and the outer circle of the seal (recommended to use HP-R of the Great Wall, temperature resistance of 180 degrees of grease) to strengthen the performance of lubrication and temperature resistance, increase the sealing and waterproof effect of the oil seal, and at the same time, appropriate amount of grease is also applied to the motor shaft oil seal location, front end cover and oil seal chamber.
- 2.When installing the oil seal, the side of the oil seal with the spring should be oriented towards the motor, select the appropriate sleeve specification, and put the oil seal on the sleeve to ensure that the seal lip will not be damaged.
- 3.After the sleeve with the oil seal is put into the motor shaft, the oil seal is slowly pressed into the oil seal chamber with the press assembly tool. Pay attention to ensure that the oil seal is perpendicular to the machine shaft. The purpose of using the framework oil seal press assembly tool is to ensure that the oil seal is pushed into the chamber with uniform force.
- 4.After successful installation, check whether the oil seal will tilt, the oil seal should be fitted with the motor bearing cover, and the lip of the oil seal should be completely closed to ensure the tightness of the oil sea.

70

Shaft Seal Installation
油封安装示意图

USE
SUITABLE
PRESSURE
TOOL
TO
INSTALL
THE
SEAL

Flange Size ID OD L

60mm 16mm 32mm 50mm

80mm 21mm 42mm 50mm

100

100

70

Oil seal installation instructions
油封安装说明

1、Seal must be fitted with gaiter spring visible from inside of motor.
油封安装时，使油封有弹簧的一面朝向电机。

2、Ensure that seal is lubricated sufficiently for installation and initial start-up. (HP-R lubricants of Sinopec is recommended)
确保油封安装前充分润滑。(推荐使用长城HP-R润滑油)

3、Sealing lip must be protected from damage during installation by use of suitable sleeve over shaft. Specifications of sleeve refer to shaft seal installation flow chart printed on opposite page.
在安装过程中必须用合适的套筒，确保密封唇不被损坏，套筒规格见反面《油封安装示意图》中所示。

4、Firmly support rear of motor on suitable press, taking care to avoid damage to connectors.
在电机后端用合适的压力支撑，同时避免损坏电机插头。

WARNING:

Under no circumstances attempt to hammer seal into position.

警告:

任何情况下，严禁敲击油封至油封室内。

100

70

38

**Note**

- Before installing the oil seal, ensure that experienced personnel install the oil seal according to correct steps to prevent damage to the motor or the oil seal.
- Do not use a hammer to directly hit the oil seal for assembly. Direct tapping will cause deformation of the oil seal, scratches on the outer diameter of the oil seal and the seal lip, and the spring of the oil seal.
- When installing the oil seal, the oil seal must be placed horizontally and then installed under uniform pressure. If the oil seal is forcibly installed in the skewed state, it will cause the deformation of the oil seal and thus fail.
- If no professional tools can be installed, use a stop plate on the oil seal, and evenly apply force to knock the oil seal into the oil seal chamber.

3.2 EMC configuration description

iSMD meets the relevant requirements of EMC certification standards EN IEC 61000-6-2 , EN IEC 61800-3 , but in the course of product use, it may be affected by the actual installation method, connector and wiring status. Results in different EMC test levels.

**Warning**

- In the civil environment, this product may cause interference which may degrade the reception quality of radio signals. Users shall take suppression measures if necessary to prevent interference.
- Products must be installed and certified by professionals who are familiar with safety and EMC requirements, and EMC engineers are responsible for ensuring that the products and systems produced comply with relevant laws.

Site condition

iSMD device is not directly connected to the mains and meet the requirements of overvoltage class I (OVC I) and Pollution degree (PD 2) use sites.

Installing environmental condition

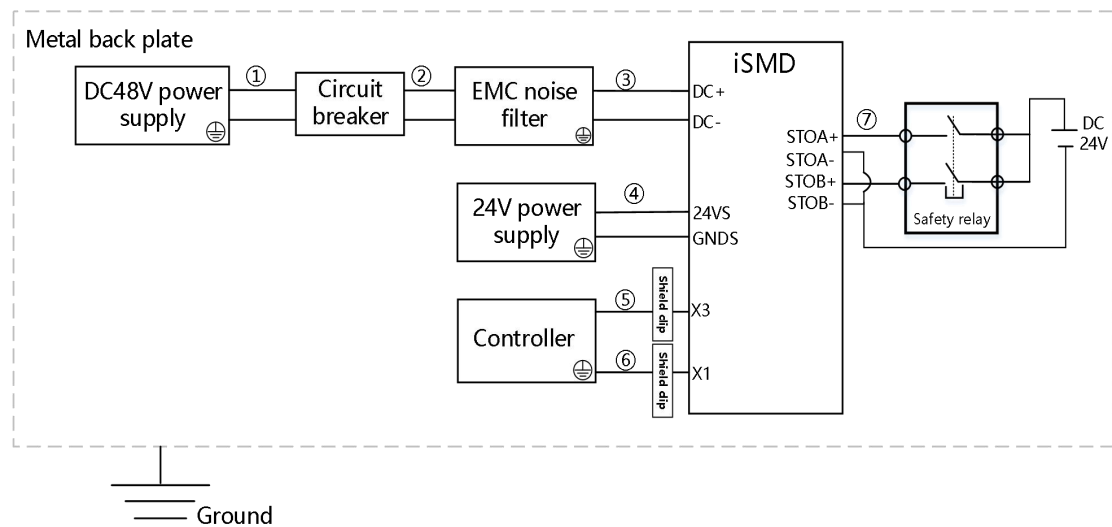


Figure 3-1 iSMD installation environment diagram



Note :

1. Select the shielded cables in the diagram with a shielding coverage rate of more than 85%.

Serial number	Cable name	Specification
①②③④	Power cable	Without shielded cable
⑤	Input/Output signal cable	Shielded cable
⑥	Communication cable	Shielded + twisted pair cable
⑦	STO cable	Shielded cable

2. Please install the device in the control cabinet. The device should be installed on the same metal backplane with good electrical conductivity in the cabinet.

(1) EMC noise filter

- In order to ensure that the product meets the EMC standard requirements, install the EMC noise filter at the power input near the servo wheel. For the noise filter model, see **Section 3.2**.
- The length of the cable between the output of the noise filter and the input power supply of the servo wheel should be less than 30cm;
- The servo wheel and filter are installed in the same metal backplane and the metal backplane is well grounded;
- Do not lay the input and output cables of the filter in the same slot or bind them together.

(2) Grounding

- In order to prevent electric shock, be sure to ground the noise filter, servo wheel;

(3) Shielded layer

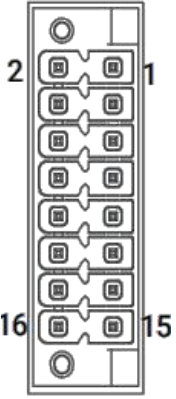
- Input and output signal cables and communication cable please use shielded cable.

Chapter 4 Wiring terminal

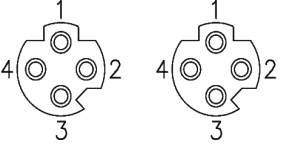
4.1 Interface terminal description

4.1.1 iSMD IP20 model

4.1.1.1 Bus communication interface (X1)

iSMD60D&iSMD80D-AA/AS model (IP20)			
 F8CVM-THR-3.5-16P-7-L30	Pin	Name	Description
	1	CANH	Differential CAN signal
	2		
	3	CANL	
	4		
	5	GNDC	CAN isolated ground
	6		
	7	RS485A	Data send/receive
	8		
	9	RS485B	
	10		
	11	GNDR	485 signal ground
	12		
	13	5VAUX	Internal 5V DC power supply output, voltage range +4.5 ~ 5.5V, Max output 500mA, only for use with photoelectric sensors ※
	14	GNDAUX	Internal 5V DC power supply output reference ground
	15	24VAUX	Internal 24V DC power supply output, voltage range +23 ~ 25V, Max output 500mA, only for use with photoelectric sensors ※
16	GNDAUX	Internal 24V DC power supply output reference ground	

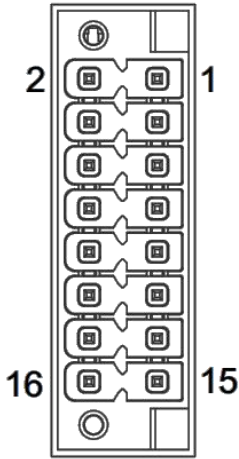
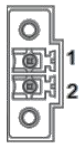
Note: The auxiliary power supply output type depends on the model, please consult local sales for specific information

iSMD60D&iSMD80D-EA/ES/PA/PS model (IP20)			
 X1A (IN) X1B(OUT) M8-S4D-PTT	Pin	Name	Description
	1	TD+	Send data +
	2	RD+	Receive data+
	3	TD-	Send data -
	4	RD-	Receive data -
	Shell	Shielded	Connect to the integrated motor shell

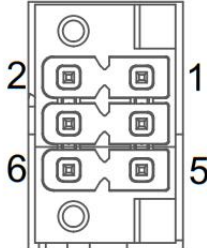
4.1.1.2 PC debugging interface (X2)

	Description
	Standard USB2.0 Type-C interface Note: To prevent signal loss or disconnection affecting debugging, the USB debug cable should not exceed 3 meters.

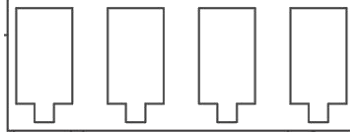
4.1.1.3 IO interface and battery interface (X3A&X3D)

 F8CVM-THR-3.5-16P-7-L30	Pin	Name	Description
	1	AI1+	1-channel voltage type analog input, 12bit, -10V ~ +10V, differential connection method
	2	AI1-	
	3	PUL+	Pulse+direction, CCW+CW, phase A+phase B (12V~24V) Max frequency:500KHz Min turn-on voltage < 11V, Max cut-off voltage > 5V, Max input voltage:30V,It can be multiplexed as 2-channel DI (DI4 and DI5) by software setting
	5	PUL-	
	7	DIR+	
	9	DIR-	
	4	COMI	Input common terminal
	6	DI1	3-channel DI, Common COMI port, High level : 12.5V~30VDC, Low level: 0~5VDC, Max frequency : 1KHz ,Min turn-on voltage < 11V, Max cut-off voltage > 5V, Max input voltage:30V, DI2 can be configured for probe function, Response time ≤10 uS,Input impedance: 5.1K
	8	DI2 (Fast)	
	10	DI3	
	11	DO1+	Open collector output, Max control voltage:30V, Max load current:100mA
	12	DO1-	
	13	DO2+	
	14	DO2-	
	15	24VS	24V DC logical power input. voltage range +18 ~ 30V, maximum input current 1A
	16	GNDS	24V DC logic power input reference ground
Note1:DIR and PUL can be multiplexed as DI4 and DI5 through software settings(20101E address) 0: Pulse mode (factorydefault), 1: DI mode Note2:Analog signal only exist in AA/AS model Note 3: The logic power supply of STO model must be connected, and the logic power supply of non-STO model is optional			
 F8A-1M-3.5-02P-A	Pin	Name	Description
	1	BAT+	Battery interface (only applicable to multi-turn magneto-electric encoder motor)
	2	BAT-	

4.1.1.4 STO interface (X4)

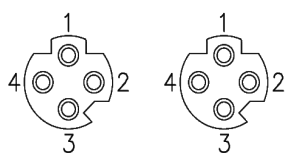
 F8CVM-THR-3.5-06P-4-L30	Pin	Name	Description
	1	STOA+	STO safety interface A STO input level limit value:
	2	STOA-	● Low: $\leq 11V$ ● High: $\geq 15V$ at 2.0 ~2.5mA. STO input level recommend value: ● Low: $\leq 7V$ ● High: $\geq 18V$
	3	STOB+	STO safety interface B STO input level limit value:
	4	STOB-	● Low: $\leq 11V$ ● High: $\geq 15V$ at 2.0 ~2.5mA. STO input level recommend value: ● Low: $\leq 7V$ ● High: $\geq 18V$
	5	24V	Internal 24V DC power supply output, voltage range +18~30V, maximum output current 100mA, only for STO use.
	6	GND	Internal 24V DC power supply output reference ground

4.1.1.5 Power supply and braking resistor interface (X5)

 RB- RB+ DC- DC+ F9CP-7-7.5-04P-A	Pin	Name	Description
	1	RB-	External brake resistance input
	2	RB+	
	3	DC-	DC power supply input-
	4	DC+	DC power supply input+

4.1.2 iSMD IP65 model

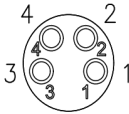
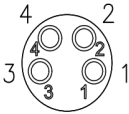
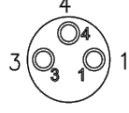
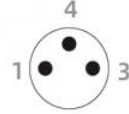
4.1.2.1 Bus communication interface (X1)

 X1A (IN) X1B (OUT) M8-S4D-PTT	Pin	Name	Description
	1	TD+	Send data +
	2	RD+	Receive data+
	3	TD-	Send data -
	4	RD-	Receive data -
	Shell	Shielded	Connect to the integrated motor shell

4.1.2.2 PC debugging interface (X2)

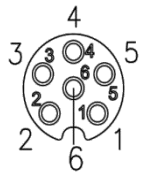
	Description
	Standard USB2.0 Type-C interface Note: To prevent signal loss or disconnection affecting debugging, the USB debug cable should not exceed 3 meters.

4.1.2.3 IO interface and battery interface (X3A&X3B&X3C&X3D)

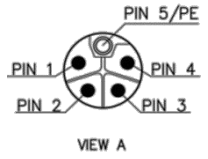
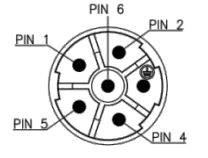
 X3A M8-S4A-GPFM11	Pin	Name	Description
	1	24V	Internal 24V DC power supply output, voltage range +18~ 30V, maximum output current 200mA, only for digital input use. When not connected to logic power, it is the source type common terminal
	2	DI1	Digital input 1, which can be configured as NPN/PNP input
	4	DI2	Digital input 2, which can be configured as NPN/PNP input and can be configured as a probe input
	3	GND	Internal 24V DC power supply output reference ground; When not connected to logic power, it is the sink type common terminal
	Shell	Shielded	Connect to the integrated motor shell
 X3B M8-S4A-GPFM11	Pin	Name	Description
	1	24V	Internal 24V DC power supply output, voltage range +18~ 30V, maximum output current 200mA, only for digital input use. When not connected to logic power, it is the source type common terminal
	2	DI3	Digital input 3, which can be configured as NPN/PNP input and can be configured as a probe input
	4	DI4	Digital input 4, which can be configured as NPN/PNP input
	3	GND	Internal 24V DC power supply output reference ground; When not connected to logic power, it is the sink type common terminal
 X3C M8-S3A-GPFM11	Pin	Name	Description
	1	DO1	2-channel digital output, software can set active/passive output . For active output, external 24V logic power supply needs to be connected to the power socket terminal (X5). DO1 load current : 200mA, DO2 load current : 100mA, Max control voltage : 30V
	3	DO2	
	4	COMO	Output common terminal
	Shell	Shielded	Connect to the integrated motor shell
 X3D M8-P3A-GPFM11	Pin	Name	Description
	1	BAT+	Battery interface (only applicable to multi-turn magneto-electric encoder motor)
	3	BAT-	
	4	-	-
	Shell	Shielded	Connect to the integrated motor shell

Note: For the IP65 model, the DI/DO mode is selected through software settings.
20101D set the DO output (0:no power output ,default value;1:Power output)
20101F set the DI input (0 : PNP input ,default value; 1:NPN input)

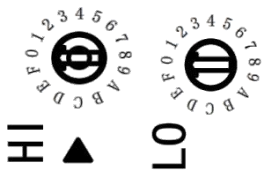
4.1.2.4 STO interface (X4)

 X4 M8-S6A-GPFM11	Pin	Name	Description
	1	24V	Internal 24V DC power supply output, voltage range +18~ 30V, maximum output current 100mA, only for STO use.
	2	GND	Internal 24V DC power supply output reference ground
	3	STOA-	STO safety interface A STO input level limit value: ● Low: ≤ 11V ● High: ≥ 15V at 2.0 ~2.5mA.
	4	STOA+	STO input level recommend value: ● Low: ≤ 7V ● High: ≥ 18V
	5	STOB+	STO safety interface B STO input level limit value: ● Low: ≤ 11V ● High: ≥ 15V at 2.0 ~2.5mA.
	6	STOB-	STO input level recommend value: ● Low: ≤ 7V ● High: ≥ 18V
	Shell	Shielded	Connect to the integrated motor shell

4.1.2.5 Power supply interface (X5)

 iSMD60D M12-P5L-GWFM16	Pin	Name	Description
	1	GNDS	24V DC logic power input reference ground
	2	24VS	24V DC Logical power input. voltage range +18 ~ 30V, maximum input current 1A
	3	DC-	DC power supply input-
	4	DC+	DC power supply input+
	Shell	Shielded	Connect to the integrated motor shell
 iSMD80D M923-P6-GCL	Pin	Name	Description
	1	GNDS	24V DC logic power input reference ground
	2	24VS	24V DC Logical power input. voltage range +18 ~ 30V, maximum input current 1A
	4	DC+	DC power supply input+
	5	DC-	DC power supply input-
	Shell	Shielded	Connect to the integrated motor shell
Note: Applicable to the iSMD60D/iSMD80D PG socket model P000			

4.1.2.6 iSMD Rotary DIP switch

	Model	Description
	AA/AS	Rotaty DIP switch is used to set the CANopen&RS485/Modbus communication station number and baud rate
	EA/ES	Rotaty DIP switch is used to set the network alias for EtherCAT communication
	PA/PS	PA/PS model does not have a rotary dip switch

**Note**

- The logic power supply of STO model must be connected, while that of non-STO model can be optionally connected

CANopen & RS485 communication baud rate setting

To set the baud rate, please follow the steps below:

- (1) First, power off the drive. The drive can only detect the change in the baud rate value when the power is restarted next time;
- (2) Using a 2mm/2.5mm flathead screwdriver, turn the **HI** rotary dip switch to position **C** to prepare for accepting the new baud rate;
- (3) Using a 2mm/2.5mm flathead screwdriver to turn the **LO** rotary dip switch to the position corresponding to the selected bit rate;
- (4) Power on the drive. The drive only reads the rotary switch settings when the power is on.
- (5) Set the baud rate of CANopen&RS485 on the upper computer according to the value selected by the **LO** rotary dip switch;
- (6) By observing the light ring status, if the green light flashes twice, it indicates that the baud rate setting is successful

Table 4-1 Baud rate value corresponding table

LO position	CANopen	RS485/Modbus
0	Default value 500kbps	Default value 38400bps
1	20kbps	2400bps
2	50kbps	4800bps
3	125kbps	9600bps
4	250kbps	19200bps
5	500kbps	38400bps
6	800kbps	57600bps
7	1Mbps	115200bps
8...F	Not used	Not used

**Note**

- When **HI** is set to **C**, **LO** is used to set the baud rate. When **HI** is set to a value between 8 ~ B and D~ F, the setting is invalid and the drive will alarm , at this time, the light ring remains constantly on with a red , and the drive alarms an CAN communication interruption error.

CANopen & RS485 communication station number setting

The combination of **HI** and **LO** represents the station number address from 0 to 0X7F (Hexadecimal), that is, from 0 to 127 in decimal.

To set CANopen & RS485/Modbus communication station number, please follow the step below:

- (1) First, power off the drive. The drive can only detect the change of station number when the power is restarted next time;
- (2) Using a 2mm/2.5mm flathead screwdriver, turn the **HI** rotary dip switch to the position between 0 and 7;
- (3) Using a 2mm/2.5mm flathead screwdriver, turn the **LO** rotary dip switch to the position between 0 and F;
- (4) Power on the drive. The drive only reads the rotary switch settings when the power is on;
- (5) According to the value selected by the **HI/LO** rotary dip switch, set the station number of CANopen&RS485 on the upper computer at the same time;
- (6) Check whether the station number setting is successful by observing the light ring status . If the setting is incorrect, the light ring will remains constantly on with a red, and the drive alarms an CAN communication interruption error. If the setting is successful, it will enter the standby state.



Note

- When setting the CANopen & RS485 communication station number, the range of the **HI** rotary switch should be between 0 and 7, and the **LO** rotary switch should be between 0 and F. If the HI rotary switch is not within the range of 0 to 7, the station number setting will be invalid.

EtherCAT communication network alias setting

The combination of **HI** and **LO** represents the station number address from 0 to 0XFF (Hexadecimal), that is, from 0 to 255 in decimal.

To set EtherCAT communication station number, please follow the step below:

- (1) First, power off the drive. The drive can only detect the change of station number when the power is restarted next time;
- (2) Using a 2mm/2.5mm flathead screwdriver, turn the **HI** rotary dip switch to the position between 0 and F;
- (3) Using a 2mm/2.5mm flathead screwdriver, turn the **LO** rotary dip switch to the position between 0 and F;
- (4) Power on the drive. The drive only reads the rotary switch settings when the power is on;
- (5) According to the value selected by the **HI/LO** rotary dip switch, set the network alias of EtherCAT on the upper computer at the same time;
- (6) Check whether the network alias setting is successful through indicator light ring .



Note

- When setting the EtherCAT communication network alias, the range of the **HI** rotary switch should be between 0 and F, and the **LO** rotary switch should be between 0 and F.
- The Profinet station number is set in the software

4.2 STO function description

4.2.1 Failure-safety of iSMD drive equipment

iSMD is designed to fail-safe principles. This means that in case of failure, the device will enter a safe state. The safe state is defined as the de-energized state of the motor.

STO (safety torque off) function refers to the use of safety input signals to forcibly turn off the signal of the internal power part of the servo driver circuit, so as to achieve the safety function of turning off the motor current and turning off the output torque of the motor. This safety function has a two-channel architecture and ensures the safe blocking of the PWM gate control by two independent digital low-active 24 V input. It should be noted that the safety function ensures the cut-off of motor torque, but it cannot ensure the immediate stop of a running motion. If the STO function is triggered while the motor is running, the drive cannot actively brake the load and the load continues to run until it is stopped by friction, a mechanical brake or another external measure.

The safe pulse blocking function Safe Torque Off (STO) deactivates the signal of the servo drive's internal electric circuit, so that the motor current and the motor's output torque are both switched off in order to reach a safe state. iSMD has two channels for STO input signals. The drive will switch off the motor current and motor output when one of the two STO signals is triggered.

The STO function interrupts the power supply to the motor by two-channel inhibition of the control pulses to the power stage transistors. Thus, a rotating field and electrical torque can no longer be generated in the motor.

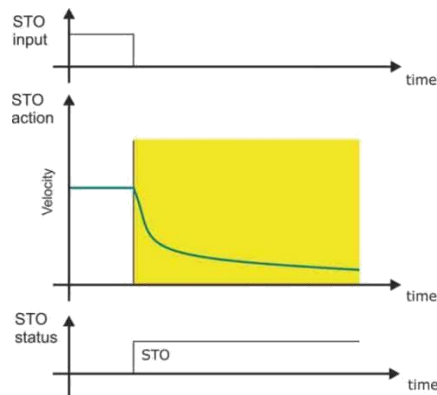


Figure 4-1 STO safety function

4.2.2 Implementation

The following figure shows the implementation of the safe pulse blocking function STO in the iSMD, including its external wiring with an emergency stop device.

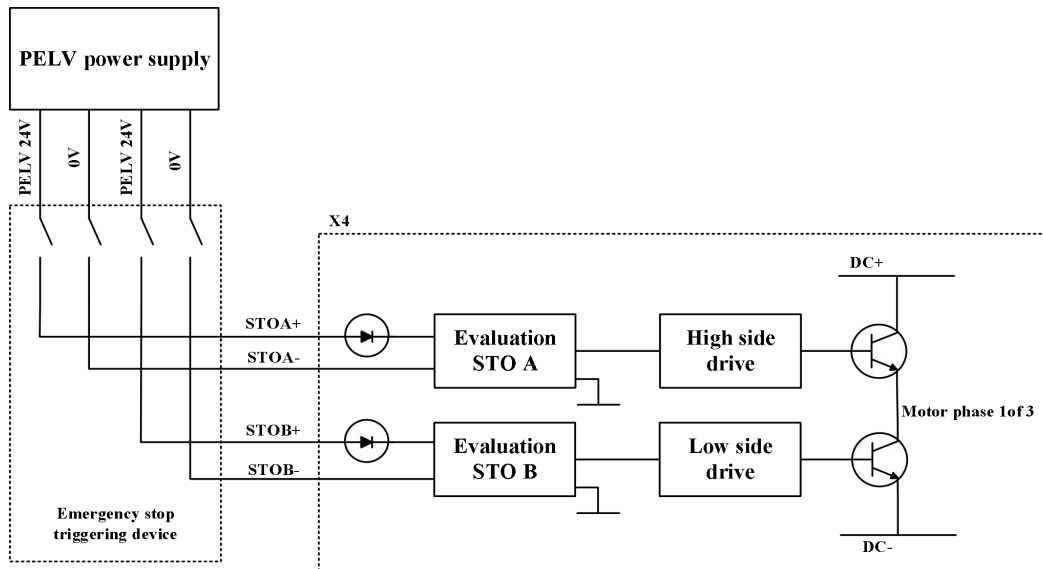


Figure 4-2 STO wiring

After you reconnect the 24VDC, you must enable the integrated motor again!

The following limit values apply to switching levels of the STO input:

- Low: $\leq 11V$
- High: $\geq 15V$ at 2.0 ~2.5mA

Recommended STO input level:

- Low: $\leq 7V$
- High: $\geq 18V$

4.2.3 Requirements



Note

- The availability of safety functions taken by itself does not guarantee safety. Carry out a comprehensive risk assessment in order to make sure that the machine or plant is safe. It is the responsibility of the end product or application designer to ensure safety and compliance with relevant regulations. Safety elements must be correctly integrated into the safety concept of the machine or plant. Make sure that the safety concept of the machine or plant complies with all applicable laws and standards.
- In applications with vertical loads under gravity, the STO function is not sufficient as the sole safety function.
- Before activating the STO function, it is essential to reserve a maximum break time of 20ms. This break time will decisively influence the reaction time of the safety functions, and thus the residual paths and periods you must consider! Be sure to validate the overrun time of the complete system as a basis for calculating the overall reaction time of the safety system!



Warning

- Voltage on the machine or plant remains when STO function is active
- Death or severe injuries by electrocution:
 - The STO function can not de-energize the device and therefore does NOT provide protection against electrocution!
 - You must take other suitable measures in order to de-energize the machine or plant.
- If the drive system is in motion when the STO function is triggered, it will coast to standstill. Consider this residual movement and timespan when you calculate the safety circuit of the machine or line !
- In processes without personnel protection, it is strictly prohibited to use the STO function to stop the drive for the following reasons:
 - After the equipment triggers the STO function in the running state, the load will continue to slide to a stationary state due to inertia. The shutdown process is completely uncontrolled, which can easily cause safety accidents such as collisions and pinches. It does not have the safety protection level required for the emergency stop function.
 - The STO function cannot guarantee the deceleration control logic of the brake. After the STO function is triggered, the brake will directly close without sufficient deceleration of the motor, and it will bear a dynamic load that far exceeds the design. Frequent use of the STO function to stop the drive will increase the wear of the brake.
- Dangerous movement through external force (for example gravity acting on vertical loads) may lead death or severe injuries (limb loss, bone fractures, bruising) .
 - Prevent uncontrolled movement by suitable precautions (for example using mechanical brakes).
 - Make sure that the measures you take are in accordance with the necessary safety level.
- Jolts and coast-down movement of the machine may lead to Death or severe injuries (limb loss, bone fractures, bruising)
 - When designing your safety measures, always consider the maximum residual paths and periods that are possible in case of accidental movement (jolts and coast-down). Calculate buffer times on the basis of the maximum possible velocity.
 - The maximum possible velocity of the drive consists of the maximum possible acceleration and the fault reaction time as well as the actively monitored speed limit.

Checks

Check the STO function when commissioning the machine or machining line. Document the check and its result in an acceptance report.

For a regular check, trigger the STO function at least once every three months.

PELV power supply

To ensure that the electrical values for extra-low voltage with safe isolation cannot be exceeded, use a 24V or 48V power supply unit that complies with PELV (Protective Extra Low Voltage) regulations in accordance with IEC 62368-1 ES1.

**Note**

- The 24 V or 48V power supply unit must master the maximum voltage interruption of 3ms as specified in IEC 62368-1 ES1.
- Failure or accidental triggering of STO function can cause damage to persons and machinery
 - Only use PELV-conforming power supply units.

Cable protection

Install all cables so that they are adequately protected.

Protective cable installation is a precondition for the correct performance of the STO function.

**Note**

- Cables must never be damaged by extraneous causes. Mechanical damage can accidentally trigger the STO function. Also, mechanical damage can cause current leakage between damaged cables, which again can accidentally trigger the STO function.

Specification parameter

The specification parameters in the following table are applicable to the STO function of JD2:

Table 4-2 STO function parameters

Item	Specification
Maximum Performance Level according to EN ISO 13849	PL e
Maximum Safety Integrity Level according to IEC 61508/IEC 62061	SIL 3
PTI (Proof Test Interval)	20 year
PFH (Probability of dangerous Failure per Hour)	2.55 e-8
DC (Diagnostic Coverage) average	88.45%
MTTFd (Mean Time To dangerous failure)	1989 year
Systematic capability (SC)	3
Category	Cat. 3
Mission time of the PDS (SR)	20 year
Response time	< 20ms
Note: The temperature condition for parameter calculation is 75°C	

Diagnostic function

There is a continuously running diagnosis function which checks the correct function of the STO related hardware and there are drive firmware functions which check the consistency of the two STO input signals. In case of any detected malfunction, a detection of signal inconsistency or unexpected loss of one or both STO signals while the drive is enabled, a STO related drive fault is issued.

The actual status of the STO function can be monitored by looking up the object dictionary parameter "STO _Status" (0X60F7.11) in the upper computer software.

Table 4-3 STO Status

STO Status	Error code	Error_state2 (Note1) Bit6:STO1 Bit7:STO2 (Note2)	Check when	Fault resetable	Description
0	-	0	-	Yes	STO A = H, STO B = H, no diagnosis error
1	0xFF21	Bit7=1	Always	NO	Self diagnosis CPU can't find STO "H" signal
2	0xFF22	Bit7=1	Always	NO	Impossible (theoretically)
3	0xFF1A	Bit6=1	Always	Yes	STO A = H -> L while STO B = H for >= 100ms
4	0xFF24	Bit7=1	Always	NO	Impossible (theoretically)
5	0xFF1B	Bit6=1	Always	Yes	STO B = H -> L while STO A = H for >= 100ms
6	0xFF10	Bit6=1	Attempt to be enabled/ enabled drive	Yes	STO A and STO B = H->L within 100ms while the drive was enabled or STO A and/or STO B = L when the drive is going to be enabled (Note3)
7	0xFF20	Bit7=1	Always	NO	Hardware fault

Note1: more than 1 bit can be set at a time

Note2: errors with STO2 (bit 7) = 1 are caused by severe faults in the STO unit, fault reset by control word is not possible

Note3: if this status persists despite H signal on both STO inputs, damages inside the STO circuitry have to be considered

Chapter 5 System wiring

5.1 Electrical wiring diagram

5.1.1 iSMD IP20 model electrical wiring diagram

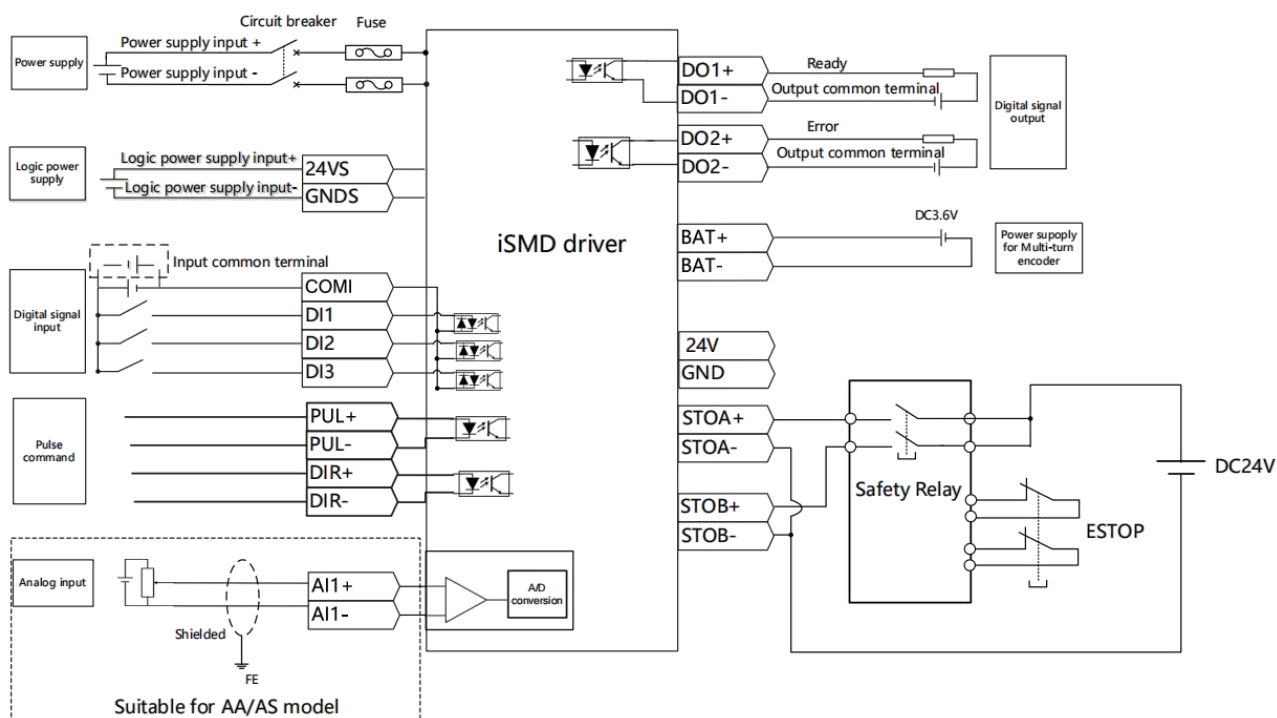


Figure 5-1 IP20 model electrical wiring diagram (Use STO)

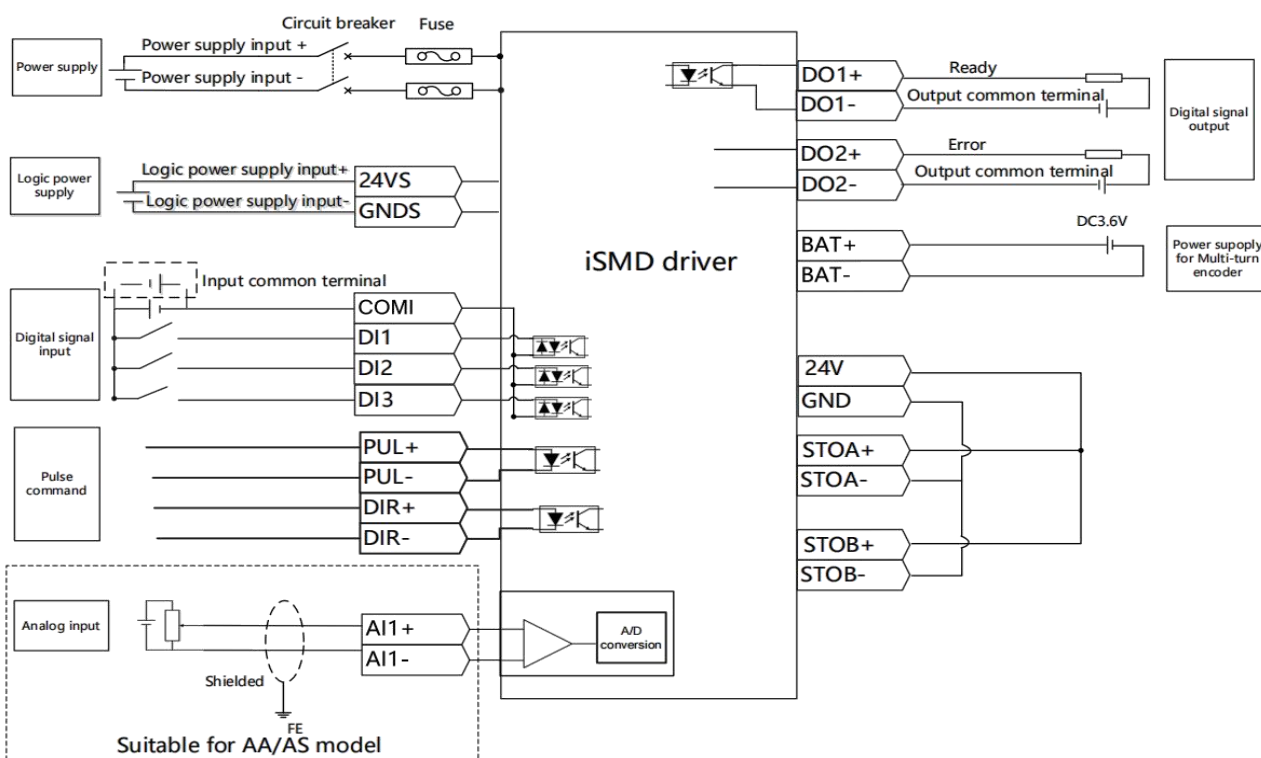


Figure 5-2 IP20 model electrical wiring diagram (Not use STO)

5.1.2 iSMD IP65 model electrical wiring diagram

For the iSMD IP65 model, the I/O needs to be set through software to select the corresponding mode. The electrical wiring of this model will be explained in several cases.

5.1.2.1 Digital input signal

Taking DI1~DI2 as an example, the DI1~DI4 interface circuits are the same.

- When the X5 interface is connected to a logic power supply, 24V and GND are the logic power supply output. When the input interface is connected to a PNP type device, 20101F (DI input mode) needs to be set to 0 (PNP mode) in the software. The wiring method is as follows:

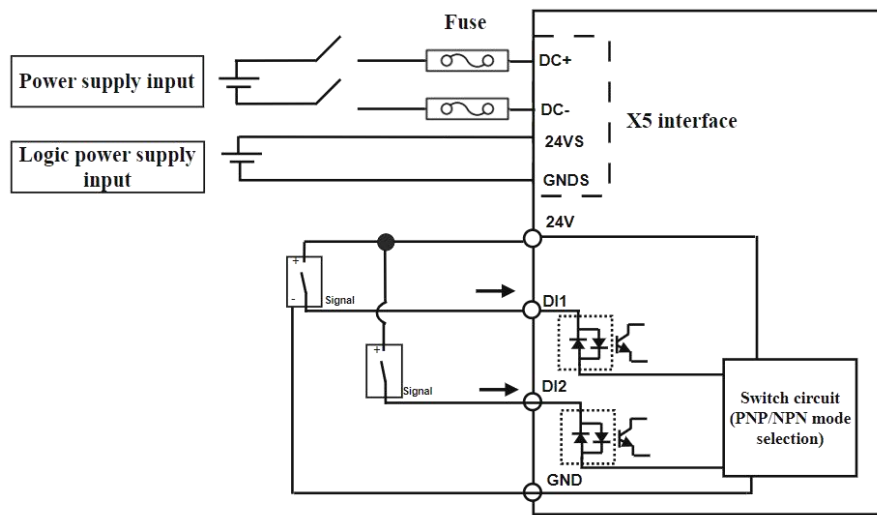


Figure 5-3 PNP mode (With logic power supply)

- When the X5 interface is connected to a logic power supply, 24V and GND are the logic power supply output. When the input interface is connected to a NPN type device, 20101F (DI input mode) needs to be set to 1 (NPN mode) in the software. The wiring method is as follows:

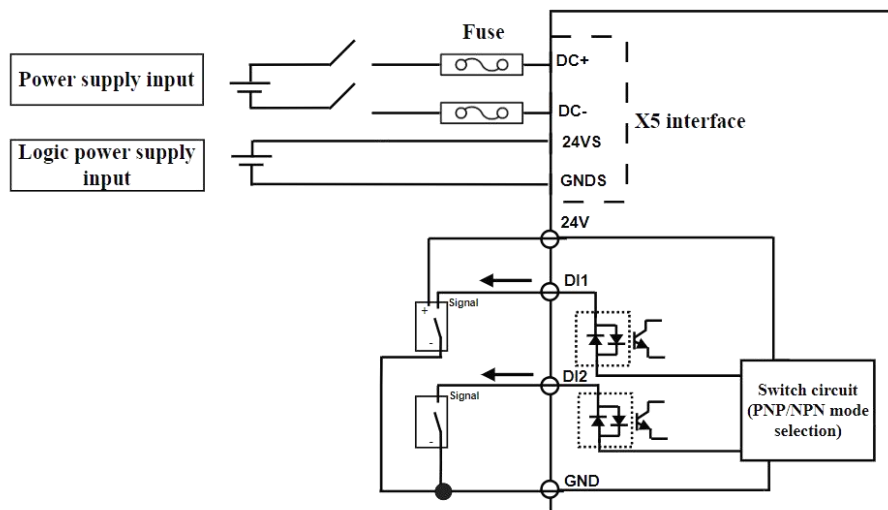


Figure 5-4 NPN mode (With logic power supply)

- When the X5 interface is not connected to the logic power supply, the input interface only supports PNP type device. set 20101F (DI input mode) to 0 (PNP mode) in the software. The wiring method for PNP mode is as follows:

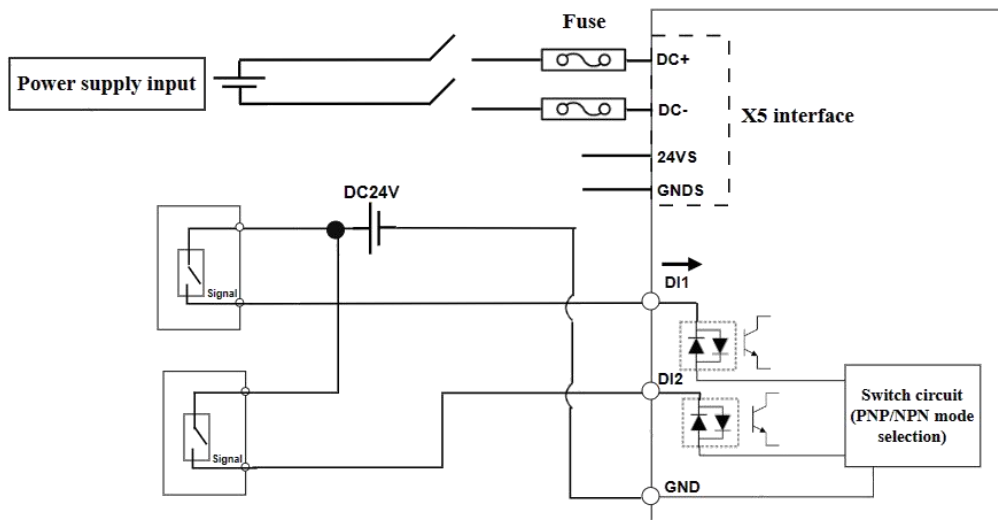


Figure 5-5 PNP mode (Without logic power supply)

5.1.2.2 Digital output signal

Taking DO1 as an example, the DO1~DO2 interface circuits are the same.

- When the X5 interface is connected to the logic power supply, The source output connection method requires setting 20101D (DO output mode) to 1 (source output) in the software. The wiring method is as follows:

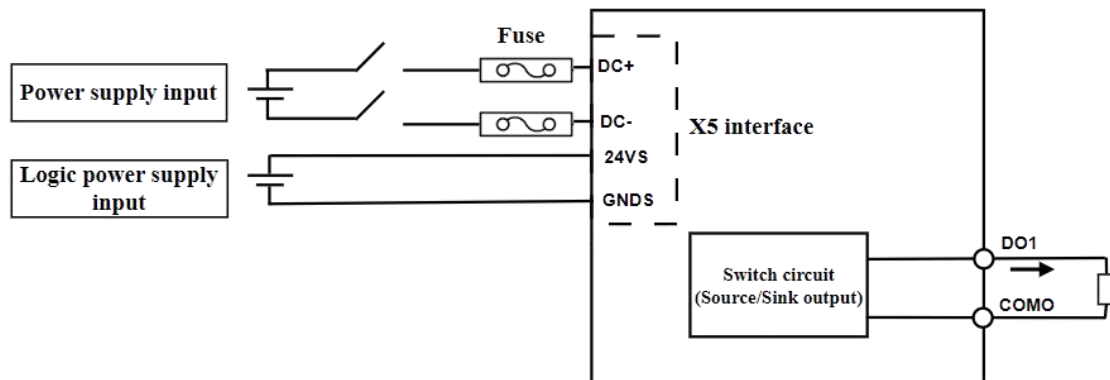


Figure 5-6 Source type output connection method(With logic power supply)

- When the X5 interface is connected to the logic power supply, The sink output connection method requires setting 20101D (DO output mode) to 0 (sink output) in the software. The wiring method is as follows:

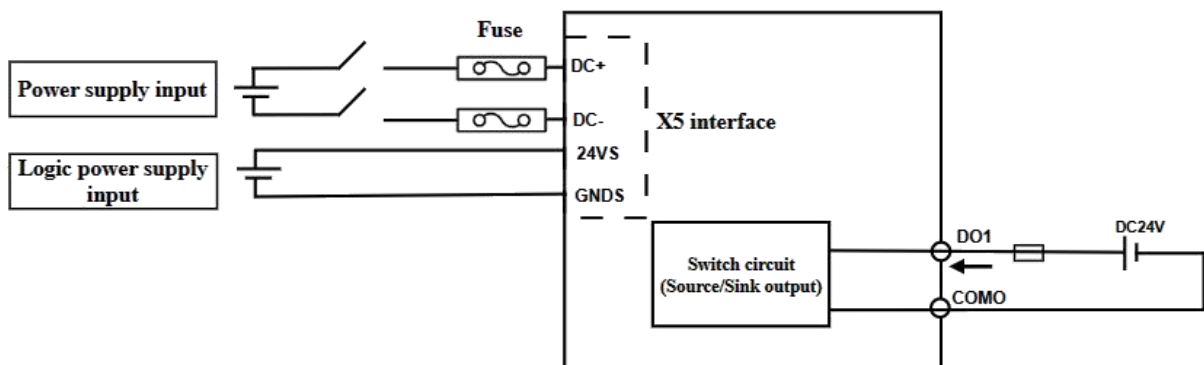


Figure 5-7 Sink type output connection method(With logic power supply)

- When the X5 interface is not connected to the logic power supply, at this time, the output interface only supports the sink output connection method. 20101D (DO output mode) needs to be set to 0 (sink output) in the software. The wiring method is as follows:

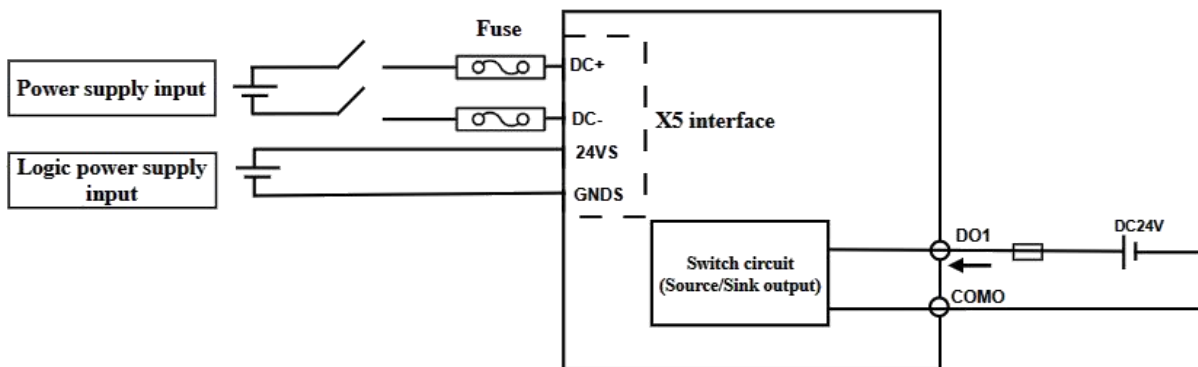


Figure 5-8 Sink type output connection method (Without logic power supply)

5.1.3 STO electrical wiring diagram

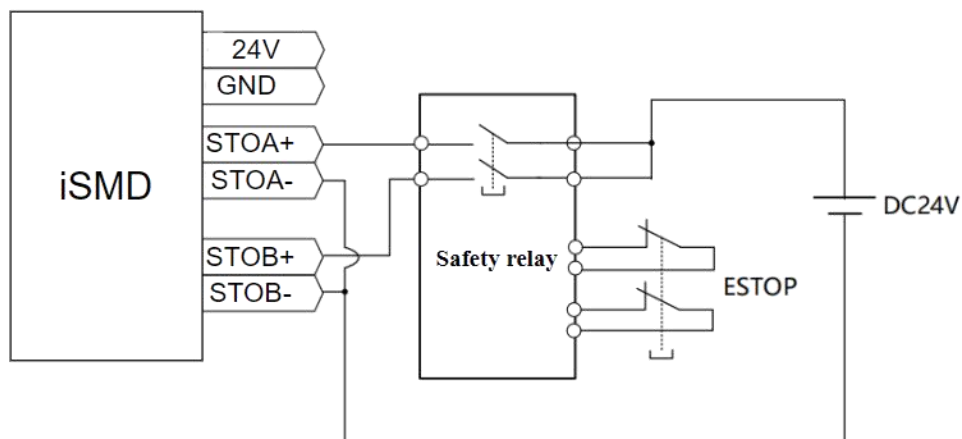


Figure 5-9 STO wiring diagram (Use STO)

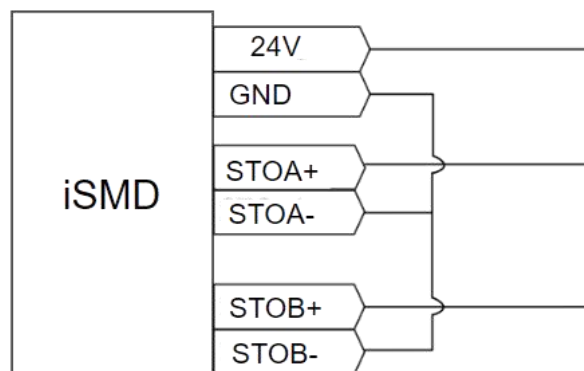
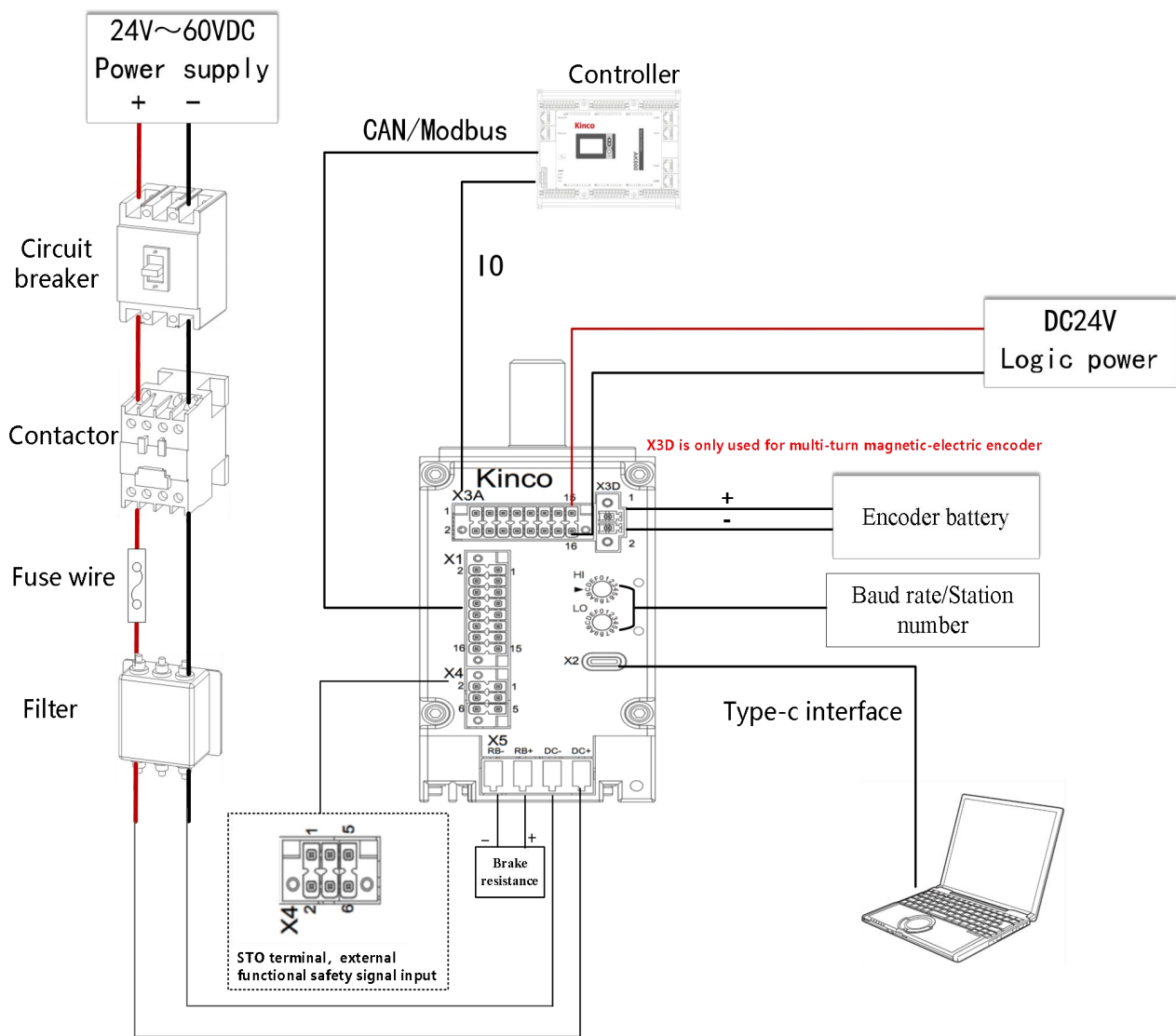


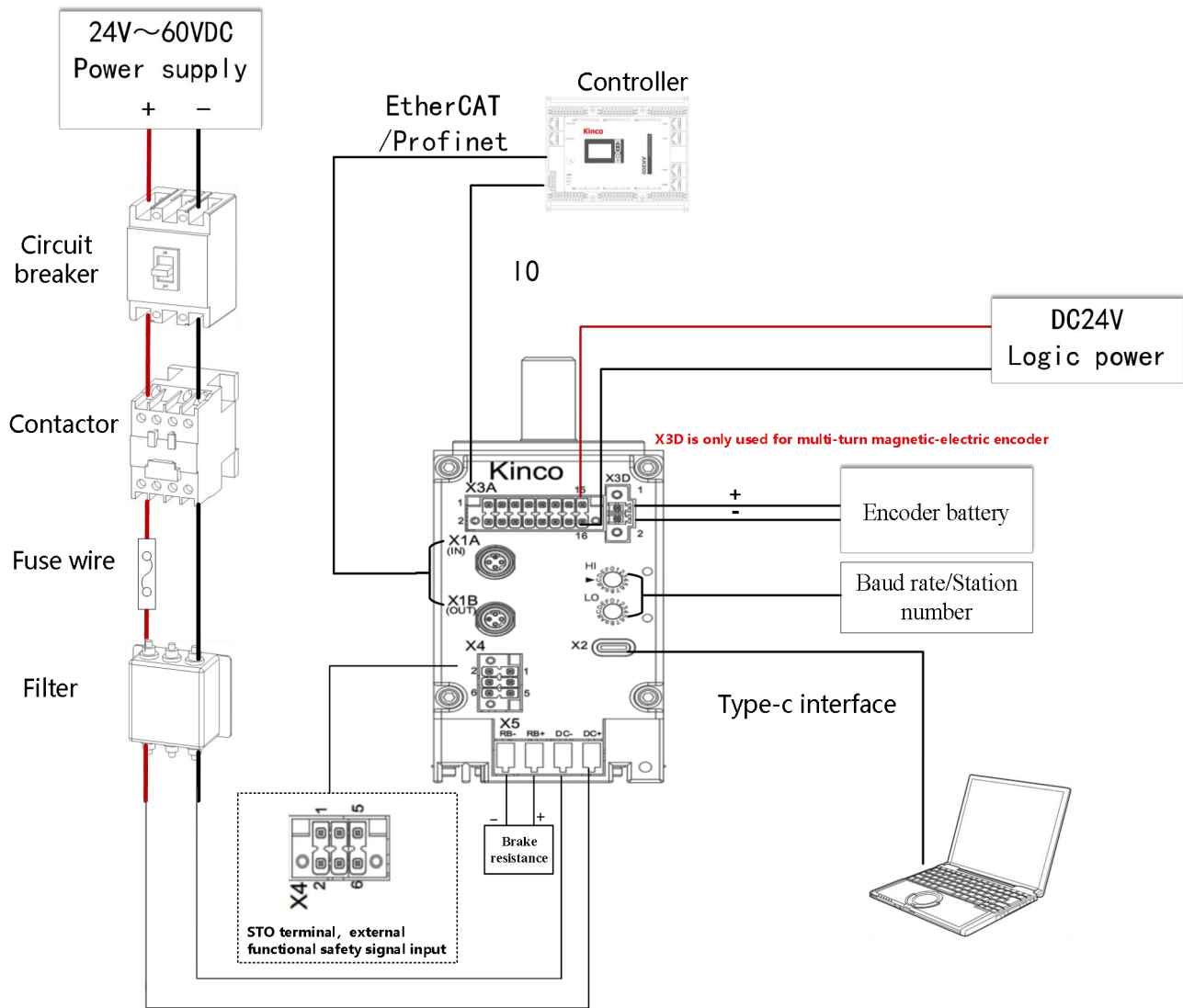
Figure 5-10 STO wiring diagram (Not use STO)

5.2 External wiring diagram

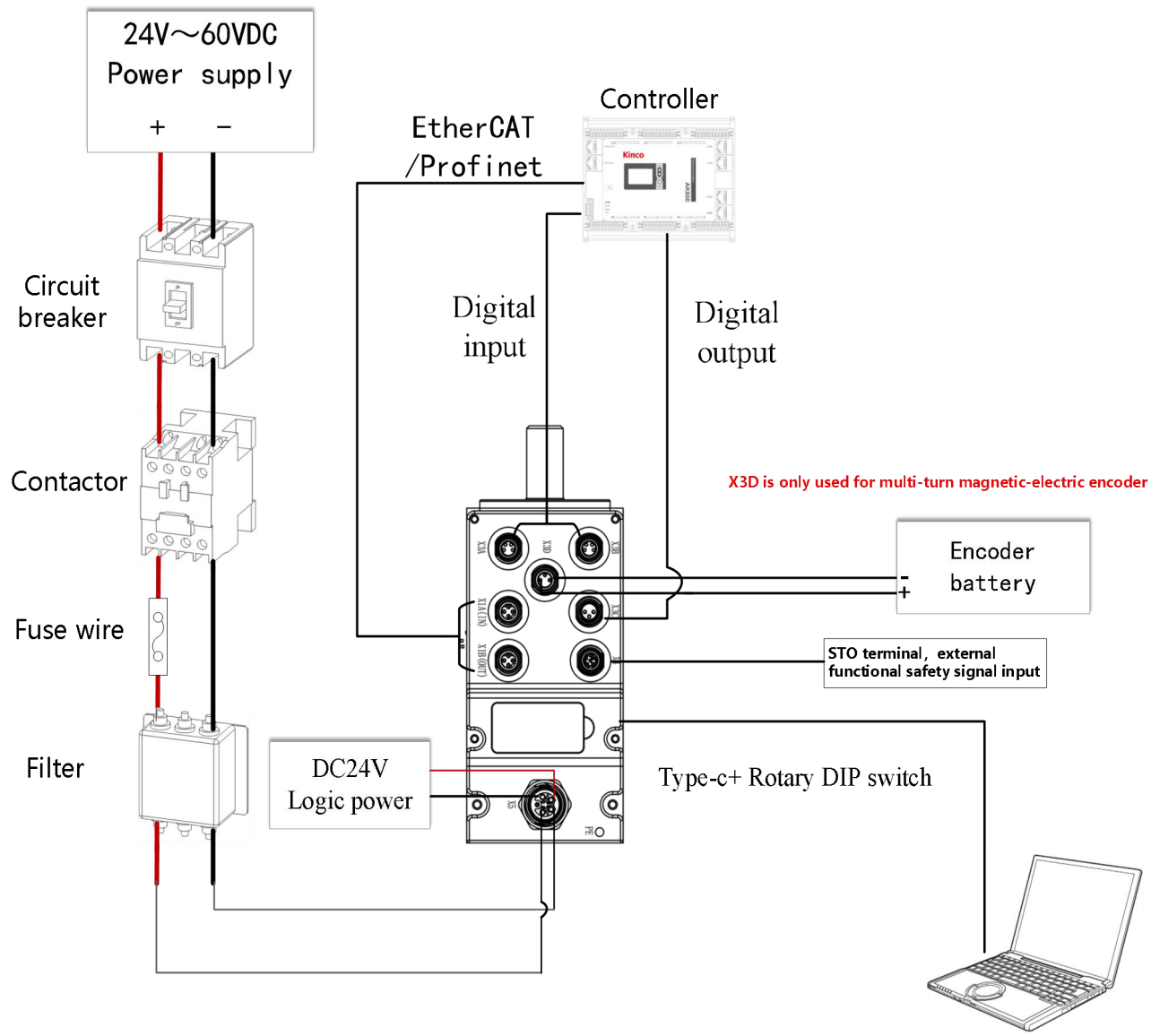
IP20 AS model



IP20 ES/PS model



IP65 ES/PS model



5.3 Power supply description

iSMD series drive must use the monotonic power supply shown in Figure 5-11 when powering on. If the non-monotonic power supply shown in Figure 5-12 is used, when the deviation is close to any threshold voltage, the non-monotonic slope may cause problems such as abnormal start of the driver, resulting in an error failure.

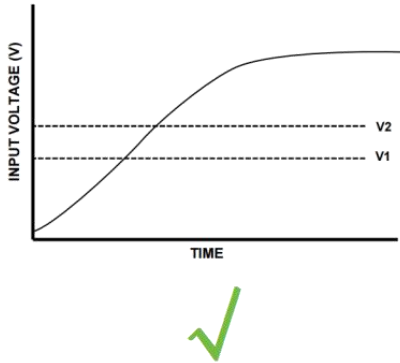


Figure 5-11 Monotonic power supply

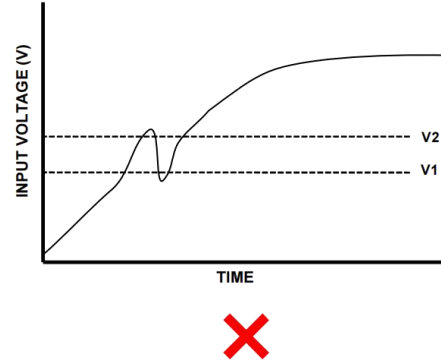


Figure 5-12 Non-monotonic power supply

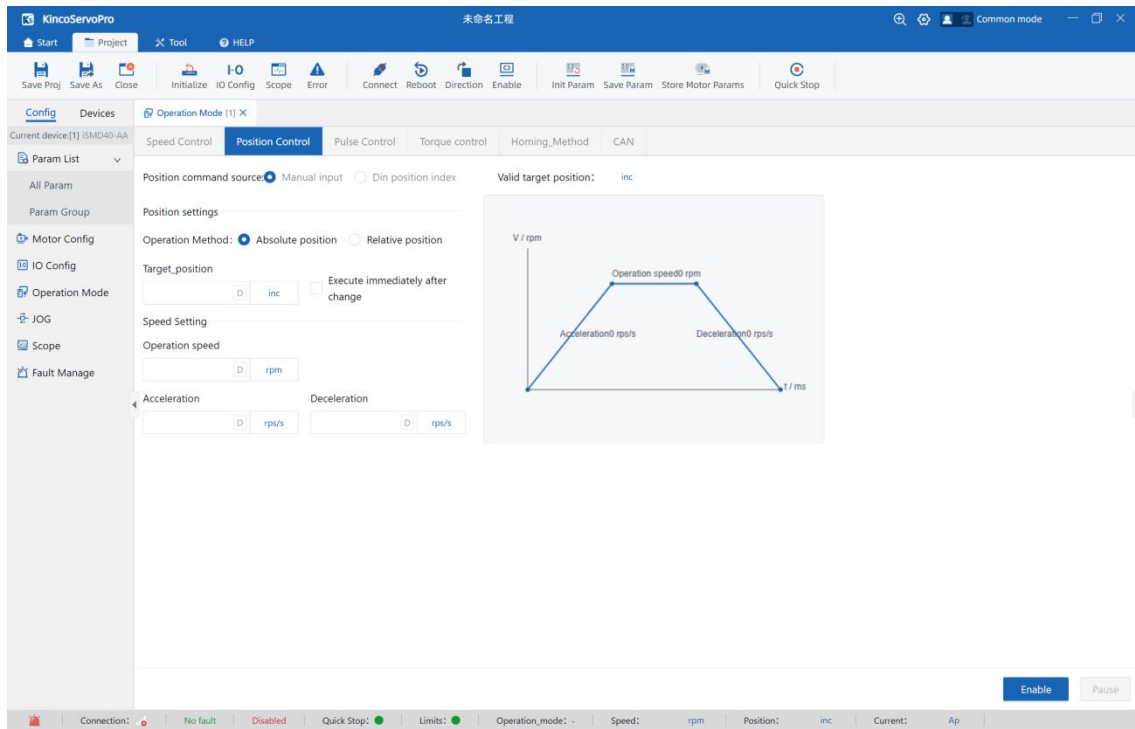


Note

- In figure 5-11 and figure 5-12, V1 indicates 10V voltage, and V2 indicates 17V voltage.
- The drive has an electrolytic capacitor. Ensure that the input power is working properly before powering on the driver.

Chapter 6 Kincoservo PRO software

This chapter mainly introduces the usage method of the KincoServo PRO upper computer debugging software (hereinafter referred to as "KS PRO").



6.1 Software installation

6.1.1 Software acquisition

The software can be downloaded from the Kinco official website.

Installation requirement

KincoServo PRO can run on the following versions of Windows clients:

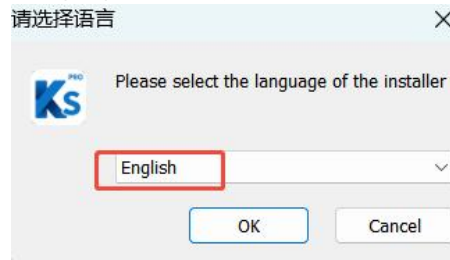
- Windows 10 SAC 1709 and later version
- Windows 10 enterprise edition 2015 LTSC
- Windows 10 enterprise edition 2016 LTSC
- Windows 10 enterprise edition 2019 LTSC
- Windows 10 enterprise edition 2021 LTSC
- Windows 10 enterprise edition multi-session
- Windows 10 IOT enterprise edition SAC
- Windows 10 IOT enterprise edition 2019 LTSC
- Windows 10 IOT enterprise edition 2021 LTSC
- Windows 11

- Windows 11 enterprise edition multi-session
- Windows 11 IOT Enterprise 2024 LTSC

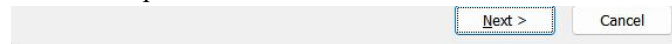
6.1.2 Software installation

Double-click [**servo-pro-amd64-installer**] to start the installation, and you will see the following installation interface;

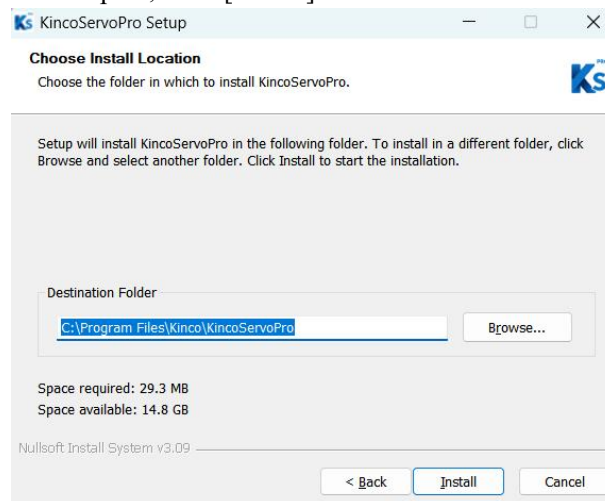
- First, select the language(Default language:Chinese);



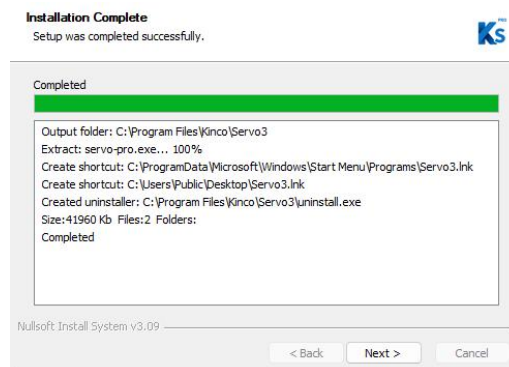
- Click [**Next**] to enter the installation process



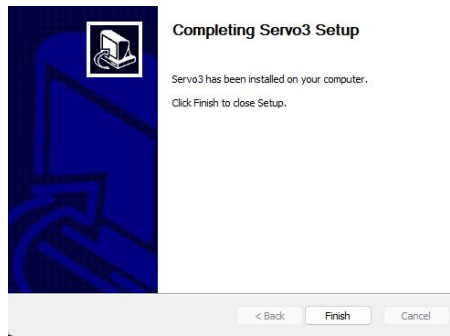
- After choosing the installation path, click [**install**] to start the installation.



- After the installation is completed, click [**Next**];



- Finally, click [**Finish**] to complete the installation



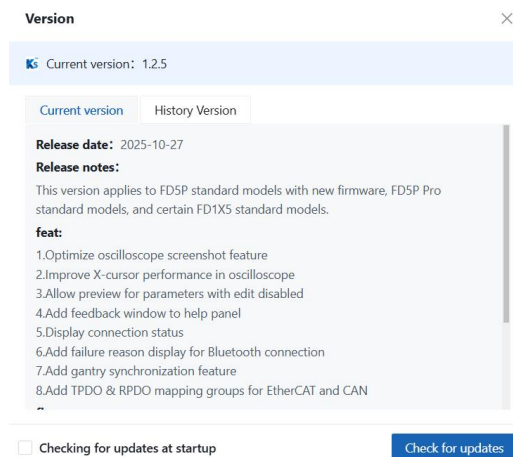
6.1.3 Version update

After the installation of KS PRO is completed, subsequent version update will be done within the software. The specific operation steps are as follows

- After entering the KS PRO software interface, click the [**Help**], and then select [**Version**].

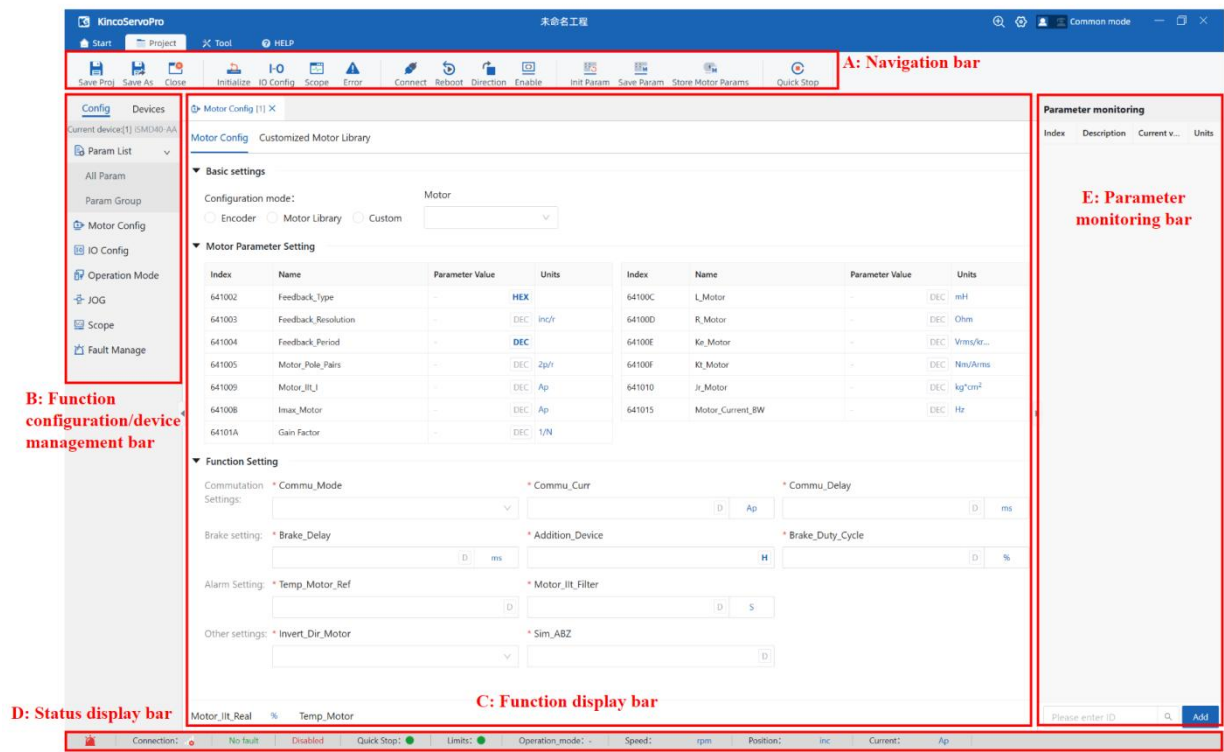


- In the pop-up version interface, click the [**Check for updates**] at the lower right corner. If there is a new version of the software, the [**Update**] button will appear at the lower right corner and click it to complete the update.



6.2 Quick start guide

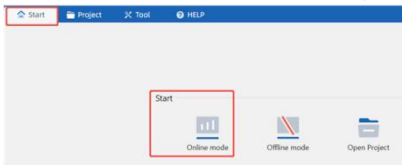
6.2.1 Software window introduction



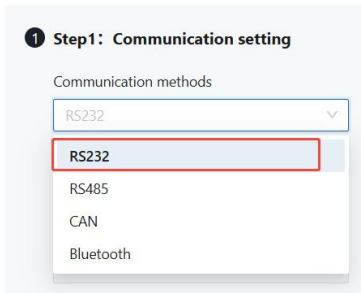
A	[Navigation bar]: Provide quick navigation for common basic servo function
B	[Function configuration/Device management bar]: Provide relevant navigation for servo function configuration and equipment management
C	[Function display bar]: Display the interaction interface between the upper computer and the lower computer
D	[Status display bar]: Display the current operating status of the servo, including data such as speed, position and current feedback
E	[Parameter monitoring bar]: Used for monitoring servo parameter

6.2.2 Connect to the computer

- First, in the KS PRO software, find the [Start] menu in the navigation bar and click [Online mode].



- In the pop-up [Online mode] window, select the communication methods.

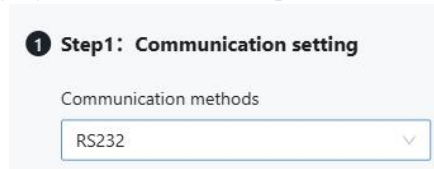


485 communication connection

- iSMD communicates with the PC: Communicate using a USB-type C cable via the X2 interface of the drive or via the X1 interface with a 232/485 to USB isolation converter.

(1) When using the USB-TypeC cable to connect to the host computer, note that:

- Select the RS232 interface on the communication settings page. Since the Type-c nterface of iSMD uses the 232 communication protocol by default, it can be successfully connected to the host computer without changing the communication protocol through the internal address.



- If only the USB cable is used to communicate with the host computer and there is no power supply outside the drive, the debugging cable suddenly drops during data storage. In this case, the drive will report to EEPROM.
- If the power supply suddenly cuts off during the process of downloading the firmware using the Type-C interface, resulting in communication interruption, the drive needs to be re-powered before scanning can be performed again to download the firmware.

(2) When using X1 interface connect to the host computer, note that:

- The default communication protocol of the X1 interface of the iSMD is MODBUS. You need to change the RS485 communication protocol (modbus address mapping is 0X86C0) to 232 by modbus. Perform the following steps to restart the driver after the operation is complete.

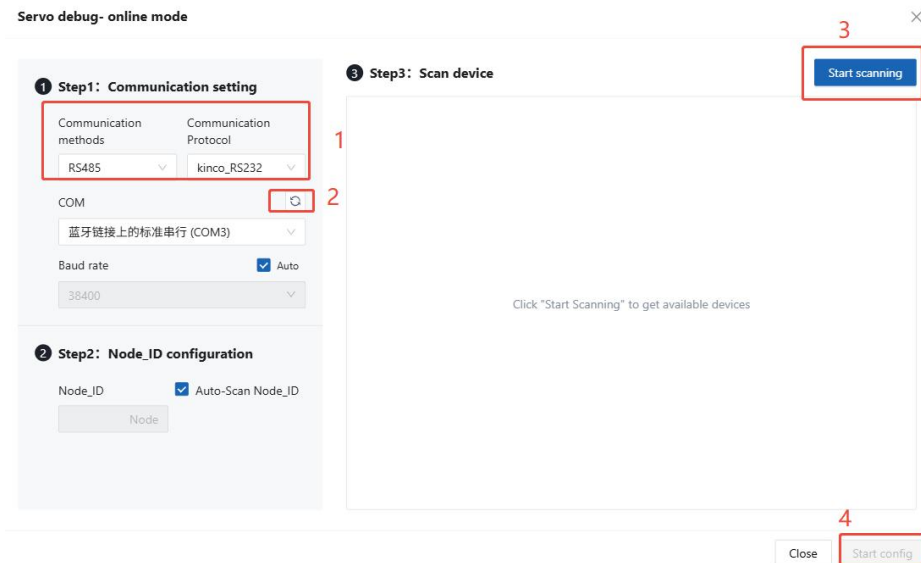
Step	MODBUS message	Message specification (ID=1)
1	01 06 86 C0 00 01 61 7E	65100C write 1 , switch to the 232 protocol immediately
	01 06 86 C0 00 03 E0 BF	65100C write 3, switch to the 232 protocol, save the Settings, and take effect after restart
2	01 06 29 10 00 01 41 93	Storage control parameter

Relevant parameter:

Internal address	Parameter name	Meaning	Default
100B0008	Equipment station number	Driver station number	1
2FE20010	RS485 baud rate	Set the baud rate of the RS485 port (Modbus address: 0X2600) Set value Baud rate 1080———9600 540———19200 270———38400 90———115200 Note: Need to save and restart.	270
65100C08	RS485 Communication protocol selection	bit0: 0: Using MODBUS protocol 1: using RS232 communication protocol bit1: 0: Changing bit0 takes effect immediately 1: Changing bit0 takes effect when the storage restarts	0
65100E10	RS485 mode	Data bit =8, stop bit =1, no parity check	Fixed value

CANopen communication connection

iSMD can also communicate with the computer through the CANopen protocol. Since one CAN bus can mount multiple devices, this function realizes the cascaded debugging of multiple drives. The communication operation is as follows:



(1) Select the CAN interface on the communication setting page and select the type of CAN device to be used.

Currently, two kinds of CAN devices are supported to connect with the KSPRO:

- **Peak CAN device**
- **Chuangxin CAN**

- (2) Click refresh serial port button  and then select the serial port ;
- (3) Click **[Start scanning]** , the addable device will be displayed below ;
- (4) In the scanning device column, select the device you want to connect, click **[Start config]** to complete the device connection and project creation.



Note

- The baud rate of CAN communication (default :500 KBit/s) , and the object dictionary address is [2F8100].
- Select the device station number, the default is 1, the object dictionary address is [100B00].
- The communication between KSPRO and drive adopts CANopen protocol. Therefore, during the communication of the upper computer, do not use other devices (such as PLC or Linux devices) to send CAN messages (such as SDO/PDO messages) to the drive, otherwise the communication of the upper computer will be abnormal or even unable to communicate
- The software will automatically match the default configuration file for the standard product. If necessary, you can also manually switch by clicking the drop-down box.

6.2.3 Save & Close project

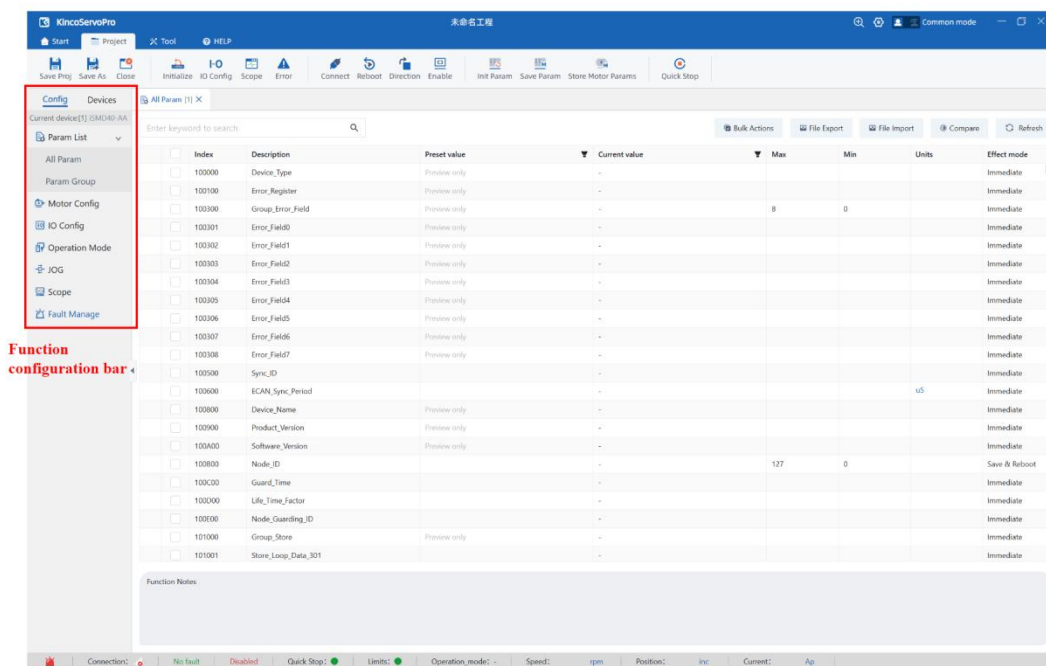
In the navigation bar, find the **[Project]** and perform the operation.



- Click the **[Save Proj]** button under the **[Project]** , enter the project name and folder path, and click "Confirm". Then the saving project is completed successfully. When you click **[Save Proj]** again later, there is no need to redefine the name and path.
- Click the **[Save As]** button under the **[Project]** , enter the project name and folder path, and click "Confirm". Then the saving project is completed successfully.
- Click the **[Close]** button under the **[Project]** to choose to **close without saving** or **save and close** the current project.

6.2.4 Parameter monitoring and management

In the function configuration bar, find the **[Param List]** menu, which includes the "All Param" and "Param Group" option.



All parameter list

By using the **[All Param]** option, you can access all the object dictionaries of the servo drive, and you can quickly locate and filter the target parameter through keyword search. The parameter preset values will retain the value by the user. After the parameters are issued, the current parameter value will be updated. The interactive buttons support operations such as parameter modification, parameter monitoring, import/export parameter file, parameter compare, initializing control parameters, and saving control parameters, etc.

Enter keyword to search Search Box

Function Interaction Button

☐	Index	Description	Preset value	Filter Button	Current value	Max	Min	Units	Effect mode
☐	329600	Service_Request			0	HEX			Immediate
☐	329700	Service_Status			14	HEX			Immediate
☐	329800	Service_Timeout			0	DEC		ms	Immediate
☐	400400	Pos_Abs			40609	DEC		inc	Immediate
☐	600700	Abort_Connection_Mode			0no process	DEC			Immediate
☐	603F00	Error_Code			0	HEX			Immediate
☐	604000	Controlword			6 Q	Base Conversion			Immediate
☐	604100	Statusword			4231	HEX			Immediate
☐	605400	Quick_Stop_Mode			0stop without control	DEC			Immediate
☐	605800	Shutdown_Stop_Mode			0stop without control	DEC			Immediate
☐	605C00	Disable_Stop_Mode			0stop without control	DEC			Immediate
☐	605D00	Halt_Mode			1stop by current ramp	DEC			Immediate
☐	605E00	Fault_Stop_Mode			0stop without control	DEC			Immediate
☐	606000	Operation_Mode			-4Pulse train Control	DEC			Immediate
☐	606100	Operation_Mode_Buff			0	DEC			Immediate
☐	606300	Pos_Actual			-4	DEC		inc	Immediate
☐	606400	Pos_Actual			-4	DEC		inc	Immediate
☐	606500	Max_Following_Error			524288	DEC	268435455	0	inc
☐	606700	Target_Pos_Window			327	DEC	268435455	0	inc
☐	606800	Position_Window_time			10	DEC	32767	0	ms

Function Notes **Function Comment Interface**

driver controlword
bit0: Switch_on
bit1: Enable_voltage
bit2: Quick_stop
bit3: Enable_operation

6.2.4.1 Parameter modification

- Double-click the preset value cell or click the [Edit] button;

☐	60FE01	Digital_Outputs	Write Prohibition	0	HEX
☐	60FF00	Target_Speed	Click [Edit]	0	DEC
☐	640300	Motor_Catalogue_Number		no data	String
☐	641000	Group_Motor	Write Prohibition	32	DEC
☐	641001	Motor_Num		BX	ASCII
☐	641002	Feedback_Type		4	HEX

- Enter a value in the preset value field. The base of the input data is displayed on the right. For some parameters, you can click to switch between base / other data type;

☐	60FE01	Digital_Outputs	Write Prohibition	0	HEX
☐	60FF00	Target_Speed	1000	DEC	DEC
☐	640300	Motor_Catalogue_Number	Enter preset value	no data	String
☐	641000	Group_Motor	Write Prohibition	32	DEC
☐	641001	Motor_Num		BX	ASCII
☐	641002	Feedback_Type		4	HEX

- Some parameters are composed of multiple Bit. You can click the [+] on the left side of the parameter row to expand the parameter and view/edit it in more detail. If you only need to edit the data of a certain Bit , you can click the "Copy" button in the current value cell to copy the current value to the preset value, and then edit it.

☐	101100	Group_Restore_Data	Write Prohibition	1	DEC
☐	101101	Restore_All_Data_301		1	HEX
☐	101400	Emergency_Mess_ID		81	HEX
☐	101601	Consumer_Heartbeat		0	HEX
	bit0~7	heartbeat message ID number		0	
	bit16...	interval time(ms)		0	
☐	101700	Producer_Heartbeat_Time		0	DEC

- Press the Enter key (in editing mode) or click the [Send] button (in exiting editing mode) to send parameters.

☐	60FE01	Digital_Outputs	Write Prohibition	0	HEX
☐	60FF00	Target_Speed	1000	DEC	0
☐	640300	Motor_Catalogue_Number	Press Enter Or Click Send	no data	String
☐	641000	Group_Motor	Write Prohibition	32	DEC
☐	641001	Motor_Num		BX	ASCII
☐	641002	Feedback_Type		4	HEX

- If the parameter modification is successful, the current value will be updated to the preset value.

Preset value	Current value
Write Prohibition	-4
Write Prohibition	180000
Write Prohibition	1
Write Prohibition	0
1000	1000

- If the current value is different from the preset value, the preset value input box will turn light yellow.

6.2.4.2 Parameter monitoring

- Select the target parameter with the left mouse button, right-click and choose [Add to watch]. The target parameter will appear in the parameter monitoring window on the right;

The screenshot shows the Kincoservo PRO software interface. The main window displays a list of parameters with columns for Index, Description, Preset value, Current value, Max, Min, Units, and Effect mode. A red box highlights the 'Add to Watch' button in the context menu. Another red box highlights the 'Parameter monitoring' window on the right, which shows a list of monitored parameters with columns for Index, Description, Current value, and Units. A third red box highlights the 'View Monitoring Parameters' button in the context menu.

- You can also search for and check the target parameter in the search box at the bottom of the parameter monitoring bar, and then click [Add];

The screenshot shows the Kincoservo PRO software interface. The search box at the bottom of the parameter monitoring bar contains the text 'Target_Speed'. The 'Add' button is highlighted. The context menu is open, showing the 'Add to Watch' button.

- In the parameter monitoring bar, you can modify parameters (except for those that are prohibited from being written). Enter the modified value in the current value box and press the Enter key to send it.

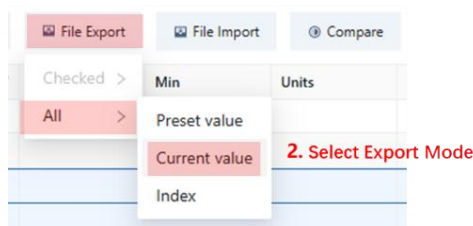
Parameter monitoring			
Index	Description	Current v...	Units
605E00	Fault_Sto...	0	D
60FF00	Target_Sp...	1000	D rpm

Press Enter To Send After Modifying The Parameters

6.2.4.3 Export parameter file

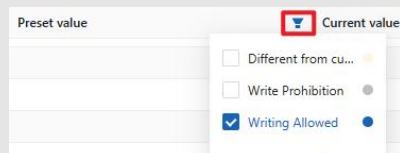
- First, click [**File Export**], and select to export the selected parameters or all parameters. You can choose to export the preset or current value of the parameters, or you can choose not to export the parameter values and only export the parameter index.

1. Click [File Export]



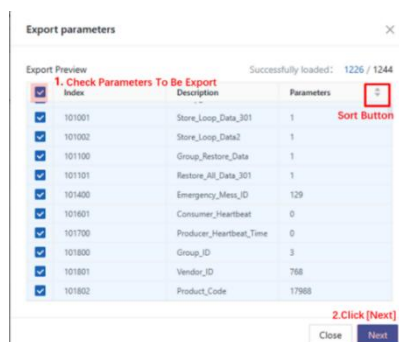
Note

- You can use the preset value filter button to display only the parameters that are allowed to be written. After checking, click [**Export Selected Parameter**].



- If you choose to export Index , the exported index file will not contain the parameter values but only the parameter indexes, which can be used for operations such as automatic marking.

- In the parameter export interface, reconfirm the parameters that need to be exported, and click [**Next**] to export the selected parameters to a local file.



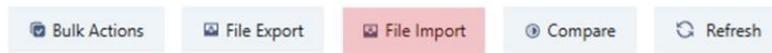


Note

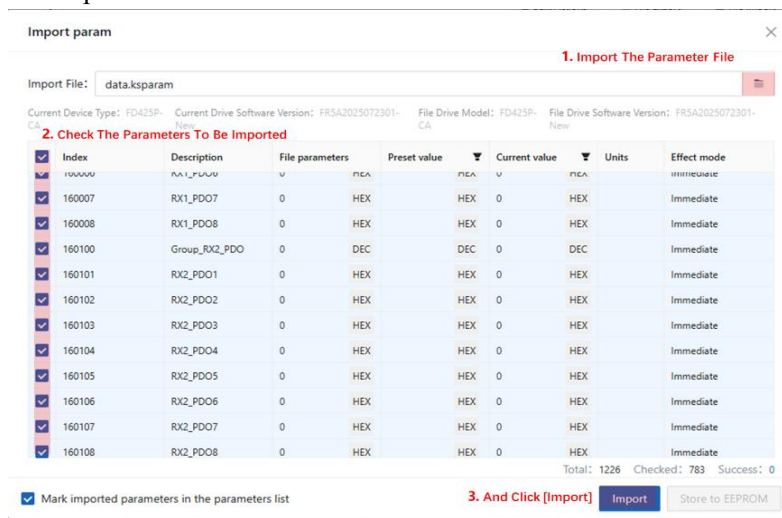
- The software only exports the parameters that have been successfully loaded. Click the Sort button at the head of the parameter column to filter out the parameters that have not been successfully loaded.

6.2.4.4 Import parameter file

- Find [File Import] in the function interaction button;



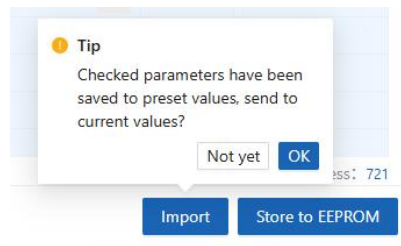
- Click the button, select the parameter file to be imported, and then click the [Import]. The parameters in the file will be saved to the preset values. If "Mark imported parameters in the parameters list" (default) is checked, the relevant parameters in the file will be checked in the list.



Note

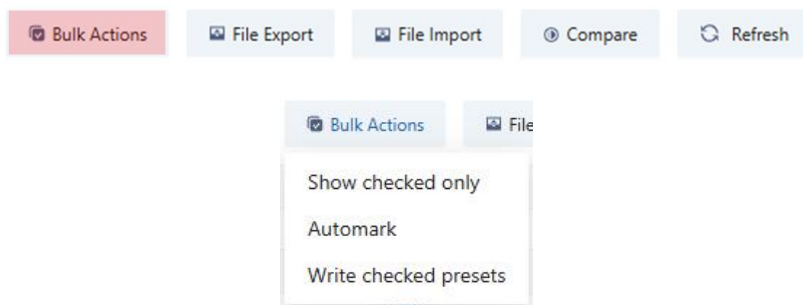
- The file suffix after exporting "Indexed" is.ksidx and it cannot be imported. The file suffix after exporting the "preset value" or "current value" is.ksparam, which can be imported.
- The parameter file will store the drive model and software version used during the export process. When importing, the software will check if the information in the file is consistent with the current drive. Since the parameter definitions exported under different software versions may vary, please ensure you are aware of the risks associated with these differences.
- For a large number of identical applications, to avoid setting the drive parameter one by one, user can import the.ksparam file for drive configuration. If the user needs to import the.ksparam file before connecting the encoder, they need to set the **Use_Inner_MTLlib** (3041.06) to 1 before import the file, which avoid the problem of abnormal feedback cycle. If the encoder cable is connected first and then import the file, there is no such requirement.

- After clicking the [**Import**], you can choose whether to write the imported parameters to the current value of the drive. Once the import is successful, if you wish to store the current value in non-volatile memory that won't be lost in case of power failure, please click the "Store to EEPROM".



6.2.4.5 Bulk actions

- After checking the target parameters, click "Batch Operation", and you can choose to **show checked only** or **write checked presets**.

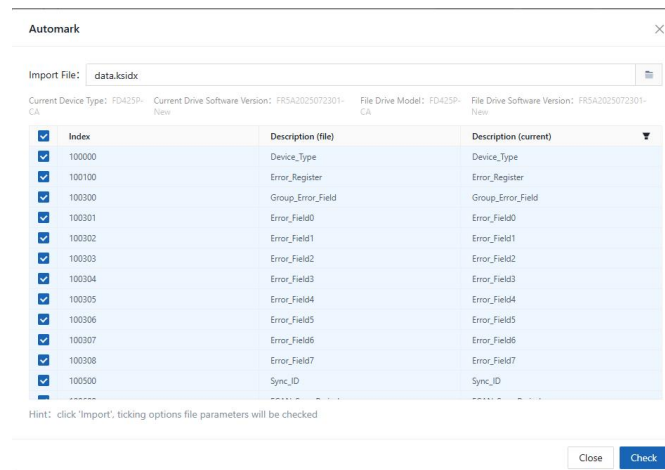


Note

- Hold down the Shift key and click the target parameters at the initial position and the final position with the left mouse button. All the parameters between these two positions will be selected simultaneously.

Enter keyword to search								
	Index	Description	Preset value	Current value	Max	Min	Units	Effect mode
☐	641012	Brake_Delay	150	DEC 150	DEC 32767	0	ms	Immediate
☐	641013	Invert_Dir_Motor	1: CW	DEC 1: CW	DEC 1	0		Save & Reboot
☐	641014	Motor_Num2	8X	ASCII 8X	ASCII			Immediate
☐	641015	Motor_Current_BW	2000	DEC 2000	DEC 2500	500	Hz	Save & Reboot
☐	641017	Addition_Device	0	HEX 0	HEX			Immediate
☒	641018	Temp_Motor_Ref	8192	DEC 8192	DEC			Immediate
☒	64101A	Gain_Factor	1	DEC 1	DEC 15.94	1	1/N	Immediate
☒	64101C	Index_Width	8	DEC 8	DEC			Immediate
☒	64101D	Encoder_Power_Delay	5	DEC 5	DEC			Immediate
☒	64101E	QEL_Filter	13	HEX 13	HEX			Immediate
☒	64101F	Sim_ABB	0	DEC 0	DEC			Immediate
☒	651001	Driver_ILI	5.94	DEC 5.94	DEC 13.18	0.01	Ap	Immediate
☒	651002	Driver_ILI_Filter	233.98	DEC 233.98	DEC 8388.35	0.51	S	Immediate
☒	651003	IL_Max	18	DEC 18	DEC 3276.7	0	Ap	Immediate
☐	651004	DCBus	310	DEC 310	DEC 32767	0	V	Immediate
☐	651005	Voltage_200_DC	200	DEC 200	DEC 32767	0	V	Immediate
☐	651006	Voltage_360_DC	360	DEC 360	DEC 32767	0	V	Immediate
☐	651007	Voltage_Under	200	DEC 200	DEC 32767	0	V	Immediate
☐	651008	Chop_Voltage	380	DEC 380	DEC 32767	0	V	Immediate
☐	651009	Voltage_Over	400	DEC 400	DEC 32767	0	V	Immediate

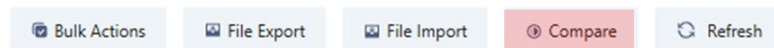
- After clicking the **Automark** option, in the pop-up window, select the .ksidx index file and click the check button. Then, you can check the relevant parameters in the entire parameter list.



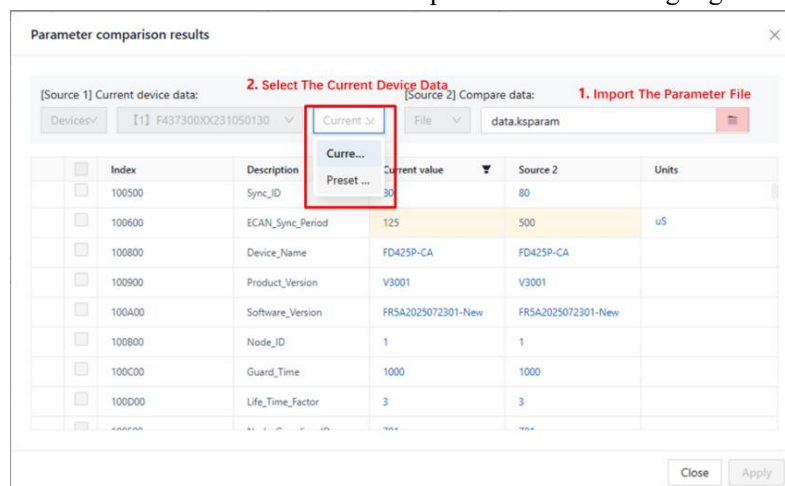
6.2.4.6 Compare

The parameter compare function can compare the differences between two parameter files.

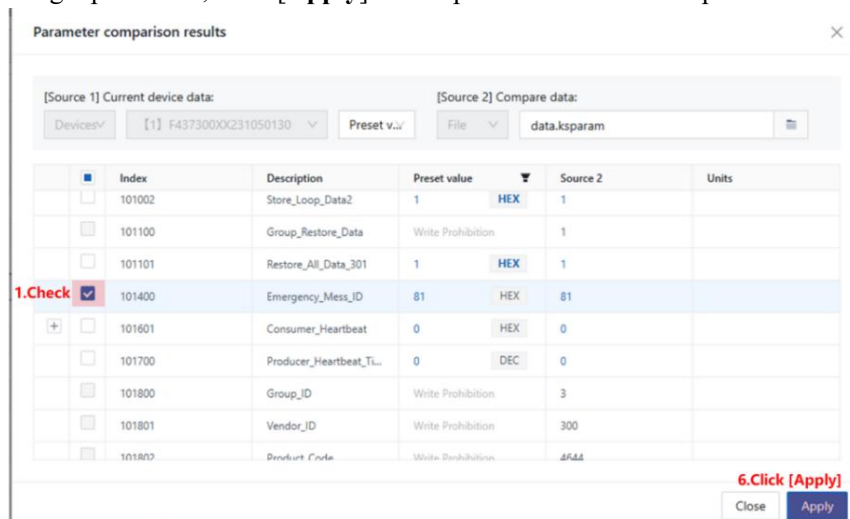
- Find [**Compare**] in the function interaction buttons;



- After clicking the "Import File" button, you can select the current device data to compare it with the file parameters. The differences between the two sets of parameters will be highlighted.

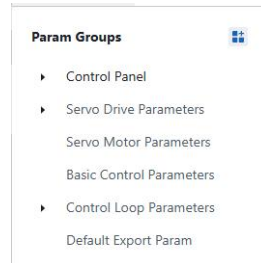


- If the current device data selects the preset value, you can modify the corresponding preset value. After checking the target parameter, click [**Apply**]. This operation will send the preset value to the drive.



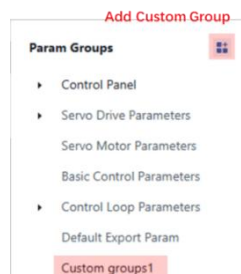
6.2.4.7 Param Groups

The parameter group can be used to categorize servo parameters according to requirements, making it easier to quickly locate them. The custom group contains a series of preset groups, and all related operations can be performed within these groups.



If custom grouping is required, the operation steps are as follows:

- Click "Add Custom Group", and a custom parameter group will be automatically generated.



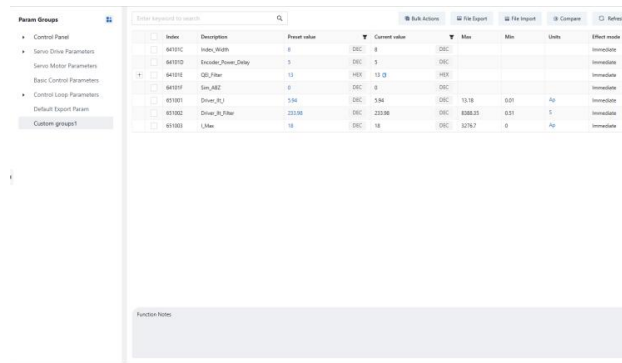
- User can rename, create subgroup or delete custom groups.



- In the parameter list, select the target parameter with the left mouse button, right-click on it and add this parameter to the custom group.

☐	64101C	Index_Width	8	DEC	8	DEC				Immediate
☐	64101D	Encoder_Power_Delay	5	DEC	5	DEC				Immediate
☒	64101E	QEI_Filter	13	HEX	13 0	HEX				Immediate
☐	64101F	Sim_RRZ	0	DEC	0	DEC				Immediate
☐	651001	Driver_Rt_I	5.94	DEC	5.94	DEC	13.18	0.01	Ap	Immediate
☐	651002	Driver_Rt_Filter	233.98	DEC	233.98	DEC	8368.35	0.51	S	Immediate
☐	651003	I_Max	18	DEC	18	DEC	3276.7	0	Ap	Immediate
☐	651004	DCbus		DEC	310	DEC	32767	0	V	Immediate
☐	651005	Voltage_200_DC	190	DEC	200	DEC	32767	0	V	Immediate
☐	651006	Voltage_360_DC	360	DEC	360	DEC	32767	0	V	Immediate

- The selected target parameters will appear in the custom group.

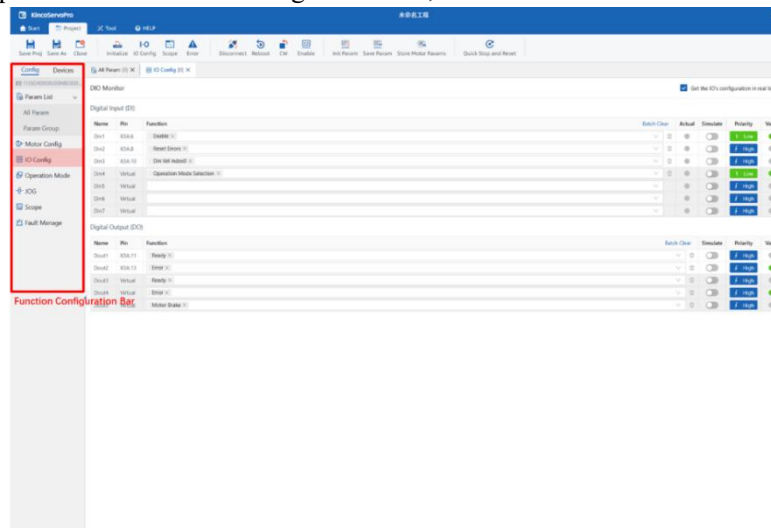


Note

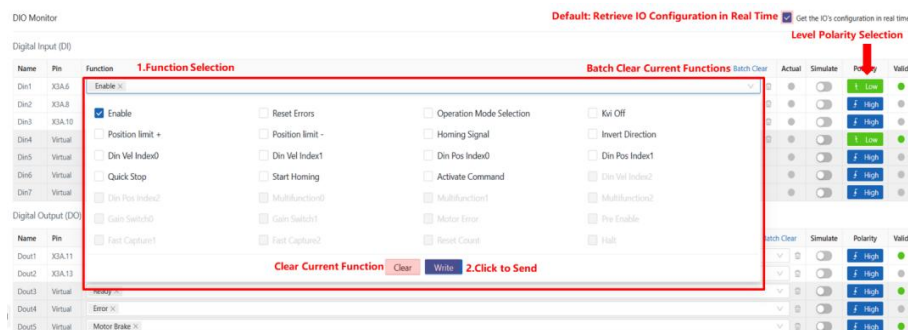
"Selected" and "checked" are two different states. Please pay attention to the distinction.

6.2.5 Digital input and output setting

Find the [IO Config] option in the function configuration bar;

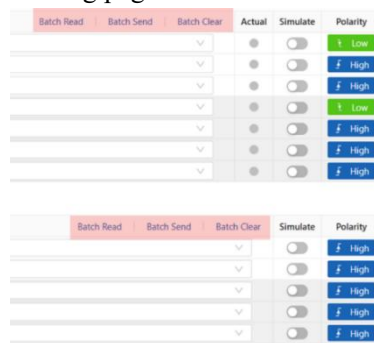


- Using the digital IO settings, user can define the trigger conditions for each IO port. First, user need to configure the function and trigger polarity of the IO port. The specific operation process is shown in the following diagram:



- If the option of "Get the IO's configuration in real time" is unchecked, batch operations can be performed on the definition of IO port function, including **Batch Read**, **Batch Clear** and **Batch Send**. If the option of "Get the IO's configuration in real time" is checked, the IO configuration will be automatically batch-read

and batch-send; batch read can obtain the current driver-set function, and batch send can write the configuration function of the DIO setting page into the drive.



- After configuring the IO, when the corresponding functional event occurs, the IO port will trigger an input or output high level. User can also click the simulation input button to forcibly trigger the input and output of the IO port.



6.2.5.1 Digital input

Name	Pin	Function	Batch Clear	Actual	Simulate	Polarity	Valid
Din1	X1.4	Enable				High	
Din2	X1.6					High	
Din3	X1.8					High	
Din4	X1.10					Low	
Din5	X1.12					Low	
Din6	X1.14					High	
Din7	X1.16					High	

Simulation: Simulate digital input signal

Actual: Display the actual digital input status

Polarity: High indicates a high-level input, the valid input is 1. Low indicates a low-level input, and the effective input is 1.

Valid: The results of simulation, actual input and polarity interaction; means active, logic status is 1; means "not active", logic status is 0.

The relationship among the Actual input , Simulation input , Polarity, Valid input is shown as following table:

Actual input / Simulation input	Polarity	Effective input
High level	High level effective	High level
Low level	High level effective	Low level
High level	Low level effective	Low level
Low level	Low level effective	High level

Table 6-1 Digital input function

DIN function	Description
Enable	Controller enabling 1: Enable controller = Din control word selection(2020.0F) 0: Disable controller = 0x06
Reset errors	Set the Controlword to reset errors, active edge (bit7) = 1
Operationg mode	Operation_mode selection 1:Operation_mode = EL.Din_Mode1 (2020.0E), Default value=-4 0:Operation_mode = EL.Din_Mode 0 (2020.0D), Default value =-3
Kvi off	1: Velocity control loop integrating gain off 0: Velocity control loop integrating gain has been set.
Position limit+	Positive/Negative position limit switch input for "normally closed" limit switches 0: position limit is active, the related direction is blocked
Position limit-	
Homing signal	Home switch signal for homing
Invert Direction	Inverts command direction in the velocity and torque mode
DinVel Index 0	Din_Speed Index in the DIN speed mode
DinVel Index 1	
DinVel Index 2	
Din Pos Index0	Din position index in Din position mode
Din Pos Index1	
Din Pos Index 2	
Quick stop	Set control word bit2=0 to enable emergency stop. If you want to re-enable after emergency stop, you need to first set the control word to 0x06 and then to 0x0F (if the input port is configured with drive enable, you only need to trigger the enable signal again).
Start homing	Starts homing. Only makes sense if the controller is enabled. The controller returns to the previous operation mode after homing.
Activate command	Activates the position command. Controls bit 4 of the Controlword, e.g. Controlword=0x2F—>0x3F
Motor error	If an external error occurs (for example, the motor itself is too hot), the error signal can be transmitted to the driver through the input port
Fast capture 1	Fast capture, refer to chapter 12.7 for more details
Fast capture 2	
Pre enable	For safety reasons, pre-enable can serve as a signal for indicating whether or not the entire system is ready , 1: Drive can be enabled; 0: Drive can not be enabled
Reset count	Clear the number of pulses the drive has received but not completed
Halt	1: Pause motor, code: bit8 = 1 0: Motor continues to execute unfinished instructions
Positive pulse regulation	Activate the position fine-tuning data in the positive direction in the pulse mode
Negative pulse regulation	Activate the position fine-tuning data in the negative direction in the pulse mode

**Note**

DIN control word selection (2020.0F) is set to 0x2F by default.

6.2.5.2 Digital output

Digital Output (DO)

Name	Pin	Function	Batch Clear	Simulate	Polarity	Valid
Dout1	X1.1	Ready ×	▼	☐	High	
Dout2	X1.5	Motor Brake ×	▼	☐	High	
Dout3	X1.9	Speed Limit × Speed Reached × Pos Reached ×	▼	☐	High	
Dout4	X1.11	Zero Speed ×	▼	☐	High	
Dout5	X1.20	Error ×	▼	☐	High	

Simulation : Simulate digital output signal

Polarity: High indicates a high-level output, the valid input is 1. Low indicates a low-level input, and the effective input is 1.

Valid: The results of simulation, actual output and polarity interaction; means active, logic status is 1; means "not active", logic status is 0.

The relationship among the Simulation output , Polarity, Valid output is shown as following table:

Simulation output	Polarity	Effective output
High level	High level effective	High level
Low level	High level effective	Low level
High level	Low level effective	Low level
Low level	Low level effective	High level

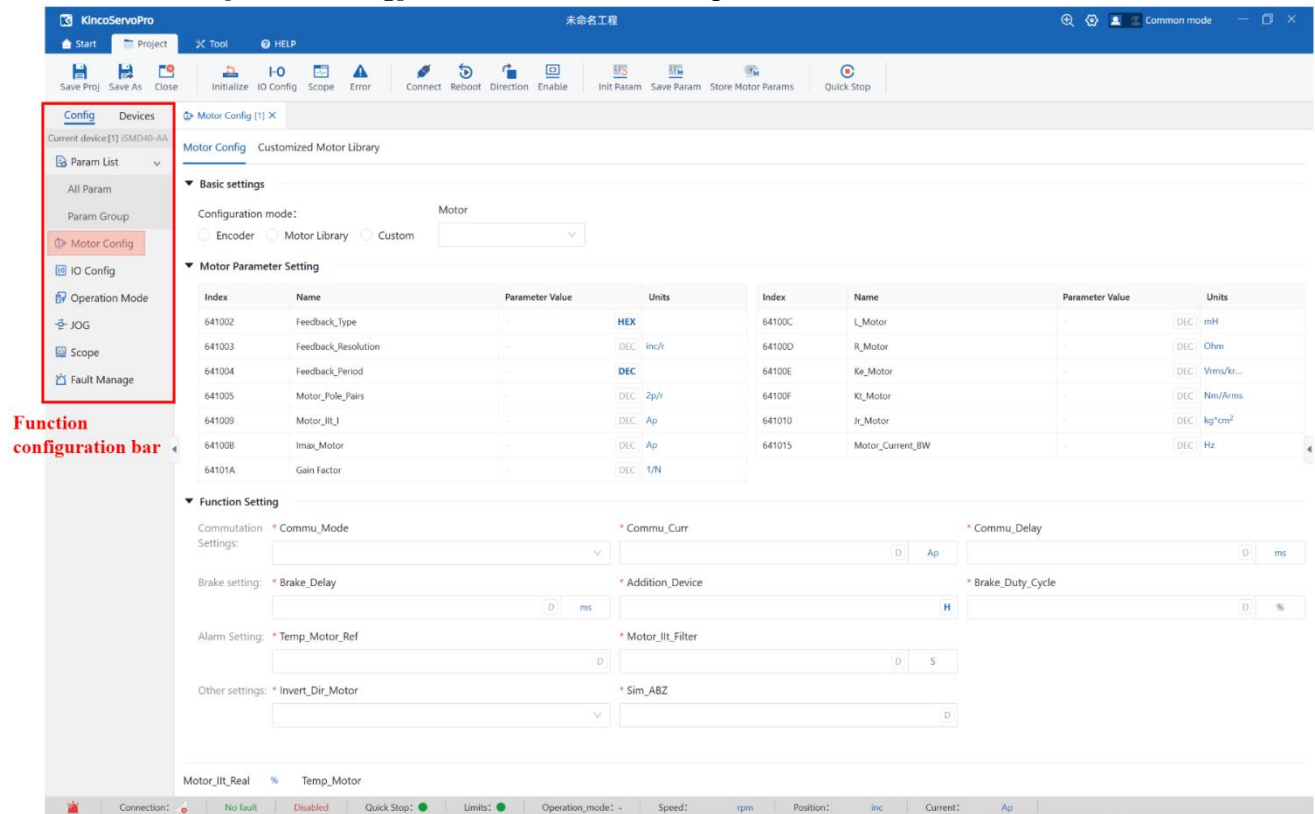
Table 6-2 Digital output

OUT function	Description
Ready	Controller is ready to be enabled
Error	Controller error
Pos reached	In position mode, when the difference between the actual position and the target position is less than the Target_Pos_Window (6067.00), and the duration is greater than or equal to the Position_Window_time (6068.00), the the position reach function is output
Zero speed	When the absolute value of the Speed-1ms (60F9.1A) is less than or equal to the Zero_Speed_Window (2010.18), and the duration is greater than or equal to the Zero_Speed_Time (60F9.14), the zero speed function is output
Motor beake	The motor brake control output signal can be used to connect an external relay that controls the motor lock. If the brake motor is used, this function must be set, otherwise it will damage the motor. An effective output of green indicates that the brake is opened, and an effective output of gray indicates that the brake is closed
Speed reached	Output speed reach function when the Speed_Error (60F9.1C) is less than theTarget_Speed_Window (60F9.0A)
Enc Index	Motor index signal appear
Speed limit	In torque mode ,actual speed reached Max_Speed(607F.00)
Driver enabled	Drive is enabled. Motor locks shaft
Position limit	In the positive and negative position limit
Home found	Home found
Ref Torque Reached	When the Current_Out_Real (60F5.08) reaches the Torque_Out_Ref (60F5.06) and the duration exceeds the Torque_Out_Filiter (60F5.07), the output torque reaches the limit, and the torque reaches the reference (60F5.06) is set to 0, indicating that the torque limit detection is not enabled .
Multi Dout 0	Din multi-segment position control position to output function
Multi Dout 1	
Multi Dout 2	

6.2.6 Motor configuration

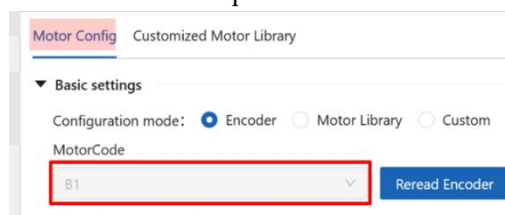
Before the motor starts its commissioning, the motor parameters need to be correctly configured. The drive offers three configuration methods. For Kinco servo drive product, the parameters stored internally in the encoder or those stored in the motor can be read; for third-party motor, custom configuration can be done based on the motor parameter drawings.

First, find the **[Motor Config]** menu in the function configuration bar.

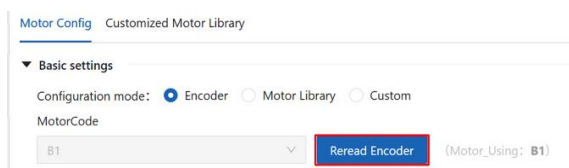


Read the parameters stored inside the encoder

- In the function configuration section find the **[Motor Config]**. First, verify that the motor code in the basic settings is the same as that on the motor nameplate.



- If there is any inconsistency, you can click **[Reread Encoder]**. After restarting, the software will automatically complete the motor configuration.





Note

- This operation will initialize the drive, and all drive parameters will be changed to default values (including the station number).

- If you wish to modify the current motor parameter, user can make the changes and then click [**Apply**]. After that, click [**Confirm**] in the pop-up window, and the parameters will be sent to the drive.

The screenshot shows the 'Motor Config' window with the 'Motor Parameter Setting' tab selected. It displays a table of motor parameters with columns for Index, Name, Parameter Value, and Units. Below the table are sections for 'Function Setting' (Commutation, Brake, Alarm, and Default Direction) with various input fields and dropdown menus. At the bottom right, there is a 'Click [Apply]' button. Below the main window, an 'Apply' dialog box is open, showing a list of parameters that will be restored after a reboot.

Index	Name	Parameter Value	Units
641002	Feedback_Type	4	HEX
641003	Feedback_Resolution	85536	DEC
641004	Feedback_Period	327702	DEC
641005	Motor_Rise_Freq	3	DEC
641009	Motor_ILI	3.4	DEC
641008	Imax_Motor	10.2	DEC
64101A	Gain Factor	1	DEC

Index	Name	Parameter Value	Units
64100C	L_Motor	11.5	DEC
64100D	R_Motor	5.8	DEC
64100E	Ka_Motor	33.95	DEC
64100F	Kb_Motor	0.56	DEC
641010	Jr_Motor	0.4	DEC
641015	Motor_Current_RW	2000	DEC

Index	Function Area	Name	Remarks
641006	Function Setting	Commu_Mode	Restore after reboot
641007	Function Setting	Commu_Curr	
641008	Function Setting	Commu_Delay	
641018	Function Setting	Temp_Motor_Ref	Restore after reboot
64100A	Function Setting	Motor_ILI_Filter	Restore after reboot
641012	Function Setting	Brake_Delay	Restore after reboot
641017	Function Setting	Addition_Device	Restore after reboot
641011	Function Setting	Brake_Duty_Cycle	Restore after reboot



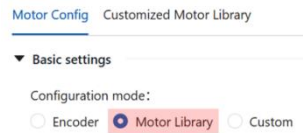
Note

- In encoder mode, only some parameters can be modified, and some parameters will be restored after a restart.
- The [**Read**] button only reads the motor parameters in the servo drive, but does not read the encoder information.

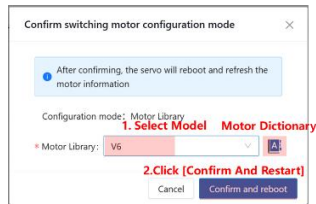
Use the internal motor library

If the parameters stored internally in the encoder cannot be read, the parameters can be configured through the motor library and the motor code.

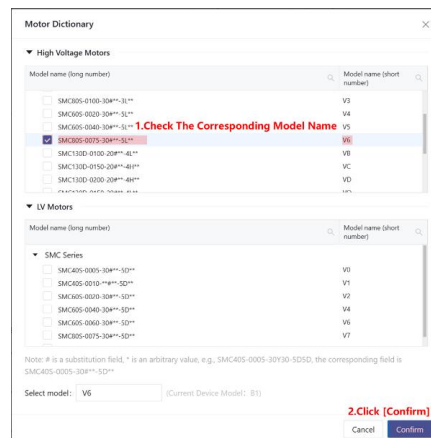
- In the basic settings, change the configuration method to [**Motor Library**].



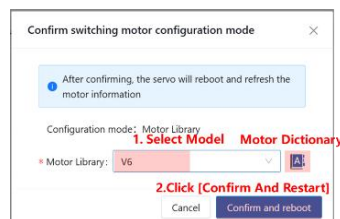
- In the pop-up configuration interface, enter the motor code.



- Or click the "Motor Dictionary" on the right, and select the specific model based on the model name on the motor nameplate.



- Finally, just click [Confirm and reboot].



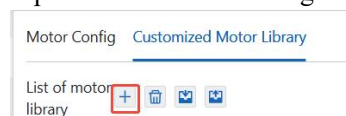
Note

- If it shows that writing to the motor library failed, it indicates that the drive does not support this model.

Customized motor library

When configuring a third-party motor, user can create a custom motor parameter configuration by following the steps below:

- Click the "Customized Motor Library" option in the motor configuration bar, and then click the [+] button.



- In the pop-up interface for creating the motor, fill in the custom motor name, select the communication protocol, and input the motor parameters according to the motor parameter drawing. You can also click the **[Read current parameters]** button on the right side to modify the current motor parameters, and finally click **[Save]**.



Note

- The custom motor name is the user's designation for the custom motor. It can be any text / letters / numbers and is independent of the MotorCode.

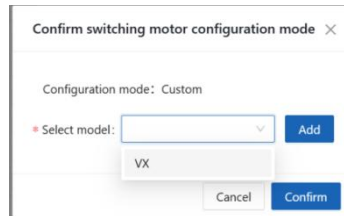
- After completing the above operation, the custom motor library will appear in the motor library list. Subsequently, user can modify the parameters in the right parameter bar, and then click **[Save]** to save the motor. The saved motor can be viewed in all local projects created

Other function	Operation
Delete motor library	2. Click [delete] 1. Check The Motor Model
Import motor library	2. Click [Import] 1. Check The Motor Model
Export the motor library	2. Click [Export] 1. Check The Motor Model

After creating a new custom motor library, on the motor configuration interface, click the **[custom]** in the basic setting bar.



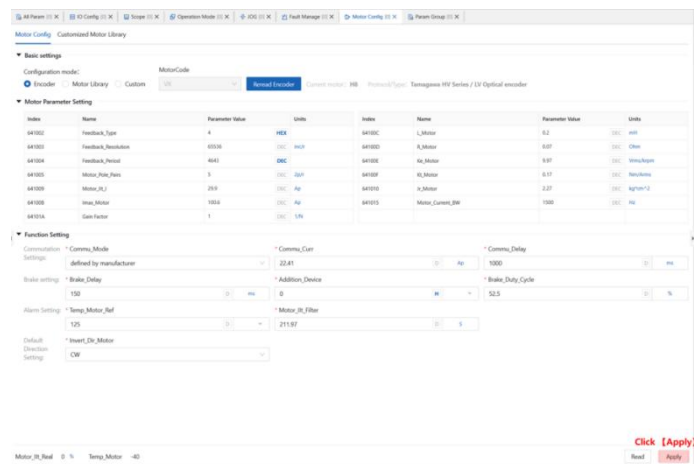
In the popped-up interface, after selecting the custom motor model, click **[Confirm]**.



Note

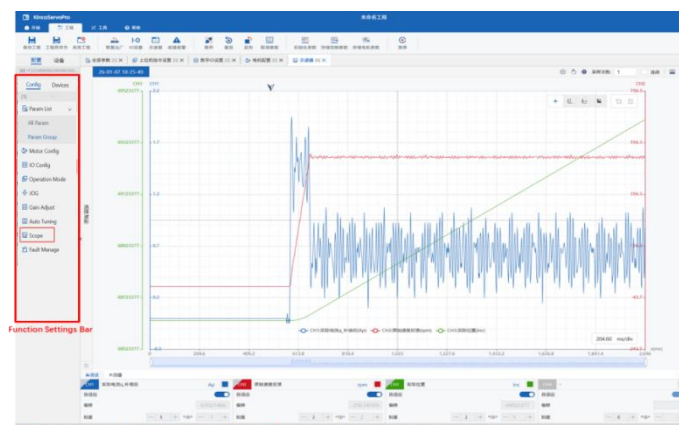
- Clicking **[Add]** will redirect to the motor creation interface.

Finally, use the mouse scroll wheel to scroll down to the bottom of the motor configuration interface, then click **[Apply]**.



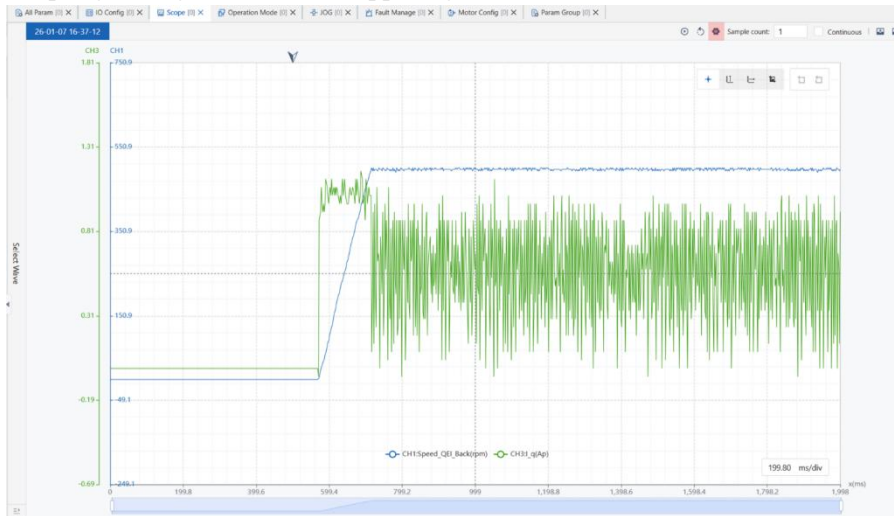
6.2.7 Scope

The oscilloscope is an important tool for monitoring, analyzing and recording system signals. First, find the **[Scope]** in the function configuration section.

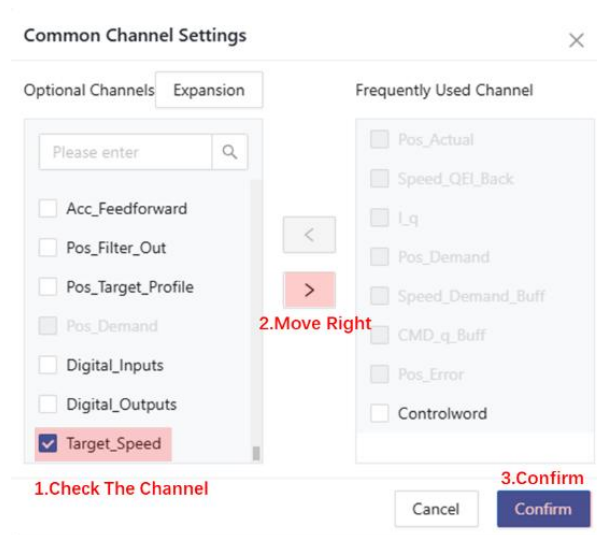


6.2.7.1 Sampling settings and execution

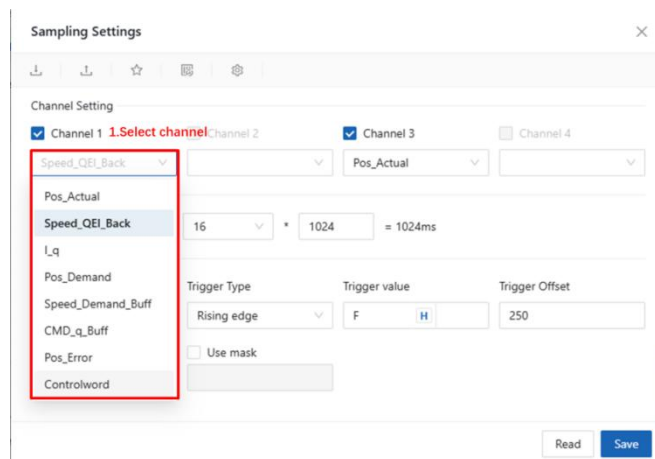
- Click the sampling settings button in the upper right corner of the interface;



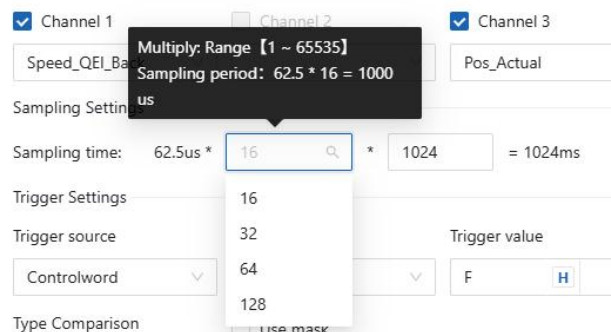
- Add sampling objects: First, add the objects to be collected to the list. The software has preset some commonly used object.
- In the pop-up **[Settings]** window, check the optional sampling objects;
- Click to add the data channel to frequency used channel;
- Click **[Confirm]**;
- Some special objects require clicking the **[Expansion]** button for selection. This operation may cause abnormal data collection.



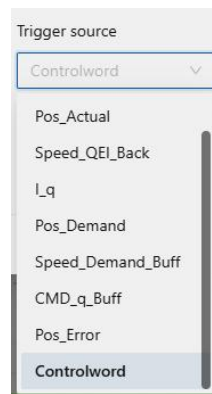
- After adding the commonly used sampling objects, you can find the corresponding objects in the drop-down menu.



- Sampling settings: The sampling time is determined by the sampling period and the number of samples. The sampling period unit is a current loop time of 62.5us. If the sampling period is set to 64, the data change time interval is $62.5 \times 64 \text{us}$; the number of samples is the number of data collections. The upper limit changes with the channel settings, and the software automatically calculates it.



- Trigger source: Select the trigger source;

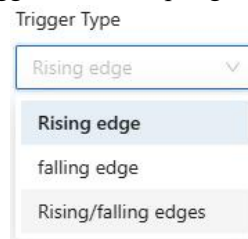


- Trigger type: Rising edge, falling edge, rising/falling edge trigger modes can be selected.

Rising edge: When the data rises above the trigger value, sampling begins.

Falling edge: When the data drops from above the trigger value to the trigger value, sampling begins.

Rising/falling edges: When the data rises from below the trigger value to the trigger value and then falls from above the trigger value to the trigger value, sampling begins.



- Trigger value: When the data reaches the trigger value, the timing of sampling is determined based on the trigger type.
- Trigger offset: When the trigger offset is set to 250, the collected waveform will start from 250 data points before the trigger moment.



- Type comparison: During signed data comparison, the data changes from 0 to -100, which is defined as the data decreasing; during absolute value comparison, the data changes from 0 to -100, and the absolute value is equivalent to the data changing from 0 to 100, which is defined as the data increasing;



- Use mark: The mask can shield the specified data bits. The trigger source comparison value = the actual value of the trigger source & the trigger mask. For example, the 16-bit control word data 0x002F uses the mask 0x000F, and only the lower 4 bits are retained, that is, the data becomes 0x000F. This function is generally used when only observing (a few) data bits, and generally chooses not to use a mask.



- Other function: The red part of the sampling settings interface, from left to right, are **import**, **export**, **favorite**, and **automatic collection settings**.

Export:The current oscilloscope configuration can be exported as a reusable file.

Import: The oscilloscope configuration file can be imported into the software, eliminating the need for manual reconfiguration.

Favorite:The function belongs to the export channel settings, but it is not exported as a file; instead, it is directly added to the template.

Automatic collection settings:Ready-made channel settings and custom channel settings can be imported from the template.



Note

- The templates imported by the collection/automatic collection settings are only available in the current project.

The Sampling Settings dialog box contains the following sections:

- Channel Setting:** Includes checkboxes for Channel 1, Channel 2, Channel 3, and Channel 4. Channel 1 is selected with 'Speed_QEI_Back' as the source. Channel 3 is selected with 'Pos_Actual' as the source.
- Sampling Settings:** Shows 'Sampling time: 62.5us * 16 * 1024 = 1024ms'.
- Trigger Settings:** Includes 'Trigger source' (Controlword), 'Trigger Type' (Rising edge), 'Trigger value' (F H), and 'Trigger Offset' (250).
- Type Comparison:** Includes 'Signed Data Comparison' and a checked 'Use mask' option.

Buttons at the bottom: Read, Save.

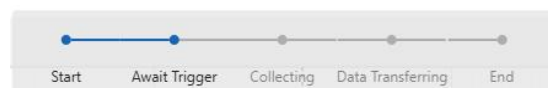
6.2.7.2 Start sampling

- After the channel settings are completed, if you need to repeat the sampling, you can change the sampling count. If the sampling frequency is not known, you can check the "Continuous" option and then click

1.Enter the number of sampling times and choose whether to sample continuously.
2.Click **[Start]**

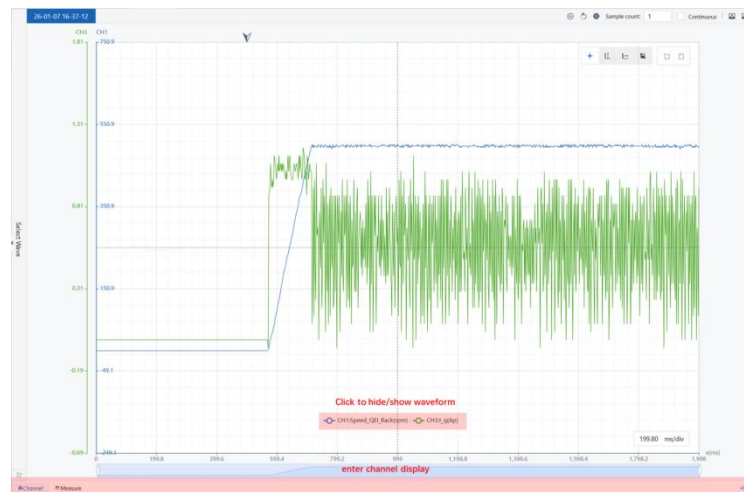
The control bar shows: Stop, Refresh, Settings, Sample count: 1, Continuous (unchecked), and icons for saving and loading templates.

The window will pop up the display of the collection status.



6.2.7.3 Waveform display settings

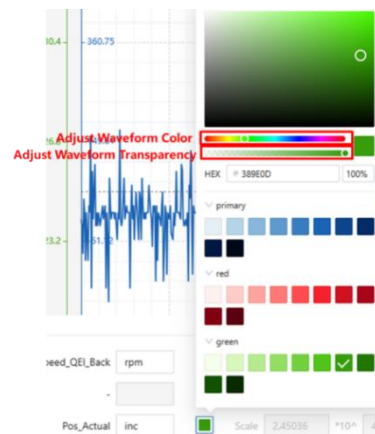
- After the sampling is completed, click on the legend below the waveform to hide or display the specified waveform, then click on the channel display bar to enter the channel display.



- If you need to manually adjust the waveform size and position, first cancel the window auto adaptation;
- The offset input box can shift the waveform up and down;
- The scale frame can adjust the waveform amplitude display;
- The color change window allows you to modify the waveform color and transparency.



- Click the color change window. In the pop-up interface, you can adjust the waveform color and also change the waveform transparency.



- The waveform display range can be modified by using the "Time Adjustment" option at the bottom right corner. You can also drag the time cursor or view the waveform in the scope display interface to observe the waveforms of different time period.



- You can also use the area zoom function to select and zoom in on a specified area waveform, and restore the waveform through the restore area zoom function.

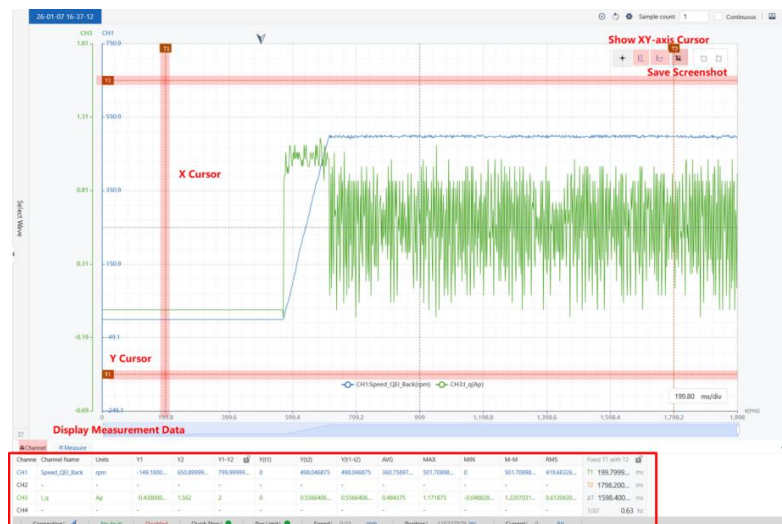


6.2.7.4 Cursor

- The waveform data of different channels at the same time can be viewed through the curve cursor.



- You can click the X and Y-axis cursor display buttons in the upper right corner to bring them up. Then, use the mouse to drag the X and Y-axis cursors and view the data of each channel and the time interval in the data display box.



6.2.7.5 Waveform export and view historical waveform

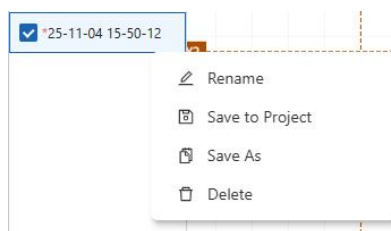
- Click on the left side [Select Wave];



- Select and view the waveforms to be exported in the historical waveform interface. They are sorted from top to bottom according to the acquisition time. The newly acquired waveforms are at the bottom. By default, the historical waveforms are in an unsaved state. After saving the project, the historical waveforms are also saved synchronously.



- Right-click on the historical waveform to perform operations such as rename, save to project, saving as , and delete.



- The [Export] button in the upper right corner of the waveform display interface can be used to save the waveform file.



6.2.7.6 Waveform import

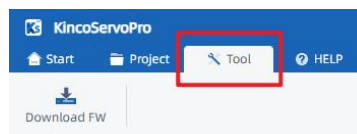
- The **[Import]** button in the upper right corner of the waveform display interface allows you to import the waveform file.



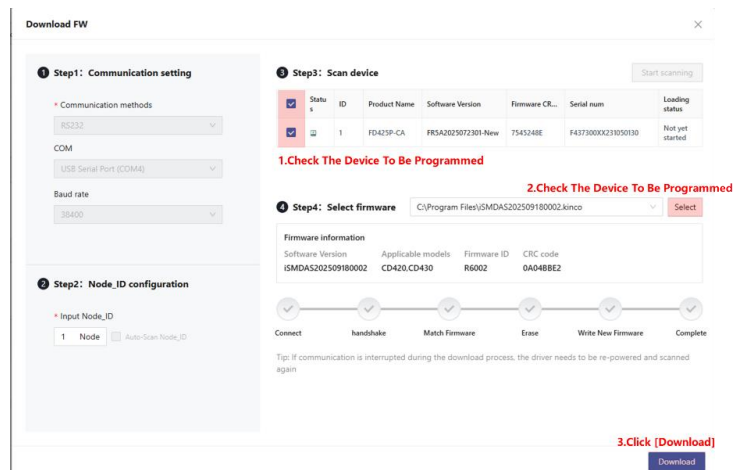
The imported waveform files are displayed in the list on the left

6.2.8 Download FW

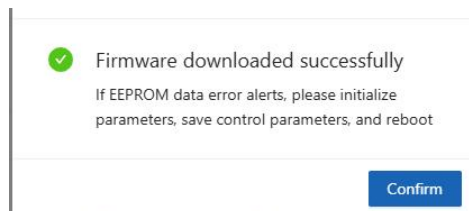
Find the **[Tool]** menu in the navigation bar and click **[Download FW]**.



- Check the scan device;
- Select firmware;
- Click **[Download]** and wait for the download to complete



- After the download is complete, follow the prompts to operate.



Note

- Users can also directly download the firmware without creating a project, but they need to perform communication settings and scanning operations first. For specific operations, please refer to Section 6.2.2

1 Step1: Communication setting

* Communication methods

RS232

COM

USB Serial Port (COM4)

Baud rate ☒ Auto-Baudrate

38400

- If there is no firmware in the drive, the firmware check code will display "No firmware". The download steps are the same as the previous ones.

3 Step3: Scan device Start scanning

☒	Status	ID	Product Name	Software Version	Firmware CR...	Serial num	Loading status
☒		-	-	RF001 bootloader V1.0!	No firmware	-	Not yet started

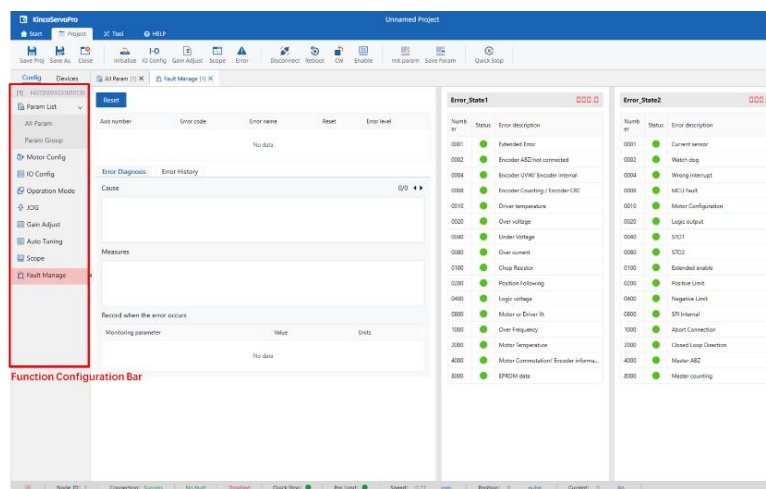


Note

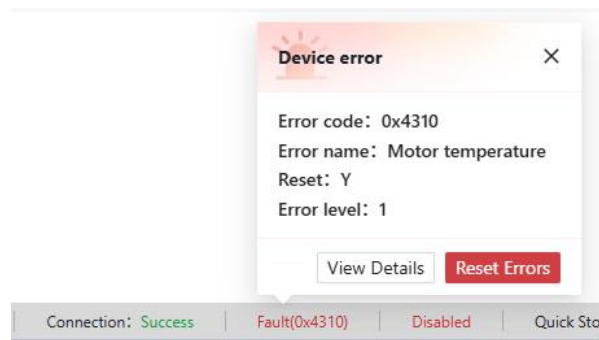
- When the original firmware is erased during the download process and the new firmware has not been successfully written, if the drive is restarted, the device will be in a state without firmware

6.2.9 Fault manage

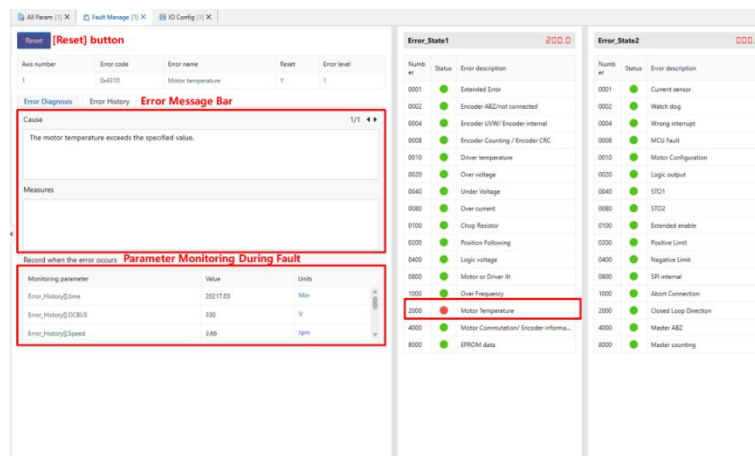
Find the [Fault Manage] menu in the function configuration bar.



- After clicking, you will enter the fault management interface. Or when the servo alarms, clicking on the [View Details] in the bottom status bar can also jump to the fault management interface.



- The error diagnosis interface includes the **Error code**, **Error diagnosis** and **the servo status at the time of the failure**;
- Clicking the [Reset] button can reset the fault. If the digital input is configured with functions related to control words such as enabling and resetting fault, this button will not take effect.
- The error status word may be composed of multiple states superimposed, that is, multiple faults occur simultaneously. In the list below each error status word, a red light indicates that the error has occurred



- Click [Error History] to view historical fault codes and the servo status when the fault occurred.

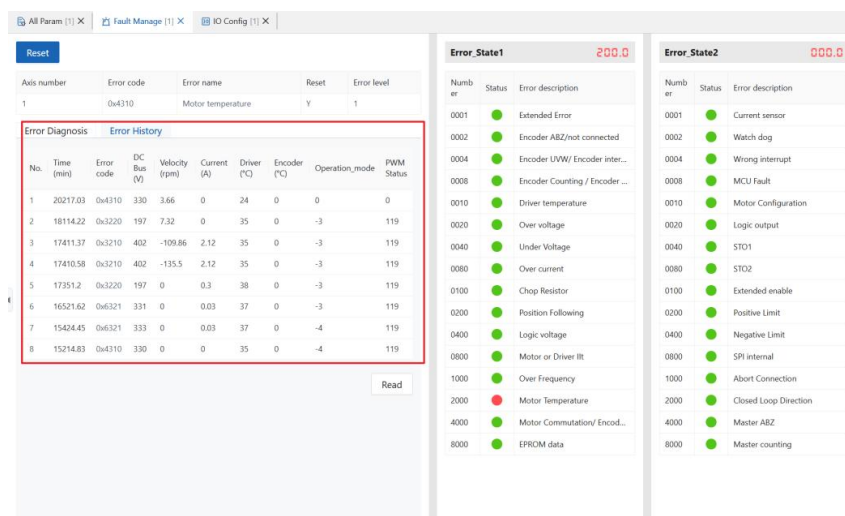


Table 6-3 Error_state (2601.00) information

Bit	Error name	Error code	Description
0	Extended Error		Refer to object "Error_State 2"(2602.00)
1	Encoder not connected	0x7331	No communication encoder connected
2	Encoder internal	0x7320	Internal encoder error
3	Encoder CRC	0x7330	Communication with encoder disturbed
4	Controller Temperature	0x4210	Heatsink temperature too high
5	Overvoltage	0x3210	DC bus overvoltage
6	Undervoltage	0x3220	DC bus undervoltage
7	Overcurrent	0x2320	Power stage or motor short circuit
8	Chop Resistor	0x7110	Overload, brake chopper resistor
9	Following Error	0x8611	Max. following error exceeded
10	Low Logic Voltage	0x5112	Logic supply voltage too low
11	Motor or controller Ilt	0x2350	Motor or power stage Ilt error
12	Overfrequency	0x8A80	Pulse input frequency too high
13	Motor Temperature	0x4310	Motor temperature sensor alarm
14	Encoder information	0x7331	No encoder connected or no encoder communication reply
15	EEPROM data	0x6310	EEPROM checksum fault

Table 6-4 Error_state2 (2602.00) information

Bit	Error name	Error code	Description
0	Current sensor	0x5210	Current sensor signal offset or ripple too large
1	Watchdog	0x6010	Software watchdog exception
2	Wrong interrupt	0x6011	Invalid interrupt exception
3	MCU ID	0x7400	Wrong MCU type detected
4	Motor configuration	0x6320	No motor data in EEPROM / motor never configured
5	Reserved		
6			
7			
8	External enable	0x5443	DIN "pre_enable" function is configured, but the DIN is inactive when the controller is enabled / going to be enabled
9	Positive limit	0x5442	Positive position limit (after homing) – position limit only causes error when Limit Function (2010.19) is set to 0.
10	Negative limit	0x5441	Negative position limit (after homing) position limit only causes error when Limit Function(2010.19) is set to 0.
11	SPI internal	0x6012	Internal firmware error in SPI handling
12	CAN bus interrupt	0x8100	Communication interruption mode (6007.00) is set to 1 to generate fault alarm
13	Closed loop direction	0x8A81	Different direction between motor and position encoder in closed loop operation by a second encoder.
14	Master ABZ	0x7382	Master encoder connection error
15	Master counting	0x7306	The main encoder index signal is abnormal

Table 6-5 Error extension (2605.07) information

Bit	Error name	Error code	Description
0	Origin record error	0x5210	Current sensor signal drift or ripple is too large
1	Internal braking resistor overtemperature	0x7111	The actual power of the internal braking resistor is too high

2	Internal braking resistor short circuit	0x7112	Internal braking unit damaged, braking circuit short circuit
3	Motor phase loss	0x6321	A phase in the motor power line UVW is not connected
4	ADC sampling saturation	0x2321	Current sampling ADC reaches limit, current runaway
12	Service timeout	0x81FF	Communication Bus Error Extension
13,14	Driver input missing a phase detection	0x3221	bit[13:14] : 11: power input missing two phases (power failure); 01: Power input missing a phase (single-phase input without this); 00: Power input normal
	Driver input missing two/three phase detection	0x3222	

Table 6-6 Error mask

Object index	Data type	Name	Meaning (Bit meaning please see table5-7 and table 5-8)	Default
2605.01	uint 16	Error_Mask	Mask of Error_State(2601.00). Bit = 0 means related error will be ignored.	0xFFFF
2605.02	uint 16	Store_Mask_ON	Error mask for Error_History of Error_State(2601.00) when controller is enabled. Bit = 0 means related error won't be stored in the Error_History	0xFBFF
2605.03	uint 16	Store_Mask_OFF	Error mask for Error_History of Error_State(2601.00) when controller is not enabled. Bit = 0 means related error won't be stored in the Error_History	0x0000
2605.04	uint 16	Error_Mask2	Mask of Error_State2(2602.00). bit = 0 means related error will be ignored	0xFFFF
2605.05	uint 16	Store_Mask_ON2	Error mask for Error_History of Error_State2(2602.00) when controller is enabled. Bit = 0 means related error won't be stored in the Error_History	0xF1FF
2605.06	uint 16	Store_Mask_OFF2	Error mask for Error_History of Error_State2(2602.00) when controller is not enabled. Bit = 0 means related error won't be stored in the Error_History	0x003F

Chapter 7 Commissioning

7.1 Pre-operation preparation

The following table items need to be checked before running the drive and servo motor.

7.1.1 Cable connection and environment check

Table 7-1 Check list

Serial number	Wiring check items
1	The power input terminals (DC+, DC-) of the servo driver must be properly connected
2	The control signal cables of the servo driver are correctly connected, and the external signal cables such as lock and over-range protection are reliably connected
3	The servo drive and servo motor must be reliably grounded
4	The stress on all cables is within the specified range
5	The distribution terminals have been insulated
Serial number	Environmental inspection items
1	There are no foreign objects such as wire heads and metal shavings inside and outside the servo drive that will cause short circuit of signal lines and power lines
2	The servo drive and external brake resistors are not placed on combustible objects
3	The installation, shaft and mechanical connection of the servo motor must be reliable
4	The servo motor and the connected machinery must be in operational condition

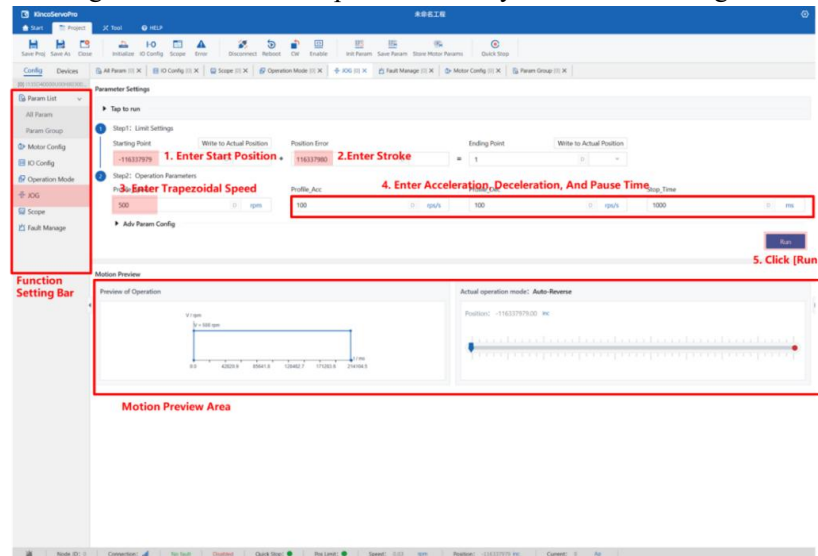
7.2 Commissioning

After the motor parameters are configured, you control the motor for commissioning through the debugging software, but please pay attention to the following matters:

- Please ensure that there are no safety hazards in the movement mechanism accessories. When the run button is pressed, the motor will run at the set "target speed". Please confirm that the motor will not cause danger when running before using the jog function.
- If the parameter configuration is not set in line with the current mechanical structure, the mechanical equipment may vibrate, jitter or otherwise malfunction, and may even cause danger. Please make sure you have set the appropriate control parameters before using this function.
- Please ensure that the drive is not enabled; otherwise, this function cannot be used.
- The pulse model is configured with digital IO input function by default at the factory, which may affect the operation of the kspro. Please cancel the corresponding configuration according to the KS PRO prompt. If you do not need to use the corresponding IO function, you can clear the digital input function in batches.

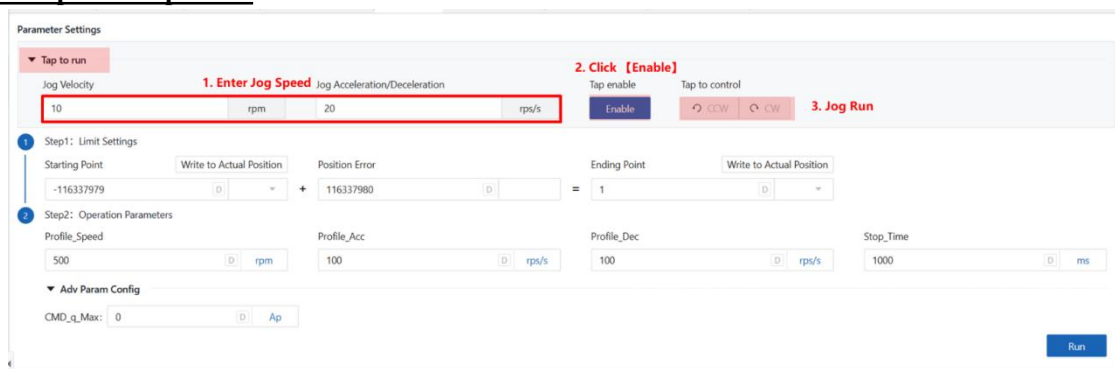
Position commissioning process

The position commissioning function enables the servo motor to perform reciprocating motion between the preset starting point and ending point. Through this function, users can simulate the movement state of the motor during actual operation, ensuring that the motor can operate smoothly within the set range.



- First, click [JOG] in the function configuration bar;
- **Motion range setting:** After selecting the unit as "r", input "1" to indicate one revolution of the motor shaft; select "internal unit" to indicate the number of encoder position pulses (DEC value). Then enter the starting point and stroke, or directly enter the end point, the motion range is the distance between the starting point and the end point;
- **Operating parameter setting:** Enter the trapezoidal speed, and the servo motor will run at this speed. At the same time, set operating parameters such as trapezoidal acceleration and deceleration;
- **Pause time setting:** The pause time is the duration required for the servo motor to reverse in both directions. After the setting is completed, the "Motion Preview" will display the planned speed curve and show the real-time running position.
- Finally, click the [Run] button on the right to start the position trial operation.

Tap to run operation process



- First, input the jog velocity;
- Click [Enable];

Long press the mouse to control the CW or CCW button, and the motor will run at the jog speed.

Chapter 8 Servo basic function

The servo drive is the core control unit of the servo system and is responsible for receiving and processing input signals and feedback signals to achieve precise control of the servo motor. Its control modes mainly include position control, speed control, torque control, homing control mode and pulse mode. This chapter introduces the basic operation process of each mode in detail. For performance adjustment, please refer to Chapter 10.

8.1 Server status

8.1.1 Server status

The use of iSMD must follow the DS402 protocol to guide the servo drive in order to operate in the specified state.

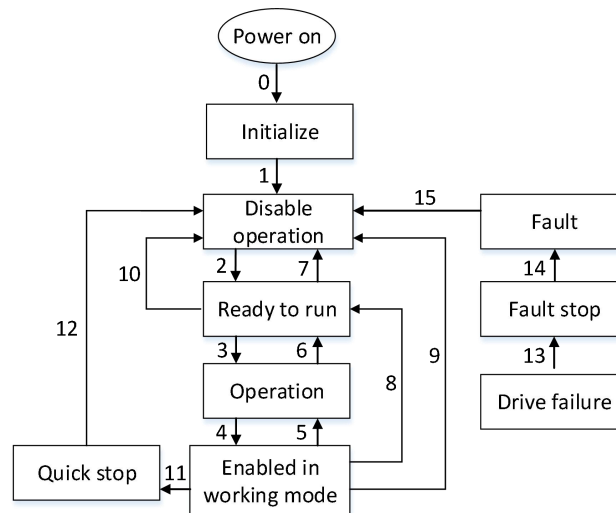


Figure 8-1 DS402 state machine

Each state is explained in the following table:

Table 8-1 State description

State	Explanation
Initialize	The drive is in the initializer state and cannot set parameters.
Disable operation	The drive is in the initializer state and cannot set parameters.
Ready to run	The drive is ready to run, you can set parameters to allow the drive to be enabled.
Operation	The driver main power circuit is ready but not powered, parameters can be set and allow the driver to be enabled.
Enabled in working mode	The driver is in the enable state, the main power circuit is energized to start the drive function, and the motor is running. Some parameters cannot be set in this state.
Quick stop	Execute the quick stop command and the motor stops according to the quick stop mode.
Fault stop	The driver fails and the motor stops according to the error stop mode.
Fault	The drive is in the faulty state and the fault shutdown has been completed. Enable the drive after troubleshooting and resetting the fault.

Table 8-2 Control commands and status switching description

Serial number	Status switching	Trigger event (bit3~bit0)	Drive state (bit7~bit0)
0	Power-on state → Initialization state	After the driver is powered on, it automatically switches the status without sending the control word	The drive initializes itself
1	Initialize state → Disable state	Automatically switches the status without delivering the control key	The drive communicates normally
2	Forbidden State → Ready State	Control word Enter 6	The status word is changed to 31
3	Ready State → Running state	Control word enter 7	The status word is changed to 33
4	Running State → Working Mode Enable state	Control word enter F	The status word is changed to 0037
5	Working Mode Enable state → Running state	Control word enter 7	The status word is changed to 0033
6	Running State → Ready state	Control word Enter 6	The status word is changed to 0031
7	Ready State → Disable State	Control word Enter 0	The status word is changed to 0070
8	Working Mode Enable status → Ready state	Control word Enter 6	The status word is changed to 0031
9	Working Mode Enable state → Disable State	Control word Enter 0	The status word is changed to 0070
10	Ready State → Disable State	Control word Enter 0	The status word is changed to 0070
11	Working Mode Enable state → Fast Stop state	Control word Enter 2	The status word is changed to 0050
12	Fast Stop state → Disable state	When the quick stop function is complete and the quick stop mode is 0, 1, 2, or 18, the drive automatically switches to the disabled state	Disable the driver and turn off the main power supply
13	→ Fault shutdown state	In the initialization, disable running, running ready, running, working mode enabled, fast stop state automatically switched when the fault condition is reached	Stop according to the error stop mode
14	Fault Stop state → Fault state	Automatic switchover	The status word is changed to 0038 The driver function needs to be disabled and the main power supply needs to be turned off
15	Fault state → Disable state	Control word Enter 86 Switch status Error reset command from I/O or remote control	The status word is changed to 0031 If the drive exception has been cleared, use the control word bit to clear the fault state
16	Quick Stop State → Working Mode Enable state	When the quick stop mode is 5 or 6, the control word input F switches the status	The drive function needs to be enabled

8.1.2 Control word and status word description

The following table describes the control words and status words:

Table 8-3 Control word and status word

Name	Object index	Property	Data type	Unit	Explanation	Factory default
Control word	6040.00	RWLTM	uint 16	HEX	bit0: Switch_on bit1: Enable_voltage bit2: Quick_stop bit3: Enable_operation bit4: Set_Point bit5: Immed_Change bit6: Related_Abs bit7: Fault_reset bit8: Halt bit9: Reserved0 bit10: Reserved1 bit11: Manufacture0 bit12: Manufacture1 bit13: Manufacture2 bit14: Manufacture3 bit15: Manufacture4	6
Status word	6041.00	RLTM	uint 16	HEX	bit0: Ready_on bit1: Switched_on bit2: Operation_enable bit3: Fault bit4: Voltage_enable bit5: Quick_stop bit6: Switchon_disabled bit7: Warning bit8: Maunufacture0 bit9: Remote bit10: Target_reached bit11: Intlim_active bit12: Setpoint_Ack bit13: Fllowing_Error bit14: Commutation_Found bit15: Reference_Found	-

8.1.2.1 Control word description 604000

Table 8-4 Control word declaration

Control word	State	Meaning	Note
bit0	0	Drive running - invalid	For details, see Table 8-2
	1	Drive running - valid	
bit1	0	Main power circuit ready - invalid	
	1	Main power circuit ready - valid	
bit2	0	Quick stop - valid	
	1	Quick stop - invalid	
bit3	0	Servo running - invalid	
	1	Servo running - valid	

bit4	0→1	Indicate that a new displacement command is pre-enabled. Whether the command is successfully enabled depends on the servo state	Position mode
	1→0	Indicates that bit12 of control word 6041h is pre-cleared. Whether it is successfully cleared depends on the servo status	
	0	Origin mode is not activated	Origin mode
	0→1	Active origin mode	
	1	The origin return to zero is in progress	
	1→0	Interrupt origin return to zero	
	0	Interrupt interpolation	Interpolation mode
	1	Enable interpolation	
bit5	0	Acceleration and deceleration, trapezoidal speed, target position is not updated	F→2F
	1	Acceleration and deceleration, trapezoidal speed, target position update immediately	
bit6	0	Target position 607Ah is a position instruction for absolute positioning	Position mode
	1	Target position 607Ah is a position instruction for relative positioning	
bit7	0→1	Fault reset - Rising edge effective	Fault reset
	1	Fault reset complete	
bit8	1	The servo motor is suspended	Applicable to operating modes other than torque mode (4 mode)
	0	Servo motor continues to run (continue to perform the previous task)	
bit9	Reserve		
bit10			
bit11			
bit12	0	The target position 607Ah is a position instruction for absolute positioning and does not update the contour trajectory according to the change of the target position	Position mode
	1	Target position 607Ah is a position instruction for absolute positioning, which updates the contour trajectory in real time according to the change of target position	
bit13	Reserve		
bit14			
bit15			
Motor lock shaft: 0x2F/0x0F Motor loose shaft: 0x06 Error reset: 0x86 Absolute position mode: 0x2F→0x3F Absolute positioning mode immediately as target position changes: 0x103F Relative position mode: 0x4F-->0x5F Find the origin: 0x0F->0x1F			

8.1.2.2 Status word description 604100

Table 8-5 Status word description

Control word	State	Meaning	Note
bit0	0	Servo no fault - invalid	For details, see Table 5-2
	1	Servo fault-free - valid	
bit1	0	Waiting to turn on the servo enable - invalid	
	1	Wait to turn on the servo enable - Valid	
bit2	0	Servo ready to run - invalid	
	1	Servo ready to run -valid	
bit3	0	Fault-invalid	
	1	Fault-valid	
bit4	0	Power on the main circuit-invalid	
	1	Power on the main circuit-valid	
bit5	0	Quick stop - valid*	
	1	Quick stop - invalid*	
bit6	0	Servo ready- invalid	
	1	Servo ready- valid	
bit7	0	The servo has no warning status word	
	1	Servo in warning status word	
bit8	Reserved, undefined		
bit9	0	DIN defines enable/reset faults, IO controls	
	1	Telecommunication control	
bit10	0	Target position, zero speed, target speed not reached	Target arrival
	1	Position, speed , zero speed reached	
bit11	0	The actual position does not reach the soft limit or hard limit	Positive/negative limit
	1	The actual position reaches the soft limit/hard limit position, and input the reverse shift instruction can make the motor exit the position limit state and clear the bit	
bit12	0	The servo can receive new displacement commands	Position mode
	1	The servo cannot receive new shift instructions	Homing mode
		Homing error	
	0	User speed is not zero	Speed mode
	1	User speed is zero	
bit13	0	The position following error does not exceed the maximum following error	Position following error
	1	The position following error exceeds the maximum following error	
bit14	0	No motor excitation found or motor excitation missing	Excitation state
	1	Motor excitation found	
bit15	0	The origin return to zero has not been performed or completed	Origin state
	1	The origin has been returned to zero. The reference point has been found	

8.2 Speed control mode (-3, 3)

The speed control mode controls the movement speed of the machine by inputting speed commands. The servo driver adjusts the speed of the motor according to the received speed command to achieve precise control of the mechanical speed. This mode is suitable for occasions that require constant speed. There are two speed modes: 3 and -3. In speed mode, the servo motor runs according to the speed command. The command issuance method is divided into: **Manual input**, **Din speed index** and **Simulate input**.

8.2.1 Speed mode function block diagram

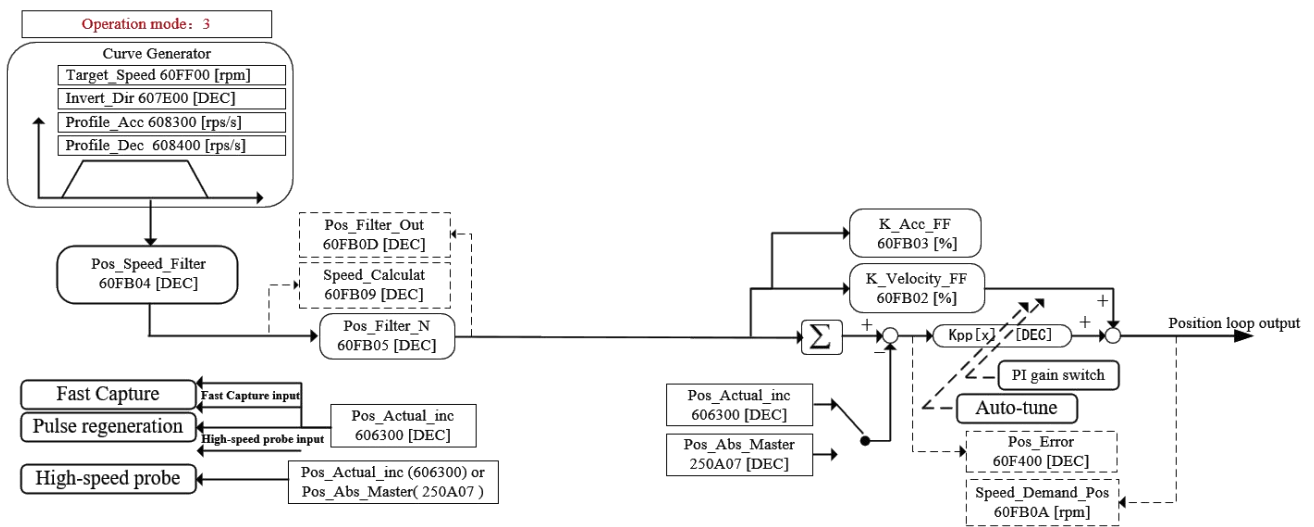


Figure 8-2 Speed mode position loop block diagram

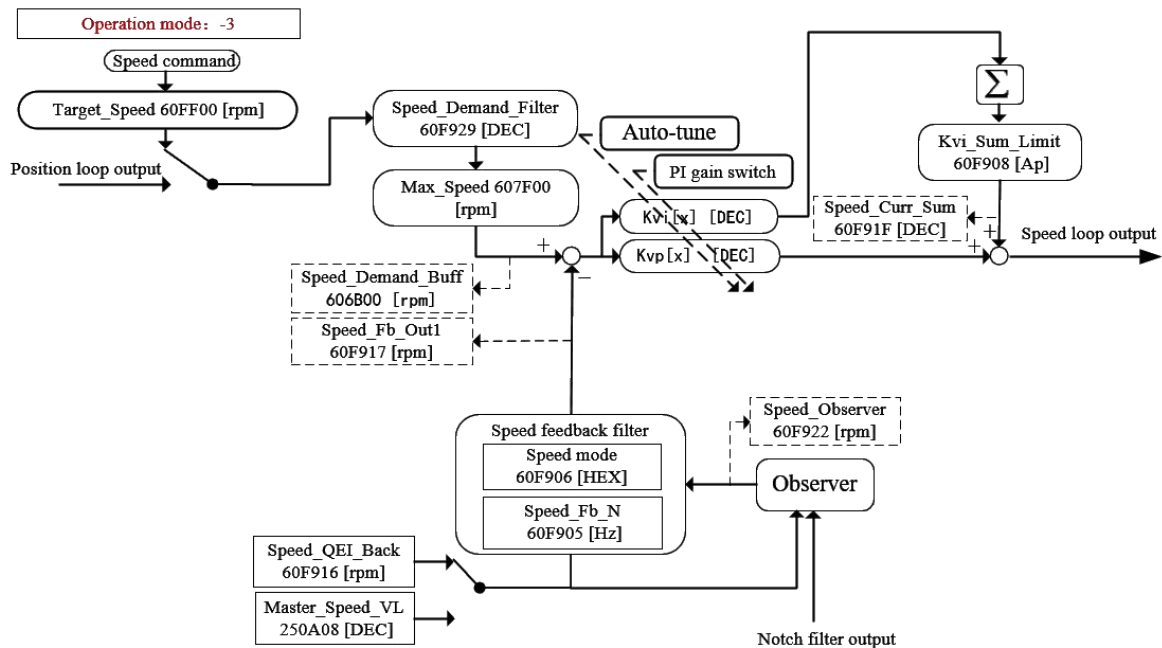


Figure 8-3 Speed mode speed loop block diagram

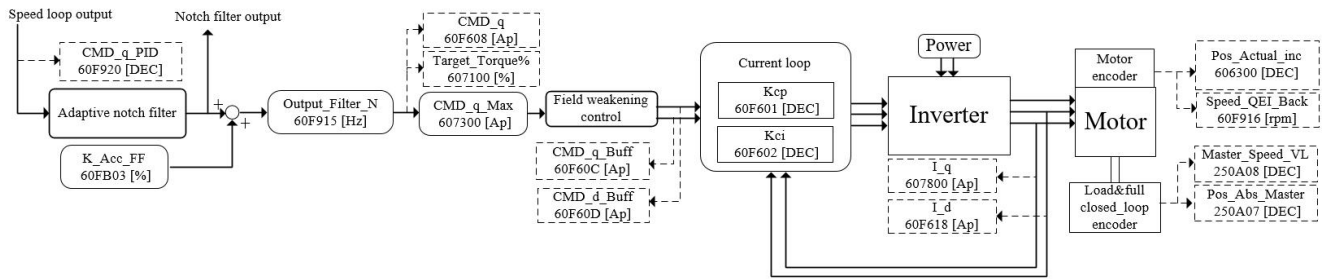


Figure 8-4 Speed mode current loop block diagram

8.2.2 Configuration process and panel operation

Step 1: Select the operation mode

The operation mode can be set through the following object:

Object index	Data type	Name	Description	Effect mode
6060.00	int 8	Operation_Mode	-3: Immediate speed mode, not involved in position loop control. The actual speed will immediately reach the target speed. 3: The speed mode with acceleration and deceleration is controlled by the position loop and the speed loop. The actual speed is planned as the target speed by trapezoidal acceleration/trapezoidal deceleration.	Immediate

Step 2: Select the running direction

The running direction can be set through the following object:

Object index	Data type	Name	Description	Effect mode
607E.00	uint 8	Invert_Dir	Run polarity reversal 0: Counterclockwise is the positive direction (default value) 1: Clockwise is the positive direction	Immediate

Step 3: Set the target speed

There are two sources for the target speed command :1. Manual input; 2.din speed index.

1. Manual input

The target speed can be manually set through the following object:

Object index	Data type	Name	Description	Effect mode
60FF.00	int 32	Target_Speed	User defined unit:rpm	Immediate

2. din speed index

The DIN speed can be manually set through the following objects:

Object index	Data type	Name	Description	Effect mode
2020.05	int 32	Din_Speed[0]	The speed command is specified via Din_Speed[x]. x is the BCD code of Bit 0: Din Vel Index0 Bit 1: Din Vel Index1 Bit 2: Din Vel Index2 As soon as one Index is configured to Din,the index which is not configured means 0	Immediate
2020.06	int 32	Din_Speed[1]		
2020.07	int 32	Din_Speed[2]		
2020.08	int 32	Din_Speed[3]		
2020.14	int 32	Din_Speed[4]		
2020.15	int 32	Din_Speed[5]		
2020.16	int 32	Din_Speed[6]		
2020.17	int 32	Din_Speed[7]		

The Din Vel Index 0, 1, 2 needs to be configured in the digital input port; for the configuration process, refer to **Chapter 6.2.5**; after the configuration is completed, the corresponding Din speed command can be obtained by triggering the BCD code composed of the corresponding digital input signal.

For example:

If Din Vel Index 0 is valid, and Din Vel Index 1 and 2 are invalid, the resulting BCD code is 001 (decimal value is 1), then Din_speed [1] is the valid speed command input;

If Din Vel Index 0 and 1 is valid , and Din Vel Index 2 is invalid, then the composed BCD code is 011 (decimal value is 3), then Din_speed [3] is the valid speed command input.

Step 4: Set the acceleration and deceleration

In Mode 3, acceleration and deceleration can be set through the following objects; Mode -3 is not effective.

Object index	Data type	Name	Description	Effect mode
6083.00	uint 32	Profile_Acc	User defined, unit: rps/s	Immediate
6084.00	uint 32	Profile_Dec	When the operation_mode is 3 mode, the trapezoidal acceleration and trapezoidal deceleration must be set, otherwise the response speed segment will not be executed.	

Step 5: Set the Max_Speed_RPM

In speed mode, the servo drive can limit the maximum operating speed of the motor. The specific object is as follows:

Object index	Data type	Name	Description	Effect mode
6080.00	uint 16	Max_Speed_RPM	Motor max speed limit User defined, unit: rpm	Immediate

Step 6: Enable and run

After the control word is written to 0x0F, the drive is enabled and the speed command is executed.

Object index	Data type	Name	Description	Effect mode
6040.00	uint 16	Controlword	0x0F Enabled 0x06 Disabled	Immediate

Step 7: Monitor relevant parameter

The following parameters can be monitored to observe the operating status of the motor:

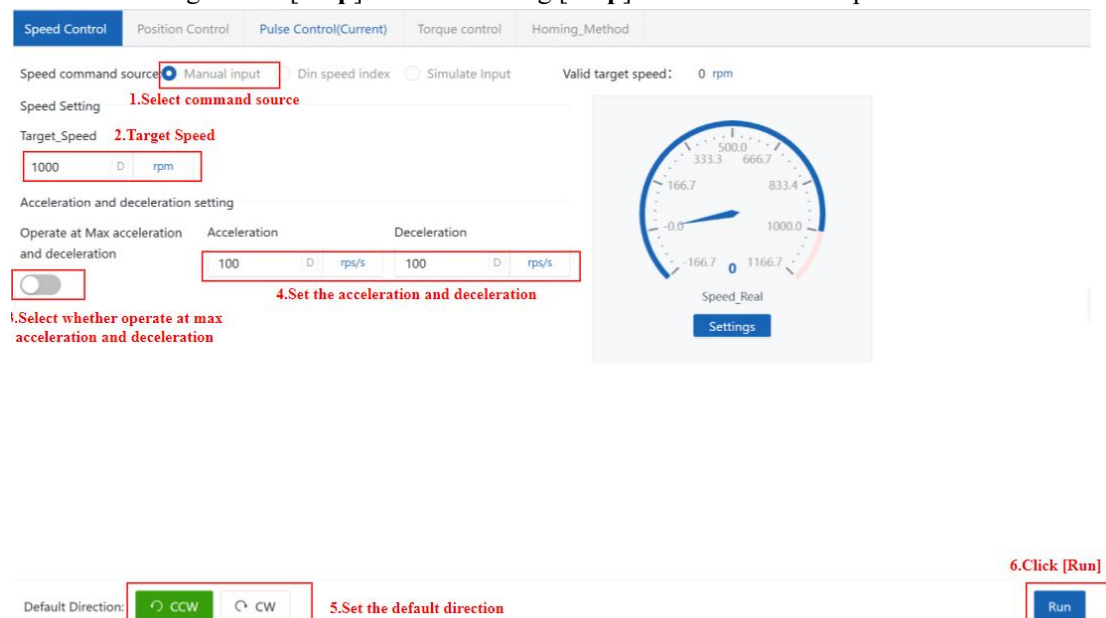
Object index	Data type	Name	Description
606B.00	int 32	Speed_Demand_Buff	In mode 3, the effective target speed is the speed planned as a trapezoidal curve. In mode -3, the effective target speed is equal to the target speed unit: rpm.
606C.00	int 32	Speed_Real	The actual speed of the motor, unit: rpm.
2FF0.14	int 32	Real_Speed_RPM	The speed value of the actual speed after filtering. unit: rpm

8.2.3 KS PRO operation

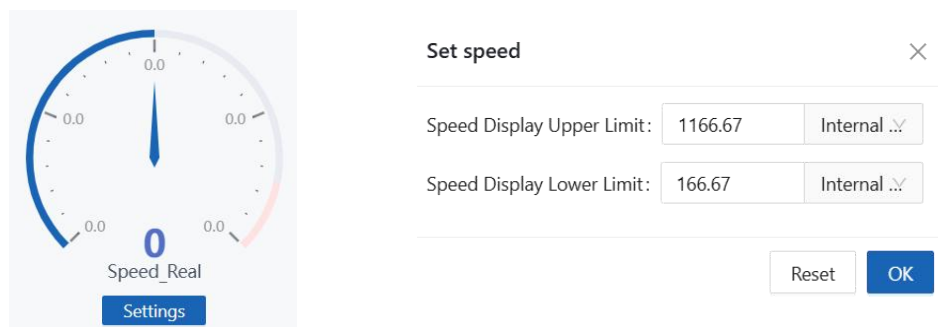
Manual input

In this mode, you can manually enter the speed command, configure the acceleration and deceleration, and then click Run.

- Set the **speed command source** to manual input;
- Set the Target_Speed;
- Choose whether to operate at the max acceleration or deceleration, If not selected, the acceleration or deceleration needs to be entered.
- Set the default direction;
- After clicking **[Run]**, the servo motor will start to respond to the speed command. At this point, the **[Run]** button will change to the **[Stop]** button. Clicking **[Stop]** can terminate the operation of the servo motor.



The upper and lower limits of the actual speed instrument panel will be automatically set according to the target speed. If you need to customize the upper and lower limits of the speed display, you can click the **[Settings]** button in the speed display instrument panel to enter the configuration page for settings;



DIN speed index

In din speed index mode, the servo motor executes the specified speed command according to the BCD code represented by DIN 0, DIN1 and DIN2. For example, when DIN0 input is high level and DIN1 and DIN2 are low level, the BCD code is 001, and the speed command will run at DIN speed [1]. The specific steps are as follows:

- Set the **speed command source** to Din speed index;
- Set the DIN speed;
- Choose whether to operate at the max acceleration or deceleration, If not selected, the acceleration or deceleration needs to be entered.
- Set the default direction;
- Click on [**Index Settings**], and configure the IO input ports as DIN vel index 0, 1, and 2. For the specific operation procedure, please refer to Chapter 6.2.5.
- After the configuration is completed, click [**Run**] to change the BCD value through IO input high-level or simulation input, thereby executing the corresponding speed command at different times.

Speed Control

Position Control Pulse Control(Current) Torque control Homing_Method

Speed command source: ☐ Manual input ☒ Din speed index ☐ Simulate Input Valid target speed: 0 rpm

Speed Setting

1. Select command source

2. Set DIN speed

Din Speed[0] 0 D rpm

Din Speed[1] 100 D rpm

Din Speed[2] 200 D rpm

Din Speed[3] 300 D rpm

Din Speed[4] 0 D rpm

Din Speed[5] 0 D rpm

Din Speed[6] 0 D rpm

Din Speed[7] 0 D rpm

Acceleration and deceleration setting

3. Select whether operate at max acceleration and deceleration

☒ Operate at Max acceleration and deceleration

Acceleration 100 D rps/s

Deceleration 100 D rps/s

4. Set the acceleration and deceleration

Default Direction: ☒ CCW ☐ CW

5. Set the default direction

6. Click [Index Settings]

7. Click [Run]

DIN input simulation

Din Speed Index 0

Din Speed Index 1

Din Speed Index 2

Din1

Din2

Din3

Digital Input (DI)

Name	Pin	Function	Batch Clear	Actual	Simulate	Polarity	Valid
Din1	X1.4	Din Vel Index0				High	
Din2	X1.6	Din Vel Index1				High	
Din3	X1.8	Din Vel Index2				High	
Din4	X1.10					High	
Din5	X1.12					High	
Din6	X1.14					High	
Din7	X1.16					High	

**Note**

- The green line segments in the figure represent the current valid levels of each DIN index. The corresponding DIN speed that is in effect is displayed in green, and the field above the input box on the left side will also be displayed in green.

Din Speed[0]

0

D

rpm

Din Speed[1]

0

D

rpm

Analog speed mode

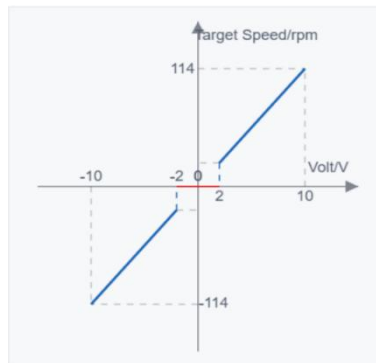
- Set the **speed command source** to Simulate Input;
- Set the **Simulate speed factor**, that is, the speed value corresponding to the 1V analog input voltage;
- Set the **Simulate speed offset**. For example, when set to +2V, the speed-analog correspondence shifts 2V to the right, as shown in the figure below.

Simulate input offset

0

D

V

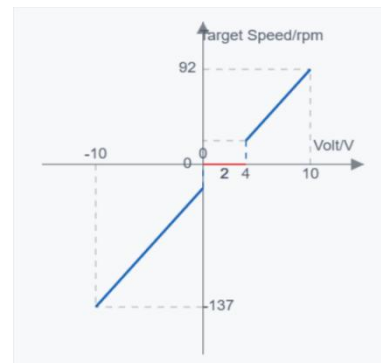


Simulate input offset

2

D

V



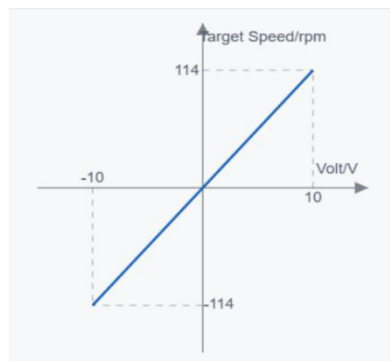
- Set the **Simulate input dead zone**, when the analog is in the dead zone, the corresponding velocity is 0;

Simulate input dead zone

0

D

V

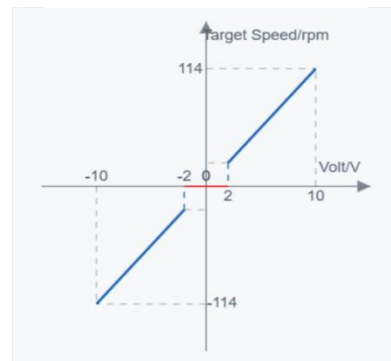


Simulate input dead zone

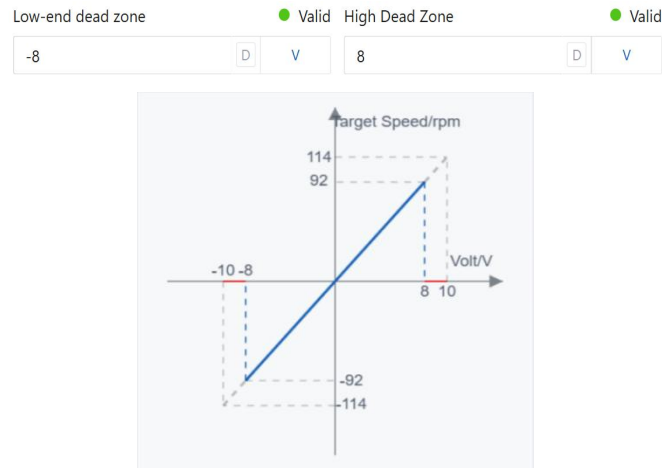
2

D

V



- Set the **Low-end dead zone** and **High-end dead zone**, When the analog input is less than the low-end dead zone or greater than the high-end dead zone, the target speed is 0.



- Choose whether to operate at the max acceleration or deceleration, If not selected, the acceleration or deceleration needs to be entered.
- Set the default direction;
- Click **[Enable]**, after inputting the analog voltage, the motor operates at the target speed mapped by the speed-analog voltage.

Speed Control | Position Control(Current) | Pulse Control | Torque control | Homing_Method

Speed command source: ☐ Manual input ☐ Din speed index ☒ **Simulate Input** 1. Select command source Valid target speed: 0 rpm

Simulate Speed Settings

2. Set the simulate speed factor Simulate speed factor: 11.44 D rpm/V

3. Set the simulate input filter Simulate input filter: 5 D V

4. Set the simulate input offset Simulate input offset: 0 D V

Simulate input dead zone: 2 D V Low-end dead zone: 0 D V High Dead Zone: 0 D V 5. Set the dead zone

Acceleration and deceleration setting

Operate at Max acceleration and deceleration: ☒ 6. Select whether operate at max acceleration and deceleration

Acceleration: 100 D rps/s Deceleration: 100 D rps/s 7. Set the acceleration and deceleration

Speed - Analog voltage mapping diagram Speed and Analog voltage display

Default Direction: CCW CW 8. Set the default direction 9. Click [enable] Enable

Related parameter:

Object index	Data type	Name	Description	Value
2501.06	uint 16	ADC1_Buff[1]	AIN1 input original ADC data	Only read
2502.0F	int 16	Analog1_out	AIN1 valid input; analog input signal1 (AIN1) input voltage after filter, deadband and offset	
2502.01	uint 16	Analog1_Filter	Analog input signal 1 (AIN1) filter	User defined
2FF0.1D	int 16	Analog1_Dead_V	AIN1 deadband (unit: 0.01V)	
2FF0.1E	int 16	Analog1_Offset_V	AIN1 offset (unit: 0.01V)	

2502.0A	int 16	Analog_Speed_Factor	AIN speed factor	
2502.07	uint 8	Analog_Speed_Con	Analog signal control speed, valid at operation mode 3 or -3 1: Ain1 Control Speed	1
2502.0D	int 16	Analog_Dead_High	Default: 0 if it's NOT 0, Analog_out> Analog_Dead_High is treated as 0	User defined
2502.0E	int 16	Analog_Dead_Low	Default: 0 if it's NOT 0, Analog_out< Analog_Dead_Low is treated as 0	
6060.00	int 8	Operation mode	Select the operation mode according to the actual control mode	
6040.00	uint 16	Controlword	Enable drive	
Note: Every time Analog1_out (2502.0F) increases or decreases by 0.1V, the corresponding theoretical value of DEC change is 285DEC (2501.06). Due to temperature drift, zero point drift, environmental electromagnetic interference, etc., the theoretical value may have certain errors in actual use.				

For convenience, some new names are used in the formula. Definitions:

AIN1_in : AIN1 input voltage after filter and offset

Analog1_out: AIN1 valid input ,which depends on the user's hardware wiring and the selection of the analog input channel. This data value is obtained after deadband and offset processing followed by filtering.

Computational formula:

If Analog_out is not limited by Analog_Dead_High or Analog_Dead_Low:

Target speed[rpm]=Analog_out *Analog_Speed_Factor; otherwise Target speed[rpm]=0.

For example:

Set

Analog1_Dead_V = 1, Analog1_Offset_V = 2, Analog_Speed_Factor = 100;

Analog_Speed_Con = 1, Analog_Dead_High = 0; Analog_Dead_Low = 0;

Where AIN1 input voltage is 5V:

AIN1_in=5V-2V=3V, |AIN1_in| >Analog1_Dead, so Analog1_out=3V-1V=2V;

Target speed=2*100=200rpm.

Where AIN1 input voltage is -5V:

AIN1_in=-5V-2V=-7V, |AIN1_in|>Analog1_Dead, so Analog1_out=-7V+1V=-6V;

Target speed=-6*100=-600rpm.

8.2.4 IO function related to speed mode

The IO function related to speed mode need to be configured through the upper computer. For the configuration process, please refer to Section 6.2.5.

Enable and reset fault

After configuring the enable signal in the digital input, the control word can be switched between the 0x06

->Din_control_word (20200F) setting value, thereby enabling the drive;

Reset the fault can restore the drive error.

Input function	Description
Enable	Drive enable 1: Controlword = Din_control_word (20200F) 0: Controlword = 0x06
Reset fault	Controlword.bit7(Fault_reset) = 1

Relevant parameter:

Object index	Name	Description	Set value
2020.0F	Din_control_word	If "Enable" function is configured to Din, the "Control word" (6040.00) will be set to this value when the corresponding Din valid input is 1	0x2F
6040.00	Controlword	0x06: Disable 0x0F: Enable 0x0B: Quick stop, load stop - voltage disconnection 0x2F→3F: Enter the absolute positioning mode 0x4F→5F: Relative positioning 0x103F: Immediate absolute positioning based on changes in target position 0x0F→1F:Homing 0x86: Error reset	User defined

Din speed index [x]

This output function is used in the Din speed indexing mode. Each digital input port can only be configured with one Din speed index. When a digital input port is activated, the BCD code composed of Din speed index [x] can trigger the corresponding speed command in the Din speed indexing mode.

Input function	Description
DinVel Index0	Din speed index in Din speed mode
DinVel Index1	
DinVel Index2	

Zero-speed detection

This function is used to detect whether the motor has stopped moving.

Output function	Description
Zero_Speed	When the absolute value of the actual velocity -ms (60F9.1A) is less than or equal to the zero velocity window (2010.18), and the duration is greater than or equal to the zero velocity time (60F9.14), the zero speed function is output

Relevant parameter:

Object index	Name	Description	Set value
2010.18	Zero_Speed_Window	if Speed_1ms (60F9.1A)<="Zero_Speed_Window"(2010.18) then it's treated as zero speed, only if the zero speed lasting time >="Zero_Speed_Time"(60F9.14),the Dout with "Zero_Speed" function is activated	3inc/ms
60F9.14	Zero_Speed_Time		10ms
60F9.1A	Speed_1ms	Speed data[inc/1ms]	Read only

Speed reach detection

This function is used to detect whether the actual speed of the motor has reached the specified value.

Output function	Description
Speed reach	Target_Speed_Window(unit:ms),Under velocity mode,if Speed_Error (60F9.1C)<Target_Speed_Window means target reached,Statusword.bits.Target_reached=1;

Relevant parameter:

Object index	Name	Description	Set value
60F9.0A	Target_Speed_Window	Target_Speed_Window(unit:ms),Under velocity mode,if Speed_Error (60F9.1C) < Target_Speed_Window means target reached,Statusword.bits.Target_reached=1;	10rpm
60F9.1C	Speed_Error	Actual speed error	Read only

Max Speed RPM

The maximum speed limit can be set through the internal address 608000

608000				Bulk Actions	
☐	Index	Description	Preset value	Current value	
☐	608000	Max_Speed_RPM	5000	5000	



Note

- The maximum speed limit is also effective in all other mode.

8.3 Position control mode (1)

Position control mode is the most commonly used control method in servo systems. By inputting a position command, the servo drive controls the motor to move to the specified position. This mode is widely used in situations where precise positioning is required. In position mode, the driver controls the motor to perform two positioning methods: absolute position positioning and relative position positioning. According to the source of speed and position instructions, it is divided into: **Manual input** and **Din position index**.

8.3.1 Position mode function block diagram

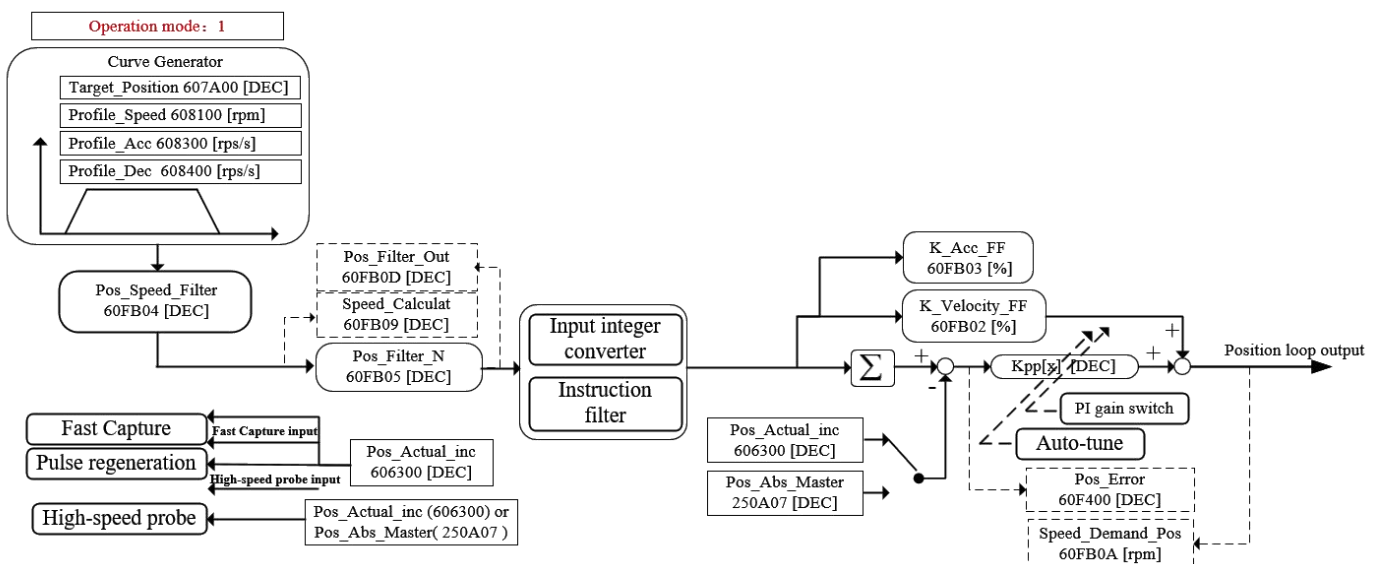


Figure 8-5 Position mode position loop block diagram

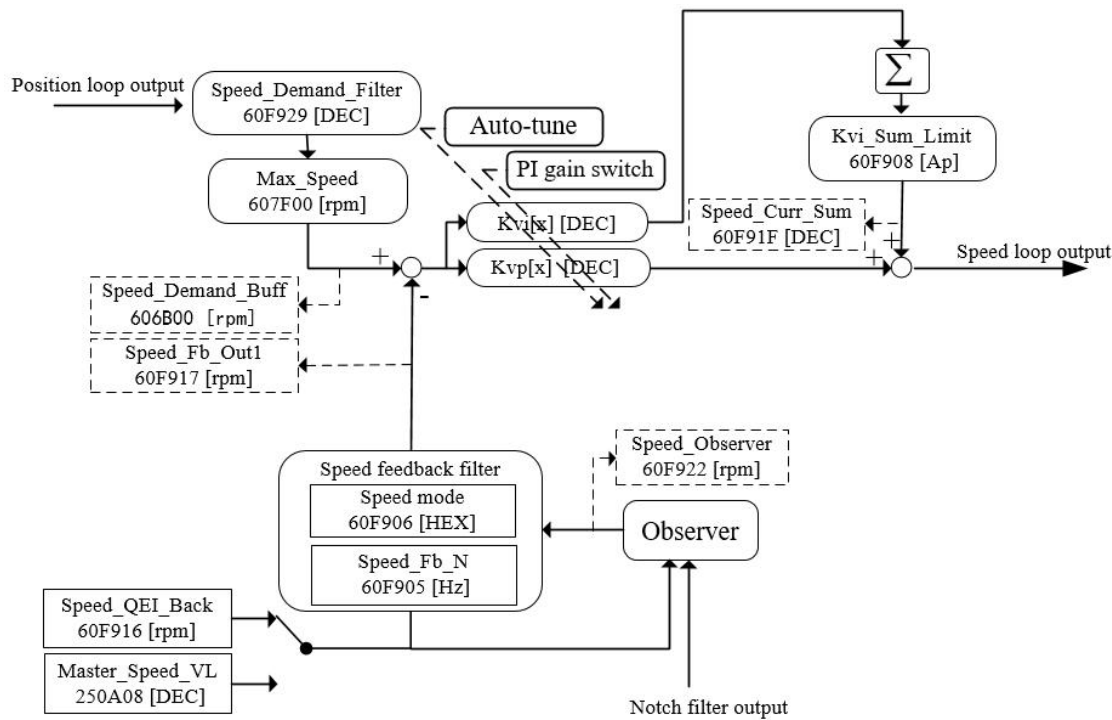


Figure 8-6 Position mode speed loop block diagram

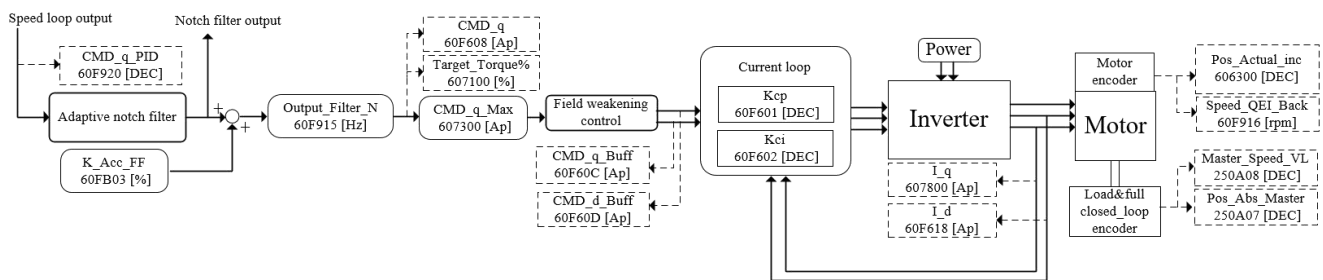


Figure 8-7 Position mode current loop block diagram

8.3.2 Configuration process and panel operation

Step 1: Select the operation mode

The operation mode can be set through the following object:

Object index	Data type	Name	Description	Effect mode
6060.00	int 8	Operation_Mode	1: Set the operation mode to absolute/relative position mode	Immediate

Step2: Set the Target_Position

There are two sources for the target position command :1. Manual input; 2.Din position index.

1.Manual input

The target position can be manually set through the following object:

Object index	Data type	Name	Description	Effect mode
607A.00	int 32	Target_Position	Target absolute/relative position instruction value User defined, unit:DEC	Immediate

2.Din position index

The DIN position can be manually set through the following objects:

Object index	Data type	Name	Description	Effect mode
2020.01	int 32	Din_pos[0]	The position command is specified via Din_Speed[x]. x is the BCD code of Bit 0: Din Pos Index0 Bit 1: Din Pos Index1 Bit 2: Din Pos Index2 The case where the digits are all 0 cannot occur; User defined unit:rpmmm	Immediate
2020.02	int 32	Din_pos[1]		
2020.03	int 32	Din_pos[2]		
2020.04	int 32	Din_pos[3]		
2020.10	int 32	Din_pos[4]		
2020.11	int 32	Din_pos[5]		
2020.12	int 32	Din_pos[6]		
2020.13	int 32	Din_pos[7]		
2FF1.01	uint 8	Din_position_select_L	Select the position segment L to be set (L range is 0-7, corresponding to the internal position segment 0-7 in turn)	
2FF1.02	int 16	Din_position_M	The number of pulses set for the position segment (L) =M*10000+N	
2FF1.03	int 16	Din_position_N		

The Din Pos Index 0, 1, 2 needs to be configured in the digital input port; for the configuration process, refer to **Chapter 6.2.5**; after the configuration is completed, the corresponding Din position command can be obtained by triggering the BCD code composed of the corresponding digital input signal.

For example:

If Din Pos Index 0 is valid, and Din Pos Index 1 and 2 are invalid, the resulting BCD code is 001 (decimal value is 1), then Din_pos [1] is the valid position command input;

If Din Pos Index 0 and 1 is valid , and Din Pos Index 2 is invalid, then the composed BCD code is 011 (decimal value is 3), then Din_pos [3] is the valid position command input.

Step 3: Set the acceleration and deceleration

Set the trapezoidal speed and trapezoidal acceleration and deceleration for position planning through the following objects:

Object index	Data type	Name	Description	Effect mode
6081.00	uint 32	Profile_Speed	Speed command in position mode, user defined Unit:rpm	Immediate
6083.00	uint 32	Profile_Acc	Acceleration of trapezoidal curve user defined Unit:rps/s	Immediate
6084.00	uint 32	Profile_Dec	Deceleration of trapezoidal curve user defined Unit:rps/s	Immediate

If the DIN position mode is used, the position planning speed is the speed index of DIN[X], and it is set according to the following objects:

Object index	Data type	Name	Description	Effect mode
2020.05	int 32	Din_Speed[0]	The speed command is specified via Din_Speed[x]. x is the BCD code of Bit 0: Din Vel Index0 Bit 1: Din Vel Index1	Immediate
2020.06	int 32	Din_Speed[1]		
2020.07	int 32	Din_Speed[2]		

2020.08	int 32	Din_Speed[3]	Bit 2: Din Vel Index2 As soon as one Index is configured to Din,the index which is not configured means 0	
2020.14	int 32	Din_Speed[4]		
2020.15	int 32	Din_Speed[5]		
2020.16	int 32	Din_Speed[6]		
2020.17	int 32	Din_Speed[7]		

Step 4: Enable and run

The position operation mode is defined by the control word and is divided into absolute position, phase position and immediate response to target position change mode:

Object index	Data type	Name	Description	Effect mode
6040.00	uint 16	Control word	0x2F->0x3F: When the absolute position command is activated, the absolute positioning command will not be executed immediately according to the change of the target position. It is used when the operation mode is 1. 0x4F->0x5F: Activate relative position command, it is used when the operation mode is 1. 0x103F:Immediately execute absolute positioning instructions according to changes in target position, it is used when the operation mode is 1.	Immediate

Step 5: Monitor relevant parameter

The following parameters can be monitored to observe the operating status of the motor:

Object index	Data type	Name	Description
6063.00	int 32	Pos_Actual_inc	Motor actual position
606B.00	int 32	Speed_Demand_Buff	Planning speed value
2FF0.14	int 32	Real_Speed_RPM	The speed value of the actual speed after filtering

8.3.3 KS PRO operation

In position mode, the servo motor operates according to position instruction. The instruction issuance methods are divided into **Manual input** and **Din position index** mode

Manual input

1.Absolute position

- Select operation method: **Absolute position**;
- Set the Target_position;
- Choose whether to execute immediately after change.If this option is not checked, the position command will only be responded to once when locking the shaft and running subsequently. After entering a new command, you need to click [**Disable**] first and then click[**Enable**]and [**run**]. If this option is checked, after enabling and click [**Run**] , when entering a new target command, press Enter to respond to the target position command.

Speed Control **Position Control** Pulse Control(Current) Torque control Homing_Method

Position command source ☒ Manual input ☐ Din position index Valid target position: 0 inc

1.Select command source

Position settings

Operation Method: ☒ Absolute position ☐ Relative position

2.Select operation method

Target_position 100000 D inc Execute immediately after change

3.Set the target position 4.Choose whether to execute immediately after change

Speed Setting

Operation speed 1000 D rpm

5.Set the operation speed

Acceleration 100 D rps/s Deceleration 100 D rps/s

Speed planning display

6.Click [Enable]

Enable Pause

- After clicking [Enable], the [Enable] button will change to [Disable] button, and the drive will enter the enabled state;



- After clicking [Run], the motor will run to the target position. While the motor is running, the [Run] button will change to the [Pause] button. Click the [Pause] button, the drive controls the servo motor to pause, but is still in the enabled state, and can continue to respond to command after resuming operation.



2.Relative position

- Select operation method: **Relative position**;
- Set the position increment;
- Set the operation speed;

Speed Control **Position Control** Pulse Control(Current) Torque control Homing_Method

Position command source ☒ Manual input ☐ Din position index Valid target position: 0 inc

1.Select command source

Position settings

Operation Method: ☐ Absolute position ☒ Relative position

2.Select operation method

Position increment 100000 D inc 3.Set the position increment

Speed Setting

Operation speed 1000 D rpm

4.Set the operation speed

Acceleration 100 D rps/s Deceleration 100 D rps/s

Speed planning display

5.Click [Enable]

Enable Pause

- After clicking **[Enable]**, the **[Enable]** button will change to **[Disable]** button, and the drive will enter the enabled state;



- After clicking **[Run]**, the motor will run to the distance corresponding to the "position increment" based on the current position. While the motor is running, the **[Run]** button will change to the **[Pause]** button. Click the **[Pause]** button, the drive controls the servo motor to pause, but is still in the enabled state, and can continue to respond to command after resuming operation.



Din position index

In din position index mode, the servo motor executes the specified position command according to the BCD code represented by DIN 0, DIN1 and DIN2. For example, when DIN0 input is high level and DIN1 and DIN2 are low level, the BCD code is 001, and the speed command will run at DIN Speed[1]. The specific steps are as follows (Taking the absolute position operation method as an example) :

- Set the **speed command source** to Din position index;
- Set the **operation method** to Absolute position;
- Set the DIN Target position;
- Set the operation speed;
- Set the Acceleration and Deceleration;
- Click on **[Index Settings]**, and configure the IO input ports as DIN pos index 0, 1, and 2. For the specific operation procedure, please refer to **Chapter 6.2.5**
- After the configuration is completed, click **[Enable]** to change the BCD value through IO input high-level or simulation input, thereby executing the corresponding position command at different times.

The screenshot displays the Kinco servo control software interface with the following components and annotations:

- Top Tabs:** Speed Control, **Position Control** (selected), Pulse Control(Current), Torque control, Homing_Method.
- Position command source:** Manual input, **Din position index** (selected).
- Valid target position:** 0 inc.
- 1. Select command source:** A red box highlights the "Din position index" selection.
- 2. Select operation method:** A red box highlights the "Absolute position" radio button.
- 3. Set the DIN target position:** A red box highlights the "Din Target position" input fields (0 to 7).
- 4. Set the DIN speed:** A red box highlights the "Din Speed" input fields (0 to 7).
- 5. Set the Acceleration and Deceleration:** A red box highlights the "Acceleration" and "Deceleration" input fields.
- 6. Click [Index Settings]:** A red box highlights the "Index Settings" button.
- 7. Click [Enable]:** A red box highlights the "Enable" button at the bottom right.
- DIN input simulation:** A red box highlights the "DIN input simulation" section, which includes a graph of Din Position (0 to 7) and three digital input waveforms for Din pos index 0, 1, and 2.

Digital Input (DI)						Batch Clear	Actual	Simulate	Polarity	Valid
Name	Pin	Function								
Din1	X1.4	Din Pos Index0	✕	▼	🗑	●	●	⚙	High	●
Din2	X1.6	Din Pos Index1	✕	▼	🗑	●	●	⚙	High	●
Din3	X1.8	Din Pos Index2	✕	▼	🗑	●	●	⚙	High	●



Note

- The green line segments in the figure represent the current valid levels of each DIN index. The corresponding DIN speed that is in effect is displayed in green, and the field above the input box on the left side will also be displayed in green.

Din Target position[0]	Din Target position[1]
0	10000

8.3.4 IO function related to position mode

The IO function related to position mode need to be configured through the upper computer. For the configuration process, please refer to **Section 6.2.5**.

Enable and reset fault

After configuring the enable signal in the digital input, the control word can be switched between the 0x06 -> Din_control_word (20200F) setting value, thereby enabling the drive;

Reset the fault can restore the drive error.

Input function	Description
Enable	Drive enable 1: Controlword = Din_control_word (20200F) 0: Controlword = 0x06
Reset fault	Controlword.bit7(Fault_reset) = 1

Relevant parameter:

Object index	Name	Description	Set value
2020.0F	Din_control_word	If "Enable" function is configured to Din, the "Control word" (6040.00) will be set to this value when the corresponding Din valid input is 1	0x2F
6040.00	Controlword	0x06: Disable 0x0F: Enable 0x0B: Quick stop, load stop - voltage disconnection 0x2F→3F: Enter the absolute positioning mode 0x4F→5F: Relative positioning 0x103F: Immediate absolute positioning based on changes in target position 0x0F→1F:Homing 0x86: Error reset	User defined

Command activation

The command activation function enables position mode absolute positioning or relative positioning commands to be activated when the drive is enabled.

Input function	Description
Command activation	Activate position instructions, such as Controlword: 0x2F→3F and 0x4F→5F

Din pos index [x]

This output function is used in the Din position indexing mode. Each digital input port can only be configured with one Din speed index. When a digital input port is activated, the BCD code composed of Din pos index [x] can trigger the corresponding position command in the Din position indexing mode.

Input function	Description
Din Pos Index0	Din Pos index in Din position mode
Din Pos Index1	
Din Pos Index2	

Zero-speed detection

This function is used to detect whether the motor has stopped moving.

Output function	Description
Zero_Speed	When the absolute value of the actual velocity -ms (60F9.1A) is less than or equal to the zero velocity window (2010.18), and the duration is greater than or equal to the zero velocity time (60F9.14), the zero speed function is output

Relevant parameter:

Object index	Name	Description	Set value
2010.18	Zero_Speed_Window	if Speed_1ms (60F9.1A)<="Zero_Speed_Window"(2010.18) then it's treated as zero speed, only if the zero speed lasting time >="Zero_Speed_Time"(60F9.14),the Dout with "Zero_Speed" function is activated	3inc/ms
60F9.14	Zero_Speed_Time		10ms
60F9.1A	Speed_1ms	Speed data[inc/1ms]	Read only

Speed reach detection

This function is used to detect whether the actual speed of the motor has reached the specified value.

Output function	Description
Speed reach	Target_Speed_Window(unit:ms),Under velocity mode,if Speed_Error (60F9.1C)<Target_Speed_Window means target reached,Statusword.bits.Target_reached=1;

Relevant parameter:

Object index	Name	Description	Set value
60F9.0A	Target_Speed_Window	Target_Speed_Window(unit:ms),Under velocity mode,if Speed_Error (60F9.1C) < Target_Speed_Window means target reached,Statusword.bits.Target_reached=1;	10rpm
60F9.1C	Speed_Error	Actual speed error	Read only

Position reach detection

Output function	Description
Position reach	In positioning mode, if position difference between Pos_Actual and Pos_Target is smaller than "Target_Pos_Window"(6067.00) and lasting time >= Position_Window_time(6068.00) then Status_Word.bits.Target_reached is set to 1

Relevant parameter:

Object index	Name	Description	Set value
6067.00	Target_Pos_Window	In positioning mode, if position difference between Pos_Actual and Pos_Target is smaller than "Target_Pos_Window"(6067.00) and lasting time >= Position_Window_time(6068.00) then Status_Word.bits.Target_reached is set to 1	327DEC
6068.00	Position_Window_time		10ms

Multi-function signal

In the multi-segment position mode, the position reach signal of Din position 1 to 7 segment is represented by the BCD code composed of multi-function signal 0 to 2

Output function	Description
Multi DOUT0	The driver executes the selected position segment program and outputs a multifunction signal after reaching the Din target position [x], where the x is a BCD code composed of the following signal Bit 0: Multi DOUT0 ; Bit 1: Multi DOUT1; Bit 2: Multi DOUT 2;
Multi DOUT1	
Multi DOUT2	

8.4 Torque control mode (4)

The torque control mode controls the motor torque output by inputting a torque command. Since the current and torque of the servo motor have a linear relationship, controlling the current can achieve control torque. This mode is suitable for devices that have strict requirements on material stress, such as rewinding and unwinding devices and other tension control occasions, to ensure that the material stress is not affected by changes in the winding radius. In torque mode, the drive will control the motor to output the user-set torque during operation.

8.4.1 Torque mode function block diagram

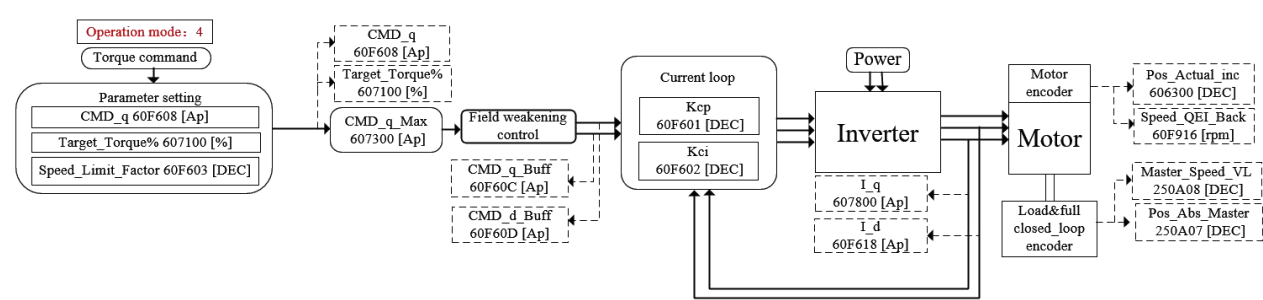


Figure 8-8 Torque mode block diagram

8.4.2 Configuration process

Step 1: Select the operation mode

The operation mode can be set through the following object:

Object index	Data type	Name	Description	Effect mode
6060.00	int 8	Operation_Mode	4: Set the operation mode to torque mode	Immediate

Step 2: Set the target torque

The target torque can be set through the following object:

Object index	Data type	Name	Description	Effect mode
6071.00	int 16	Target_Torque%	Target_torque/rated_torque*100 (unit:%)	Immediate

Step 3: Set the max speed limit for the torque mode

First, the speed limit factor needs to be set. When it is set to 0, the speed loop is used for speed limit; when it is not 0, the speed limit factor is used for speed limit. The specific object description is as follows:

Object index	Data type	Name	Description	Effect mode
60F6.03	uint 16	Speed_Limit_Factor	When set to 0, the maximum speed is limited through speed closed-loop control. That is, in torque mode, when the actual speed is greater than the maximum speed limit, the torque is automatically adjusted through PID control, so that the actual speed is maintained near the limit value. Using this mode requires tuning the PID parameters in advance. When it is not 0, the speed limit factor is used for speed limitation, that is, when the actual speed is greater than the maximum speed limit, the torque is linearly adjusted according to the speed factor. The larger the speed factor, the faster the adjustment, but at the same time, it is easy to cause poor speed fluctuation effects.	Immediate

Then set the Max_Speed according to the following object:

Object index	Data type	Name	Description	Effect mode
607F.00	uint 32	Max_Speed	Motor operating maximum speed limit unit:rpm	Immediate

Step 4: Torque reach configuration

Set the Torque_Out_Ref, Torque_Out_Filter, and when the actual torque $\geq$ Torque_Out_Ref, the **Ref torque reached** is effective

Object index	Data type	Name	Description	Effect mode
60F5.06	int 16	Torque_Out_Ref	The limit value, the actual torque exceeds Torque_Out_Ref and the duration exceeds the Torque_Out_Filter, Dout port output Ref torque reached signal, unit:Ap	Immediate
60F5.07	int 16	Torque_Out_Filter	the actual torque exceeds Torque_Out_Ref and the duration exceeds the Torque_Out_Filter, Dout port output Ref torque reached signal, unit:ms Max value:32767	Immediate
60F5.08	int 16	Current_Out_Real	Monitors the actual torque, unit:Ap	Immediate

Step 5: Enable and run

Enable the drive and run the torque mode through the following object:

Object index	Data type	Name	Description	Effect mode
6040.00	uint 16	Controlword	0x0F Enabled	Immediate

Step 6: Monitor relevant parameter

The following parameters can be monitored to observe the operating status of the motor:

Object index	Data type	Name	Description
6077.00	int 16	Actual_Torque	Motor Actual torque

8.4.3 KS PRO operation

- First, set the Target_Torque% (expressed as a percentage of the rated torque). When set to a positive value, the motor rotates in the positive direction. When set to a negative value, the motor rotates in the opposite direction.
- If the Speed_Limit_Factor(60F603) is written to 0, the speed limit will be implemented through PID control. Otherwise, speed limit factor should be used for speed control.

☐	Index	Description	Preset value	Current value
☐	60F603	Speed_Limit_Factor		10 DEC

- In torque mode, set the Max_speed as needed; in torque mode 4, enter the corresponding torque command and enable the drive, the speed limit function will automatically take effect according to the parameters set by the Speed_Limit_Factor (60F6.03), and the Max_speed (607F.00) will be set below the Max value.

☐	Index	Description	Preset value	Current value
☐	607F00	Max_Speed		5000 DEC

- (Optional) Set the Torque_Out_Ref and Torque_Out_Filter, and configure the ref torque reached signal to DOUT port. (For the configuration process, please refer to Section 9.2.5)
- After clicking [Enable], the servo motor will output torque to reach the target value and ensure that the actual speed does not exceed the maximum speed limit. When the actual torque reaches the reference value range and remains within the filter time, the **Ref torque reached** signal becomes valid.

Speed Control Position Control Pulse Control(Current) **Torque control** Homing_Method

1. Select command source

Torque command source: ☒ Manual input ☐ Simulate Input

Torque Setting

Target_Torque: 0 % Max_Torque: 314.4 %

2. Set target torque 3. Set max torque

Run settings

Direction: CCW CW Max Speed: 5000 rpm Speed Limit Factor: 0

4. Set max speed

Torque Reach Setting

Torque_Out_Ref: 0.5 Ap T Torque_Out_Filter: 100 D Ref Torque Reached: Invalid

5. Set Torque_Out_Ref 6. Set Torque_Out_Filter 7. Configuration torque to digital output

Actual Torque: 157.5, 78.75, 236.25, 0, 315

Speed Real: 1250, 2500, 3750, 0, 5000

Rated_Current: 6200 mA Driver Current Limit: 19.49 Ap

8. Click [Enable]

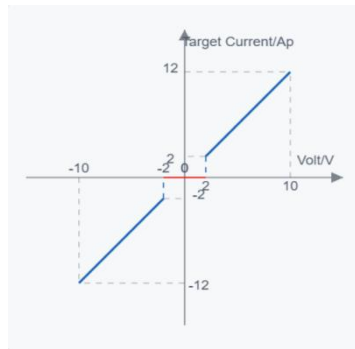
Enable

Analog torque mode

- Set the **Torque command source** to simulate input;
- Set the Analog_Torque_Factor, that is, the target current value corresponding to the 1V analog input voltage;
- Set the Simulate input offset, for example, when set to +2V, the torque-analog correspondence shifts 2V to the right, as shown in the following figure;

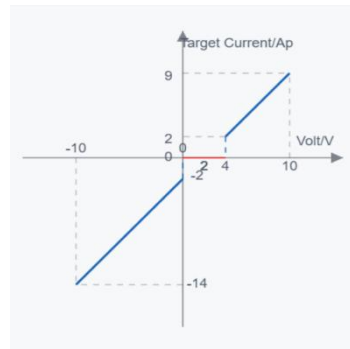
Simulate input offset

0



Simulate input offset

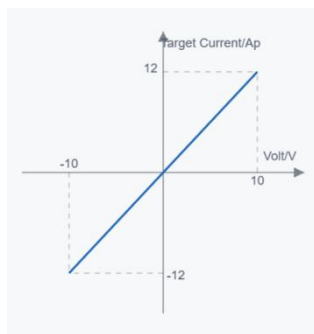
2



- Set the Simulate input dead zone. When the analog voltage is within the dead zone, the corresponding torque is 0.

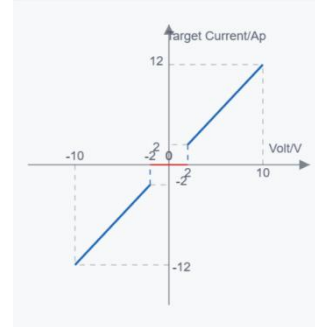
Simulate input dead zone

0



Simulate input dead zone

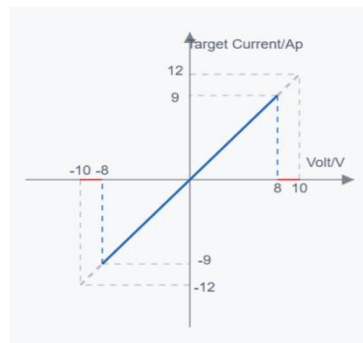
2



- Set the Low-end dead zone and High-end dead zone, When the analog input is less than the low-end dead zone or greater than the high-end dead zone, the target speed is 0.

Low-end dead zone ● Valid High Dead Zone ● Valid

-8 8



- Set the running direction;
- Set the max speed;
- Set the speed limit factor(Default:10).If the speed jitter is too large during operation, it can be appropriately reduced. Generally, no changes are needed;
- (Optional) Set Torque_Out_Ref and Torque_Out_Filter,and configure torque to digital output;

- Click [Enable], after the analog voltage is input, the motor operates at the target torque mapped by torque-analog .

Speed Control | Position Control | Pulse Control(Current) | **Torque control** | Homing_Method

Torque command source: ☐ Manual input ☒ **Simulate Input** 1.Select command source

Simulation Rectangle settings

Simulate Force Factor: 1.17 Ap/v 2.Set the simulate force factor 3.Set the simulate input filter

Simulate input filter: 5 4.Set the simulate input offset

Simulate input offset: 0 V

Simulate input dead zone: 0 V

Low-end dead zone: 0 V

High Dead Zone: 0 V

Invalid

Run settings

Direction: CCW CW 6.Set the direction

Max Speed: 5000 rpm 7.Set the max speed and speed limit factor

Speed Limit Factor: 10

Torque_Out_Ref: 0 Ap

Ttorque_Out_Filter: 100

Ref Torque Reached: Not config

Invalid

8.Set the Tprque_Out_Ref 9.Set the Tprque_Out_Filter 10.Configuration torque to digital output

Current_Out_Real: 0 Ap

Valid 10 V Actual 10 V

Actual Torque

Speed, Real

11.Click [Enable]

Enable

Rated_Current 6200 mA Driver Current Limit 19.49 Ap

Related parameter:

Object index	Data type	Name	Description	Value
2501.06	uint 16	ADC1_Buff[1]	AIN1 input original ADC data	Only read
2502.0F	int 16	Analog1_out	AIN1 valid input; analog input signal1 (AIN1) input voltage after filter, deadband and offset	
2502.01	uint 16	Analog1_Filter	Analog input signal 1 (AIN1) filter	User defined
2502.02	int 16	Analog1_Dead	Analog input signal 1 (AIN1) deadband	
2502.03	int 16	Analog1_Offset	Analog input signal 1 (AIN1) offset	
2502.0B	uint 16	Analog_Torque_Factor	AIN-Torque factor (Ap/V)	User defined 0,1
2502.08	uint 8	Analog_Torque_Con	0:Invalid 1:Ain1 Control Torque	
2502.0C	uint 16	Analog_MaxT_Factor	AIN-MaxTorque factor (Unit: Ap/V)	User defined 0, 1
2502.09	uint 8	Analog_MaxT_Con	Analog signal control Max. torque 0:Invalid 1:Ain1 Control Max.Torque	
60F6.03	uint 16	Speed_Limit_Factor	Influence max speed limit 0x60800010, if value is bigger, limit is better, but if it is too big, it will cause noise	10
6080.00	uint 16	Max_Speed_RPM	Limit motor max speed	

Note: Every time Analog1_out (2502.0F) increases or decreases by 0.1V, the corresponding theoretical value of DEC change is 285DEC (2501.06). Due to temperature drift, zero point drift, environmental electromagnetic interference, etc., the theoretical value may have certain errors in actual use.



Note

- Analog_MaxT_Con is not only used in operation mode 4. All operation modes can use analog output to limit max torque output.

For convenience, some new names are used in the formula. Definitions:

AIN1_in : AIN1 input voltage after filter and offset

Analog1_out : AIN1 valid input , which depends on the user's hardware wiring and the selection of the analog input channel. This data value is obtained after deadband and offset processing followed by filtering.

Result:

If Analog_out is not limited by Analog_Dead_High or Analog_Dead_Low:

Target torque[Nm]=Analog_out[V] *Analog_Torque_Factor [Ap/V]* torque coefficient (Nm/Ap); otherwise Target torque=0.

8.4.4 IO function related to torque mode

The IO function related to torque mode need to be configured through the operation panel or the upper computer. For the configuration process, please refer to Section 6.2.5.

Enable and reset fault

After configuring the enable signal in the digital input, the control word can be switched between the 0x06

->Din_control_word (2020.0F) setting value, thereby enabling the drive;

Reset the fault can restore the drive error.

Input function	Description
Enable	Drive enable 1: Controlword = Din_control_word (2020.0F) 0: Controlword = 0x06
Reset fault	Controlword.bit7(Fault_reset) = 1

Relevant parameter:

Object index	Name	Description	Set value
2020.0F	Din_control_word	If "Enable" function is configured to Din, the "Control word" (6040.00) will be set to this value when the corresponding Din valid input is 1	0x2F
6040.00	Controlword	0x06: Disable 0x0F: Enable 0x0B: Quick stop, load stop - voltage disconnection 0x2F→3F: Enter the absolute positioning mode 0x4F→5F: Relative positioning 0x103F: Immediate absolute positioning based on changes in target position 0x0F→1F:Homing 0x86: Error reset	User defined

Ref torque reached

This function is used to determine whether the actual torque of the motor has reached the set value.

Output function	Description
Ref torque reached	The limit value, the Current_Out_Real(60F5.08) exceeds Torque_Out_Ref(60F5.06) and the duration exceeds the Torque_Out_Filter(60F5.07),Dout port output Ref torque reached signal Setting Torque_Out_Ref (60F5.06) to 0 indicates that the torque limit detection is not enabled.

Relevant parameter:

Object index	Name	Description	Set value
60F5.06	Torque_Out_Ref	Limit limit value. The limit value, the actual torque exceeds Torque_Out_Ref and the duration exceeds the Torque_Out_Filter,Dout port output Ref torque reached signal, unit:Ap	0
60F5.07	Torque_Out_Filter	Unit: ms, Max value:32767	100
60F5.08	Current_Out_Real	Monitor the actual torque parameters, Unit: Ap	Read only

8.5 Homing mode (6)

The homing mode is used for the initial positioning of the servo system to ensure that the mechanical system can accurately find its physical origin position when it is started. This mode is usually executed automatically after the device is powered on. Through the feedback signal of the mechanical limit switch or encoder, the motor is driven to move in a predetermined direction until it reaches the mechanical origin position. This process is often called "Back to homing" and is the first step in starting the servo system to ensure that the system can operate accurately in subsequent control modes. In some applications, the system requires that each movement of the mechanical load starts from the same position, so users can use the homing mode to meet the needs.

8.5.1 Homing mode function block diagram

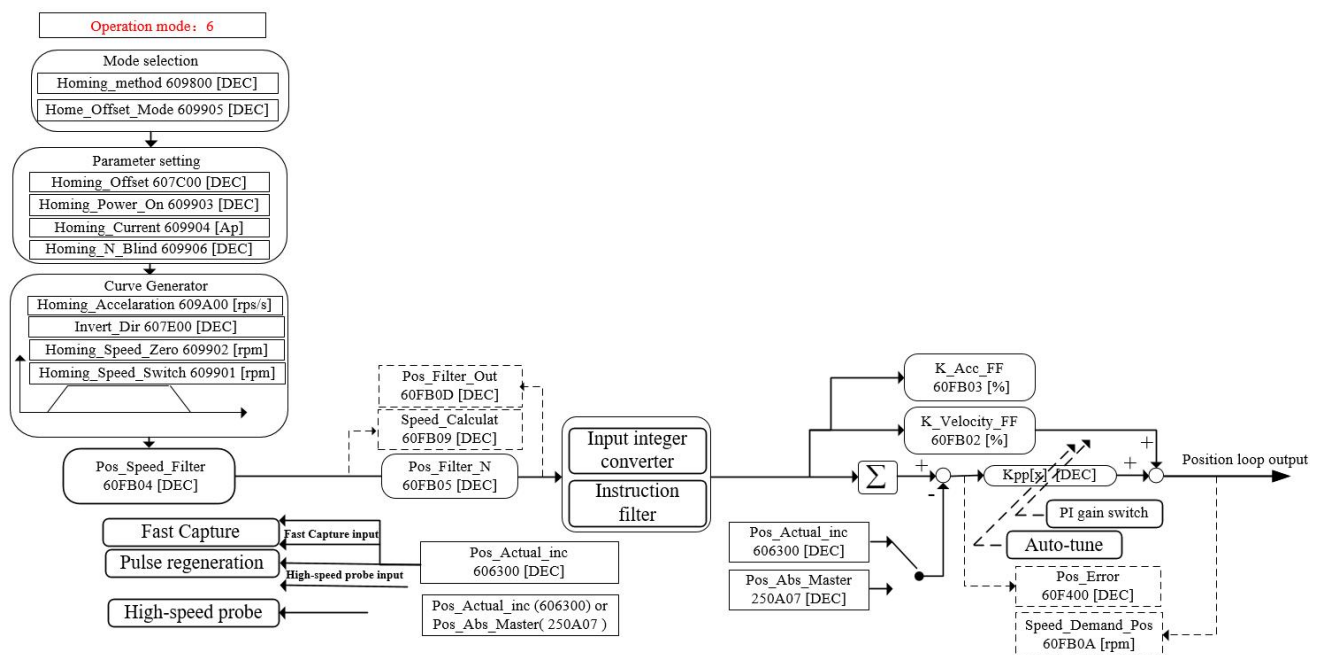


Figure 8-9 Homing mode position loop block diagram

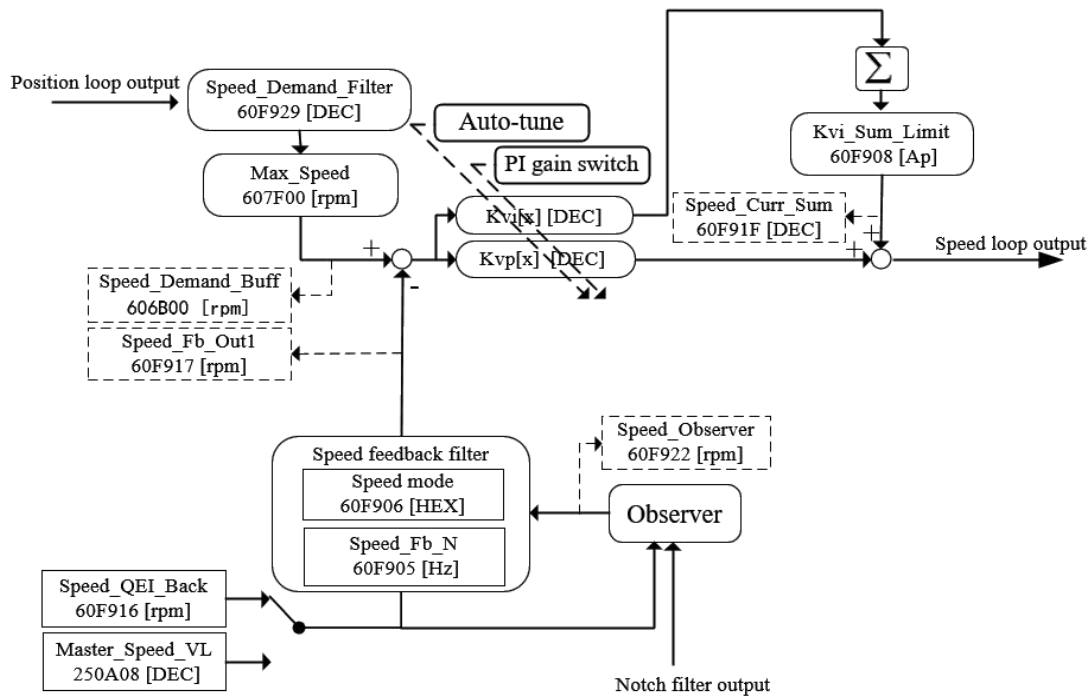


Figure 8-10 Homing mode speed loop block diagram

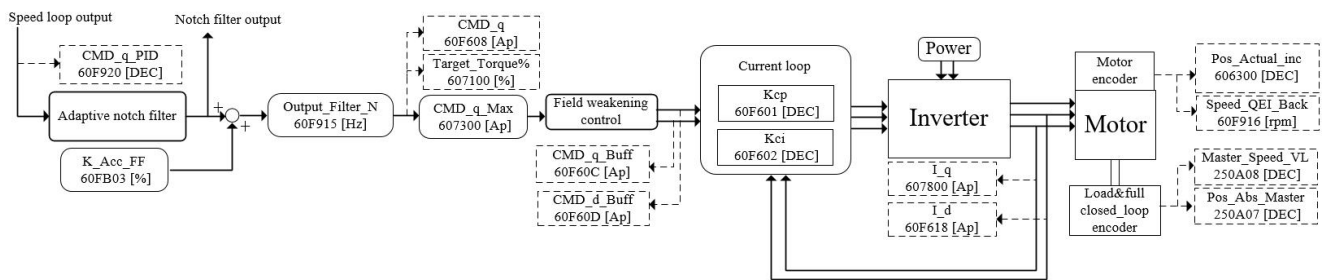


Figure 8-11 Homing mode current loop block diagram

8.5.2 Configuration process

Step 1: Select the operation mode

The operation mode can be set through the following object:

Object index	Data type	Name	Description	Effect mode
6060.00	int 8	Operation Mode	6: Set the operation mode to homing mode	Immediate

Step 2: Select the homing mode

The homing mode can be set through the following object:

Object index	Data type	Name	Description	Effect mode
6098.00	int 8	Homing Method	User defined,the homing mode table is as follow	Immediate

The detailed introduction of the origin mode is as follows:

Homing_Method	Description	Schematic
1	Homing with negative position limit switch and index pulse	
2	Homing with positive position limit switch and index pulse	
3	Homing with home switch and index pulse	
4	Homing with home switch and index pulse	
5	Homing with home switch and index pulse	
6	Homing with home switch and index pulse	

7	Homing with positive position limit switch, home switch and index pulse	Index Signal Home Signal Positive Limit
8	Homing with positive position limit switch, home switch and index pulse	Index Signal Home Signal Positive Limit
9	Homing with positive position limit switch, home switch and index pulse	Index Signal Home Signal Positive Limit
10	Homing with positive position limit switch, home switch and index pulse	Index Signal Home Signal Positive Limit
11	Homing with negative position limit switch, home switch and index pulse	Index Signal Home Signal Negative Limit
12	Homing with negative position limit switch, home switch and index pulse	Index Signal Home Signal Negative Limit

13	Homing with negative position limit switch, home switch and index pulse	Index Signal Home Signal Negative Limit
14	Homing with negative position limit switch, home switch and index pulse	Index Signal Home Signal Negative Limit
17	Homing with negative position limit switch	Negative Limit
18	Homing with positive position limit switch	Positive Limit
19	Homing with home switch	Home Signal
20	Homing with home switch	Home Signal
21	Homing with home switch	Home Signal

22	Homing with home switch	Home Signal
23	Homing with positive position limit switch and home switch	Home Signal Positive Limit
24	Homing with positive position limit switch and home switch	Home Signal Positive Limit
25	Homing with positive position limit switch and home switch	Home Signal Positive Limit
26	Homing with positive position limit switch and home switch	Home Signal Positive Limit
27	Homing with negative position limit switch and home switch	Home Signal Negative Limit
28	Homing with negative position limit switch and home switch	Home Signal Negative Limit

29	Homing with negative position limit switch and home switch	
30	Homing with negative position limit switch and home switch	
33, 34	Homing with index pulse	
35	Homing to actual position	
-17, -18	Homing via mechanical limit	

**Note**

In homing mode, the drive will limit the output current to the Homing_Current (609904). For inertial loads (like vertical axes), the Homing_Current must be set according to the actual load before running, otherwise the load might drop during homing.

Step 3: Set the Homing_Speed_Zero and Homing_Acceleration

Set the homing signal velocity and acceleration through the following parameters:

Object index	Data type	Name	Description	Effect mode
6099.02	uint 32	Homing_Power_On	Find the homing signal velocity. Unit : rpm	Immediate
609A.00	uint 32	Homing_Accelaration	Acceleration when looking for homing. Unit: rps/s	Immediate
6099.01	uint 32	Homing_Speed_Switch	Velocity for searching position_limit switch / home_switch signal Unit: rpm	Immediate

Step4: Set the Home_offset

Object index	Data type	Name	Description	Effect mode
607C.00	int 32	Home_Offset	Offset after homing. User defined	Immediate
6099.05	uint 8	Home_Offset_Mode	homing offset mode control 0: Run to the homing offset point. 1: Run to the home event happen point. The actual position will be "homing offset"	Immediate

Step5: Set the Home_N_Blind

If the homing mode needs home signal (position limit / home switch) and index signal, Home_N_Blind function can avoid the homing result being different with the same mechanics, when the Index signal is very close to the home signal. By setting to 1 before homing, the controller detects a suitable blind window for homing automatically. It can be used to assure that homing results are always the same.

During homing, the index signal inside this blind window is ignored after the home signal is found. Home_N_Blind (0:0rev;1:0.25rev;2:0.5rev) is defaulted to 0. If it's set to 1, it's changed to 0 or 2 after homing depending on the index signal position relative to the homing signal. This parameter needs to be saved. If the mechanical assembly is changed or the motor has been replaced, just set it to 1 again for initial homing.

Object index	Data type	Name	Description	Effect mode
6099.06	uint 8	Home_N_Blind	Homing index signal blind area : 0, 1	Immediate

Step6: Set the Homing_Power_On

When the Homing_Power_On(609903) is set to 1, the **Switch_On_Auto** must be set to 1 at the same time. The driver will immediately enable the motor and start finding the homing after power-on. User must fully consider safety factor before use.

Object index	Data type	Name	Description	Effect mode
2000.00	uint 8	Switch_On_Auto	Define the driver enables automatically after reboot or first time logic power on 0: Invalid 1: Drive enables automatically Note:Only can be written 1 when the "Enable" is not configured to Din	Immediate
6099.03	uint 8	Homing_Power_On	0: Default, turn off the Homing_Power_On 1: Start homing after power on or reboot and first controller enable 2: Automatically save the origin position of the multi-turn absolute encoder motor	Immediate

Step7: Set the Homing_Current

Object index	Data type	Name	Description	Effect mode
6099.04	int 16	Homing_Current	Max current for finding homing. Unit:Ap	Immediate

Step 8: Enable and run

Enable the drive and run the homing mode through the following object:

Object index	Data type	Name	Description	Effect mode
6040.00	uint 16	Controlword	Homing:0x0F->0x1F	Immediate

Step 9: Monitor relevant parameter

Statusword BIT15 can be used to determine whether the homing has been found:

Object index	Data type	Name	Description																
6041. 00	uint 16	Statusword	The status byte shows the status of the drive <table><tr><td>bit0: Ready_on</td><td>bit1: Switched_on</td></tr><tr><td>bit2: Operation_enable</td><td>bit3: Fault</td></tr><tr><td>bit4: Voltage_enable</td><td>bit5: Quick_stop</td></tr><tr><td>bit6: Switchon_disabled</td><td>bit7: Warning</td></tr><tr><td>bit8: Internal retention</td><td>bit9: Remote</td></tr><tr><td>bit10: Target_reached</td><td>bit11: Internal limit activation</td></tr><tr><td>bit12: Pulse response</td><td>bit13: Following_Error</td></tr><tr><td>bit14: Commutation_Found</td><td>bit15: Reference_Found</td></tr></table>	bit0: Ready_on	bit1: Switched_on	bit2: Operation_enable	bit3: Fault	bit4: Voltage_enable	bit5: Quick_stop	bit6: Switchon_disabled	bit7: Warning	bit8: Internal retention	bit9: Remote	bit10: Target_reached	bit11: Internal limit activation	bit12: Pulse response	bit13: Following_Error	bit14: Commutation_Found	bit15: Reference_Found
bit0: Ready_on	bit1: Switched_on																		
bit2: Operation_enable	bit3: Fault																		
bit4: Voltage_enable	bit5: Quick_stop																		
bit6: Switchon_disabled	bit7: Warning																		
bit8: Internal retention	bit9: Remote																		
bit10: Target_reached	bit11: Internal limit activation																		
bit12: Pulse response	bit13: Following_Error																		
bit14: Commutation_Found	bit15: Reference_Found																		

8.5.3 KS PRO operation

The homing mode is usually used to determine and set the initial position of the servo motor. The main purpose of this mode is to rotate the motor to a predetermined reference position (i.e., the homing position) through the drive system, and then use this position as the reference point for motor movement. The specific operation process is as follows:

- Set the homing mode, and check whether the current homing mode meets expectations in the homing mode description. User can also directly set the homing mode in the preset homing method. The legend and parameters in the homing mode description will change accordingly.
- If the function meets expectation, click **[Write]** to send the preset homing mode to the drive.

Homing Mode 1. Set Homing mode

Encoder Index signal: ☒ Yes ☐ No

Trigger signal: Limit & Origin

Position Limit Signal: ☒ Negative ☐ Positive

Relative Home Signal Pol: ☒ Negative side ☐ Positive side

Homing Method: ☒ Leave ☐ Contact

Preset Homing: 14 Actual Homing: 0 **Write**

Homing parameters

Homing Speed Switch: 300 rpm Homing Acceleration: 50 rpm/s

Homing Speed Zero: 100 rpm Homing Current: 2.46 A

Offset mode: run to the homing offset position Home Offset: 0 inc

Home N_Blind: 0 Rev

☐ start searching homing when power on and the first enable driver

☐ auto save homing information

Homing Method Explanations 2. View current Homing method explanation

Methods 7 to 14: Homing on the home switch and index pulse

These methods use a home switch which is active over only portion of the travel, in effect the switch has a 'momentary' action as the axle's position sweeps past the switch.

Using methods 7 to 10 the initial direction of movement is to the right, and using methods 11 to 14 the initial direction of movement is to the left except if the home switch is active at the start of the motion. In this case the initial direction of motion is dependent on the edge being sought. The home position is at the index pulse on either side of the rising or falling edges of the home switch, as shown in the following two diagrams. If the initial direction of movement leads away from the home switch, the drive must reverse on encountering the relevant limit switch.

Pos Actual 0 inc Homing Status: * Not yet started **Enable** **Start homing**

- Set the homing parameters. The descriptions of each parameter are as follows:

Parameter name	Description
Homing_Speed_Switch	Search for the initial velocity of the homing signal
Homing_Speed_Zero	After the homing signal or limit signal is triggered, look for the speed of the encoder index signal
Homing Acceleration	Acceleration when looking for homing
Homing_Current	Max current for finding homing
Offset mode	1.Run to the homing offset point 2.Run to the homing event homing point
Home_Offset	Offset after homing
Home_N_Blind	If the homing mode needs home signal (position limit / home switch) and index signal, Home_N_Blind function can avoid the homing result being different with the same mechanics. During the return period, after the homing signal is found, the index signal within this blind window will be ignored.
Start searching homing when power on and the first enable driver	Each time the power is restored, the function of finding the homing is executed once
Auto save homing information	Start homing after power on or reboot and first controller enable Automatically save the origin position of the multi-turn absolute encoder motor, it will not be lost after power failure

- First click [Enable], and then click [Start homing].

Homing Method Explanations

Methods 3 and 4: Homing on the positive home switch and index pulse
Using methods 3 or 4 the initial direction of movement is dependent on the state of the home switch. The home position is at the index pulse to either to the left or the right of the point where the home switch changes state. If the initial position is sited so that the direction of movement must reverse during homing, the point at which the reversal takes place is anywhere after a change of state of the home switch.

Pos_Actual: 0 inc Homing Status: * Not yet started

1.Click [Enable] 2.Click [Start homing]

[Enable] [Start homing]

- If the homing is successfully found, the homing status will display "Homed".

Homing Status: ● Homed

8.5.4 IO function related to homing mode

The IO function related to homing mode need to be configured through the operation panel or the upper computer. For the configuration process, please refer to Section 6.2.5.

Enable and reset fault

After configuring the enable signal in the digital input, the control word can be switched between the 0x06

->Din_control_word (20200F) setting value, thereby enabling the drive;

Reset the fault can restore the drive error.

Input function	Description
Enable	Drive enable 1: Controlword = Din_control_word (20200F) 0: Controlword = 0x06
Reset fault	Controlword.bit7(Fault_reset) = 1

Relevant parameter:

Object index	Name	Description	Set value
2020.0F	Din_control_word	If "Enable" function is configured to Din, the "Control word" (6040.00) will be set to this value when the corresponding Din valid input is 1	0x2F
6040.00	Controlword	0x06: Disable 0x0F: Enable 0x0B: Quick stop, load stop - voltage disconnection 0x2F→3F: Enter the absolute positioning mode 0x4F→5F: Relative positioning 0x103F: Immediate absolute positioning based on changes in target position 0x0F→1F:Homing 0x86: Error reset	User defined

Start homing

When the digital input is configured with this function, the drive starts to find the homing after activating the valid input.

Input function	Description
Start homing	Start searching for the homing signal. This operation must be performed under the condition that the driver is enabled. Once the homing is found, the operation mode will be switched back to the one before searching for the homing.

Encoder Index

When the motor index (Z phase) signal appears, the digital output configured for this function is valid.

Output function	Description
Encoder Index	The motor index signal appear

Home Found

This function is used to determine whether the homing has been found.

Output function	Description
Home found	Homing have been found

8.6 Pulse mode (-4)

Pulse mode is typically used in applications requiring precise position control and high-speed response. In this mode, the servo drive controls the position of the motor based on the input pulse signal. Usually each pulse corresponds to a fixed angle or distance of the motor rotation. The pulse frequency determines the motor rotation speed, and the number of pulses determines the motor movement distance. This mode is widely used in CNC machine tools, automation equipment and other systems that require precise positioning.

8.6.1 Pulse mode function block diagram

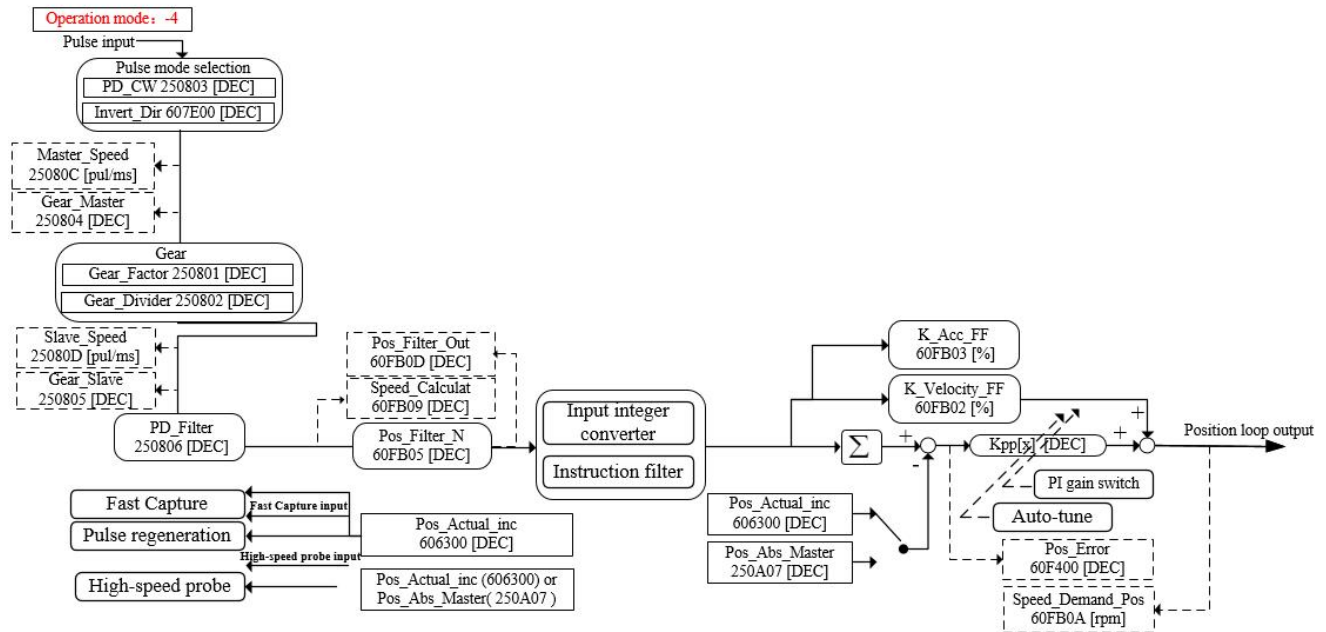


Figure 8-12 Pulse mode position loop block diagram

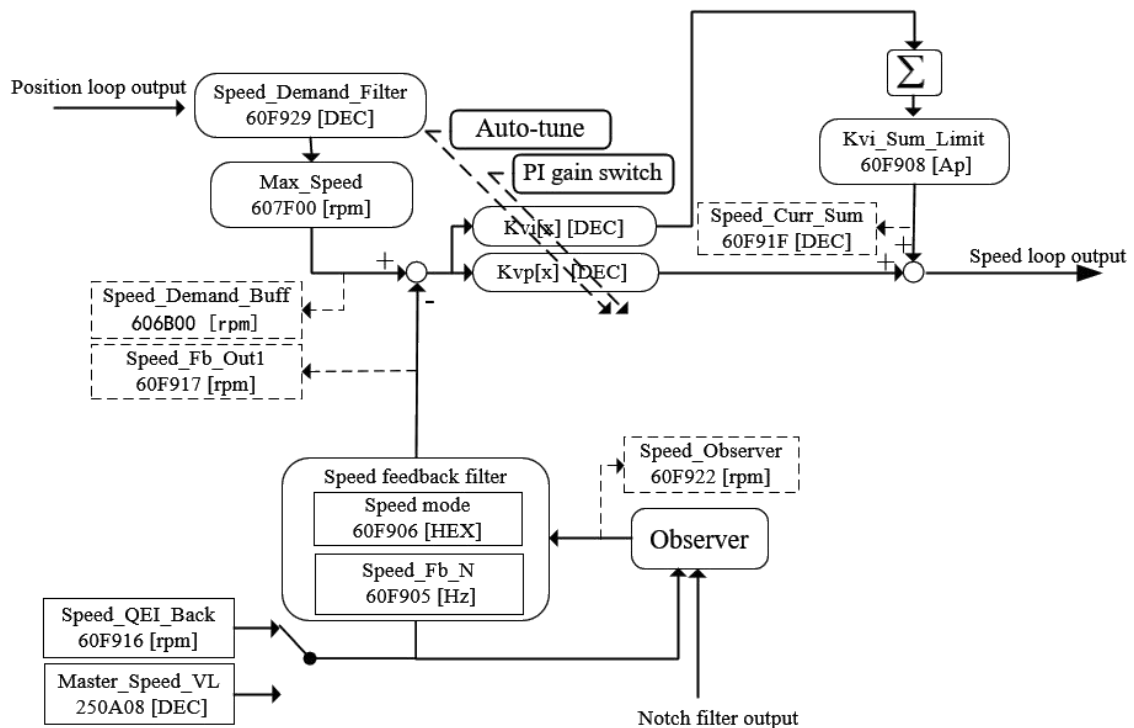


Figure 8-13 Pulse mode speed loop block diagram

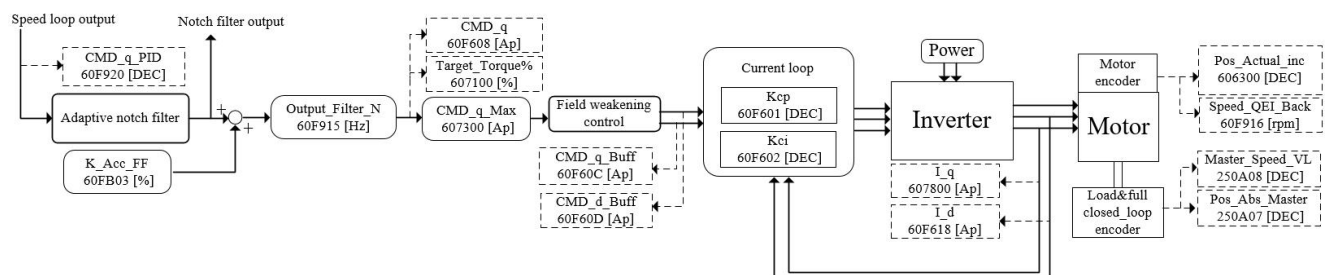


Figure 8-14 Pulse mode current loop block diagram

8.6.2 Configuration process and panel operation

Step 1: Select the operation mode

The operation mode can be set through the following object:

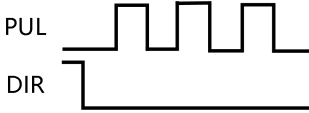
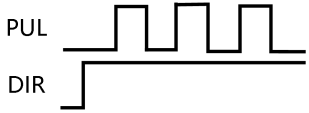
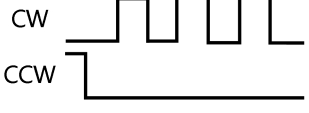
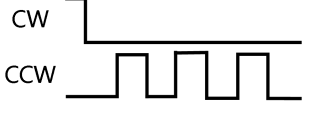
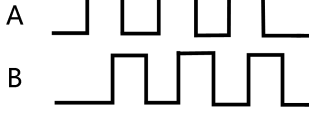
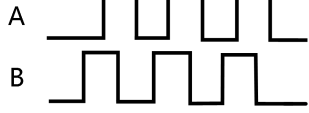
Object index	Data type	Name	Description	Effect mode
6060.00	int 8	Operation_Mode	-4: Set the pulse mode	Immediate

Step2: Confirm pulse mode

Select the corresponding pulse mode according to the type of externally input pulse signal. After the pulse mode is modified, it needs to be stored and restarted to take effect.

Object index	Data type	Name	Description	Effect mode
2508.03	uint 8	PD_CW	Pulse train mode 0: CW / CCW 1: Pulse / direction 2: A / B (incremental encoder)	Save&Reboot

The pulse inputs supported by the drive are as follows:

Pulse mode	Forward	Reverse
P/D		
CW/CCW		
A/B		

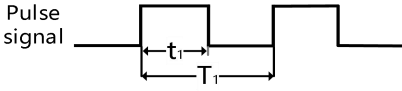
➔

Note

Forward rotation means positive position counting, the default is CCW direction, you can set the Invert_Dir (607E.00) = 1, change the rotation direction of the motor shaft

Step3: Confirm CPLD Pulse Filter Setting

The CPLD filter inside the drive is used to filter the high-frequency clutter at the pulse input port to prevent pulse interference. This filter is only valid for pulsed signals with a 50% duty cycle. It should be noted that the filter frequency must be greater than the pulse frequency before the gear, otherwise the effective pulse signal will be filtered.

50% Pulse duty cycle	$50\% \text{ duty cycle} = (t_1/T_1) \times 100\%$ 
----------------------	---

The pulse filter properties can be configured via the following parameters:

Object index	Data type	Name	Description	Effect mode
2508.06	uint 16	PD_Filter	Pulse filter (Unit:ms) , user defined	Immediate
2508.08	uint 16	Frequency_Limit	Frequency limit (inc / ms), if pulse count (in 1 ms) is greater than Frequency_Limit, overfrequency error occurs.	Immediate
3046.00	uint 8	CPLD_Filter	CPLD internal filter is used for pulse input port, only for pulse signal with 50% duty cycle, the filter frequency is: 0: 4MHz 1: 3.2MHz 2: 2MHz 3: 1.6MHz 4: 500KHz(Default) 5: 330KHz 6: 250KHz 7: 125KHz Note: The filter frequency must be higher than the Master_Speed	Immediate

Step4: Confirm the electronic gear ratio

There are two ways to set the electronic gear ratio in pulse mode. You only need to select one of them:

1.Electronic gear numerator/ Electronic gear denominator=65536/number of external input pulse

Example:

If the number of external input pulse is 10000 to make the motor rotate once, the electronic gear ratio should be 65536:10000, the numerator of the electronic gear should be set to 4096, and the denominator of the electronic gear should be set to 625.

Object index	Data type	Name	Description	Effect mode
2508.01	int 16	Gear_Factor[0]	Gear_ratio=Gear_Factor/Gear_Divider	Immediate
2508.02	uint 16	Gear_Divider[0]		

2.Set the number of command pulse per revolution of the motor

Example:

After the PulseCMD_Unit is set to 1, set the number of pulse in a single revolution of the motor to 10000, then when the number of external input pulse is 10000, the motor will rotate once.

Object index	Data type	Name	Description	Effect mode
2508.11	uint 32	Pulse/Rev	Number of position pulse commands required for one revolution of the motor shaft	Immediate
2508.12	uint 32	PulseCMD_Unit	Pulse mode Command unit: 0(encoder unit): 0:Position unit:	Save&Reboot

			inc; speed unit : (inc/0.25ms)*65536; acceleration unit:(inc/0.25ms/ms)*65536; 1:Position unit: pulse ; speed unit: pulse/s; acceleration unit: pulse/s²	
--	--	--	---	--

Step 5: Basic parameter settings before operation

After setting the parameters of the first three steps, you can check whether there is data in the Gear_Master, and whether the Master_Speed is consistent with the pulse frequency sent by the host device. After confirmation, the operation mode and the control word can be set to run the motor. The running direction of the motor can be adjusted by modifying the direction control.

Step 6: Enable and run

Enable the drive and run the pulse mode through the following object:

Object index	Data type	Name	Description	Effect mode
6040.00	uint 16	Controlword	0x0F : Enabled	Immediate

Step 7: Parameter Monitoring

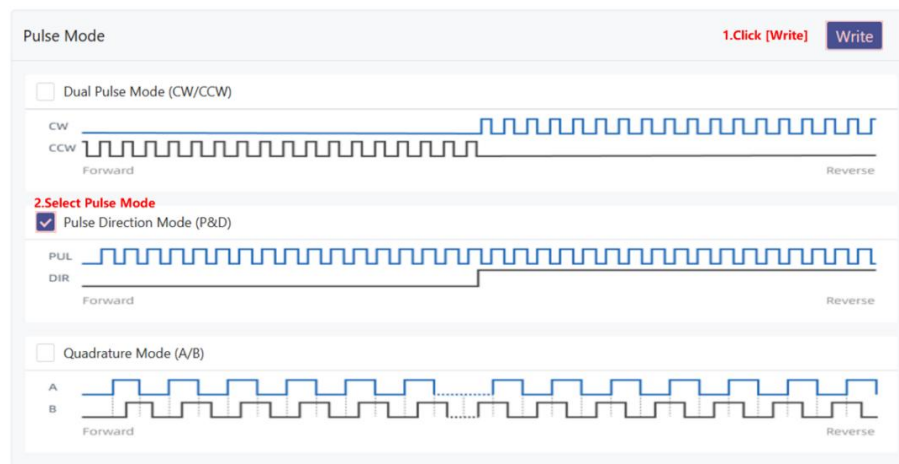
The following parameters can be monitored to check whether the pulse mode is operating normally:

Object index	Data type	Name	Description
2508.04	int 32	Gear_Master	Master_encoder pulse input counting without gear ratio
2508.05	int 32	Gear_Slave	Master_encoder pulse input counting with gear ratio
2508.0C	int 16	Master_Speed	Master input pulse speed without gear ratio(pulse/ms)
2508.0D	int 16	Slave_Speed	Master input pulse speed with gear ratio (pulse/ms)
60FC.00	int 32	Pos_Demand	Inner target position
6063.00	int 32	Pos_Actual_inc	The actual position of the motor
606B.00	int 32	Speed_Demand_Buff	Planing speed value
2FF0.14	int 32	Real_Speed_RPM	The speed value of the actual speed after filtering

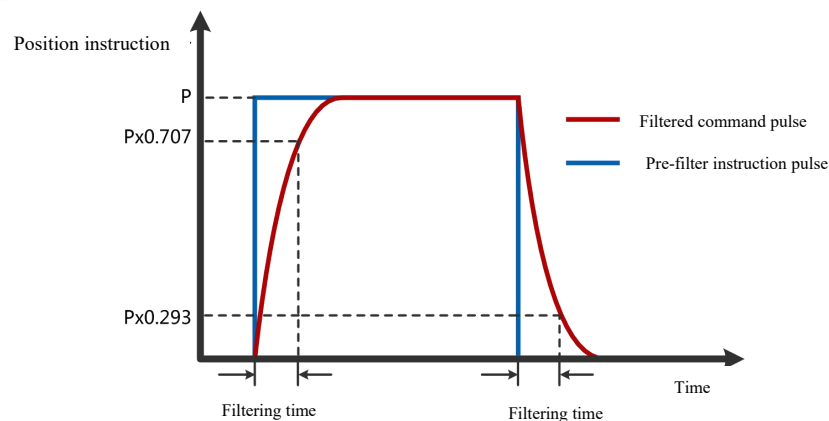
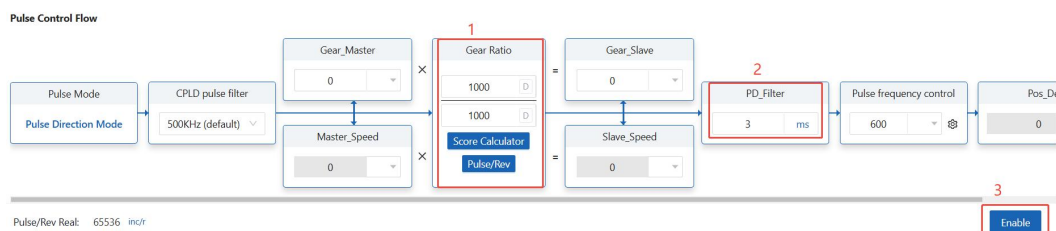
8.6.3 KS PRO operation

In the pulse control, the target speed command is determined by the external pulse frequency and electronic gear ratio. The target position increment is the number of received pulses multiplied by the electronic gear ratio.

- Select the pulse mode;
- Click **[Write]**, after the device restarts automatically, the pulse mode takes effect.



- Write the **Gear Ratio**, and then click the [Score Calculator] button below. **Slave_Speed** (target speed) = **Master_Speed** * **Gear Ratio**, **Gear_Slave** (target position increment) = **Gear_Master** * **Gear Ratio**. Or keep the Gear Ratio at the default value and switch the **PulseCMD_Unit** (2508.12) to pulse; in the disabled state, write the value to the **Pulse/Rev** (2508.11), then the number of pulses received by the drive (unit: inc) = PLC pulse output number (unit: pulse) * Gear Ratio (1inc = Gear ratio * 1Pulse), Gear ratio motor rotation number / (Pulse/Rev);
- Write the **PD_Filter** to smooth and filter the received input pulse signal to ensure that the system only responds to valid pulses and avoid erroneous responses caused by noise or interference;
- Click [**Enable**], and the servo motor will start responding to the pulse signal.



Other parameter description:

Name	Description
Gear_Master	Master_encoder pulse input counting without gear ratio
Gear_Slave	Master_encoder pulse input counting with gear ratio
Master_Speed	Master input pulse speed without gear ratio
Slave_Speed	Master input pulse speed with gear ratio

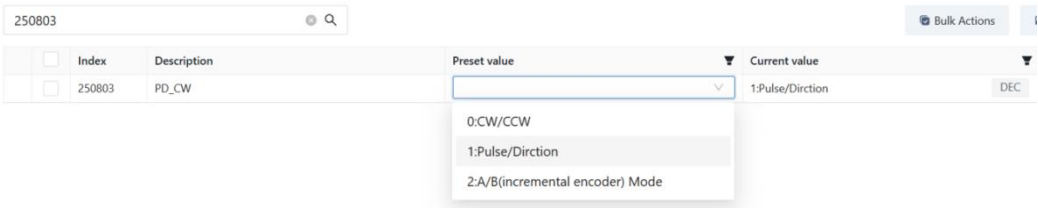
8.6.4 Encoder pulse output

The slave drive runs synchronously following the instructions of the master drive. The pulse input of the slave drive comes from an external encoder signal or the encoder output of the master drive. The encoder output signal resolution of the main drive is specified by the encoder division resolution.

Operational process:

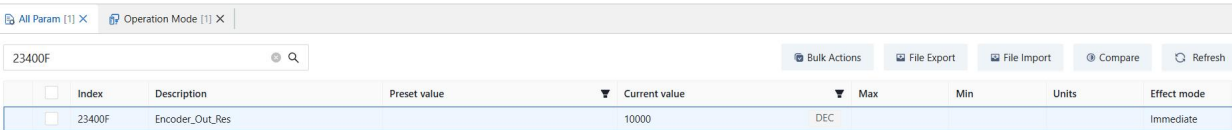
Search for the PD_CW (250803) in the All Param of the upper computer, set the pulse mode to 2, incremental encoder mode:

Object index	Data type	Name	Description	Effect mode
2508.03	uint 8	PD_CW	2:A/B(incremental encoder) Mode	Save&Reboot



Search for the Encoder_Out_Res (23400F) in the All Param of the upper computer, set the number of output pulses of the drive

Object index	Data type	Name	Description	Effect mode
2340.0F	int 32	Encoder_Out_Res	It can be used to set the number of encoder pulses output by the driver after the motor has completed one revolution. User defined	Immediate



The connection between the master and slave is shown in the following figure:

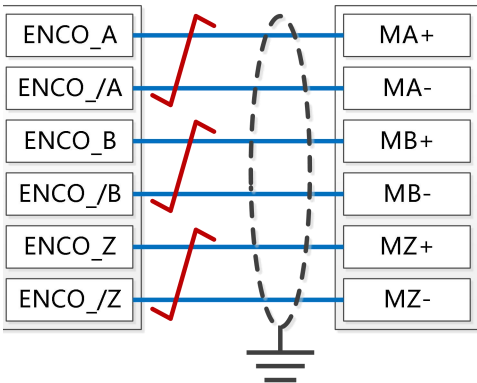


Figure 8-15 Encoder cable connection

8.6.5 IO function related to pulse mode

The IO function related to pulse mode need to be configured through the operation panel or the upper computer. For the configuration process, please refer to Section 6.2.5.

Enable and reset fault

After configuring the enable signal in the digital input, the control word can be switched between the 0x06 -> Din_control_word (20200F) setting value, thereby enabling the drive;

Reset the fault can restore the drive error.

Input function	Description
Enable	Drive enable 1: Controlword = Din_control_word (20200F) 0: Controlword = 0x06
Reset fault	Controlword.bit7(Fault_reset) = 1

Relevant parameter:

Object index	Name	Description	Set value
2020.0F	Din_control_word	If "Enable" function is configured to Din, the "Control word" (6040.00) will be set to this value when the corresponding Din valid input is 1	0x2F
6040.00	Controlword	0x06: Disable 0x0F: Enable 0x0B: Quick stop, load stop - voltage disconnection 0x2F→3F: Enter the absolute positioning mode 0x4F→5F: Relative positioning 0x103F: Immediate absolute positioning based on changes in target position 0x0F→1F:Homing 0x86: Error reset	User defined

Reset Count

Clear the Gear_Master through the following functions.

Input function	Description
Reset Count	Clear the Gear_Master that the drive has received but not executed

Chapter 9 Servo application function

9.1 Absolute value encoder motor configuration

9.1.1 Multi-turn motor power-on configuration

When the absolute encoder motor is connected to the drive for the first time, the drive will appear 000.4 encoder UVW fault or communication fault alarm, which can be cleared by resetting the status marker of the encoder.

Object index	Data type	Name	Description	Effect mode
2690.00	uint 8	Encoder_Data_Reset	Communication encoder command Write: 1: clear encoder status 2: Read the fault word, the fault status can be viewed at 0x305101 3: Clear encoder status and multi-turn data 8 Reset the status flag of the encoder 9 Clear multi-lap data 10 Reset encoder status flag and clear multi-turn data operation 12 Clear the single-turn data (requires the manufacturer's password) Read: bit 0: 1 = Command can be entered bit 1: 1 = Last command execution completed Note: Clearing the single-turn data will cause the motor phase error. It is forbidden to clear the single-turn data when the motor can run normally!	Immediate

The software operation is as follows:

- Search for the "Encoder_Data_Reset" in the all param and set the corresponding value

269000

Index

Description

269000

Encoder_Data_Reset

▼

▼

▼

▼

8:reset flags

9:clear MT data

10:like 8 + 9

12:clear ST data(password required)

0unfinished

DEC

➔

Note

- After the communication encoder data is reset, the drive alarm will not be cleared directly. It is necessary to clear the drive alarm through the control word or restart the drive.
- After clearing the multi-turn fault, if the connection between the encoder wire and the motor end is disconnected, the drive will have fault 000.4 again, and the encoder state needs to be reset.
- After clearing the multi-turn data through the communication encoder data reset, the actual position single-turn data will not be cleared at this time. To reset the actual position, you need to use the 35 homing mode to return to the origin.

9.1.2 Multi-turn data range

QAK/QBK absolute encoder motor single-turn resolution 131072 (17 bit), multi-turn number 65536, the actual position value after the positive multi-turn data overflow changes from 2147418112 to -2147418112, the negative multi-turn data overflow after the actual position value Changed from -2147483648 to 2147483648, the motor rotation direction remains unchanged after overflow.

NAK/NBK absolute encoder motor single-turn resolution 8388608 (23bit), multi-turn number 4096, the actual position value after the positive multi-turn data overflow changes from 134217727 to -134217728, the negative multi-turn data overflow after the actual position value Changed from -134217728 to 134217727, the motor rotation direction remains unchanged after overflow.

9.1.3 Multi-turn prohibition

Set Encoder_MT_Disable (0x23400E) to 1, absolute encoder can be used as incremental encoder without connecting battery. However, it should be noted that when the multi-turn prohibition is turned on, the position of the encoder cannot be saved after the power is turned off.

Object index	Data type	Name	Description	Effect mode
2340.0E	uint 8	Encoder_MT_Disable	0: Default 1: Multi-turn encoder used for single-turn encoder	Immediate

The software operation is as follows:

- Search for the "Encoder_MT_Disable" or "23400E" in the all param and set the corresponding value

23400E			Bulk Actions	
☐ Index	Description	Preset value	Current value	
☐ 23400E	Encoder_MT_Disable		0:default	DEC

The encoder status can be viewed based on the following information:

Object index	Data type	Name	Description	Effect mode
2680.00	uint 16	Warning status word	Encoder alarm status word bit 0: battery alarm bit 1: Hybrid alarm (overspeed, overtemperature) bit 2: Encoder busy (motor speed too high at power-on)	Immediate
3051.02	int 16	Encoder warning message	Help:encoder warning messages Nikon internal error of encoder Bit0: BATT Bit1: MTERR Bit2: 0 Bit3: OVSPD Bit4: MEMERR Bit5: STERR Bit6: PSERR Bit7: BUSY Bit8: MEMBUSY Bit9: OVTEMP	Immediate

The software operation is as follows:

- Search for the "Enc_ALM (305102)" and "Warning_Word (268000)" in the all param and set the corresponding value

268000					Bulk Actions	
	Index	Description	Preset value	Current value		
[-]	268000	Warning_Word	Write Prohibition	0	HEX	
	bit0	Battery Warning		0:0		
	bit1	Mixed Warning		0:0		
	bit2	Encoder Busy		0:0		

305102					Bulk Actions	
	Index	Description	Preset value	Current value		
[-]	305102	Enc_ALM	Write Prohibition	0	HEX	
	bit0	BATT		0:0		
	bit1	MTERR		0:0		
	bit2	0		0:0		
	bit3	OVSPD		0:0		
	bit4	MEMERR		0:0		
	bit5	STERR		0:0		
	bit6	PSERR		0:0		
	bit7	BUSY		0:0		
	bit8	MEMBUSY		0:0		
	bit9	OVTEMP		0:0		

9.1.4 Battery description

When the battery voltage of the multi-turn absolute value encoder is below 3.1V, the drive appears the warning status word 0001, but the drive does not alarm for shutdown at this time.

When the battery voltage of the multi-turn absolute value encoder is lower than 2.5V, the encoder UVW fault or internal fault (fault code 000.4) occurs in the drive. At this time, the motor stops running, the data is lost in multiple turns, and the absolute encoder needs to be re-initialized after the encoder cable is replaced.



Note

- The battery must be used within a limited temperature range, away from high temperature and high humidity environments, and without dust or inflammable and explosive substances.
- During transportation and assembly, be careful not to impact and squeeze the battery, so as not to damage the battery.
- Before using a battery that has been placed for a long time, check whether the battery is swollen or not.
- Do not short-circuit the positive and negative poles of the battery, otherwise the battery may burst.
- The battery is a one-time item, do not charge the battery.
- Please dispose of batteries according to local regulations.

The battery specifications are as follows:

Battery name	Lithium thionyl chloride battery
Standard Voltage	3.6V
Standard capacity	2700mH
Maximum continuous discharge current	60mA
Maximum pulse capacity	150mA
Operation temperature	-55°C~85°C
Storage temperature	-55°C~115°C

9.2 Limit function

In order to ensure that the motor runs within the range allowed by the mechanical structure, the motor can be prevented from running beyond the range by inputting a limit signal.

9.2.1 Limit function configuration

1. Digital input limit digital

By installing the sensor in the machine and connecting the external sensor signal to the digital input signal port of the drive, when the sensor level changes, the positive and negative limit signals take effect, and the motor stops. For safety reasons, the positive/negative limit signals are normally closed. When the valid input is 1 (the valid input is green), the limit function is invalid. When the valid input is 0 (the valid input is gray), it means that the limit is activated and the corresponding movement direction is prohibited.

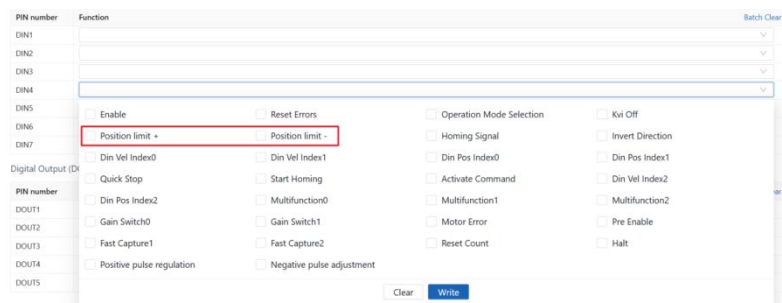
Step 1: Configure digital input

Position limit+ and Position limit- function need to be configured on the digital input port:

Object index	Data type	Name	Description
2010.06	uint 16	Din4_Function	Define the DIN4 function (Default 0010 Position limit+)
2010.07	uint 16	Din5_Function	Define the DIN5 function (Default 0020 Position limit-)

The software operation is as follows:

- In the **DIO Setting**, configure the Position limit+ and Position limit-



Step 2: Activate the digital input of the external sensor

When the corresponding limit digital input port is activated, the actual position when the positive limit and negative limit are triggered will be recorded.

Object index	Data type	Name	Description
2010.1B	int 32	Pos_L_Pos	The actual position when the positive limit of the digital input signal is triggered
2010.1C	int 32	Pos_L_Neg	The actual position when the negative limit of the digital input signal is triggered

The software operation is as follows:

- Search for the " Pos_L_Pos" (20101B) or "Pos_L_Neg"(20101C) in the all param and set the corresponding value

20101B		Bulk Actions		File Export		File Import		Compare		Refresh	
Index	Description	Preset value	Current value	Max	Min	Units	Effect mode				
20101B	Pos_L_Pos	Preview only	-				Immediate				

20101C



Bulk Actions

File Export

File Import

Compare

Refresh

☐	Index	Description	Preset value	Current value	Max	Min	Units	Effect mode
☐	20101C	Pos_L_Neg	Preview only	-				Immediate

Step 3: Limit function definition

This function defines whether the driver will alarm after the limit occurs .

Object index	Data type	Name	Description	Effect mode
2010.19	uint 8	Limit_Function	Define creating error or not after position limit 0: Will be error if position limit happen after home found 1: do nothing	Immediate

The software operation is as follows:

- Search for the "Limit_Function" (201019) in the all param and set the corresponding value

☐	Index	Description	Preset value	Current value
☐	201019	Limit_Function	▼ 0:error 1:do nothing	1:do nothing DEC

- If Limit_Function= 0, after the drive is enabled and the position limit+ and position limit- are activated, the driver will report an error

The screenshot displays the Motor Control GUI with two error state tables. At the top, there are controls for 'Position limit +' and 'Position limit -'. The 'Error_State1' table on the left shows a red header and a red value '000.1'. The 'Error_State2' table on the right shows a red header and a red value '020.0'. Both tables list error codes, status indicators (red or green circles), and descriptions.

Number	Status	Error description
0001	Red	Extended Error
0002	Green	Encoder ABZ/not connected
0004	Green	Encoder UVW/ Encoder internal
0008	Green	Encoder Counting / Encoder CRC
0010	Green	Driver temperature
0020	Green	Over voltage
0040	Green	Under Voltage
0080	Green	Over current
0100	Green	Chop Resistor
0200	Green	Position Following
0400	Green	Logic voltage
0800	Green	Motor or Driver Ilt
1000	Green	Over Frequency
2000	Green	Motor Temperature
4000	Green	Motor Commutation/ Encoder information
8000	Green	EPROM data

Number	Status	Error description
0001	Green	Current sensor
0002	Green	Watch dog
0004	Green	Wrong interrupt
0008	Green	MCU Fault
0010	Green	Motor Configuration
0020	Green	Logic output
0040	Green	STO1
0080	Green	STO2
0100	Green	Extended enable
0200	Red	Positive Limit
0400	Green	Negative Limit
0800	Green	SPI internal
1000	Green	Abort Connection
2000	Green	Closed Loop Direction
4000	Green	Master ABZ
8000	Green	Master counting

2. Soft limit position

Step 1: Set the limit function parameter

To prevent overtravel by setting the position of the software limit, it should be noted that the position of the software limit only takes effect after homing. When the actual position reaches the set software limit position, the motor stops.

Object index	Data type	Name	Description	Effect mode
607D.01	int 32	Soft Positive Limit	Software limit positive setting	Immediate

607D.02	int 32	Soft_Negative_Limit	Software limit negative setting negative data must be smaller than positive data. Otherwise it will not work.	Immediate
6085.00	uint 32	Quick_Stop_Dec	Stop deceleration after the limit signal takes effect	Immediate

The software operation is as follows:

- Search for the "Soft_Positive_Limit" (607D01) and "Soft_Negative_Limit" (607D02) in the all param and set the corresponding value

607D01

Bulk Actions

☐	Index	Description	Preset value		Current value	
☐	607D01	Soft_Positive_Limit	100000	DEC	100000	DEC

607D02

Bulk Actions

☐	Index	Description	Preset value		Current value	
☐	607D02	Soft_Negative_Limit	-1000000	DEC	-1000000	DEC

- Search for the "Quick_Stop_Dec" (608500) in the all param and set the corresponding value

608500

Bulk Actions

☐	Index	Description	Preset value	Current value
☐	608500	Quick_Stop_Dec		610

- When the actual position reaches the soft limit value, the motor stops running at a rapid deceleration

9.2.2 IO function about limit

1. Positive limit and negative limit

The positive and negative limit function digital input port is usually connected to an external sensor. It will be activated when the limit position is reached.

Input function	Description
Positive limit	Normally closed type positive/negative limit switch signal input 0: limit is activated, the corresponding direction have been banned
Negative limit	

2. Limit position reach

This function indicates that the current motor position has reached the limit position and the drive is in the limit state.

Input function	Description
Limit position reach	Positive/ negative position limit in progress

9.3 Motor brake control

In order to ensure that the motor will not rotate under the influence of gravity or other external forces after the power is turned off, the load equipment needs to use a brake motor.

9.3.1 Brake control configuration

Step 1: Set the Addition_Device

If the motor is equipped with a brake, the motor accessory should be written as 1. After the drive is enabled, it is necessary to wait for the brake to fully open before the motor starts running.

Object index	Data type	Name	Description	Effect mode
6410.17	uint 8	Addition_Device	Whether the motor has a brake accessory 0: Motor without brake 1: The motor with brake, and the drive needs to wait for a period of time (delay time) to fully open/close the brake	Immediate

The software operation is as follows:

- Search for the "Addition_Device" (641017) in the all param and set the corresponding value

641017			Bulk Actions	
Index	Description	Preset value	Current value	
+	641017	Addition_Device	1	HEX 1 HEX

Step 2: Set the Brake_Duty_Cycle and Brake_Delay

After the control word is written into the enable command, the servo motor is energized to lock the shaft, and the driver outputs 24V DC to the brake after the relay pull-in delay for a certain period of time (brake delay) to make the brake open as soon as possible. After the delay, the speed command takes effect and the motor starts to run.

After the brake is opened, the brake power supply is reduced to $24V \times \text{brake duty cycle}$ to reduce the brake power consumption. If the brake duty cycle is too low, it may cause the brake to close.

Object index	Data type	Name	Description
6410.11	uint 16	Brake_Duty_Cycle	Duty cycle of the lock signal. By changing the on-time (pulse width) and switching frequency of the switching device, the output voltage or current is controlled to reduce the temperature rise. The duty cycle of the lock is adjustable from 0 to 100%, corresponding to the average chopper voltage from 0 to 24V. The default value is 70%, that is, the lock supply voltage is $24 \times 70\% = 16.8V$
6410.12	uint 16	Brake_Delay	Brake signal delay time. The default value is 150ms



Note

- It is not recommended that the duty cycle of the brake be less than 40% or higher than 90%. Otherwise, the brake cannot be opened or the lock heats up.

The software operation is as follows:

- Search for the "Brake_Duty_Cycle" (641011) and "Brake_Delay" (641012) in the all param and set the corresponding value.

☐	Index	Description	Preset value	Current value	
☐	641011	Brake_Duty_Cycle		89.96	DEC

☐	Index	Description	Preset value	Current value	
☐	641012	Brake_Delay		150	DEC

- The operations of setting the Addition_Device, Brake_Duty_Cycle, and Brake_Delay can also be completed in the motor configuration interface. Usually, the KINCO motor configuration can be accomplished by reading the encoder information or by configuring the motor library. Third-party motors can be configured through the brake settings in the custom motor library. After the configuration is completed, click "Apply", and it will take effect after a restart.

Function Setting

Commutation Settings:

* Commu_Mode
 defined by manufacturer

* Commu_Curr
 4.11

* Commu_Delay
 1000 ms

Brake setting:

* Brake_Delay
 150 ms

* Addition_Device
 1

* Brake_Duty_Cycle
 89.96 %

Alarm Setting:

* Temp_Motor_Ref
 8192

* Motor_Ilt_Filter
 76.03 S

Default Direction Setting:

* Invert_Dir_Motor
 CCW(default setting)

Motor_Ilt_Real % Temp_Motor

Step 3: Configure the quick stop mode

Object index	Data type	Name	Description	Effect mode
605A.00	int 16	Quick_Stop_Mode	0: Uncontrolled stop 1: Curve stop 2: Quick stop deceleration stop 5: The curve stops, and finally stops in the rapid stop state 6: Rapid deceleration stop, and finally stop in rapid stop state 18: Use the motor's own resistance to brake, even if there is a problem with the encoder Factory default is uncontrolled stop mode	Immediate
6085.00	uint 32	Quick_Stop_Dec	Deceleration when quick stop mode is 2	Immediate

After the control word is written into the enable command, the servo motor is energized to lock the shaft, and the driver outputs 24V DC to the brake after the relay pull-in delay for a certain period of time (brake delay) to make the brake open as soon as possible. After the delay, the speed command takes effect and the motor starts to run.

When the stop enable command is written in the control word, the control of the holding brake is related to the set quick stop mode.

When the quick stop mode is 0 (uncontrolled stop), the drive will automatically switch to the immediate speed mode (-3) and set the target speed to 0 internally to make the motor stop as soon as possible, in the uncontrolled stop mode. After the drive changes the target speed, it will immediately cut off the 24V power supply to the brake. After the brake power is cut off and lasts for a certain period of time (the brake delay), the drive actually enters the

disabled state, and the motor is powered off and the shaft is released.

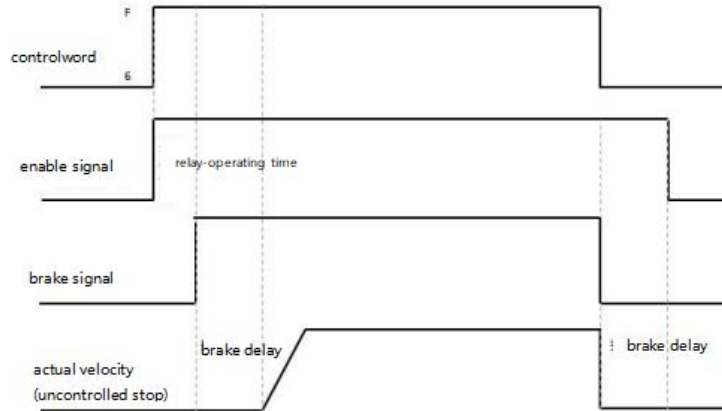


Figure 9-1 Motor holding brake sequence when the fast stop mode is 0

When the quick stop mode is 2 (quick stop deceleration stop), the drive automatically switches to speed mode (3) and decelerates to stop at quick stop deceleration (60850020) when it is disabled. Only after the drive judges that the effective target speed is zero speed, the drive will cut off the 24V brake power supply. After the brake power supply has been cut off for a certain period of time (the brake delay time), the drive will enter the disabled state, and the motor will be disconnected from the power supply.

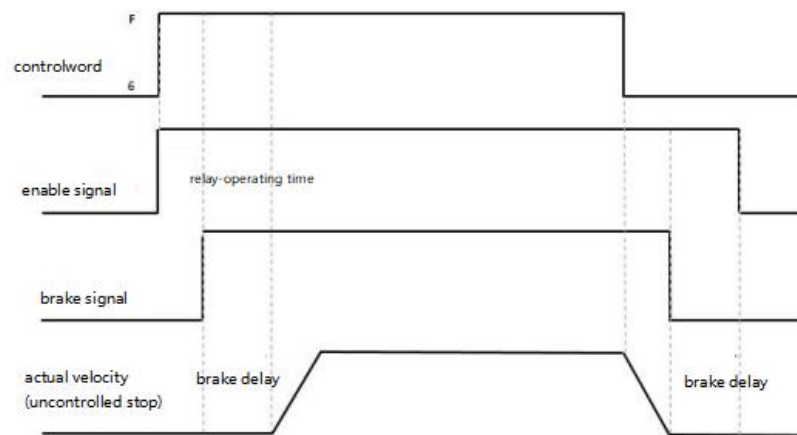


Figure 9-2 Motor holding brake sequence when the fast stop mode is 2



Note

- After the motor accessory is set to 1, and then disabled, load drop will occur. The brake delay can be appropriately increased. If adjusting the brake delay cannot improve the problem, check whether the brake torque meets the load demand.

9.3.2 IO function about brake control

Motor brake

The default brake control port is OUT5. When enabled, the motor braking output is effective and the brake is opened. When disabled, the motor braking output is ineffective and the brake is closed.

Output	Description
Motor brake	The motor brake control output signal can be used to connect the external relay that controls the motor brake. If the motor with brake is used, this function must be set; otherwise, it will damage the motor. The valid output in green indicates that the brake is open, and the valid output in gray indicates that the brake is close.

9.4 Stop mode control

9.4.1 Brake control configuration

Step 1: Configure the motor stop mode

The stop mode are divided into the following types.

Object index	Data type	Name	Description	Effect mode
605A.00	int 16	Quick_Stop_Mode	Quick stop mode Stop mode when bit2 in the control word is set to 0. For example: the control word is switched from 0x0F to 0x0B 0: Uncontrolled stop 1: Curve stop 2: Quick stop deceleration stop 5: The curve stops, and finally stops in the fast stop state 6: Rapid deceleration stop, and finally stop in rapid stop state 18: Use the motor's own resistance to brake, even if there is a problem with the encoder	Immediate
605B.00	int 16	Shutdown_Stop_Mode	Shutdown mode When the control word is switched to 0x06, the stop mode when the motor switches from the locked axis to the loose axis state 0: Uncontrolled stop 1: Curve stop 2: Quick stop deceleration stop 18: Use the motor's own resistance to brake, even if there is a problem with the encoder	Immediate
605C.00	int 16	Disable_Stop_Mode	Disable stop mode Stop mode when bit3 in the control word is set to 0. For example: the control word is switched from 0x0F to 0x07 0: Uncontrolled stop 1: Curve stop 2: Quick stop deceleration stop 18: Use the motor's own resistance to brake, even if there is a problem with the encoder	Immediate
605D.00	int 16	Halt_Mode	Pause mode Pause mode when bit8 in the control word is set to 1. For example: the control word is switched from 0x0F to 0x10F 1: Decelerate and stop at the current deceleration 2: Decelerate to stop at emergency stop deceleration	Immediate

605E.00	int 16	Fault_Stop_Mode	Error emergency stop mode Error stop mode will be activated after an alarm in the motor shaft lock state. 0: stop immediately 1: Decelerate to stop 2: Use emergency stop deceleration to stop 18: Use the motor's own resistance to brake, even if there is a problem with the encoder	Immediate
---------	--------	-----------------	--	-----------

The software operation is as follows:

- Search for the "Quick_Stop_Mode" (605A00) in the all param and set the corresponding value

The screenshot shows a software interface for parameter configuration. At the top, there is a search bar containing '605A00' and a 'Bulk Actions' button. Below this is a table with columns: Index, Description, Preset value, and Current value. The table contains one row for '605A00 Quick_Stop_Mode'. The 'Preset value' column has a dropdown menu open, showing several options: '0:stop without control', '1:stop by using ramp, then switch off', '2:stop by using quick stop deceleration, t...', '5:stop with profile deceleration, stay in qu...', '6:stop with quick stop deceleration, stay i...', and '18:using motor winding for brake even en...'. The 'Current value' column shows '0:stop without control' and a 'DEC' button.

- The operation steps for other stop modes are the same as above

Step 2: Configure the stop method

There are several stop methods available for each stop mode:

0, Stop without control: the motor will directly cut off the power to loosen the shaft without any control, and stop naturally by friction. After the stop is completed, the motor loosens the shaft. The uncontrolled stop mode has a longer deceleration time, but less mechanical impact.

1, Curve stop: Curve stop according to the Profile_Dec (608400). After the shutdown is completed, loosen the motor shaft.

5, Stop with profile deceleration(608400) : After the shutdown is completed, the motor retains the lock shaft and stops in the quick stop state. It can be enabled only after the control word 0x06 is issued to exit the quick stop state.

The relevant parameters for the stop methods of 1 and 5 are as follows:

Object index	Data type	Name	Description	Effect mode
6084.00	uint 32	Profile_Dec	Deceleration of trapezoidal curve	Immediate

The software operation is as follows:

- Search for the "Profile_Dec" (608400) in the all param and set the corresponding value

The screenshot shows a software interface for parameter configuration. It displays a table with columns: Index, Description, Preset value, and Current value. The table contains one row for '608400 Profile_Dec'. The 'Preset value' column has a dropdown menu. The 'Current value' column shows '100' and a 'DEC' button.

2, Stop by using quick stop deceleration: Stop according to the quick stop deceleration (608500), and the motor loosens the shaft after the stop is completed.

6, Stop with quick stop deceleration (608500) :After the shutdown is completed, the motor retains the lock shaft and stops in the quick stop state. It can be enabled only after the control word 0x06 is issued to exit the quick stop state.

The relevant parameters for the stop methods of 2 and 6 are as follows:

Object index	Data type	Name	Description	Effect mode
6085.00	uint 32	Quick_Stop_Dec	Deceleration for quick stop	Immediate

The software operation is as follows:

- Search for the "Quick_Stop_Dec" (608500) in the all param and set the corresponding value

608500				Bulk Actions	
☐	Index	Description	Preset value	Current value	
☐	608500	Quick_Stop_Dec		610	DEC

18, Short-circuit braking stop mode:use the motor's own resistance to stop. After the stop is completed, the motor keeps the shaft locked and stops in the quick stop state. It can be enabled after the control word sends 0x06 to exit the quick stop state. A stop mode that can quickly complete a stop even in the event of an encoder failure.

There is also a special step stop mode, which only takes effect in position mode:

Object index	Data type	Name	Description	Effect mode
2340.01	uint 8	Step_Stop_Mode	Step Mode Stop Enable It only takes effect under position control. When the position reaches the signal output, the motor uses the stepping method to lock the shaft (that is, the driver applies DC power to the motor to keep the motor in the locked shaft state), and the shaft locking current is set by the step stop current (2340.02). . 0: Do not use step mode stop enable 1: Use step mode to stop enable	Immediate
2340.02	uint 16	Step_Stop_Amp	Stop current after step stop mode is turned on	Immediate

The software operation is as follows:

- Search for the "Step_Stop_Mode" (234001) in the all param and set the corresponding value

234001				Bulk Actions	
☐	Index	Description	Preset value	Current value	
☐	234001	Step_Stop_Mode		0	DEC

- Search for the "Step_Stop_Amp" (234002) in the all param and set the corresponding value

234002				Bulk Actions	
☐	Index	Description	Preset value	Current value	
☐	234002	Step_Stop_Amp		0	DEC

9.4.2 IO function about stop mode

Emergency stop

This function can connect the emergency stop button through the digital input port, causing the motor to stop running. The stop mode is a quick stop mode.

9.5 Field weakening control

9.5.1 Field weakening control configuration

Step 1: Turn on the Field Weak mode

For the FieldWeak_Mode (0x250D.01), write 1 to enable the field weakening function. When the target speed exceeds the rated value, the field weakening control automatically intervenes and dynamically adjusts the Id and Iq currents to maintain voltage balance, allowing the motor to run at a higher speed.

Object index	Data type	Name	Description	Effect mode
250D.01	uint 8	FieldWeak_Mode	Field-Weakening mode selection: 1: Enable field-weakening function 0: Disable field-weakening function	After disabled

The software operation is as follows:

- Search for the "FieldWeak_Mode " (250D01) in the all param and set the corresponding value

	Index	Description	Preset value	Current value
☐	250D01	FieldWeak_Mode	0 0: Disable field-weakening function 1: Enable field-weakening function	0: Disable field-weakening function DEC

Step 2: Adjust the Field weakening parameter (usually keep the default)

The field weakening parameters can be set via the following objects:

Object index	Data type	Name	Description	Effect mode
250D.02	int 16	FieldWeak_BEMF_Factor	Back-EMF feedforward factor in Field-Weakening mode	Immediate
250D.03	uint 16	FieldWeak_ki	Field-Weakening PI control integral gain Ki	
250D.04	uint 8	FieldWeak_ki_micro	Field-Weakening PI control integral gain Ki fine-tuning value; range:0-64	
250D.05	uint 16	FieldWeak_kp	Field-Weakening PI control proportional gain Kp	
250D.06	int 16	FieldWeak_Filter	Field-Weakening output filter; unit:2567Hz/N	
250D.07	uint 16	FieldWeak_Id_Max	Field-Weakening D-axis current Id limit, recommended setting is 0.3 to 0.5 times of motor rated current, to avoid motor demagnetization risk	
250D.08	uint 32	FieldWeak_Trigger_Speed	Field-weakening activation triggered speed value. Set via formula: $(V_{dc} / (\sqrt{3} * K_e)) * 0.85$. V _{dc} :DC bus voltage ; K _e : Counter electromotive force coefficient.	

9.6 Fast capture and high-speed probe

9.6.1 Fast capture configuration

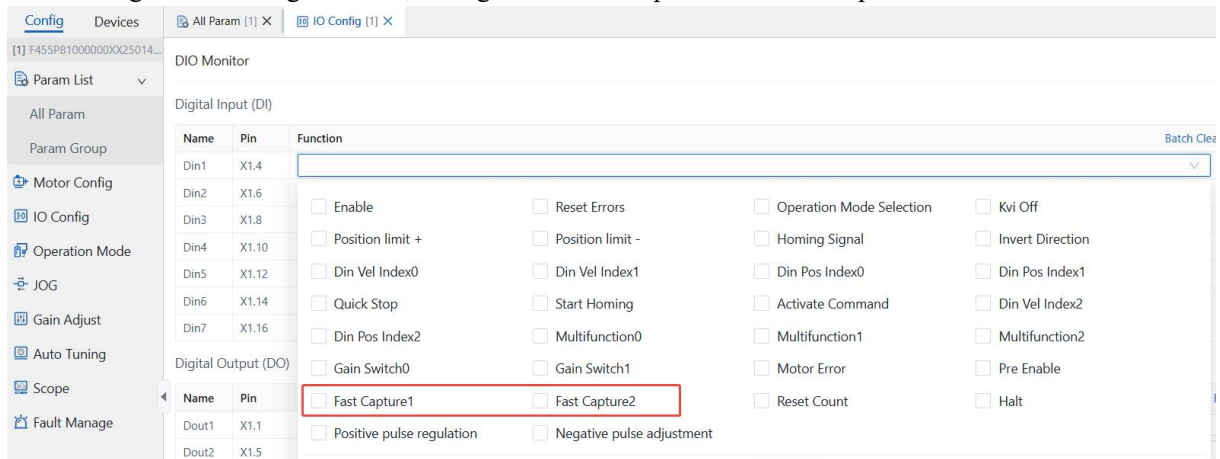
Step 1: Configure the digital input port

First, you need to configure Fast capture 1 or Fast capture 2 in the digital input port.

Object index	Data type	Name	Description	Effect mode
2010.03	uint 16	Din1_Function	8400 : Fast capture1 8800 : Fast capture2	Immediate
2010.04	uint 16	Din2_Function		
2010.05	uint 16	Din3_Function		
2010.06	uint 16	Din4_Function		
2010.07	uint 16	Din5_Function		
2010.08	uint 16	Din6_Function		
2010.09	uint 16	Din7_Function		

The software operation is as follows:

- In the digital IO Config interface, configure the Fast capture1 and Fast capture2 function



Step 2: Trigger fast capture

Internal address	Data type	Name	Description
2010.20	Uint 8	Rising_Captured1	Rising capture1 status
2010.21	Uint 8	Falling_Captured1	Falling capture1 status
2010.22	Uint 8	Rising_Captured2	Rising capture2 status
2010.23	Uint 8	Falling_Captured2	Falling capture2 status
2010.24	int 32	Rising_Capture_Position1	Captured Pos_Actual(6063.00) on Din rising edge1
2010.25	int 32	Falling_Capture_Position1	Captured Pos_Actual(6063.00) on Din falling edge1
2010.26	int 32	Rising_Capture_Position2	Captured Pos_Actual(6063.00) on Din rising edge2
2010.27	int 32	Falling_Capture_Position2	Captured Pos_Actual(6063.00) on Din falling edge2

When the DIN is configured as fast capture 1, if the rising edge of the input port arrives, the Rising_Captured1 will become 1, and the actual position will be stored in the Rising_Capture_Position1 ; if the falling edge of the input port arrives, the Falling_Captured1 will become 1, and the actual position will be stored in the Falling_Capture_Position1.

Once the Rising_Captured1 or Falling_Captured1 changes to 1, users need to reset them to 0 for the next capture, otherwise the next position will not be captured.

The usage of Fast Capture 2 is similar to that of Fast Capture 1.

The software operation is as follows:

- Activate the quick capture of valid input



- Search for the 201020 ~ 201023 in the all param ,If the value is 1, the actual position value is recorded successfully. If it is necessary to capture again, the corresponding capture status should be written to 0

☐	201020	Rising_Captured1	0	Waiting for Din rising edge	DEC
☐	201021	Falling_Captured1	0	Waiting for Din falling edge	DEC
☐	201022	Rising_Captured2	0	Waiting for Din rising edge	DEC
☐	201023	Falling_Captured2	0	Waiting for Din falling edge	DEC

- After successful capture, the corresponding capture position of 201024~201027 in the parameter list can be viewed

☐	201024	Rising_Capture_Position1	Write Prohibition	0	DEC
☐	201025	Falling_Capture_Position1	Write Prohibition	0	DEC
☐	201026	Rising_Capture_Position2	Write Prohibition	0	DEC
☐	201027	Falling_Capture_Position2	Write Prohibition	0	DEC

9.6.2 High-speed probe configuration

Step 1: Configure the probe control mode

First, you need to configure the probe function, select single trigger or continuous trigger, and rising edge trigger or falling edge trigger, etc.

Object index	Data type	Name	Description	Effect mode
3070.01	uint 16	ProbeCtrlMode	Probe control: Bit0: Probe1 enable (0: off, 1: on) Bit1: Probe1 mode (0: single-trigger, 1: continuous) Bit2-3: Probe1 trigger source (00: DIN6 fixed) Bit4-5: Probe1 trigger edge (00: rising, 01: falling, 10: falling) Bit8: Probe2 enable (0: off, 1: on) Bit9: Probe2 mode (0: single-trigger, 1: continuous) Bit10-11: Probe2 trigger source (00: DIN7 fixed) Bit12-13: Probe2 trigger edge (00: rising, 01: falling, 10: falling) Other bits: Reserved.	Immediate

The software operation is as follows:

- Select the corresponding high-speed DIN as the high-speed probe input.
- Search the ProbeCtrlMode mode or 307001 in the all param. Depending on the requirement, set the enable, capture edge, capture mode and other parameters for the two probe function.

Index	Description	Preset value	Current value
307001	ProbeCtrlMode		0 HEX
bit0	Probe1 enable		0: off
bit1	Probe1 mode		0: single-trigger
bit2-3	Probe1 trigger source		
bit4-5	Probe1 trigger edge		
bit8	Probe2 enable		0: off
bit9	Probe2 mode		0: single-trigger
bit10-11	Probe2 trigger source		
bit12-13	Probe2 trigger edge		

Step 2: Check the probe status

Confirm whether the position latch is completed according to the probe status flag.

Object index	Data type	Name	Description	Effect mode
3070.02	Unt16	ProbeStatus	Probe status: Bit0: Probe1 enabled (0: no, 1: yes) Bit1: Probe1 rising-edge captured (0: no, 1: yes) Bit2: Probe1 falling-edge captured (0: no, 1: yes) Bit8: Probe2 enabled (0: no, 1: yes) Bit9: Probe2 rising-edge captured (0: no, 1: yes) Bit10: Probe2 falling-edge captured (0: no, 1: yes) Other bits: Reserved.	Immediate

The software operation is as follows:

- Search the ProbeStatus or 307002 in the all param and confirm whether the position latch is completed

307002	Index	Description	Preset value	Current value
307002	307002	ProbeStatus	Read Only	0
	bit0	Probe1 enabled		0: no
	bit1	Probe1 rising-edge captured		0: no
	bit2	Probe1 falling-edge captured		0: no
	bit8	Probe2 enabled		0: no
	bit9	Probe2 rising-edge captured		0: no
	bit10	Probe2 falling-edge captured		0: no

Step 3: Get the captured value

The corresponding probe capture count and the position latch value of the rising and falling edges can be obtained based on the following objects.

Object index	Data type	Name	Description	Effect mode
3070.03	uint 16	Probe1_CaptureCNT	Probe1 capture count: Bit0-7: Rising-edge count Bit8-15: Falling-edge count	Immediate
3070.04	int 32	Probe1_RisingCapturePos	Probe1 rising-edge captured position value	Immediate
3070.05	int 32	Probe1_FallingCapturePos	Probe1 falling-edge captured position value	Immediate
3070.08	uint 16	Probe2_CaptureCNT	Probe2 capture count: Bit0-7: Rising-edge count Bit8-15: Falling-edge count	Immediate
3070.09	int 32	Probe2_RisingCapturePos	Probe2 rising-edge captured position value	Immediate
3070.0A	int 32	Probe2_FallingCapturePos	Probe2 falling-edge captured position value	Immediate

Adjust the probe delay compensation value to make the latch value more accurate (usually keeping the default value).

Object index	Data type	Name	Description	Effect mode
3070.06	int 16	Probe1_Rising Edge Delay Compensation	Probe1 rising-edge delay compensation value	Immediate

3070.07	int 16	Probe1 Falling Edge Delay Compensation	Probe1 falling-edge delay compensation value	Immediate
3070.0B	int 16	Probe2 Rising Edge Delay Compensation	Probe2 rising-edge delay compensation value	Immediate
3070.0C	int 16	Probe2 Falling Edge Delay Compensation	Probe2 falling-edge delay compensation value	Immediate

The software operation is as follows:

- Search the 307003~307005(Probe 1 capture value) and 307008~30700A(Probe 2 capture value) in the all param and confirm whether the position latch is completed

☐	☐	307003	Probe1_CaptureCNT	Read Only
		bits0-7	Rising-edge counts.	
		bits8-15	Falling-edge counts.	
☐	☐	307004	Probe1_RisingCapturePos	Read Only
☐	☐	307005	Probe1_FallingCapturePos	Read Only
☒	☐	307008	Probe2_CaptureCNT	Read Only
		bits0-7	Rising-edge counts.	
		bits8-15	Falling-edge counts.	
☐	☐	307009	Probe2_RisingCapturePos	Read Only
☐	☐	30700A	Probe2_FallingCapturePos	Read Only

- 307006, 307007 are the rising and falling edge compensation values of probe 1, 30700B, 30700C are the rising and falling edge compensation values of probe 2

☐	307006	Din6_RiseDelayComp	
☐	307007	Din6_FallDelayComp	
☐	30700B	Din7_RiseDelayComp	
☐	30700C	Din7_FallDelayComp	

Chapter 10 Performance adjustment

The servo system generally includes three control loops: current loop, velocity loop and position loop. For the servo system, good control loop parameters can improve the performance of the servo and better meet the field process requirements. Therefore, it is necessary to adjust good control loop parameters.



Remind

- kaff: Position loop acceleration feedforward
- kvff: Position loop speed feedforward
- kvp: Velocity loop proportional gain
- kvi: Velocity loop integral gain
- kpp: Position loop proportional gain

10.1 Online auto-tuning

The online auto-tuning function of the servo drive does not need to be separated from the controller program, and can be turned on at the touch of a button during the movement of the machine, by automatically calculating the load inertia and automatically adjusting the gain to the load state.

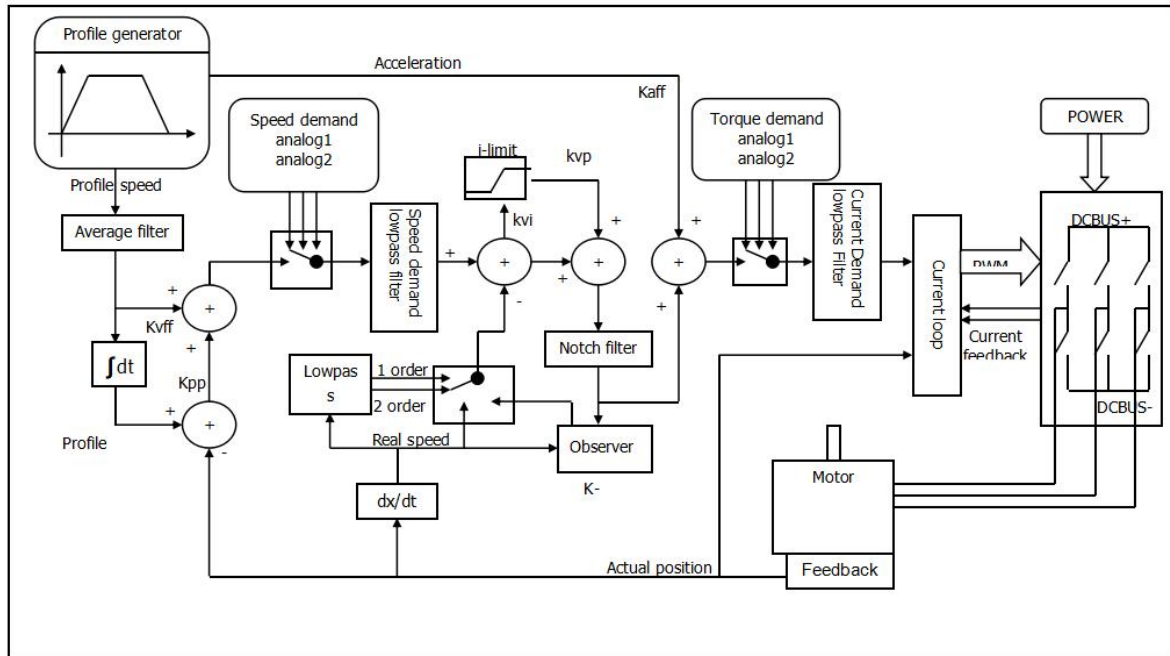
Table 10-1 Online auto-tuning function related parameters

Object index	Name	Description	Default	Range	R: read W: write S: save
2340.10	Auto_Tuning_Control	The BCD code consists of the following bits to determine the self-tuning mode: Bit0: Enables online self-tuning Bit1: Automatically adjusts kvp after online self-tuning Bit2: The acceleration feedforward is automatically adjusted after the online self-tuning Bit4: No self-tuning when running forward Bit5: No self-tuning when running in reverse Bit7: Use friction to calculate the load	12	0-31	RWS

10.2 Manual parameter tuning

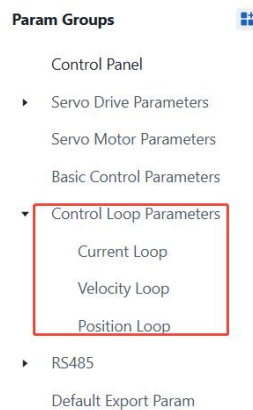
Due to constraints such as load conditions, when the automatic gain adjustment cannot achieve the expected results, the gain can be manually fine-tuned to optimize the control effect. During the debugging process, the speed loop and position loop parameters mainly need to be adjusted. The speed loop parameters are related to the load inertia of the entire mechanical system converted to the motor shaft. The position loop is the outermost control loop

of the servo system and is related to the motor action mode, that is, the field application. The current loop is the innermost control loop in the servo system, and the current loop parameters are related to the motor parameters. After the motor is correctly configured, the system will default the current loop parameters to the optimal parameters of the equipped motor, so there is no need to adjust again.



10.2.1 Operation process and related parameters

First, find the **[Param groups]** function in the function configuration bar of the KS PRO. The interface is as follows:



The specific adjustment methods are as follows:

Step1: Speed loop setting method

By adjusting the parameters of the speed loop and the position loop, better control performance can be achieved.

Step 1: Adjustment for Kvp

The proportional gain of the speed loop can be adjusted in the **[Velocity loop]**.

60F901				Bulk Actions				File Export		File Import		Compare	
	Index	Description	Preset value	Current value	Max	Min	Units						
	60F901	Kvp[0]		-	32767	1							

Relevant parameter:

Object index	Data type	Name	Description	Effect mode
60F9.01	uint 16	Kvp[0]	The proportional gain setting value of the speed loop	Immediate

Method:

Increasing Kvp can improve the bandwidth of the velocity loop and make the speed response ability faster. To calculate the Kvp, please refer the following formula.

$$kvp = \frac{1.853358080 \times 10^5 \times J \times \pi^2 \times B}{I_{Max} \times Kt \times Encoder}$$

Kt	Motor torque constant, unit N.m/Arms*100
J	Inertia, unit kg*m^2*10^6 B
B	Velocity loop bandwidth unit: Hz
Imax	MAX current(6510.03), unit: DEC
Encoder	Resolution of the encoder

Because the -3 of Operation_mode has no acceleration and deceleration and does not participating the speed loop control, the -3 mode and automatic flip mode can be used to run the motor when adjusting the speed loop parameters. During the reciprocating motion, the responsiveness of the servo can be monitored by collecting the speed step curve .

If the Kvp too low, the bandwidth of the velocity loop and the position loop will not match, which will lead to oscillation. If the Kvp is too high, it will cause the speed to overshoot and cause the speed loop to oscillate. In some rigid connection loads (such as ball screw, rack and pinion, etc.) The Kvp should be as high as possible. By comparing the oscillograms under different Kvp, find the optimal curve – the actually speed curve should be tight follows the target command and does not have velocity oscillations.

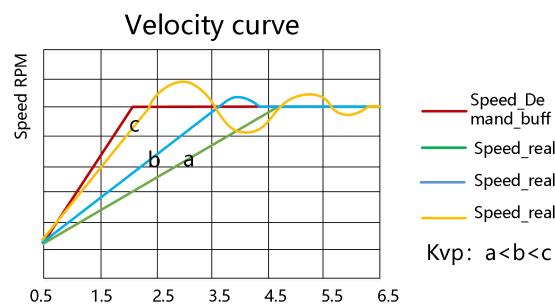


Figure 10-1 Speed step curve after Kvp adjustment

**Remind**

- Speed step curve: Curve that shows rise/fall changes.
- The vibration and noise of the motor can be felt by hand touch and listenin

Step 2: Adjustment for Kvi

The speed loop integral gain can be adjusted in the [Velocity loop].

☐	Index	Description	Preset value	Current value	Max	Min	Units	Effect mode
☐	60F902	Kvi[0]		-	1023	0		Immediate

☐	Index	Description	Preset value	Current value	Max	Min	Units	Effect mode
☐	60F907	Kvi/32		-	32767	0		Immediate

Relevant parameter:

Object index	Data type	Name	Description	Effect mode
60F9.02	uint 16	Kvp[0]	The set value of the speed loop integral gain	Immediate
60F9.07	uint 16	Kvi/32	It is 1/32 of Kvi, used for high resolution encoder.	Immediate

Method:

Kvi is designed to eliminate static errors. It can strengthen the low frequency gain of the velocity loop, and a large Kvi can reduce the response ability from low-frequency disturbance, thereby improving the anti-disturbance capability in low-frequency.

When Kvi is too small, The integral time will be long. The error adjustment will be slow, and it will take longer time to make the system entering steady state. When the integral gain of the speed loop is too large, the error will appear and in severe cases, the motor may oscillate. Kvi should be adjusted to the servo eliminating the error and enter a stable state efficiently.

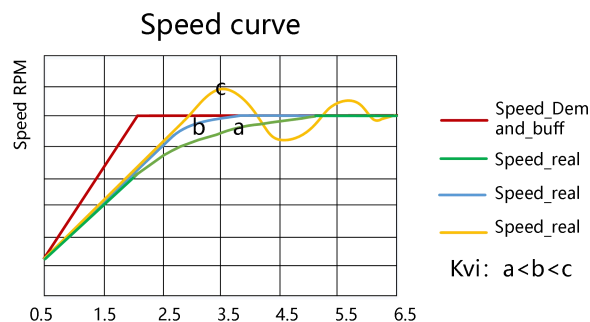


Figure 10-2 Speed step curve after Kvi adjustment

Generally, if the machine has more friction, the kvi should be set larger. If the entire system needs to respond quickly, kvi should be set as small as possible. If you need to dynamically close kvi during operation, you can use the kvi close in Din function.

**Note**

- $Kvi32 = 1/32$ of $Kvi[0]$, In high resolution encoder, directly increasing the data in the $kvi[0]$ will lead to overshoot. At this time, you can reduce the data in $kvi[0]$, and fine-tune it in the $kvi32$
- Generally, the parameters of kpp and kvi are relative. When the kpp data is adjusted to be much larger than the factory value, the kvi should also be appropriately increased on the basis of the factory value.

Step 3: Adjustment of Kvi_Sum_Limit

Search for the Kvi_Sum_Limit (60F9.08) in the [**Velocity loop**] and set the corresponding value.

Relevant parameter:

Object index	Data type	Name	Description	Effect mode
60F9.08	int 32	Kvi_Sum_Limit	Speed loop integral output limit	Immediate

Method:

The Kvi_sum_Limit is the maximum current limit when performing adjustment. Usually the default value can satisfy most applications. However, if the application system has a large resistance or the acceleration is too large, so that the actual current has reached the Kvi_sum_limit current, and at the same time, the actual speed is far less than the target speed. The value of this parameter should be increased. If the output current is prone to saturation, but the saturated output current will cause low frequency oscillations, this parameter should be reduced.

The Kvi_sum_limit should be as small as possible on the basis of satisfying the application. The smaller the Kvi_sum_limit value, the shorter the adjustment time.

Step 4: Adjustment for Speed_fb_N

The speed feedback filter can be set through Other Filter in the [**All param**].

☐	60F905	Speed_Fb_N		1000	100	Hz	Immediate
☒	60F906	Speed_Mode		FF	0		Immediate
	bit0~3	feedback mode					
	bit4	speed command filtering					
	bit5	Controls the enablement of the current command fi					

Relevant parameter:

Object index	Name	Description	Default	Range	Effect mode
60F9.05	Speed_Fb_N	bandwidth of speed feedback filter $BW = \text{Speed_Fb_N} * 20 + 100 [\text{Hz}]$	7	0-45	Immediate
60F9.06	Speed_Mode	0:2nd Order FB LPF 1:No FB LPF 2:Observer FB 4:1st Order FB LPF 10:2nd LPF+SPD_CMD FT 11:SPD_CMD FT 12:SPD_CMD FT+Observer 14:1st LPF+Observer Bit4: 1 enables the speed command filter. Bit5: Controls the enablement of the current command filter, 0 to disable the filter, 1 to enable the filter	1	0-255	Immediate

Method:

Feedback filter can reduce noise from the feedback path, eg, encoder resolution noise.

For different applications, the speed feedback filter can be converted to 1st and 2nd order by Speed_Mode.

1st order can reduce more noise, but produce less phase shift, allowing the higher Kvi or Kvp.

2nd order filter reduces more noise, but also provide more phase shift, which can limit the Kvi or Kvp.

Usually, if the machine and the load use soft connection, it is recommended to use a 1st order low-pass feedback filter or turn off the feedback filter. If the rigid connection is used or the load is too heavy, the 2nd order low-pass feedback filter could be used. Factory default is a 2nd order low-pass feedback filter, which is suitable for most applications.

If the motor noise is too loud when adjusting the Kvi or Kvp, you can appropriately reduce the speed feedback filter parameter(60F9.05). However, the velocity loop feedback filter bandwidth F must be greater than 2 times of the velocity loop bandwidth. Otherwise, oscillation may result. Speed feedback filter bandwidth $F = \text{speed feedback filter} \times 20 + 100$ [HZ]

Step 5: Adjustment of output filter

The output filter can be set through Other Filter in the [All param].

index	description	reset value	current value	max	min	units	effect mode
60F915	Output_Filter_N	-	-	2546	20.05	Hz	Immediate

Relevant parameter:

Object index	Data type	Name	Description	Effect mode
60F9.15	uint 8	Output_Filter_N	A first-order low-pass filter located in the forward channel of the current loop to filter the current command	Immediate

Method:

The output filter is a first order low pass torque filter. It can reduce the high-frequency torque order output by the speed control loop, so as to achieve the purpose of suppressing the resonance of the entire system.

When the output filter is set to 2546Hz (1DEC), it means to close the output filter control. When the system has resonance, the user can try to adjust the DEC value of the output filter setting from small to large to reduce noise.

The output filter frequency can be calculated by the following formula:

$$f = \frac{1}{2 \times \pi \times T_s \times \text{Output_Filter_N}}$$

Output_Filter_N (60F9.15), Uint : DEC

Ts Constant 62.5us



Note

- The larger the DEC value set by the output filter, the stronger the filtering effect.
- Output filter settings [Hz]=2546/[DEC]
- Output filter settings [DEC]=2546/[Hz]

Step 6: Notch Filter Adjustment

A notch filter can suppress the resonance frequency by reducing the gain near the mechanical resonance.

If the resonant frequency is unknown, it can be obtained by following method:

Adjust the load so that the system runs in the resonance region, monitor the actual current at resonance on the oscilloscope, and then adjust the notch filter to observe whether the resonance disappears. When the current data at

resonance is collected on the software oscilloscope, the period value of adjacent harmonics can be measured by the cursor to calculate the resonance frequency. After collecting multiple sets of harmonic cycles to calculate the average value of the cycles, the resonant frequency can be calculated using the average value of the cycles.

Related parameters can be searched and configured in the [All param]:

Table 10-1 Notch filter parameter

Object index	Name	Description	Default	Range
60F9.03	Notch_N	Used to set the frequency of the internal notch filter to eliminate the mechanical resonance that occurs when the motor drives the machine	550	100-2000
250B.01	Notch_N[1]			
250B.02	Notch_N[2]			
250B.03	Notch_N[3]			
60F9.04	Notch_On	Notch filter control of Speed loop Bit0~1: The BCD code composed of two bits determines the filter to be turned on. Bit4: 1: Indicates FFT analysis using actual current, 0: Indicates that the target current is used for analysis Bit5: 1: Indicates use actual speed for FFT analysis, 0: Indicates do not use actual speed for FFT analysis Bit.7: 1 indicates that the FFT function is turned on, it will automatically become 0 after the analysis is completed, and it can only be turned on in the enabled state.	0	0-255

(1) Single-point notch filter

When the notch filter control (60F9.04) is set to 0, it indicates to close the notch filter control; when set to 1, it indicates to open a single notch filter, and the frequency of the filter is set by the notch filter (60F9.03) .

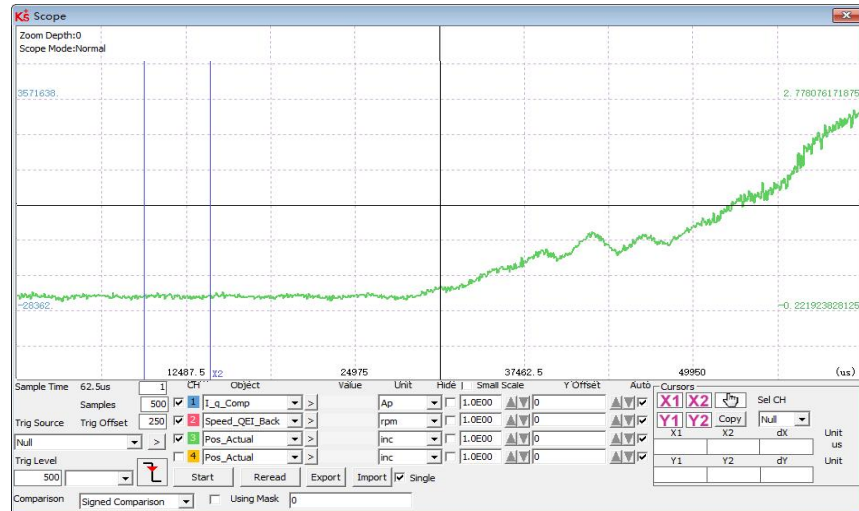
Table 10-2 Single notch filter debugging case

Description	Process
Limit output current of drive	The target current limit is adjusted through the basic operating interface of the servo software
Use the oscilloscope to collect the current waveform in the resonance region. Note that when collecting the current with the oscilloscope, the sampling period should be set as small as possible, otherwise the resonance waveform may not be collected.	

Observe the resonant frequency with an oscilloscope and calculate the notch filter setting value based on the frequency

The resonant frequency in the figure is 200Hz, the notch filter control is set to 1, and the notch filter is set to 200Hz

After adding a notch filter, collect the actual current waveform again, and adjust it until the current waveform is smooth and free of resonance.



Note

- In order to prevent machine damage caused by large resonance amplitude during debugging, the value of target current limit can be reduced and then the notch filter can be adjusted.
- During the automatic setting of notch filter, the oscilloscope function cannot be used. After the setting is successful, the oscilloscope automatically switches the acquisition object, and click the reread data to display the FFT result amplitude.

(2) FFT Multi-point Notch Filter

The high order notch filter technology is acquired, the mechanical resonance frequency of the load is measured automatically, and the tuning results are written into filter 0 and filter 1. iSMD open a total of 4 notch filters, filter 0 and filter 1 are automatically adjusted notch filters, filter 2 and filter 3 are manually adjusted notch filters, and the corresponding filters can be turned on/off by the notch filter control.

When the notch filter control (60F9.04) is set to 128DEC, it indicates that the actual current is used for FFT analysis. When the FFT is successfully tuned, the notch filter control bit7 to reset, and the BCD code combined with bit0 and bit1 determines the filter to be turned on.

Step2: Position loop setting method

Step 1: Kpp adjustment

Kpp[0] can be adjusted in the [Position loop].

☐	Index	Description	Preset value	Current value	Max	Min	Units	Effect mode
☐	60FB01	Kpp[0]		-	327.67	0	Hz	Immediate

Relevant parameter:

Object index	Data type	Name	Description	Effect mode
60FB.01	uint 16	Kpp[0]	The proportional gain setting value of the speed loop	Immediate

Method:

Increasing the Kpp can increase the position loop bandwidth to reduce the positioning time and the following error, but if the setting is too large, it will cause noise or even oscillation, and it must be set according to the load conditions. $K_{pp} = 103 \times \text{position loop bandwidth}$. The bandwidth of the position loop cannot exceed the bandwidth of speed loop. It is recommended that the setting value of the position loop bandwidth be less than one quarter of the bandwidth of the speed loop.

Step 2: Kvff adjustment

The position loop speed feedforward can be adjusted in the **[Position loop]**.

☐	Index	Description	Preset value	Current value	Max	Min	Units	Effect mode
☐	60FB02	K_Velocity_FF		-		0	%	Immediate

Relevant parameter:

Object index	Data type	Name	Description	Default	Range
60FB.02	int 16	K_Velocity_FF	Position-Loop Velocity Feedforward Setting	0	0 ~ 100%

Method:

Increasing the Kvff can reduce the position following error and improve the dynamic response characteristics of the entire system, but it may lead greater overshoot during acceleration and deceleration. When the position command signal is not smooth, reducing the Kvff can reduce the motor oscillation. When the feedforward is set to 0, if only the Kvp is used to position the servo, the in-position time will be longer.

Kvff function can be regarded as the upper controller, such as PLC, has the opportunity to directly control the velocity in position operating mode. In fact, the function will consume part of the speed loop responsiveness, so the setting value needs to match the position loop and velocity loop bandwidth.

The elastic factor of the flexible load, such as pulley, will cause the speed feedforward command loaded to the speed loop to be unsmooth and cause the load to oscillate. In this type of load, the position loop speed feedforward can be appropriately reduced. In a rigid load, if adjusting the Kpp cannot eliminate the vibration generated by the motor during acceleration and deceleration, speed loop can be gradually decrease the default 100% feedforward.

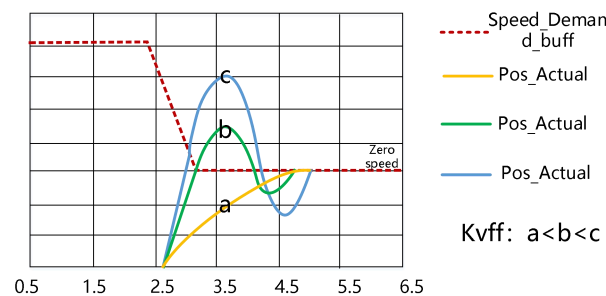


Figure 10-3 Speed step curve after Kvff adjustment

Step 3: K_Acc_FF adjustment

K_Acc_FF can be searched and adjusted in the **[Position loop]**.

☐	Index	Description	Preset value	▼ Current value	▼ Max	Min	Units	Effect mode
☐	60FB03	K_Acc_FF		-	32767	10		▼ Immediate

Relevant parameter:

Object index	Data type	Name	Description	Default	Range
60FB.03	int 16	K_Acc_FF	Position loop acceleration feedforward setting	0	0-100%

Method:

Users are not advised to adjust this parameter. When the practical application requires very high position loop response, the acceleration feedforward can be adjusted to improve the response performance.

The acceleration feedforward function can be seen as the upper controller has the opportunity to directly control the torque in the position operating mode. This function actually consumes some of the current loop responsiveness, so if the settings do not match the Kvp and velocity loop bandwidth, overshoot and oscillation will occur.

Also, the velocity feed forward to the velocity loop may not be smooth and has some noise signal inside, so a large velocity feed forward value will also amplify the noise.

Kaff can be calculated by following function:

$$\text{ACC_}\% = 6746518 / \text{K_Acc_FF} / \text{Easy_Kload} * 100$$

ACC_% This means what percentage will be used for acceleration feedforward.

K_Acc_FF Kaff(60FB.03), calculates the final internal factor of the feedforward.



Note

The smaller the value of the K_Acc_FF parameter, the greater the position loop acceleration feedforward.

Step 4: S-Curve Smoothing Factor adjustment

S-Curve Smoothing Factor can be searched and adjusted in the **[Position loop]**.

60FB05	S-Curve Smoothing Factor	-	255	1
--------	--------------------------	---	-----	---

Relevant parameter:

Object index	Data type	Name	Description	Default	Range
60FB.05	uint 16	S-Curve Smoothing Factor	This parameter adjusts the smoothness of the S-curve	1	1~255

Method:

The smoothing filter is a moving average filter. It filters the velocity commands from the velocity generator to make the velocity and position commands smoother. Using this filtering causes velocity and position commands to be delayed in the drive. So for some applications, like CNC, it is better not to use this filter, but to do the smoothing in the CNC.

Smoothing filters reduce machine influence by smoothing commands. S-Curve Smoothing Factor (60FB.05) defines the time constant of this filter in ms. Under normal circumstances, if the machine system vibrates when starting and stopping, it is recommended to increase the smoothing filter setting.

Step 5: Other adjustment

Search for Pos_Speed_Filter (60FB.04) in the **[All param]** and select to use trapezoidal curve or S curve.

60FB04

Q

Bulk Actions

	☐	Index	Description	Preset value	Current value
	☐	60FB04	Pos_Speed_Filter	▼ 1:use trapezoidal curve ... 3:use S curve mode	1:use trapezoidal curve m... DEC

Relevant parameter:

Object index	Data type	Name	Description	Default	Effect mode
60FB.04	uint 8	Pos_Speed_Filter	1: Use trapezoidal curve mode 3: Use S curve mode	1	Immediate

Method:

In position mode (operation mode 1), the position loop command filter control(60FB.04) can be set to 3 to enable the S-curve control. This curve has no command delay and is suitable for long-distance positioning control. When the speed step of the trapezoidal speed curve is large, it will cause a certain impact on the mechanical equipment. The S-shaped curve has better flexibility and smooth acceleration and deceleration, which can effectively overcome the mechanical vibration caused by the sudden change of speed.

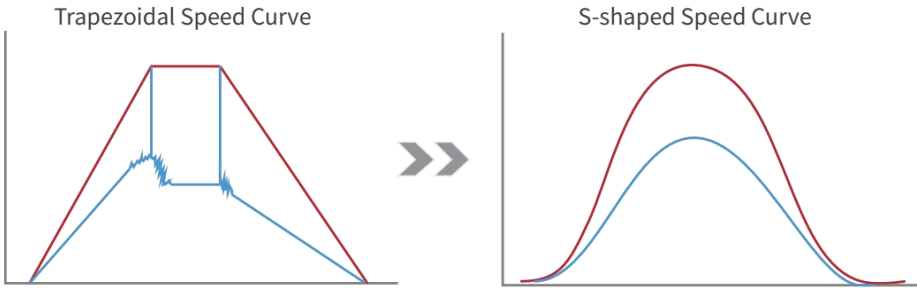


Figure 10-4 S-shaped speed step curve

➔

Note

The S-curve is valid only in position mode (operating mode 1)

10.3 Gain switch

The drive support 4 sets of PI gain settings. Each set includes Kvp, Kvi, Kpp. The actual PI parameters used are Kvp (x), Kvi (x), and Kpp (x), x=data of PI pointer.

Table 10-3 Gain switching parameter

Object index	Type	Name	Description	Unit
60F9.01	uint 16	Kvp[0]	Gain parameter called when PI pointer is 0	Dec, Hz
60F9.02	uint 16	Kvi[0]		Dec
60FB.01	int 16	Kpp[0]		Dec. Hz
2340.04	uint 16	Kvp[1]	Gain parameter called when PI pointer is 1	Dec, Hz
2340.05	uint 16	Kvi[1]		Dec
2340.06	uint 16	Kpp[1]		Dec. Hz
2340.07	uint 16	Kvp[2]	Gain parameter called when PI pointer is 2	Dec, Hz
2340.08	uint 16	Kvi[2]		Dec

2340.09	uint 16	Kpp[2]		Dec. Hz
2340.0A	uint 16	Kvp[3]	Gain parameter called when PI pointer is 3	Dec, Hz
2340.0B	uint 16	Kvi[3]		Dec
2340.0C	uint 16	Kpp[3]		Dec. Hz
60F9.28	uint 8	PI point	Indicates the PI parameter being called	Dec
60F9.09	uint 8	Auto PI switch	When the target position/target speed is reached, that is, when bits.Target_reached=1 , select the PI parameter, which is suitable for the occasions where different PI parameters need to be used during the motion process and the static state : 0: PI pointer (60F9.28)=0; 1: PI pointer (60F9.28)=1;	Dec

10.3.1 Switch PI through a digital input port

Configure the function gain switch 0 or/and gain switch 1 at the digital input port, and the value of the PI pointer is the BCD code composed of the input:

Bit0: Gain switch 0

Bit1: Gain switch 1

Correspondence between gain switching and PI pointer :

Gain switch 0 Effective input level	Gain switch 1 Effective input level	Value of PI pointer
1	0	1
0	1	2
1	1	3

For example:

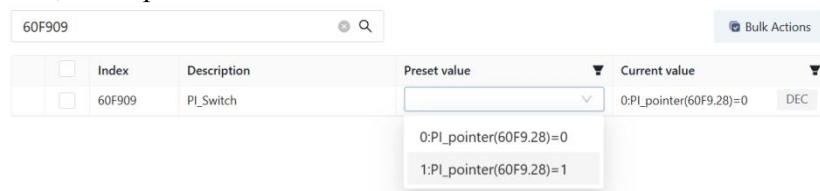
PIN number	Function	Actual input	Simulation input	Polarity	Valid input
DIN1	Gain Switch0			f High	
DIN2	Gain Switch1			f High	

Gain Switch 0=1, Gain Switch 1=0

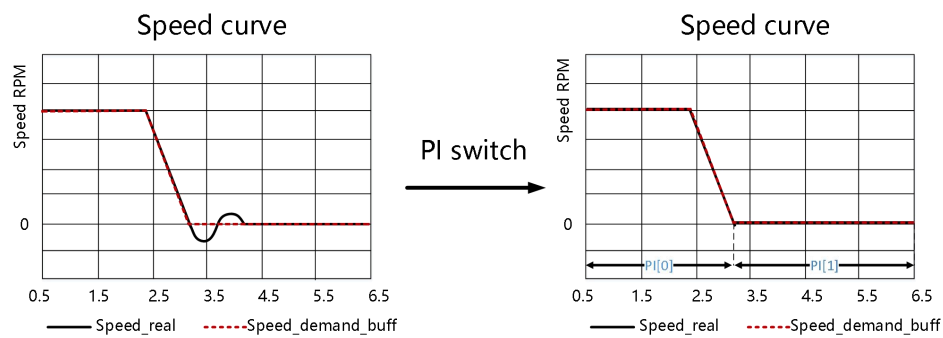
Therefore, the PI pointer =1, and the valid PI parameters are Kvp[1], Kvi[1] , Kpp[1].

10.3.2 Auto PI switch

Set PI_switch (60F9.09) = 1: When the motor is running, the PI pointer is 0; when the motor is in the position to or the motor zero speed state, the PI pointer is 1.



Some equipment with large inertia may be affected by inertia or elastic force and vibrate when it decelerates to stop. At this time, switching flexible PI parameters helps to achieve efficient shutdown. Automatic PI switching is suitable for applications where different PI parameters are required for these motors in running and stopped states. If the gain switching function is defined on the digital input port, the automatic gain switching will be invalid.



10.3.3 Directly set the PI pointer value

Search for the PI_Point_Auto (3060.20) in the entire parameter list. After setting the corresponding value, the corresponding control loop gain is used.

306020

✕

🔍

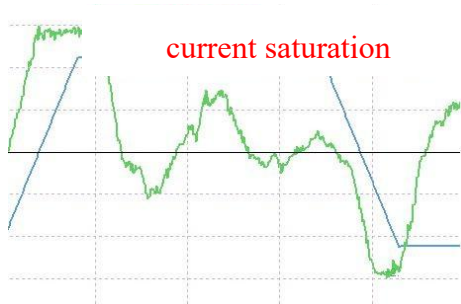
Bulk Actions

	☐	Index	Description	Preset value	Current value	
	☐	306020	PI_Point_Auto		0	DEC

10.4 Other factors affecting performance

1. Control commands created by a controller (e.g. PLC).

Control commands should be as smooth as possible and must be correct and reasonable. For example, the acceleration in the control command cannot exceed the maximum acceleration that can be produced by the motor torque. By collecting the speed and current step waveforms, during the acceleration process, when the current is saturated and clipped, the acceleration and deceleration can be appropriately reduced or the smoothing filter can be increased.



- The control command should follow the bandwidth limits of the control loop.

2. Mechanical design

In applications, performance is often limited by the machine. Various factors can affect the final control performance.

Control performance will affect the final performance of the machine such as accuracy, responsiveness and stability.

Chapter 11 List of common object parameters



Note

- CANopen address and 232 communication address are the same
- Express register addressing in the form of Index (16-bit address) and Subindex (8-bit sub-address)
- The digit 0x08 indicates that the length of the data stored in this register is 1 Byte, the digit 0x10 indicates that the length of the data stored is 2 Bytes, and the digit 0x20 indicates that the length of the data stored is 4 Bytes
- R: readable, W: writable, S: savable, M: mappable; T: traceable, L: read in the control ring, B: need to save and restart
- A complete CANopen address is in the format of: 60400010 (control word)
- Modbus address is a 4-digit hexadecimal number
- A complete Modbus address format is: 3100 (control word)

Object list

Category	Index	Sub-index	Modbus address	Data type	Name
Category 1 [Input/Output Settings]	2010	01	0810	uint 16	Din_Polarity
		02	0820	uint 16	Din_Simulate
		03	0830	uint 16	Din1_Function
		04	0840	uint 16	Din2_Function
		05	0850	uint 16	Din3_Function
		06	0860	uint 16	Din4_Function
		0A	08A0	uint 16	Din_Real
		0D	08D0	uint 16	Dout_Polarity
		0E	08E0	uint 16	Dout_Simulate
		0F	08F0	uint 16	Dout1_Function
		10	0900	uint 16	Dout2_Function
		14	0940	uint 16	Dout_Real
		18	0980	uint 16	Zero_Speed_Window
		19	0990	uint 8	Limit_Function
Category 2 [Speed/position setting]	2020	01	0C10	int 32	Din_Pos0
		02	0C20	int 32	Din_Pos1
		03	0C30	int 32	Din_Pos2
		04	0C40	int 32	Din_Pos3
		05	0C50	int 32	Din_Speed0
		06	0C60	int 32	Din_Speed1
		07	0C70	int 32	Din_Speed2
		08	0C80	int 32	Din_Speed3
		10	0D00	int 32	Din_Pos4
		11	0D10	int 32	Din_Pos5
		12	0D20	int 32	Din_Pos6
		13	0D30	int 32	Din_Pos7
		14	0D40	int 32	Din_Speed4
		15	0D50	int 32	Din_Speed5
		16	0D60	int 32	Din_Speed6
		17	0D70	int 32	Din_Speed7
		0F	0CF0	uint 16	Din_Control_Word

Category	Index	Sub-index	Modbus address	Data type	Name
Category 3 [Pulse input Settings]	2508	01	1910	int 16	Gear_Factor[0]
		02	1920	uint 16	Gear_Divider[0]
		03	1930	uint 8	PD_CW
		04	1940	int 32	Gear_Master
		05	1950	int 32	Gear_Slave
		06	1960	uint 16	PD_Filter
		09	1990	uint 16	Position_Window_time
		0C	19C0	int 16	Master_Speed
		0D	19D0	int 16	Slave_Speed
Category 4 [Error code]	2601	00	1F00	uint 16	Erroe_State
	2602	00	2000	uint 16	Erroe_State 2
Category 5 [Storage parameter setting]	2FF0	01	2910	uint 8	Store_Data
		03	2930	uint 8	Store_Motor_Data
Category 6 [Control and status of equipment]	6040	00	3100	uint 16	Controlword
	6041	00	3200	uint 16	Statusword
Category 7 [Stop mode setting]	605A	00	3400	int 16	Quick_Stop_Mode
	605B	00	3410	int 16	Shutdown_Stop_Mode
	605C	00	3420	int 16	Disable_Stop_Mode
	605D	00	3430	int 16	Halt_Mode
	605E	00	3440	int 16	Fault_Stop_Mode
Category 8 [Working mode setting]	6060	00	3500	int 8	Operation_Mode
Category 9 [Basic parameter setting]	6063	00	3700	int 32	Pos_Actual
	6065	00	3800	uint 32	Max_Following_Error
	6067	00	3900	uint 32	Target_Pos_Window
	606C	00	3B00	int 32	Speed_Real
	6071	00	3C00	int 16	Target_Torque%
	6073	00	3D00	uint 16	CMD_q_Max
	6078	00	3E00	int 16	I_q
	607A	00	4000	int 32	Target_Position
	607C	00	4100	int 32	Home_Offset
	607D	01	4410	int 32	Soft_Positive_Limit
		02	4420	int 32	Soft_Negative_Limit
	607E	00	4700	uint 8	Invert_Dir
	6080	00	4900	uint 16	Max_Speed_RPM
	6081	00	4A00	uint 32	Profile_Speed
	6083	00	4B00	uint 32	Profile_Acc
	6084	00	4C00	uint 32	Profile_Dec
	6085	00	3300	uint 32	Quick_Stop_Dec
	60F6	08	5880	int 16	CMD_q
	60FD	00	6D00	uint 32	Digital_Inputs
	60FF	00	6F00	int 32	Target_Speed
Category 10 [Origin mode setting]	6098	00	4D00	int 8	Homing_Method
	6099	01	5010	uint 32	Homing_Speed_Switch
		02	5020	uint 32	Homing_Speed_Zero
		05	5050	uint 8	Home_Offset_Mode
Category 11 [Speed loop parameters]	609A	00	5200	uint 32	Homing_Accelaration
	60F9	01	6310	uint 16	Kvp[0]
		02	6320	uint 16	Kvi[0]
		05	6350	uint 8	Speed_Fb_N
		07	6370	uint 16	Kvi/32
		0A	63A0	int 32	Target_Speed_Window
		14	6440	uint 16	Zero_Speed_Time

Category	Index	Sub-index	Modbus address	Data type	Name
Category 12 [Position loop parameter]	60FB	01	6810	int 16	Kpp[0]
		02	6820	int 16	K_Velocity_FF
		03	6830	int 16	K_Acc_FF
		05	6850	uint 16	S-Curve Smoothing Factor
Category 13 [Alarm parameters]	6410	18	7180	int 16	Temp_Motor_Ref
		19	7190	int 16	Temp_Motor
	6510	07	8670	uint 16	Voltage_Under
		08	8680	uint 16	Chop_Voltage
		09	8690	uint 16	Voltage_Over

11.1 Mode and control (0x6040)

Name	CANopen	Modbus	Command properties	Data type	Unit	Initial value	Explanation
Control word	60400010	3100	RWLTM	uint 16	HEX	0006	0x06: Loose shaft 0x0F: Lock shaft 0x0B: Quick stop, load stop - voltage disconnect 0x2F→3F: Enter the absolute positioning mode 0x4F→5F: Enter relative positioning mode 0x103F:mImmediate absolute positioning as target position changes 0x0F-1F: homing 0X86: Error reset
Status word	60410010	3200	RLTM	uint 16	HEX	0218	The status byte shows the status of the drive bit0: Ready_on bit1: Switched_on bit2: Operation_enable bit3: Fault bit4: Voltage_enable bit5: Quick_stop bit6: Switchon_disabled bit7: Warning bit8: Internal retention bit9: Remote bit10: Target_reached bit11: Internal limit activation bit12: Pulse response bit13: Following_Error bit14: Commutation_Found bit15: Reference_Found
Operating mode	60600008	3500	RWLM	int 8	DEC	-4	1: Positioning mode with position ring 3: Speed mode with position loop 4: Torque mode -3: Speed loop(Immediate velocity mode) -4: Pulse mode 6: Homing mode 7: Motion interpolation based on CANopen

Absolute/ relative position control selection	20200F10	0CF0	RWSL	uint 16	HEX	002F	When the "Drive Enable" function is configured to Din and the corresponding Din valid input is 1, the "Control word" (6040.00) is set to this value; 0x2F: Absolute position control 0x4F: Relative position control
---	----------	------	------	---------	-----	------	--

11.2 Metrical data

Name	CANopen	Modbus	Command properties	Data type	Unit	Initial value	Explanation
Actual position	60630020	3700	RLTM	int 32	inc	0	Actual motor position
Actual current value	60780010	3E00	RLTM	int 16	Ap	0.00	Rated current
Input port status	60FD0020	6D00	RLTM	uint 32	HEX	0	bit0: negative limit switch bit1: positive limit switch bit2: home switch bit4: interlock
Actual speed	606C0020	3B00	RLTM	int 32	rpm	0.00	rpm



Note

0x606C0020, The unit conversion method is $DEC = [(RPM * 512 * \text{Encoder resolution}) / 1875]$

11.3 Target object (0x607A)

Name	CANopen	Modbus	Command properties	Data type	Unit	Initial value	Explanation
Velocity position direction control	607E0008	4700	RWSL	uint 8	DEC	0	Run polarity reversal 0: Counterclockwise is the positive direction 1: Clockwise is the positive direction
Target position	607A0020	4000	RWLTM	int 32	inc	0	The target position in position mode 1, if the control word is set to start motion, transitions to the valid command position inc
Trapezoidal velocity	60810020	4A00	RWLTM	uint 32	rpm	0.00	Trapezoidal velocity (rpm) in operating mode 1
Target speed	60FF0020	6F00	RWLTM	int 32	rpm	0.00	Target speed in mode 3 and -3,
Max velocity limit	60800010	4900	RWL	uint 16	rpm	5000	Default: 5000rpm
Trapezoidal acceleration	60830020	4B00	RWSLTM	uint 32	rps/s	100.00	Default: 100rps/s
Trapezoidal deceleration	60840020	4C00	RWSLTM	uint 32	rps/s	100.00	Default: 100rps/s
Target torque	60710010	3C00	RWL	int 16	%	0.00	Torque command in torque mode, the percentage of target torque to rated torque
Target current	60F60810	5880	RWLTM	int 16	Ap	0.00	Current command in torque mode
Target current limit	60730010	3D00	RWSLTM	uint 16	Ap	48.00	Current instruction maximum

**Note**

- Velocity address: 0x60810020, 0x60800010, 0x60FF0020
 - The unit conversion method : $DEC = [(rpm * 512 * \text{encoder resolution}) / 1875]$
- Acceleration and deceleration address: 0x60830020, 0x60840020
 - The unit conversion method is $DEC = [(rps/s * 65536 * \text{encoder resolution}) / 4000000]$
- Current address: 0x60710010, 0x60730010
 - The unit conversion method is $1Arms = [2048 / (I_{peak} / 1.414)] DEC$, I_{peak} (Unit:AP) is max current (6510.03). For detailed I_{peak} data, please refer to Section 1.2 (Peak current)

11.4 Multi-segment position/Multi-segment speed (0x2020)

Name	CANopen	Modbus	Command properties	Data type	Unit	Initial value	Explanation
Multi-stage position control 0	20200120	0C10	RWSL	int 32	DEC	0	Din position [x] is meaningful only if at least one of Din Position Index 0, Din Position Index 1, or Din Position Index 2 is configured to Din; x=0~7, is the BCD code composed of the three indexes; bit0: Din Location index 0 bit1: Din Location index 1 bit2: Din Location index 2 When one of the bits is configured to Din, the unconfigured bits are equal to 0
Multi-stage position control 1	20200220	0C20	RWSL	int 32	DEC	0	
Multi-stage position control 2	20200320	0C30	RWSL	int 32	DEC	0	
Multi-stage position control 3	20200420	0C40	RWSL	int 32	DEC	0	
Multi-stage Position Control 4	20201020	0D00	RWSL	int 32	DEC	0	
Multi-stage Position Control 5	20201120	0D10	RWSL	int 32	DEC	0	
Multi-stage position control 6	20201220	0D20	RWSL	int 32	DEC	0	
Multi-stage position control 7	20201320	0D30	RWSL	int 32	DEC	0	
Multi-stage speed control 0	20200520	0C50	RWSL	int 32	rpm	0.00	The speed instruction of the driver is specified by the DIN speed [x], where the x is a BCD code composed of the following signals: bit 0: Din speed index 0; bit 1: Din speed index 1; bit 2: Din Speed index 2; The situation where all the medians are 0 cannot occur;
Multi-stage speed control 1	20200620	0C60	RWSL	int 32	rpm	0.00	

Name	CANopen	Modbus	Command properties	Data type	Unit	Initial value	Explanation
Multi-stage speed control 2	20200720	0C70	RWSL	int 32	rpm	0.00	
Multi-stage speed control 3	20200820	0C80	RWSL	int 32	rpm	0.00	
Multi-stage speed control 4	20201420	0D40	RWSL	int 32	rpm	0.00	
Multi-stage speed control 5	20201520	0D50	RWSL	int 32	rpm	0.00	
Multi-stage speed control 6	20201620	0D60	RWSL	int 32	rpm	0.00	
Multi-stage speed control 7	20201720	0D70	RWSL	int 32	rpm	0.00	

11.5 Performance object (0x6065)

Name	CANopen	Modbus	Command properties	Data type	Unit	Initial value	Explanation
Max following error	60650020	3800	RWSLM	uint 32	inc	524288	Following error value alarm value Default 524288inc
target location window	60670020	3900	RWSL	uint 32	inc	327	"Target location reached"target range, Default 10inc
reach position time window	25080910	1990	RWSLTM	uint 16	ms	10	Target (location.velocity) reach time window, Determine the reach position signal together with 0x60670020
Reach speed window	60F90A20	63A0	RWSL	int 32	inc/16s	178956	The error window when the actual speed reaches the target speed or trapezoidal speed, and determines the reach speed signal together with 0x60F91C20
Zero speed output speed window	20101810	0980	RWSL	uint 16	inc/ms	3.00	Error window when actual speed is 0
Zero speed output time	60F91410	6440	RWSL	uint 16	ms	10.00	Zero-speed output speed window 0x20101810 After reaching the set range, it takes a period of time to output the zero-speed signal, and the time is determined by the zero-speed output time
Positive setting of soft limit	607D0120	4410	RWSL	int 32	DEC	0	Soft limit positive setting
Soft limit negative setting	607D0220	4420	RWSL	int 32	DEC	0	Soft limit negative setting
Limit function definition	20101908	0990	RWSL	uint 8	DEC	1	Used to set the action after the limit is reached 0: If there is a limit after finding the origin, it will alarm 1: do nothing

11.6 Homing control (0x6098)

Name	CANopen	Modbus	Command properties	Data type	Unit	Initial value	Explanation
Homing mode	60980008	4D00	RWSLM	int 8	DEC	0	Homing function Refer origin control chapter
Origin turning signal speed	60990120	5010	RWSLTM	uint 32	rpm	300.00	After touching the trigger event, the speed of homing process (rpm)
Origin signal speed	60990220	5020	RWSLTM	uint 32	rpm	100.00	Speed when starting to find the origin
Homing acceleration	609A0020	5200	RWSL	uint 32	rps/s	50.00	Acceleration when homing process unit: rps/s
Origin offset	607C0020	4100	RWSLTM	int 32	inc	0	Offset value after homing unit: inc
Origin Offset Mode	60990508	5050	RWSL	uint 8	DEC	0	Origin Offset Mode Control 0: Run to the origin offset position, the actual position is displayed as 0 1: Run to the event trigger point, after the end the actual position will become: - Origin offset

11.7 Speed loop parameter (0x60F9)

Name	CANopen	Modbus	Command properties	Data type	Unit	Initial value	Explanation
Kvp	60F90110	6310	RWSL	uint 16	DEC	4	The larger the value, the stronger the gain, but it may cause the motor howling
Kvi	60F90210	6320	RWSL	uint 16	DEC	0	The larger the value, the stronger the gain, but it may cause the motor howling
Kvi/32	60F90710	6370	RWSL	uint 16	DEC	2	Kvi's 1/32
Output filter	60F90508	6350	RWSL	uint 8	Hz	240	velocity output filter of velocity loop $BW = \text{Speed_Fb_N} * 20 + 100$ [Hz]

11.8 Position loop parameter (0x60FB)

Name	CANopen	Modbus	Command properties	Data type	Unit	Initial value	Explanation
kpp 0	60FB 0110	6810	RWSL	int 16	Hz	10.00	Scale value of the position loop
kvff	60FB0210	6820	RWSL	int 16	%	100.00	Position loop speed feedforward
kaff	60FB0310	6830	RWSL	int 16	DEC	32767	Acceleration feedforward for position loop
S-Curve Smoothing Factor	60FB0510	6850	RWSL	uint 16	DEC	1	This parameter adjusts the smoothness of the S-curve

11.9 Input/output port parameter (0x2010)

Name	CANopen	Modbus	Command properties	Data type	Unit	Initial value	Explanation
Digital input 1	20100310	0830	RWSL	uint 16	HEX	0000	Refer to the function definition below
Digital input 2	20100410	0840	RWSL	uint 16	HEX	0000	
Digital input 3	20100510	0850	RWSL	uint 16	HEX	0000	
Digital input 4	20100610	0860	RWSL	uint 16	HEX	0000	
Digital output 1	20100F10	08F0	RWSL	uint 16	HEX	0001	
Digital output 2	20101010	0900	RWSL	uint 16	HEX	0002	
Input port status	20100A10	08A0	RLTM	uint 16	HEX	0000	bit0: Din1 bit1: Din2 bit2: Din3 bit3: Din4
Output port status	20101410	0940	RLTM	uint 16	HEX	000A	bit0: Dout1 bit1: Dout2
Change the input signal polarity definition	20100110	0810	RWSL	uint 16	HEX	FFFF	0: Turn off; 1: Turn on bit0: Din1 bit1: Din2 bit2: Din3 bit3: Din4 bit4: Din5 bit5: Din6 bit6: Din7 bit7: Din8 Default 0xFF
Polarity definition of output port	20100D10	08D0	RWSLTM	uint 16	HEX	FFFF	Polarity definition of output port
Input port signal simulation	20100210	0820	RWL	uint 16	HEX	0000	bit0: Din1 bit1: Din2 bit2: Din3 bit3: Din4 bit4: Din5 bit5: Din6 bit6: Din7 bit7: Din8
Output port signal simulation	20100E10	08E0	RWLM	uint 16	HEX	0000	bit0: Dout1 bit1: Dout2 bit2: Dout3 bit3: Dout4 bit4: Dout5

**Note**

Definition of digital input function (hexadecimal)	Output port definition (hexadecimal)
0001: Enable	0001: Ready
0002: Reset errors	0002: Error
0004: Operate mode control	0004: Pos reached
0008: Kvi off	0008: Zero speed
0010: P Limit+	0010: Motor brake
0020: P Limit-	0020: Speed reached
0040: Home signal	0040: Enc Index
0080: Invert direction	0080: Speed limit
0100: Din vel index 0	0100: Driver enabled
0200: Din vel index 1	0200: Position limit
0400: Din pos index 0	0400: Home found
0800: Din pos index 1	0800: Torque reach limit
1000: Quick stop	1000: Multi Dout 0
2000: Start homing	2000: Multi Dout 1
4000: Activate command	4000: Multi Dout 2
8001: Din vel index 2	8001: STO active
8002: Din pos index 2	
8004: Multifunction 0 (for setting multi-stage electronic gear ratio)	
8008: Multifunction 1	
8010: Multifunction 2	
8020: Gain switch 0	
8040: Gain switch 1	
8080: MaxCur switch	
8100: Motor error	
8200: Pre enable (The IO port must have an enable signal, otherwise it will alarm, which is used in some occasions where the machine needs to be determined before running the machine)	
8400: Fast capture 1	
8800: Fast capture 2	

11.10 Pulse input parameters (0x2508)

Name	CANopen	Modbus	Command properties	Data type	Unit	Initial value	Explanation
Electronic gear molecule 0	25080110	0x1910	RWSLTM	int 16	DEC	1000	Electronic gear molecule 0
Electronic gear denominator 0	25080210	0x1920	RWSLTM	uint 16	DEC	1000	Electronic gear denominator 0
Pulse mode control	25080308	0x1930	RWSLB	uint 8	DEC	1	0: Double pulse mode 1: Pulse direction mode 2: Incremental encoder mode 10: 422 Double pulse mode 11: 422 pulse direction mode 12: 422 Incremental encoder mode (After the data change takes effect, the storage control parameters must be restarted.)
Number of input pulses before electronic gear	25080420	0x1940	RWLTM	int 32	DEC	0	Number of input pulses before electronic gear

Number of input pulses after electronic gear	25080520	0x1950	RWL	int 32	DEC	0	Number of input pulses after electronic gear
Pulse filter parameters	25080610	0x1960	RWSL	uint 16	DEC	3	Pulse filter parameters
Pulse frequency before gear	25080C10	0x19C0	RLTM	int 16	DEC	0	Pulse frequency before gear (pulse/mS)
Pulse frequency after gear	25080D10	0x19D0	RLT	int 16	DEC	0	Pulse frequency after gear (pulse/mS)

11.11 Parameter for storage (0x2FF0)

Name	CANopen	Modbus	Command properties	Data type	Unit	Initial value	Explanation
Storage control parameter	2FF00108	2910	RWL	uint 8	DEC	0	1: Store all configuration parameters set 10: Initialize all configuration parameters Note: The control loop parameters are stored, excluding motor parameters.
Store motor parameter	2FF00308	2930	RWL	uint 8	DEC	0	1: Store all motor parameters set

11.12 Error code (0x2601)

Name	CANopen	Modbus	Command properties	Data type	Unit	Initial value	Explanation
Error state	26010010	1F00	RLTM	uint 16	HEX	0408	Real-time alarm error status bit 0: Extended Error bit 1: Encoder ABZ/not connected bit 2: Encoder UVW/ Encoder internal bit 3: Encoder Counting/Encoder CRC bit 4: Driver temperature bit 5: Over voltage bit 6: Under voltage bit 7: Over current bit 8: Chop Resistor bit 9: Position Following bit 10: Low logic voltage bit 11: Motor or Driver Ilt bit 12: Over frequency bit 13: Motor temperature bit 14: Motor commutation bit 15: EEPROM data
Error state 2	260200	2000	RLTM	uint 16	HEX	0000	Error state 2 bit 0: Current Sensor bit 1: Watch dog bit 2: Wrong interrupt bit 3: MCU ID bit 4: Motor configuration Bits 5 to 7: reserved bit 8: External enable bit 9: Positive limit bit 10: Negative limit bit 11: SPI internal bit 12: Abort connection bit 13: Closed loop direction bit 14: Master ABZ bit 15: Master counting

11.13 Stop mode

Name	CANopen	Modbus	Command properties	Data type	Unit	Initial value	Explanation
Quick stop mode	605A0010	3400	RWSL	int 16	DEC	0	Encountered limit switch, emergency stop switch, or control word is 0x000B 0: stop without control 1: stop by using ramp, then switch off 2: stop by using quick stop deceleration, then switch off 5: stop with profile deceleration, stay in quick stop active 6: stop with quick stop deceleration, stay in quick stop active 18: using motor winding for brake even encoder is wrong
Shutdown mode	605B0010	3410	RWSL	int 16	DEC	0	Shutdown stop mode (Drive enable cancel) 0: stop without control 1: stop by using ramp, then switch off 2: stop by using quick stop deceleration, then switch off 18: using motor winding for brake even encoder is wrong
Disable stop mode	605C0010	3420	RWSL	int 16	DEC	0	0: stop without control 1: stop by using ramp, then switch off 2: stop by using quick stop deceleration, then switch off 18: using motor winding for brake even encoder is wrong
Pause mode	605D0010	3430	RWSL	int 16	DEC	1	Control word bit8 is set to 1 The motor is paused and enabled 1: stop by current ramp 2: stop by quick stop deceleration
Error stop mode	605E0010	3440	RWSL	int 16	DEC	0	0: stop without control 1: stop by using ramp, then switch off 2: stop by using quick stop deceleration, then switch off 18: using motor winding for brake even encoder is wrong
Trapezoidal deceleration	60840020	4C00	RWSLTM	uint 32	rps/s	100.00	Working Mode Deceleration in modes 1 and 3
Quick stop deceleration	60850020	3300	RWSL	uint 32	rps/s	610.00	Deceleration for quick stop

11.14 Alarm parameter

Name	CANopen	Modbus	Command properties	Data type	Unit	Initial value	Explanation
Temp_Motor_Ref	64101810	7180	RWSL	int 16	°C	120	The default alarm point for the temperature is 120 °C
Temp_Motor	64101910	7190	RLT	int 16	°C	-	When the motor is not connected to the temperature sensor or the temperature sensor is abnormal, the motor temperature will be displayed at -40°C
Voltage_Under	65100710	8670	RWHP	uint 16	V	18	Drive low voltage alarm point
Chop_Voltage	65100810	8680	RWHP	Unsigned 16	V	63	Driver chopper voltage point
Voltage_Over	65100910	8690	RWHP	Unsigned 16	V	70	Drive overvoltage alarm point

11.15 Other parameter

Name	CANopen	Modbus	Command properties	Data type	Unit	Initial value	Explanation
Use the internal motor library	30410608	-	RWSL	uint 8	DEC	2	0: The driver automatically reads the motor model from the encoder, using the corresponding motor parameters in the driver's motor library 1: The user enters the motor model manually and uses the motor parameters corresponding to the motor library in the drive 2: Read and use the saved motor parameters in the encoder (factory default)
Device restart	2FFF0010	2F00	RW	uint 16	HEX	-	Start bootloader 55AA: start bootloader AA55: restart drive
Analog increment	64101F10	-	RWSL	int 16	DEC		>0: when the encoder is communication type, the simulated resolution of incremental encoder supports 4000, 8000, 10000 incremental encoder data. 0: Disable the simulated incremental encoder, and the feedback resolution refer to motor' default. Note: After setting the analog increment, click "Store motor parameters" to save, and it will take effect after restarting. Because the resolution has changed, you must re-initialize the control parameters and save them again for the restart to take effect

Chapter 12 Alarm exclusion

When the drive alarms, the red ERR light of the drive will be steady on.

When an alarm occurs for the product, it is necessary to investigate the fault and identify the cause of the alarm. Do not directly perform a forced reset on it. Directly forcing a reset on the product is not allowed. Any losses caused by the customer's forced reset shall be excluded from our responsibility!

If you want to query more detailed error information and error history, please use the RS485 serial port communication cable (or usb debugging cable) to connect the drive to the upper computer to view.

Table 12-1 Error status word 1 alarm code

Alarm code	Alarm information	Reason	Troubleshooting
FFF.F	The motor is not configured or the motor is incorrectly configured	1. The driver is not equipped with a motor. 2. The current motor model is different from the motor model saved by the driver	1. the driver factory parameters do not include motor configuration, connect the motor encoder line can automatically identify the motor parameters. 2. Connect the upper computer software to check whether the current motor model matches the motor code on the nameplate; When the current motor model is different from the actual connected motor, please re-identify the motor parameters.
000.1	Extended Error	Error status word 2 alarm	Look at the error status word 2 and check the meaning of the error against the alarm code.
000.2	Encoder communication failure	Encoder cable error or not connected; The presence of a 400.0 alarm indicates that the communication encoder is not responding	Step 1: Make sure that the encoder between the driver and the motor is properly connected and the cable is firmly connected. Step 2: Check whether the current motor model matches the motor code in the nameplate through the upper computer software. If the current motor model is different from the actual motor, please self-identify the motor parameters.
000.4	Encoder internal fault	Multi-turn absolute encoder multi-turn data is invalid, need to reset	Step 1: Use the host software to modify the communication encoder data reset (0x269000) to 10. Step 2: Reset the faulty or restart the drive. If the error persists, check whether the battery cable is securely connected, and try to replace the battery cable or compare the motor.
000.8	Encoder CRC error	Motor model setting error; Encoder wiring error or external interference caused;	1, Whether the motor configuration is correct Step 1: Check whether the current motor code is consistent with the motor code in the connected motor nameplate through the upper computer software. Step 2: Make sure that the encoder between the driver and the motor is properly connected and the cable is firmly connected. 2, check whether the driver grounding wire is well connected (different from the motor PE wire) 3. Check whether the grounding cable of the entire device is well connected. 4. Use an independent power supply to power the drive.

001.0	Driver overtemperature	The temperature of the drive power module reaches the alarm value	Step 1: Check whether the motor and drive power meet the requirements. Step 2: Restart the drive and check whether the cooling fan can be started during the power-on process. Check that the heat dissipation hole in the driver housing is blocked. Step 3: When the ambient temperature exceeds 40°, take heat dissipation measures or derate the power cabinet. Step 4: Increase the drive installation distance appropriately Step 5: The internal power circuit of the driver is damaged. Replace the driver.
002.0	Overvoltage	The DC bus voltage exceeds the overvoltage alarm point	Step 1: Check whether the power supply voltage is within the driver input voltage range and whether the power supply voltage is stable. Step 2: The energy generated in high-speed braking occasions will be fed back to the driver bus capacitor, and it can be considered to connect a suitable brake resistance to absorb and consume excess energy through the brake resistance. Step 3: If the brake resistance value is too large, the bus energy leakage is not timely, and the brake resistance value should be reduced. Note: iSMD brake resistors recommended range please see section 2.2
004.0	Undervoltage	The DC bus voltage is lower than the low voltage alarm point	Step 1: Check whether the power supply voltage is within the driver input voltage range and whether the power supply voltage is stable. Step 2: Power on the power supply and then the logic power supply, and ensure that the drive is connected to the power supply before enabling the power supply.
008.0	Overcurrent	Instantaneous current exceeds the overcurrent protection value.	Step 1: Check whether the motor configuration parameters are consistent with the motor drawing. Step 2: Check whether there is a short circuit between the power line UVW, and try to replace the power cable/motor/driver for comparison. Step 3: Interference causes an overcurrent alarm. For details
010.0	Chop Resistor	External brake resistance is overloaded	Step 1: Check whether the external brake resistance value and brake resistance power are set correctly on the upper computer. Step 2: The brake resistor power is insufficient, replace the brake resistor with a higher power.
		Internal brake resistance overtemperature	The power of the brake resistance inside the driver is insufficient. Please disconnect the short cable between DC+/RB1 and RB-end, and connect the appropriate external brake resistance between DC+/RB1 and RB-end.
		Internal brake unit damaged, brake circuit short circuit	After the driver is powered off, use a multimeter to measure the DC- and RB-ends. If the driver works properly, replace the driver.
020.0	Following Error	The actual following error exceeds the set maximum following error value. The possible reasons are: 1. The motor cable is improperly connected 2. The control ring is too rigid 3. The maximum following error is set too small 4. Target speed exceeds maximum speed limit 5. The target torque limit is too small 6. The lock is not opened 7. Mechanical plug/friction	Step 1: Make sure the power cable UVW is connected correctly Step 2: Readjust the gain and increase the rigidity. Step 3: Readjust the maximum follow error (0x606500) Step 4: Readjust the maximum speed limit (0x607f00) Step 5: Readjust the target current limit (0x607300), the default value of the target current limit is equal to the maximum motor current (0x64100B) Step 6: Measure whether the lock line connection is normal, measure the lock voltage, replace the motor and compare the test Step 7: Remove mechanical jams and apply lubricant

040.0	Low logic voltage	The logic voltage is lower than the lower alarm limit	Check whether the logic voltage is within the driver input voltage range and the power supply voltage is stable
080.0	Motor or drive IIT error	The motor or drive is overloaded for a long time, the possible reasons are: 1. Motor power line and encoder wiring error 2. When the motor shaft is rotating, the lock is not loosened 3. The drive control ring parameters are incorrectly set 4. The mechanical device is stuck or the friction is too large 5. Driver/motor selection error can not meet the application requirements	Step 1: Check that the encoder/power line is properly connected Step 2: Measure whether the brake line connection is normal, measure the lock voltage, replace the motor and compare the test Step 3: Adjust the parameters of the control ring Step 4: Eliminate mechanical jams and apply lubricant Step 5: Run with reduced load or replace with more powerful products
100.0	Over frequency	External input pulse frequency is too high	1 Reduce the pulse frequency 2 When using more than 600KHz pulse, need to increase the pulse frequency control 0x250808 value.
200.0	Motor temperature	The motor temperature exceeds the specified value.	User-defined alarm
400.0	Encoder information error	Encoder cable error or not connected; The presence of a 000.2 alarm indicates that the communication encoder is not responding	Step 1: Make sure that the encoder between the driver and the motor is properly connected and the cable is firmly connected. Step 2: Check whether the current motor model matches the motor code in the nameplate through the upper computer software. If the current motor model is different from the actual connected motor, please set the correct motor parameters.
800.0	EEPROM error	Data is damaged when the power is turned on and data is read from the EEPROM.	Step 1: After initializing the control ring parameters, save the control parameters and restart the drive Step 2: Import the cdi file through the host software

Table 12-2 Error status word 2 Alarm code

Alarm code	Alarm information	Reason	Troubleshooting
000.1	Current sensor	Current sensor signal offset or ripple too big	Step 1: The current sensor generates an alarm due to external interference. Step 2: Current sensor circuit is damaged, replace the driver and compare.
000.2	Watchdog	Software watchdog exception	After initializing the control ring parameters, save the control parameters and restart the drive.
000.4	Wrong interrupt	Invalid interrupt exception	After initializing the control ring parameters, save the control parameters and restart the drive.
000.8	MCU fault	1. The software program does not match the hardware 2. An MCU error is detected	Step 1: Check the software version in the drive properties and update the correct software Step 2: Replace the drive and compare.
001.0	Motor misconfiguration	1. Motor model cannot be automatically identified EEPROM has no motor data or the motor is not configured correctly 2. The motor line is missing, and a phase of the power line UVW is not connected	Step 1: Connect the host computer software to check whether the current motor model matches the motor code on the nameplate; If the current motor model is different from the actual connected motor, reset the correct motor parameters. Step 2: Make sure the motor power line UVW is connected correctly and reliably Step 3: Replace the power cable or driver comparison test.
010.0	External enable	DIN function "pre_enable" is configured, but the input is inactive when the controller is enabled or should become enabled	User-defined alarm
020.0	Positive limit	Positive position limit (after homing), position limit only causes error when Limit_Function (0x201019) is set to 0.	User-defined alarm

040.0	Negative limit	Negative position limit (after homing), position limit only causes error when Limit_Function (0x201019) is set to 0.	User-defined alarm
080.0	SPI fault	Internal firmware error in SPI handling	Please contact the supplier.
100.0	CAN bus fault	This function is enabled only when the Communication interruption mode (0x600700) is set to 1	User-defined alarm
200.0	Closed loop direction	In full closed loop mode, the counting direction of the main encoder is opposite to that of the motor encoder	Step 1: Check the count direction of the main encoder and the count direction of the motor encoder, the count direction is inconsistent, please change the count direction of the main encoder (0x250A03) Step 2: The master encoder speed is not proportional to the motor encoder speed. Check the mechanical installation, exclude mechanical slippage or jammed, check the full closed spindle check (0x250A05) and full closed slave shaft check (0x250A06) and full closed loop ratio check (0x250A09) Settings.
400.0	The main encoder ABZ fault	Master encoder connection error	Check that the main encoder signal line is connected correctly
800.0	Master counting error	Master encoder index signal is abnormal	Step 1: Correctly fill in the main encoder period (0x250A01) and set it to 0 to disable check. Step 2: Check for interference

Table 12-3 Performance anomalies and solutions

Anomalies	Alarm reason	Solutions
Motor no load operation abnormal sound or vibration	1. Wiring error 2. Improper setting of control loop parameters 3. Improper installation of motor oil seal or motor failure	Use standard power cables, encoder cables, and control cables to ensure that the cables are properly connected and are not loose or damaged. Initialize Control parameters - The storage control parameters restart before trying to run. Hand twist the motor bearing to check whether the motor is abnormal and try to reinstall the motor oil seal.
Incorrect positioning	1. Improper control loop parameters 2. The pulse data in front of the gear is inconsistent with the number of pulses delivered by the controller 3. The electronic gear ratio is set too large 4. Cumulative error of reciprocating motion 5. Mechanical system factors	Adjust drive control ring parameters Use shielded twisted pair cables for pulse cables. If the cables are long, increase the diameter of the control cables appropriately. The pulse line is as far away from the power cord as possible, and the driver is well grounded to the motor. Resume factory gear ratio operation. The machine is returned to the original where the process permits, and the origin is searched before the accumulated error exceeds the allowable. Check whether the coupling equipment is firmly installed, the pulley or gear is well fastened, the load inertia is too large, and try to reduce the load or replace a more powerful motor.

Chapter 13 Communication

13.1 RS485 communication

13.1.1 RS485 cable connection

iSMD RS485 port supports the 232 protocol and Modbus protocol, which can be used to modify the internal parameters of the servo and monitor the servo state. The default baud rate of iSMD is 38400 and Modbus is used by default.



Note

1. The device station ID and baud rate take effect only after the drive is restarted.
2. When the communication is unstable when connecting the host computer with the baud rate, reduce the communication baud rate and reconnect the software for debugging.
3. The driver has added the function of switching the 232 protocol through modbus, the internal address is 6510 0C, and the corresponding modbus address is mapped to 0X86C0, when the 485 interface uses the Modbus protocol, it can be immediately switched to the 232 protocol by sending the command (Take ID = 1 as an example): 01 06 86 C0 00 01 61 7E can be switched to the host computer immediately, or you can send 01 06 86 C0 00 03 E0 BF (232 protocol, effective after saving and restarting) 01 06 29 10 00 01 41 93 (store control parameters), and then restart the machine to connect with the host computer.
4. If FD1X5 fail to connect to the host computer using the RS485 communication protocol, it may be caused by MODBUS being selected as the communication protocol. You can send MODBUS instructions according to the third point above to replace the MODBUS protocol with 232 protocol and then reconnect to the host computer.

13.1.2 RS485 communication parameters list

Object index	Name	Meaning	Default
100B0008	Device station number	Driver station number	1
2FE20010	RS485 baud rate	Set the baud rate of the RS485 port (Modbus address: 0X2600) Set value Baud rate 1080———9600 540———19200 270———38400 90———115200 Note: Need to save and restart.	270
65100C08	RS485 Communication protocol selection	bit0:0: Using MODBUS protocol 1: using RS232 communication protocol bit1:0: Changes to bit0 take effect immediately. 1: Changes to bit0 take effect after the storage restarts	0
65100E10	RS485 mode	Data bit =8, stop bit =1, no parity check	Fixed value

65100B08	RS232 class communication	RS232 level communication control (Modbus address: 0X86B0) 0: 1 to 1 communication 1: Cascade communication	0
----------	---------------------------	---	---

13.1.3 MODBUS RTU protocol

iSMD support the MODBUS RTU communication protocol, and their internal objects are discontinuous 16-bit data registers (mapped to 4X when read and written by the upper computer). The format of the message is as follows:

Target station	Function code	Data	CRC check code
1 Byte	1 Byte	N Byte	2 Byte

Modbus common function codes introduction

Function code 0x03: Read data register

Request format:

Target station	Function code	Modbus address		Read byte		CRC
		High Byte	Low byte	High Byte	Low byte	
1 Byte	03	1 Byte	1 Byte	1 Byte	1 Byte	2 Byte

Correct response:

Target station	Function code	Return byte	Register data			CRC
			High Byte	Low byte		
1 Byte	03	1 Byte	1 Byte	1 Byte		2 Byte



Note

If a response error such as the address does not exist, the returned function code is 0x81

Example: Send packet 01 03 32 00 00 02 CA B3

Message meaning:

01——ID number

03——Function code, Read data register

32 00——The modbus address of the servo writable object "Statusword" 60410010;

00 02——The number of word data to be read ;

CA B3——Check code

Function code 0x06: Write single data register

Request format:

Target station	Function code	Modbus address		Revise content		CRC
		High Byte	Low byte	High Byte	Low byte	
1 Byte	06	1 Byte	1 Byte	1 Byte	1 Byte	2 Byte

Response format: If the setting is successful, the original text will be returned.



Note

If the written data is out of range, the address does not exist, and the response to read-only data operations is wrong, the returned function code is 0x86.

Function code 0x10: Write multi-holding register

Request format:

Target station	Function code	Modbus address	Data length (word)		Number of bytes of data written (byte)	Low level data		High level data		CRC
			High Byte	Low byte		High Byte	Low byte	High Byte	Low byte	
1 Byte	10	2Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	2Byte

Correct response:

Target station	Function code	Modbus address	Data length (word)		CRC
			High Byte	Low byte	
1 Byte	10	2Byte	1Byte	1Byte	2Byte

**Note**

If the written data is out of range, the address does not exist, or the response to read-only data operations is incorrect, the returned function code is 0x90.

Example:

Send packet 01 10 6F 00 00 02 04 F5 C3 00 28 D9 B3

Message meaning:

01——ID number

10——Function code, write multiple WORD

6F 00——The modbus address of the servo writable object "target speed" 60FF0020, the data length is 2 WORD;

00 02——write 2 WORD

04——Data length is 4 Byte (2 WORD);

F5 C3 00 28——Write data in hexadecimal 0028F5C3, decimal 2684355, converted to 150 RPM;

D9 B3——Check code

The following is an example of sending packets in different modes. The station number is 1.

Table 13-1 Message format

Object index	Modbus address	Name	Note	Message (ID=1)
60600008	3500	Operate mode	Operate mode is 3	01 06 35 00 00 03 C6 07
60FF0020	6F00	Target speed	Speed 150RPM	01 10 6F 00 00 02 04 F5 C3 00 28 D9 B3
60400010	3100	Control word	Enable to write F	01 06 31 00 00 0F C7 32
60410010	3200	Statu word	Read drive status	01 03 32 00 00 02 CA B3
Homing control mode (control word first F then 1F)				
Object index	Modbus address	Name	Value	Message (ID=1)
60400010	3100	Control word	F	01 06 31 00 00 0F C7 32
60600008	3500	Operate mode	6	01 06 35 00 00 06 06 04
60980008	4D00	Homing mode	33	01 06 4D 00 00 21 5E BE
60990120	5010	Homing transition signal speed	200RPM	01 10 50 10 00 02 04 9D 03 00 36 51 1A
60990220	5020	Homing signal speed	150RPM	01 10 50 20 00 02 04 F5 C3 00 28 CE 5A
60400010	3100	Control word	1F	01 06 31 00 00 1F C6 FE
01 03 32 00 00 02 CA B3 Read the status word. C037 indicates that the origin is found (bit15 set to 1)				
Position control mode (control word absolute positioning first 2F then 3F relative positioning first 4F then 5F, 103F update immediately)				
Object index	Modbus address	Name	Value	Message (ID=1)
60400010	3100	Control word	F	01 06 31 00 00 0F C7 32
60600008	3500	Operate mode	1	01 06 35 00 00 01 47 C6
607A0020	4000	Target position	50000inc	01 10 40 00 00 02 04 C3 50 00 00 FE 39

60810020	4A00	Trapezoid speed	30RPM	01 10 4A 00 00 02 04 31 27 00 08 03 FD
60830020	4B00	Trapezoidal acceleration	610.352rps/s	Use default value
60840020	4C00	Trapezoidal deceleration	610.352rps/s	Use default value
60400010	3100	Control word	2F	01 06 31 00 00 2F C6 EA
		Variable name	3F(Absolute positioning)	01 06 31 00 00 3F C7 26
		Control word	4F	01 06 31 00 00 4F C6 C2
		Operate mode	5F(Relative positioning)	01 06 31 00 00 5F C7 0E
01 03 32 00 00 02 CA B3 Reads the status word. D437 indicates the location recah(bit10 set to 1)				
Speed control mode				
Object index	Modbus address	Name	Value	Message (ID=1)
60600008	3500	Operate mode	3	01 06 35 00 00 03 C6 07
60FF0020	6F00	Target position	30RPM	01 10 6F 00 00 02 04 31 27 00 08 A5 6C
60400010	3100	Control word	F	01 06 31 00 00 0F C7 32
60830020	4B00	Trapezoidal acceleration	100rps/s	Use default value
60840020	4C00	Trapezoidal deceleration	100rps/s	Use default value

**Note**

In communication mode, data is transmitted in hexadecimal format.

Table 13-2 Unit conversion relation

Name	Engineering unit	Internal unit	Conversion relation
Speed	rpm	DEC	$DEC = [(RPM * 512 * \text{Encoder resolution}) / 1875]$
Acceleration	r/s ²	DEC	$DEC = [(R/S^2 * 65536 * \text{Encoder resolution}) / 4000000]$
Current	A	DEC	$1 Arms = [2048 / (I_{peak} / 1.414)] DEC$
Note: The encoder resolution is 65536, and the driver Ipeak is shown in section 1.2			

13.1.4 RS232 communication protocol**RS232 transport protocol**

iSMD RS-232C communication strictly follow the master/slave station protocol, the host computer can send any data to the driver. The drive is set to an address that calculates this data and replies with a return value.

The transport protocol used by RS232 uses a fixed ten-byte packet format:

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Byte 9
Driver ID	Data							CHKS	

Host send:

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Byte 9
Driver ID	Host send data							CHKS	

Servo send/host receive:

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Byte 9
Driver ID	Host received data							CHKS	

Key points:

The calculation process of the check code CHKS takes 01 40 F0 2F 09 00 00 00 00 97 (this command is to read the target speed of the slave station 2FF00910) as an example, the last 0x97 is the check code, which is calculated by LRC check (vertical redundancy check).

Here recommend a check code online computing website: <http://www.ip33.com/lrc.html>

**Note**

- There is a separate CHKS for every ten bytes
- If the host sends the server drive an address that does not exist on the network, the server drive will not respond.
- When the host sends a correct data, the slave station will look for the corresponding address of the slave station data and check the parity value, if the parity value does not match, the slave station will not respond.

RS232 data protocol

The data protocol is different from the transmission protocol, and its content refers to the 8 data bytes of the above 10 bytes above. iSMD internal data definition is compliant with Canopen international standards. Numerical and functions are expressed through indexes and sub-indexes.

A: Downloading (From master to slave)

Download means that the master sends a command to write a value to the slave object. If downloaded to a non-existent object, the master will generate an error.

Master station sends data format

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Function code	Index		Subindex	Data			

Function code : Specifies the direction and size of the data transfer

23(0x16): Send 4 bytes of data (bytes 4... 7 Contains 32 bits)

2B(0x16): Send 2 bytes of data (bytes 4... 5 Contains 16 bits)

2F(0x16): Send 1 byte of data (bytes 4 contains 8 bits)

Index: The address of the sending object, 16 bits

Subindex: Subaddress of the sending object, 8 bits

Data : The data to be sent (the order of the four bytes in the data is high byte after, low byte before)

For example:

Write 600rpm to the "target speed" of the slave station, the internal unit of 2FF0.09 is rpm, 600 is decimal, and the hexadecimal representation is 258. Since the length of the object to be written is 4 bytes, but the result is only two bytes, it should be filled with 0 at the high level, so the final result =00 00 02 58, the byte order sent is 58 02 00 00

Slave response data format

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Function code	Index		Subindex	Reserve			

Function code : Specifies the direction and size of the data transfer

60(0x16): Data sent successfully

80(0x16): Error, by byte 4... 7 generation

Index: The address of the sending object, 16 bits, Same as the main station

Subindex: Subaddress of the sending object, 8 bits, Same as the main station

Reserve: Reserved

B: Upload (From slave to master)

Upload means that the master sends a command to read the address of the slave object. Uploading an address that does not exist will result in an error for the master.

Master station sends data format

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Function code	Index		Subindex	Reserve			

Function code: Define the direction of the data transfer

40(0x16) : Read data

Index: The address of the sending object, 16 bits

Subindex: Subaddress of the sending object, 8 bits

Reserve: Byte 4... 7 no use

Slave receive data

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Function code	Index		Subindex	Data			

Function code: Display slave response

43(0x16): Bytes 4... 7 contains 32 bits

4B(0x16): Bytes 4... 5 contains 16 bits

4F(0x16): Bytes 4 contains 8 bits

80(0x16): Error, by byte 4... 7 generation

Index: The address of the sending object, 16 bits, Same as the main station

Subindex: Subaddress of the sending object, 8 bits, Same as the main station

Data: Data returned from the slave station

If there is no error in the data, byte 4... 7 saves the value of the slave object, low byte in front, high byte in back.

If there is an error in the data, the four bytes of data are not equal to the value of the slave object.

For example:

The master sends an "upload" command to the slave:

01 40 F0 2F 09 00 00 00 00 97 (This command is to read the target speed of the slave station 2FF00910)

Slave station response:

01 4B F0 2F 09 58 02 00 00 32

Description:

01: The slave station number is 1

4B: 2 bytes are received, saved by byte4 and byte5 in the 10 bytes of the response

F0 2F: The object index address is 2FF0

09: The address of the object subindex is 09

58 02 00 00: The value is 00 00 02 58(hex)= 600 rpm

Table 13-3 Setting homing mode, position mode, and speed mode through RS232 communication

Homing mode				
Object index	Name	Value	Message(ID=1)	Note
60400010	Control word	F	Send→01 2B 40 60 00 0F 00 00 00 25 Receive←01 60 40 60 00 0F 00 00 00 F0	Home turning signal speed and Homing signal velocity are expressed in decimal DEC by default DEC=[(RPM*512*[641003])/1875]
60600008	Operating mode	6	Send→01 2F 60 60 00 06 00 00 00 0A Receive←01 60 60 60 00 06 00 00 00 D9	
60980008	Homing mode	33	Send→01 2F 98 60 00 21 00 00 00 B7 Receive←01 60 98 60 00 21 00 00 00 86	
60990120	Home turning signal speed	200RPM	Send→01 23 99 60 01 03 9D 36 00 0C Receive←01 60 99 60 01 03 9D 36 00 CF	
60990220	Homing signal velocity	100RPM	Send→01 23 99 60 02 82 4E 1B 00 F6 Receive←01 60 99 60 02 82 4E 1B 00 B9	
60400010	Controlword	1F	Send→01 2B 40 60 00 1F 00 00 00 15 Receive←01 60 40 60 00 1F 00 00 00 E0	
Send 01 40 41 60 00 00 00 00 00 1E read status word, Drive response 01 4B 41 60 00 31 C0 FF FF 24, bit15=1 indicates homing found				
Position mode				
Object index	Name	Value	Message(ID=1)	Note
60400010	Control word	2F	Send→01 2B 40 60 00 2F 00 00 00 05 Receive←01 60 40 60 00 2F 00 00 00 D0	The trapezoidal speed is expressed in decimal DEC by default DEC=[(RPM*512*[641003])/1875]
		4F	Send→01 2B 40 60 00 4F 00 00 00 E5 Receive←01 60 40 60 00 4F 00 00 00 B0	
60600008	Operating mode	1	Send→01 2F 60 60 00 01 00 00 00 0F Receive←01 60 60 60 00 01 00 00 00 DE	
607A0020	Target position	50000inc	Send→01 23 7A 60 00 50 C3 00 00 EF Receive←01 60 7A 60 00 50 C3 00 00 B2	
60810020	Trapezoidal speed	200RPM	Send→01 23 81 60 00 03 9D 36 00 25 Receive←01 60 81 60 00 03 9D 36 00 E8	Trapezoidal acceleration and trapezoidal deceleration are expressed in decimal DEC by default.
60830020	Trapezoidal acceleration	100rps/s	Send→01 23 83 60 00 06 E A3 01 00 E7 Receive←01 60 83 60 00 06 E A3 01 00 AA	
60840020	Trapezoidal deceleration	100rps/s	Send→01 23 84 60 00 06 E A3 01 00 E6 Receive←01 60 84 60 00 06 E A3 01 00 A9	

60400010	Controlword	3F	Send→01 2B 40 60 00 3F 00 00 00 F5 Receive←01 60 40 60 00 3F 00 00 00 C0	DEC=[(RPS/S*65536*[641003])/4000000] 2F→3F: absolute location 4F→5F: relative location
		5F	Send→01 2B 40 60 00 5F 00 00 00 D5 Receive←01 60 40 60 00 5F 00 00 00 A0	
Send 01 40 41 60 00 00 00 00 1E read the status word,Drive response 01 4B 41 60 00 37 C4 FF FF 1A,bit10 representsTarget reached				
Speed mode				
Object index	Name	Value	Message(ID=1)	Note
60600008	Operation mode	3	Send→01 2F 60 60 00 03 00 00 00 0D Receive←01 60 60 60 00 03 00 00 00 DC	The target speed is expressed in decimal DEC by default DEC=[(RPM*512*[641003])/1875] Trapezoidal acceleration and trapezoidal deceleration are expressed in decimal DEC by default: DEC=[(RPS/S*65536*[641003])/4000000]
60FF0020	Target speed	-100RPM	Send→01 23 FF 60 00 7E B1 E4 FF 6B Receive←01 60 FF 60 00 7E B1 E4 FF 2E	
60400010	Control word	2F	Send→01 2B 40 60 00 2F 00 00 00 05 Receive←01 60 40 60 00 2F 00 00 00 D0	
60830020	Trapezoidal acceleration	100rps/s	Send→01 23 83 60 00 6E A3 01 00 E7 Receive←01 60 83 60 00 6E A3 01 00 AA	
60840020	Trapezoidal deceleration	100rps/s	Send→01 23 84 60 00 6E A3 01 00 E6 Receive←01 60 84 60 00 6E A3 01 00 A9	
Note: The message is expressed in hexadecimal, and the motor resolution used in this case is 65536				

13.1.5 Troubleshooting measures for communication faults

When the driver cannot communicate with the host computer, please check the driver communication parameters and wiring; When the drive is prone to disconnection, read-only no write or only write no read and other problems, you can check through the following aspects:

Serial number	Check item	Description
1	Terminal resistance	When the 485 communication rate is high and the communication distance is long, the signal will be reflected at the end of the transmission line, so it is necessary to parallel a 120Ω terminal resistor at the beginning and end of the communication network. Since the FD servo drive has its own terminal resistance, it is only necessary to dial the dip switch of the first and last drive to the ON position.
2	Cable specification	The recommended wire diameter of 485 signal cable is 24AWG, and the shielded twisted pair cable should be used, and the shielding layer of the cable should be grounded together and the grounding resistance should not be greater than 1Ω. The use of twisted pair cables can effectively eliminate adversarial interference, and the cable with a good shielding layer can effectively reduce the impact of external interference sources.
3	Reasonable wiring	The distance between communication cables and strong-current cables should be at least 20cm. If the cables can be folded into metal pipes, the anti-interference capability is better. During the wiring process, the signal line and the power line intersect vertically, and parallel laying should be avoided as far as possible.
4	Good grounding	The motor power line must be a cable with a shielded network, the motor PE must be connected to the driver ground terminal, and the driver shell must be well grounded.
5	Power interference	Power interference. The instability of the power supply of the grid will also directly affect the normal use of the servo, and the external circuit of the driver can be connected by referring to Chapter 7.

13.2 CANopen communication

CANopen is the most famous and successful of the open fieldbus standards, which has been widely recognized and widely used in Europe and the United States. In 1992, the Association of Automation CAN Users and Manufacturers (CiA) was established in Germany and began to develop CANopen, an application layer protocol for automation CAN. Since then, the members of the Association have developed a series of CANopen products, which are widely used in machinery manufacturing, pharmaceuticals, food processing and other fields.

iSMD is standard CAN slave devices that strictly follow the CANopen 2.0A/B protocol and can communicate with any host computer that supports the protocol. iSMD use a strictly defined list of objects, which we call the Object Dictionary. This object dictionary is designed based on the CANopen international standard, and all objects have clear functional definitions. The Objects mentioned here are similar to the memory address we often say, some objects such as speed and position can be modified by the external controller, and some objects can only be modified by the drive itself, such as status and error messages. These objects are hexadecimal numbers. For example, the CANopen address in working mode is 0x60400010, as shown in the following table.

Table 13-4 The example object dictionaries lists

Complete CANopen address composition			Property	Meaning
Index	Subindex	Bits(Data length)		
0x6040	00	0x10	RW	Device status control word
0x6060	00	0x08	RW	Operate mode
0x607A	00	0x20	W	Target position
0x6041	00	0x10	RO/M	Device status word

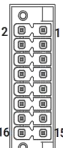
The properties of an object are as follows:

1. RW(Read and write) : Objects can be read or written;
2. RO(read-only) : The object can only be read;
3. WO (Write only) : Write only;
4. M (Mappable) : Objects can be mapped, similar to indirect addressing;
5. S (Storable) : Objects can be stored in the Flash-ROM area and are not lost when powered off.

13.2.1 Hardware description

iSMD CANopen port hardware definition is as follows:

Table 13-5 Pin name and function description table

	Pin	Signal identification	Signal name
	1&2	CAN_H	CAN_H bus (high dominant)
	3&4	CAN_L	CAN_L bus (low dominant)
	5&6	GNDC	CAN isolated ground

CAN communication protocol mainly describes the mode of information transmission between devices. The definition of CAN layer is consistent with the open system interconnection model (OSI). Each layer communicates with the same layer on another device. The actual communication occurs in the two adjacent layers of each device, and the devices are only interconnected through the physical medium of the physical layer of the model. The

specification of CAN defines the bottom two layers of the model, the data link layer and the physical layer. CAN bus physical layer is not strictly stipulated, can use a variety of physical media such as twisted pair optical fiber, etc., the most commonly used is twisted pair signal, the use of differential voltage transmission (commonly used bus transceiver), two signal lines are called CAN_H and CAN_L, static time is about 2.5V, at this time the state is expressed as logic 1, can also be called hidden bit, CAN_H is higher than CAN_L to represent logical 0, which is called display, and the usual voltage value at this time is CAN_H=3.5V and CAN_L=1.5V, and display is preferred in competition.

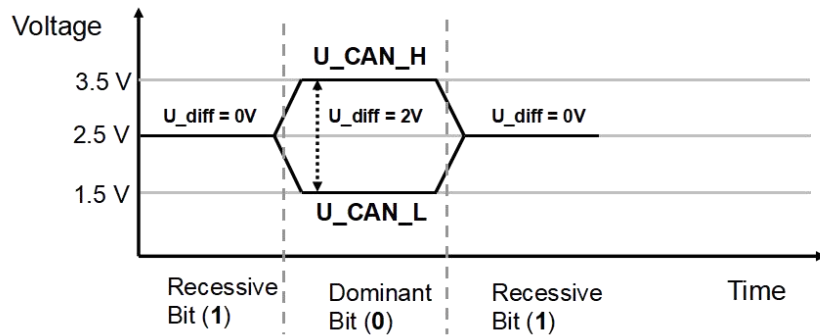


Figure 13-1 CAN Signal Identification

Note:

1. All CAN_L and CAN_H pins of the slave station can be directly connected and connected by series connection.
2. Connect the master terminal and the last slave terminal to a terminal resistor of 120 Ω .
3. Use shielded twisted pair cables for communication cables and ground them properly. (The ground cable can be disconnected for short-distance communication, but it is recommended to ground for long-distance, high-baud communication.)
4. The following table lists the theoretical maximum communication distances of different baud rates.

Table 13-6 The longest distance table that can theoretically communicate with each baud rate

Communication speed (bit/s)	Communication distance (M)
1M	25
800K	50
500K	100
250K	250
125K	500
50K	600
25K	800
10K	1000

13.2.2 Software description**13.2.2.1 EDS description**

EDS (electronic data form) file is the identification file or similar code of the slave station connected to the PLC, through which to identify the type of slave station (which is similar in 401, 402, 403, or which device belongs to 402). This file contains all the information of the slave station, such as manufacturer, serial number, software version,

supported baud rate type, OD that can be mapped and the attributes of each OD and so on, similar to the GSD file of Profibus. Therefore, before hardware configuration, we first need to import the EDS file from the station to the upper configuration software.

13.2.2.2 SDO description

SDO is mainly used to transmit low-priority objects between devices. Typically, it is used to configure and manage slave devices, such as modifying PID parameters and PDO configuration parameters of current ring, speed ring, position ring, etc. This kind of data transmission is the same as MODBUS, that is, after the master station sends out, the slave station needs to return data response. This communication mode is only suitable for parameter setting, and is not suitable for data transmission with high real-time requirements.

The communication mode of SDO is divided into upload and download, and the host computer can read and write the OD inside the servo according to the special SDO read and write instructions. In the CANopen protocol, the content of the Object dictionary can be modified through the Service Data Object (SDO). The following describes the structure of the SDO command and the guidelines to follow.

The basic structure of SDO is as follows: Client→Server/Server→Client

Byte0	Byte1-2	Byte3	Byte4-7
SDO Command specifier	Object Index	Object subindex	Maximum 4 bytes of data

SDO command word contains the following information:

- Download/upload
- Request/response
- Segmented/expedited transfer
- CAN the frame data byte length, which is used to alternately clear and set the toggle bit for each subsequent segment

There are five request/reply protocols implemented in SDO:

- Initiate domain Download
- Download Domain Segment
- Initiate Domain Upload;
- Upload Domain Segment
- Abort Domain Transfer

Among them, Download refers to write operation of the object dictionary, Upload refers to read operation of the object dictionary; Use the Initiate Domain Upload protocol when reading parameters; When setting parameters, use the Initiate Domain Download protocol; The syntax of the protocol's SDO command word (the first byte of an SDO CAN message) is described in Table 8-4 and Table 8-5, where "-" indicates that it is irrelevant and should be 0.

Table 13-7 Start domain download

Bit	7	6	5	4	3	2	1	0
Client→	0	0	1	-	n		e	s
←Server	0	0	1	-	-	-	-	-

Table 13-8 Start domain upload

Bit	7	6	5	4	3	2	1	0
Client→	0	0	1	-	-	-	-	-
←Server	0	0	1	-	n		e	s

Description:

n——Indicates the number of bytes of meaningless data in the packet data [from (8-n) bytes to the seventh byte of meaningless data] (n is valid when e=1 and s=1, otherwise n is 0);

e——Normal transmission when e=0, accelerated transmission when e=1;

s——Indicates whether the data length is specified. 0 indicates that the data length is not specified, and 1 indicates that the data length is specified.

e=0, s=0——Retained by the CiA;

e=0, s=1——The data byte is the byte counter, byte4 is the low data part (LSB), byte7 is the high data part (MSB);

e=1——ata bytes are the data to be downloaded.

SDO message format description

The transmission of SDO is an object data transmission of no more than 4 bytes, and the message consists of a COB-ID and a data segment. The following table shows that the COB-ids of the sending SDO(RSD) and the receiving SDO do not agree.

Data segments are arranged in the "low position in front, high position in back" mode. All SDO message data segments must be eight bytes. The format of the SDO transmission message is shown in the following table.

Table 13-9 SDO transmit message format table

	COB-ID	DLC	Data							
			0	1	2	3	4	5	6	7
Master →	0x600+Node_ID	8	Send command word	Object Index		Object subindex	00			
← Slave	0x580+Node_ID	8	Receive command word	Object Index		Object subindex	Maximum 4 bytes of data			

The command code specifies the transmission type and data length of the SDO, the index and subindex are the objects to be transmitted, and the data is the numerical value of the object.

SDO read message

Depending on the length of the received data, the received SDO command word changes, and the message format is as follows:

Table 13-10 SDO reads message description

		COB-ID	DLC	Data						
				0	1	2	3	4	5	6
Master →		0x600+Node_ID	8	40	Object index	Object subindex	-	-	-	-
← Slave	Normal	0x580+Node_ID	8	4F	Object index	Object subindex	Data	-	-	-
	Normal	0x580+Node_ID	8	4B	Object index	Object subindex	Data		-	-
	Normal	0x580+Node_ID	8	43	Object index	Object subindex	Data			
	Error	0x580+Node_ID	8	80	Object index	Object subindex	Error code			

The command word sent by the SDO message is 0x40 when the parameter is read. The command word received will be different depending on the length of the message data returned by the driver.

**Note**

"-" indicates that data exists but is not considered. You are advised to write 0 when writing data.

SDO write message

Using SDO to modify parameters for objects no larger than 4 bytes, the transmission message varies according to the read and write mode and the inconsistency of the content data length. SDO write messages as shown in the following table:

Table 13-11 SDO write message description

		COB-ID	DLC	Data							
				0	1	2	3		4	5	6
Master →		0x600+Node_ID	8	2F	Object index	Object subindex	Data	-	-	-	
		0x600+Node_ID	8	2B	Object index	Object subindex	Data		-	-	
		0x600+Node_ID	8	23	Object index	Object subindex	Data				
← Slave	Normal	0x580+Node_ID	8	60	Object index	Object subindex	-	-	-	-	
	Error	0x580+Node_ID	8	80	Object index	Object subindex	Error code				

When writing parameters, the command word is related to the data length of the object when the SDO message is sent. If the data to be sent is 1 byte, the command word is 0x2F. If the data to be sent is 2 bytes, the command word is 0x2B. If the data to be sent is 4 bytes, the command word is 0x23. The SDO message is successfully sent, and the received command word is 0x60. The SDO message failed to be sent with the received command word 0x80.

Example:

The slave station number is 1, the SDO is used to write the object to the target position 607A.00, and the write value is 100000, that is, 0x186A0. The master station sends a message as shown in the following table. (All in hexadecimal)

Master send SDO message

COB-ID	DLC	Data									
		0	1	2	3	4	5	6	7		
0x601	8	23	7A	60	00	A0	86	01	00		

Write normal drive receives SDO message

COB-ID	DLC	Data									
		0	1	2	3	4	5	6	7		
0x581	8	60	7A	60	00	A0	86	01	00		

If the write is abnormal, the error command word 0x80 is returned

COB-ID	DLC	Data									
		0	1	2	3	4	5	6	7		
0x581	8	80	7A	60	00	01	00	01	06		

Table 13-12 SDO message error code

Error code	Code function description
0x05040001	Invalid command, unknown or illegal Client/Server command word
0x06010001	Attempts to read only write object parameters
0x06010002	An attempt was made to write read-only object parameters

0x06020000	Invalid index. The object does not exist in the object dictionary
0x06040041	Cannot be mapped, object parameters do not support mapping to PDO
0x06060000	The drive is in an error state, causing the object parameter access failure
0x06070010	The data type and length of the service parameter do not match
0x06070012	The data type does not match, and the length of the service parameter is too large
0x06070013	The data type does not match, and the length of the service parameter is too short
0x06090011	Invalid subindex
0x06090030	Invalid data, out of range of object parameters
0x06090031	The value of the written data is too large
0x06090032	Write data value is too small
0x08000022	Data cannot be transferred or saved to the application due to the current device state

Table 13-13 Set the origin mode via SDO messages

Parameter address	Name	Value	message(ID=1)
60400010	Control word	F	Send→601 2B 40 60 00 0F 00 00 00 Receive←581 60 40 60 00 0F 00 00 00
60600008	Operate mode	6	Send→601 2F 60 60 00 06 00 00 00 Receive←581 60 60 60 00 06 00 00 00
60980008	Homing mode	33	Send→601 2F 98 60 00 21 00 00 00 Receive←581 60 98 60 00 21 00 00 00
60990120	Homing transition signal speed	200RPM	Send→601 23 99 60 01 03 9D 36 00 Receive←581 60 99 60 01 03 9D 36 00
60990220	Homing signal speed	100RPM	Send→601 23 99 60 02 82 4E 1B 00 Receive←581 60 99 60 02 82 4E 1B 00
60400010	Control word	1F	Send→601 2B 40 60 00 1F 00 00 00 Receive←581 60 40 60 00 1F 00 00 00
Send 601 40 41 60 00 00 00 00 00 Read the status word and the drive replies 581 4B 41 60 00 31 C0 FF FF, bit15=1 for origin found			

Table 13-14 Setting the position mode via SDO message

Parameter address	Name	Value	message(ID=1)
60400010	Control word	2F	Send→601 2B 40 60 00 2F 00 00 00 Receive←581 60 40 60 00 2F 00 00 00
		4F	Send→601 2B 40 60 00 4F 00 00 00 Receive←581 60 40 60 00 4F 00 00 00
60600008	Operate mode	1	Send→601 2F 60 60 00 01 00 00 00 Receive←581 60 60 60 00 01 00 00 00
607A0020	Target position	50000inc	Send→601 23 7A 60 00 50 C3 00 00 Receive←581 60 7A 60 00 50 C3 00 00
60810020	Trapezoidal speed	200RPM	Send→601 23 81 60 00 03 9D 36 00 Receive←581 60 81 60 00 03 9D 36 00
60830020	Trapezoidal acceleration	100rps/s	Send→601 23 83 60 00 6E A3 01 00 Receive←581 60 83 60 00 6E A3 01 00
60840020	Trapezoidal deceleration	100rps/s	Send→601 23 84 60 00 6E A3 01 00 Receive←581 60 84 60 00 6E A3 01 00
60400010	Control word	3F	Send→601 2B 40 60 00 3F 00 00 00 Receive←581 60 40 60 00 3F 00 00 00
		5F	Send→601 2B 40 60 00 5F 00 00 00 Receive←581 60 40 60 00 5F 00 00 00
Send 601 40 41 60 00 00 00 00 00 Read the status word and the drive replies 581 4B 41 60 00 37 C4 FF FF,bit10 representing the target location reach			

Table 13-15 Speed mode is set through SDO message

Parameter address	Name	Value	message(ID=1)
60600008	Operate mode	3	Send→601 2F 60 60 00 03 00 00 00 Receive←581 60 60 60 00 03 00 00 00
60FF0020	Target position	-100RPM	Send→601 23 FF 60 00 7E B1 E4 FF Receive←581 60 FF 60 00 7E B1 E4 FF
60400010	Control word	2F	Send→601 2B 40 60 00 2F 00 00 00 Receive←581 60 40 60 00 2F 00 00 00
60830020	Trapezoidal acceleration	100rps/s	Send→601 23 83 60 00 6E A3 01 00 Receive←581 60 83 60 00 6E A3 01 00
60840020	Trapezoidal deceleration	100rps/s	Send→601 23 84 60 00 6E A3 01 00 Receive←581 60 84 60 00 6E A3 01 00

Note: The message is expressed in hexadecimal, and the motor resolution used in this case is 65536

13.2.3 PDO description

PDO can transmit 8 bytes of data at a time, with no other protocol presetting (meaning that the data content is predefined), and is mainly used to transmit data requiring high frequency exchange.

The transmission mode of PDO breaks the existing data question-and-answer transmission concept and adopts a new data exchange mode. The two sides of the device define the data receiving and sending area in each device before transmission, and directly send the relevant data to the data receiving area of the other side during data exchange, which reduces the question-and-answer inquiry time and greatly improves the efficiency of bus communication. As a result, high bus utilization is achieved.

13.2.3.1 PDO COB-ID description

COB-ID is a unique method of CANopen Communication protocol. Its full name is Communication Object Identifier-Communication object-ID. These COB-ids define the corresponding transport levels for PDO. The controller and the servo can define the same transmission level and the transmission content in their respective software configurations, so that after the controller and the servo use the same transmission level and transmission content, the data transmission is transparent, that is, both sides know the data content to be transmitted. It is not necessary to reply whether the data is transmitted successfully when the data is transmitted.

The default ID allocation table is based on the 11-bit CAN-ID defined by CANopen 2.0A (CANopen 2.0B protocol COB-ID is 29 bits), which contains a 4-bit function code part and a 7-bit Node-ID part, as shown in the following figure .

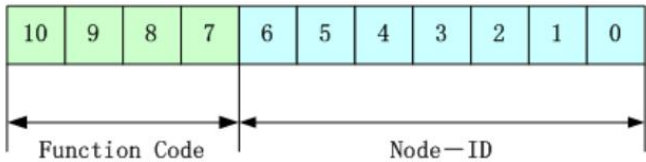


Figure 13-2 Default ID description diagram

➔

Note

Node-ID —— The station number of the servo. The Node-ID ranges from 1 to 127.

Function Code ——Function codes for data transmission, which define the transmission levels of various PDO, SDO, and management packets. The smaller the function codes, the higher the priority.

Table 13-16 CANopen predefined master/slave connection set CAN identifier assignment table

Object	COB-ID
NMT Module Control	000H
SYNC	080H
TIME SSTAMP	100H
Object	COB-ID
Emergency	081H-0FFH
PDO1 (Send)	181H-1FFH
PDO1 (Receive)	201H-27FH
PDO2 (Send)	281H-2FFH
PDO2 (Receive)	301H-37FH
PDO3 (Send)	381H-3FFH
PDO3 (Receive)	401H-47FH
PDO4 (Send)	481H-4FFH
PDO4 (Receive)	501H-57FH
SDO (Send/server)	581H-5FFH
SDO (Receive/customer)	601H-67FH
NMT Error Control	701H-77FH

**Note**

- 1、The smaller the COB-ID, the higher the priority.
- 2.The function code before the COB-ID of each level is in a fixed format.
- 3.The COB-ID 00H, 80H, 100H, 701H-77FH, and 081H-0FFH are all system management formats.

13.2.3.2 COB-ID

- Send PDO relative to the servo refers to the data sent by the servo, which is received by the PLC. The function code (COB-ID) for sending the PDO is:
 - 1.0x180+Servo station NO.
 - 2.0x280+ Servo station NO.
 - 3.0x380+ Servo station NO.
 - 4.0x480+ Servo station NO.
- Receiving PDO relative to the servo refers to the data received by the servo, which is sent by the PLC, and the function code (COB-ID) for sending PDO is:
 - 1.0x200+ Servo station NO.
 - 2.0x300+ Servo station NO.
 - 3.0x400+ Servo station NO.
 - 4.0x500+ Servo station NO.

**Note**

Since the iSMD are designed according to the standard CANopen 2.0A protocol, but also support CANopen 2.0B protocol, which means that if the above eight Pdos are not enough, you can also define a new PDO. For example, 0x43FH is used as the communication PDO of station 1, as long as the controller and the servo are defined in accordance with this.

13.2.3.3 PDO transmission type

- SYNC - Transmission triggered by a synchronization message (transmission type: 0-240)

In this transmission mode, the controller must have the ability to send synchronous messages (periodic messages with a frequency of up to 1KHZ), which the servo sends after receiving the synchronous message.

Aperiodic - pre-triggered transfer by a remote frame, or by an object-specific event specified in the device subprotocol. In this mode, the data in the PDO is sent once every time the servo drive receives a synchronization message.

Periodic - Delivery is triggered after every 1 to 240 SYNC messages. In this mode, the data in the PDO is sent once every time the servo drive receives n synchronization packets.

When CANopen communication is configured in synchronous transmission mode, the driver uploads the mapped data in TPDO only after receiving the synchronization message.

Table 13-17 TPDO configuration in synchronous mode

Name	Meaning
TPDO1 Mapping group	2, the number of objects configured in this PDO, TPDO1 is configured with two objects: actual location and status word
Mapping 1-8	Configure the servo-CANOPEN control object
TPDO1 station number	180+ Drive ID (TPDO2 station number should be set to: 280+ drive ID)
TPDO1 Transport type	Synchronous transmission mode: The driver sends TPDO to the controller after receiving synchronization packets
TPDO1 Disable time	It must be set to 0
Note: The sum of the actual position and status word length of the assigned object in TPDO1 is 4+2=6 bytes	

**Note**

- The sum of the actual position of the object and the length of the status word in TPDO1 is 4+2=6 bytes.
- The default RPDO transmission mode is 254. You do not need to set the transmission mode. The transmission mode takes effect immediately after data is received.
- The default value of synchronizing packets is

COB-ID	DLC
0x80	0

- Asynchronous (Transfer type: 254/255)

The slave station sends the packet after the change regardless of whether the master station asks for it. In addition, you can define the interval between sending the same packet twice to prevent the packets with higher

priorities from occupying the bus all the time (the lower the value of PDO, the higher the priority).

For iSMD, it supports all 256 transmission modes, and the user only needs to select the transmission mode of the drive according to the transmission mode supported by the controller.

In asynchronous transfer mode, the mapped object data in PDO is transmitted as soon as it changes.

Table 13-18 TPDO configuration in asynchronous transport mode

Name	Meaning
TPDO1 Mapping group	Indicates the number of objects configured in the PDO. TPDO1 is configured with two objects: actual location and status word
Mapping 1-8	Configure the servo-CANOPEN control object
TPDO1 station number	180+ Drive ID (TPDO2 station number should be set to: 280+ drive ID)
TPDO1 Transport type	254 or 255, asynchronous transmission mode
TPDO1 Disable time	The unit is ms. This parameter prevents the network from being blocked by frequent packets sent by the server. In multi-axis asynchronous transmission mode, set this parameter based on site requirements



Note

- The sum of the actual position of the object and the length of the status word in TPDO1 is 4+2=6 bytes.
- The default RPDO transfer mode is 254. This parameter takes effect immediately after data is received.
- A PDO can specify a forbidden time, that is, define the minimum interval between two consecutive PDO transmissions, to avoid the problem that the data of high priority information always occupies the bus because the amount of data is too large, and other low priority data cannot compete with the bus. The forbidden time is defined by a 16-bit unsigned integer, in 1ms.

● Event time reporting function

In asynchronous transfer mode, in addition to the instantaneous, the event time can be set if the driver needs to periodically upload data to the controller.

Table 13-19 Event time is used to schedule the upload in asynchronous mode

Name	Meaning
TPDO1 Mapping group	Represents the number of objects configured in the PDO. TPDO1 is configured with two objects: actual location and status word
Mapping 1-8	Configure the servo-CANOPEN control object
TPDO1 station number	180+ Drive ID (TPDO2 station number should be set to: 280+ drive ID)
TPDO1 Transport type	254 or 255, asynchronous transmission mode
TPDO1 Disable time	When uploading using event time, this object is set to 0
TPDO1 Event time	Cycle time (in ms) for the driver to send PDO to the controller
Note: The sum of the actual position and status word length of the assigned object in TPDO1 is 4+2=6 bytes	

13.2.3.4 Protection mode/monitoring type description

Monitoring type refers to the check method selected by the master station to check the slave station during operation. Through these two ways, the slave station can be judged whether there is a fault, and the corresponding treatment is made according to these faults!

1. Master heartbeat message

The slave station periodically uploads the message to the master station at the "monitoring time". If the master station does not receive the next heartbeat message from the slave station after the "heartbeat consumer time", the master station determines that the communication is wrong and the master station generates an alarm!

Table 13-20 The format of the heartbeat packet was uploaded from the slave station

COB-ID	Byte 0
0x700+Node_ID	State
Case message (slave ID=1) : 701 05	

2. Slave heartbeat message

The master station periodically sends packets to the slave station according to the "monitor time". If the slave station does not receive the next heartbeat packet from the master station after the "Heartbeat Producer time", the slave station determines that the communication is wrong! When the communication interrupt mode (0x600700 set) is 1, the driver will alarm and stop when the CAN communication fails.

Table 13-21 The format of heartbeat packets sent by the master station

COB-ID	Byte 0
0x700+master ID	Master station status
Case message (primary ID=127) : 77F 05	

Table 13-22 Status value meaning

State value	Meaning
0x00	boot-up
0x04	Stopped
0x05	Operational
0x7f	Pre-operational

When a Heartbeat node starts, its Boot-up packet is the first Heartbeat packet.



Note

The generation time of heartbeat message and the heartbeat packets of the slave station are configured by the master station when the power is turned on, and they are not saved by default when the power is off.

3. Node protection

The master station periodically sends the remote request packet to the slave station in the monitoring time. The slave station responds immediately after receiving the request packet. If the master station does not receive the response packet from the slave station after the Monitoring time x Life Factor time expires, the master station determines that the slave station is wrong. At the same time, the slave station can also monitor the remote request status of the master station and start communication protection from the first remote frame received. If the remote frame of the master station is not received beyond the time of "Node protection time * node protection factor", the slave station will also judge the communication error. The communication interrupt mode (0x600700) needs to be set to 1, and the drive will alarm and stop when CAN communication fails.

Master request message format—— (0x700+node number) (The message has no data)

Slave response message format—— (0x700+node number) +state

Table 13-23 Slave station reply message

COB-ID	Byte 0
0x700+Node_ID	Bit7: trigger bit Bit6-Bit0: status

Table 13-24 Slave station reply packet status value meaning

State value	Meaning
0	Initializing
1	Disconnected
2	Connecting
3	Preparing
4	Stopped
5	Operational
127	Pre-operational

Status-The data section includes a trigger bit (bit7), which must be alternately set to "0" or "1" in each node protection response. The trigger bit is set to 0 for the first node protection request. Bits 0 and 6 (bit0 to 6) indicate the node status. Table 13-24 describes the values.

The standard CAN slave station generally supports only one node protection mode, iSMD both support protection mode. However, a node cannot support both node protection and heartbeat message at the same time. Only one of them can be used for protection.

13.2.3.5 Startup process description

CANopen supports both extended boot-up and minimal boot-up procedures during network initialization. The initialization process can be represented by a node state transition diagram, as shown in the following figure

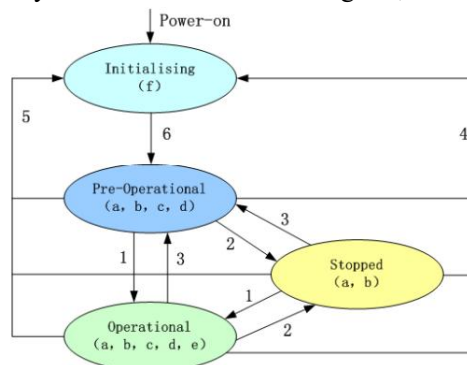


Figure 13-3 Node state transition diagram

Note: The letters in parentheses in the figure indicate communication objects that can be used in different states.

Including : a——NMT

d——Emergency

b——Node Guard

e——PDO

c——SDO

f——Boot-up

Switching between modes can be achieved through NMT management messages, only the NMT-Master node can send NMT Module Control messages, all slave devices must support the NMT Module Control service, and

NMT Module control messages do not need to be answered. After the initialization is complete, the device automatically enters the Pre_Operational state and sends Boot-up messages. NMT message format is as follows:
Nmt-master →NMT Slave(s)

Table 13-25 NMT manages the packet format

COB-ID	Byte0	Byte1
0x000	CS	Node-ID

Node-ID=0 indicates that all NMT slave devices are addressed. CS is the command word. Table 13-26 lists the values of CS.

Table 13-26 CS value table

Command word	NMT service
0x01	Enable remote node
0x02	Close remote node
0x80	Enter the pre-operation state
0x81	Reset node
0x82	Reset communication

13.2.3.6 Emergency message description

When a fatal error occurs inside the device, the application device sends an emergency packet with the highest priority to other devices. An emergency message consists of eight bytes.

Table 13-27 Emergency message format

COB-ID	Byte 0-1	Byte2	Byte4-5	Byte6-7
Emergency message station number 0x101400	Emergency error code 0x603F00	Error register 0x100100	error state 0x260100	error state 0x260200

Table 13-28 Emergency error code 0x603F00

Alarm content	Emergency Error Code (Hex)	Alarm content	Emergency Error Code (Hex)
Communication encoder is not connected	0x7331	Current sensor fault	0x5210
Communication encoder multi-turn error	0x7320	Software watchdog reset	0x6010
Communication encoder check error	0x7330	Abnormal interrupt	0x6011
Driver temperature is too high	0x4210	MCU fault	0x7400
The driver bus voltage is too high	0x3210	Motor model configure error	0x6320
The driver bus voltage is too low	0x3220	Motor power line out of phase	0x6321
Driver power part short circuit or motor short circuit	0x2320	Pre-enable alarm	0x5443
Current sampling saturation	0x2321	Positive limit error	0x5442
Driver brake resistance is abnormal	0x7110	Negative limit error	0x5441
Actual following error exceeds allowable	0x8611	SPI fault	0x6012
Logic low voltage	0x5112	Bus communication error	0x8100
Motor or drive is overloaded	0x2350	Bus communication timeout	0x81FF

Input pulse frequency is too high	0x8A80	Full closed loop check error	0x8A81
Excessive motor temperature	0x4310	Main encoder ABZ error	0x7382
Communication encoder is not responde	0x7331	Main encoder count error	0x7306
EEPROM data error	0x6310		

Table 13-29 Error register

Bit	Error type
0	Common fault
1	Current
2	Voltage
3	Temperature
4	Communication error
5	Device profile specific
6	Encoder
7	Reserve

13.2.4 CANopen Bus communication setting

This chapter will introduce the setting of CAN bus communication parameters. In the upper computer software interface, When the master station with the network management function is powered on, the parameters of the slave station are initialized by sending SDO. Generally, parameters such as synchronization ID, node protection time, node protection time coefficient, node protection station number, emergency message station number, and heartbeat message generation time do not need to be set by the user.

Table 13-30 CANopen relate parameter

CANopen address	Name	Meaning	Default
10050020	Synchronization ID	The transmission type ranges from 1 to 240. This parameter is available in synchronous mode but does not need to be set in asynchronous mode.	80
100C0010	Node protection time	Through node protection, the master station can monitor the current status of each node. The master station sends a remote frame (the default COBID is 0x700+ station number and contains no message) to query the status of the slave node according to the node protection period. The slave node needs to respond within a certain time range, otherwise the master node considers the slave node offline and the driver enters the alarm state.	1000
100D0008	Node protection time coefficient		3
100E0020	Node protection ID	700+ drive ID	
10140020	Emergency message station number	80+ drive station number	
10170010	Timestamp of heartbeat message generation	The slave node periodically sends message to the master node. If the master node does not receive the message within a certain period of time, the slave node is considered to be disconnected	

CANopen address	Name	Meaning	Default
2F810008	CAN baud rate	CAN baud rate Setting 100: 1M 50: 500k 25: 250k 12: 125k 5: 50k 1: 10k	50
30110108	ECAN synchronization period	In interpolation mode, this parameter is set based on the synchronization packet interval of the master station. In asynchronous mode, this parameter is not required. 0: 1ms 1: 2ms 2: 4ms 3: 8ms	2
30110208	ECAN Synchronous clock mode	Set to 1 in interpolation mode to enable the synchronization clock, and set 0 to turn off the synchronization clock in non-interpolation mode	0
30110410	ECAN synchronization Lost count	In synchronization mode, the communication status is monitored. If the value does not change, the communication status is good. If the value keeps changing, interference occurs or the synchronization period is incorrectly set.	
60070010	Communication interrupt mode	CAN communication interruption mode: determines the action logic that the driver still does not receive the node protection packet after the node protection time x node protection coefficient is exceeded 0: No processing 1: An error is reported	0

- Use the PLC to initialize PDO parameter

For the CANopen master station that can import EDS files, PDO Settings in the server are not required, and PDO information can be directly configured in the master station. After power-on, PLC initialization will send SDO messages to configure the PDO of the server. After the configuration is completed, the master station will send startup messages to start the slave station, and then PDO communication can be carried out. Most PLCs can be used in this way, such as: Schneider PLC, Siemens S7-1200+CM CANOPEN module, Buke F1 and so on.



Note

The EDS file can be found by searching under "Services" → "Downloads" on the Kinco official website.
EDS file name: Kinco_FD_20250417_V1.2.eds.

13.2.5 Interpolation mode based on CANopen

The interpolation mode is suitable for single-axis operation control and multi-axis synchronous control. The master station performs the motion contour planning, and the target position is periodically updated in the synchronous mode.

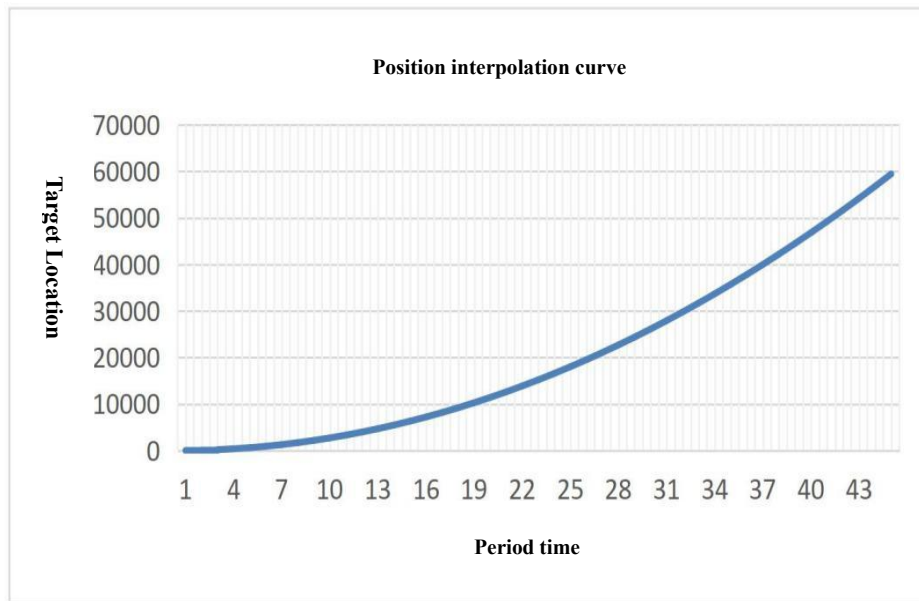


Figure 13-4 Position interpolation curve

Table 13-31 Interpolation control relate parameter

Object index	Parameter name	Description	Default value
60600008	Operation mode	Set the operation mode to interpolation mode	7
60400010	Controlword	0x1F Enable drive, run interpolation mode 0x06 Loose axis, close drive enable 0x86 Reset drive failure	1F 6 86
607A0020	Target position	Target absolute/relative position	User Defined
30110108	ECAN sync period	In interpolation mode, it is set according to the synchronization message cycle of the master station value synchronization period 0 1ms 1 2ms 2 4ms 3 8ms	User Defined
30110208	ECAN Synchronous Clock Mode	Set to 1 in interpolation mode to enable clock synchronization, and set to 0 in non-interpolation mode to disable clock synchronization.	0
30110410	ECAN Sync Loss Count	Monitor the synchronous communication status in interpolation mode, if the value keeps changing, it means that there is communication interference or the synchronization period is inconsistent with the master station setting	/

- CAN communication interruption alarm function

For the communication interruption alarm function, the following parameters need to be set

Table 13-32 Communication interrupt alarm function setting

CANopen address	Name	Meaning	Default value
100C0010	Node protection time	Through node protection, the master station can monitor the current status of each node. The master station sends a remote frame (the default COBID is 0x700+ station number and contains no message) to query the status of the slave node according to the node protection period. The slave node needs to respond within a certain time range, otherwise the master node considers the slave node offline and the driver enters the alarm state.	1000
100D0008	Node protection time coefficient		3
100E0020	Node protection ID	700+ drive ID	
10140020	Emergency message station number	80+ drive station number	
60070010	Communication interrupt mode	CAN communication interruption mode: determines the action logic that the driver still does not receive the node protection packet after the node protection time x node protection coefficient is exceeded 0: No processing 1: An error is reported	0

13.3 EtherCAT Communication

13.3.1 EtherCAT protocol

13.3.1.1 Overview

EtherCAT (Ethernet for Control Automation Technology) is a real-time industrial Ethernet technology developed by BECKHOFF and offers the following advantages:

- (1) **High speed:** Transmission rates up to 2 x 100 Mbit/s (full duplex mode) using Fast Ethernet technology.
- (2) **Flexible:** Supports various topologies such as linear and ring; inter-station distance can reach 100m, and the number of nodes can theoretically reach 65536.
- (3) **Compatible:** Conforms to Ethernet specifications, allowing the use of standard Ethernet devices such as switches.
- (4) **Synchronization:** Uses ASIC to achieve hard real-time and achieve clock synchronization accuracy of less than 1μs; The minimum synchronization period of the JD2 servo can reach 125μs.
- (5) **High efficiency:** Data refresh cycle of less than 100μs can be achieved, which is suitable for closed-loop control of the servo.

EtherCAT has a master-slave communication architecture, where the master can use a general-purpose Ethernet controller and the slaves require a special communication chip called EtherCAT Slave Controller (ESC). The PLC or controller is the master and the servo drives are the slaves.

EtherCAT uses the physical layer of standard Ethernet, modifies the data link layer to transfer data using a specialized protocol, and defines the application layer. Figure 13-5 shows a model of a Kinco Servo EtherCAT slave supporting clock synchronization and CANopen protocol, as detailed in the subsequent chapters.

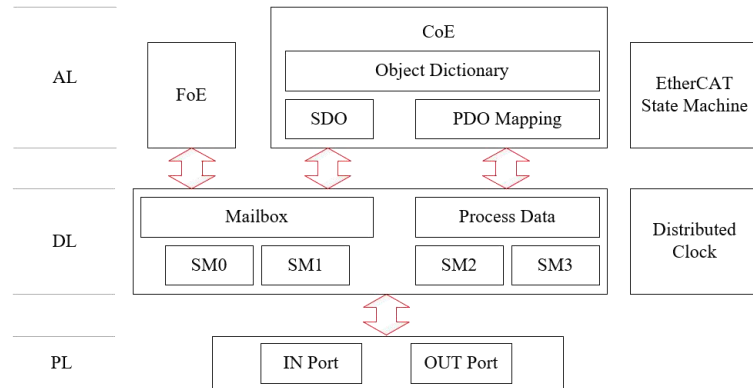


Figure 13-5 EtherCATslave model

EtherCAT uses ESI (EtherCAT Slave Information) files in XML format to fully describe the slave station devices. The ESI file for iSMD can be obtained via official website. Enter the official website page, search for XML and click on the corresponding file (File name: Kinco_FD_RX.xml) to download it.



Note

- EtherCAT[®] is a registered trademark and patented technology, licensed by Beckhoff Automation GmbH, Germany.

13.3.1.2 EtherCAT specification

iSMD servo communication specifications are as follow.

Table 13-33 EtherCAT communication specification

Item		Specification
Physical Layer	Transmission standard	IEEE 802.3
	Transmission interface	M8-S4D-PTT
	Transmission medium	100 Base-TX standard Ethernet cable
	Transmission distance	100 m
	Transmission rate	2 x 100 Mbit/s (full duplex)
	Topological structure	Linear, circular
	Number of nodes	No more than 100
	EtherCAT frame length	44 bytes ~ 1498 bytes
Data Link Layer	Communication standard	Communication standard
	DC (Distributed Clock)	DC (Distributed Clock)
	Synchronous jitter	Synchronous jitter
	Minimum synchronization period	125μs
	Refresh time	100 axes about 100 μs
	Number of FMMU	8

	Number of Sync Manager	8
	Size of DPRAM	8KB
	Communication mode	Free Run
Application Layer	Application standard	IEC 61800-7-201 (DS 402), IEC 61800-7-301
	Application Layer function	CoE (CANopen over EtherCAT)
		FoE (File Access over EtherCAT)
	CoE communication	Variable PDO mapping
		SDO request, SDO response
	CoE extended operating mode	Cyclic synchronous position mode (CSP)
		Cyclic synchronous velocity mode (CSV)
		Cyclic synchronous torque mode (CST)

13.3.1.3 EtherCAT system

The EtherCAT system consists of a master and a number of slaves. The master initiates the communication, the slave processes the received message and extracts or inserts relevant data into the message, and then transmits the message to the next slave. The last slave returns the fully processed message and passes it in reverse order to the first slave and finally to the master.

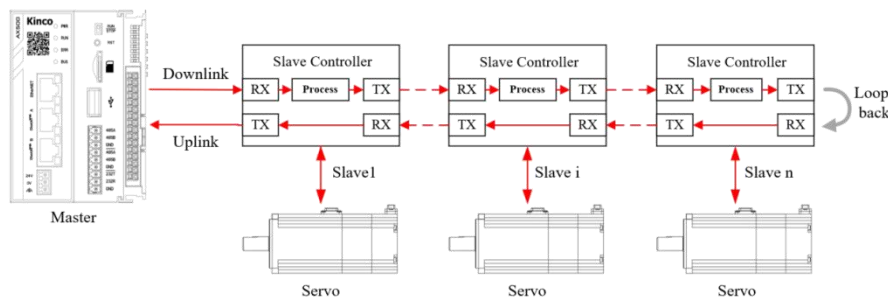


Figure 13-6 EtherCAT system operation

To make a network connection, connect the EtherCAT port of master to the IN port of slave, and the OUT port of slave to the IN port of the next slave.

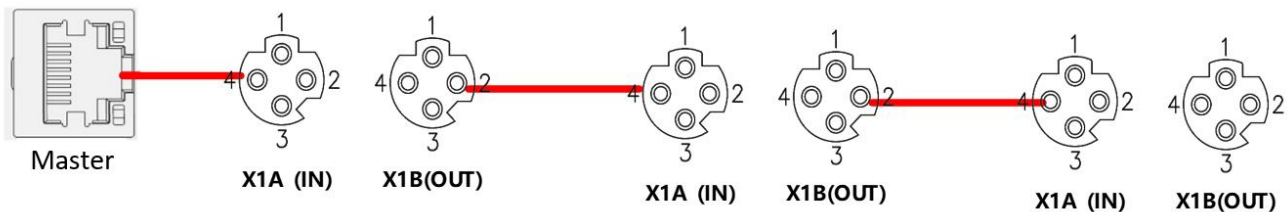
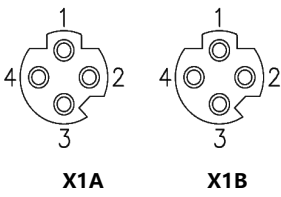


Figure 13-7 EtherCAT system topology

The EtherCAT communication interface of iSMD is shown as follows.

Table 13-34 EtherCAT communication interface

RJ45 pin definition	Pin number	Name	Description
	1	TD+	Data transmitting +
	2	RD+	Data receiving +
	3	TD-	Data transmitting -
	4	RD-	Data receiving -

The yellow light on the iSMD network port is the L/A indicator light. The L/A(Link/Activity) indicator light shows the physical layer connection status of each port.

Table 13-35 L/A indicator light

L/A indicator light status	Description
Off	The port is disconnected
Always on	This port has a connection but no data transmission
Blink	This port is connection and has data transmission

13.3.1.4 EtherCAT distributed clock

Distributed Clock enables all EtherCAT devices to use the same system time, and the servo slaves generate synchronized interrupt signals based on the synchronized system time to execute tasks. This is key to achieving strictly synchronized and coordinated motion of multiple servos for precise contour trajectory control.

Each DC slave has a local clock. EtherCAT specifies that the clock of the first DC slave connected to the master is the reference clock, and the clocks of the other slaves and the master are slave clocks, which need to be synchronized to the reference clock, as shown in Figure 13-8. The distributed clock automatically calculates the transmission delay, initial offset from the reference clock to all slave clocks and dynamic clock drift and compensates them to ensure that the local system time of all slaves remains consistent.

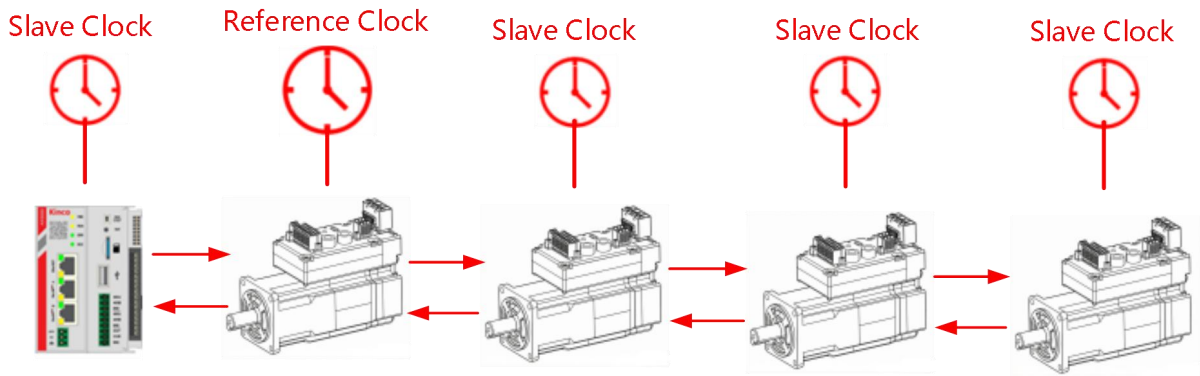
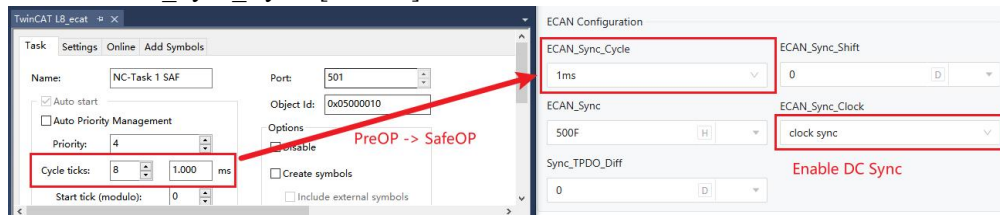


Figure 13-8 Distributed clock

When using the DC synchronous mode, the servo needs to enable the synchronous mode and set it through the ECAN_Sync_Clock [301102]. iSMD supports the self-identification function of the synchronous period. During the transition from Pre-Op to Safe-Op, the slave station will automatically update the synchronous period set by the master station to the ECAN_Sync_Cycle [301101].



The DC synchronization period of iSMD can also be manually set to 125μs, 250μs, 500μs, 1ms, 2ms, 4ms and 8ms through the ECAN_Sync_Cycle [301101].

Table 13-36 The synchronization period corresponding to the set value of ECAN_Sync_Cycle [301101]

ECAN_Sync_Cycle [301101] setting value	Synchronization period
0	125μs
1	250μs
2	500μs
3	1 ms
4	2ms
5	4ms
6	8ms



Note

The self-recognition of the synchronous period has a higher priority than manual setting and will overwrite the manually set value.

In DC mode, the servo local cycle is triggered by the synchronization signal SYNC0. The distributed clock and synchronization signal configuration related registers are shown in following table. The distributed clock is initialized, configured, started running and maintained by the master.

Table 13-37 ESC Distributed Clock related registers

Address	Bit	Description
0x0910:0x0917	0~63	Local system time
0x0920:0x0927	0~63	Offset of the local clock and reference clock
0x0928:0x092B	0~31	Transmission delay between reference and slave clocks
0x0981	0	0: Invalid; 1: Activate cycle run
	1	0: Invalid; 1: Enable SYNC0 synchronization signal
0x0990:0x0997	0~63	Cycle run start time
0x09A0:0x09A3	0~31	SYNC0 cycle time

13.3.1.5 EtherCAT state machine

EtherCAT State Machine (ESM) is responsible for coordinating the communication between master and slaves.

iSMD supports all 5 ESM states:

- (1) Init (I)
- (2) Pre-Operational (P)
- (3) Safe-Operational (S)
- (4) Operational (O)
- (5) Boot-Strap (B)

Table 13-38 ESM state

State	Meaning
Init	- No communication between master and slave at the application layer - Master initializes ESC configuration related registers - Master configures mailbox channel parameters
Pre-Operational	- Mailbox communication is possible - No process data communication
Safe-Operational	- Mailbox communication is possible - Slave transmits input process data (TxPDO) - The output process data (RxPDO) sent by master is invalid
Operational	- Mailbox communication is possible - Input process data and output process data are valid
Boot-Strap	Mailbox communication available, but only for downloading firmware via FoE protocols

The ESM is shown in Figure 13-9, and the slave transitions states according to the following rules:

- (1) When changing from Init state to Op state, the sequence "Init → Pre-Op → Safe-Op → Op" must be followed.
- (2) Cross-over transitions are allowed when returning from Op state.
- (3) Bootstrap state only allows interchanges with Init state.

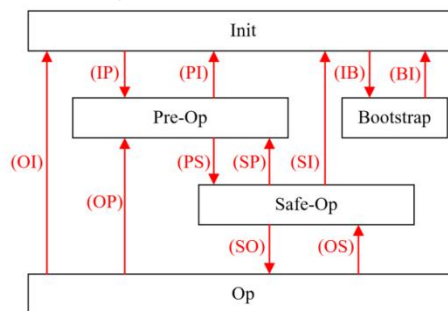


Figure 13-9 EtherCAT state machine

The meanings of each state transition in the above figure are shown in the following table.

Table 13-39 ESM state transition

Transition	Meaning
IP	- Master configures mailbox channel parameters and starts mailbox communication - Master configures Distributed Clock related registers - Master configures slave address registers

PI	• Stop mailbox communication
PS	• Master initializes process data mapping using mailbox • Master configures SM channel used for process data communication • Master configures FMMU • Start updating input process data (TxPDO)
SP	• Stop updating input process data (TxPDO)
SO	• Start updating output process data (RxPDO)
OS	• Stop updating output process data (RxPDO)
OP	• Stop process data communication
SI	• Stop process data communication • Stop mailbox communication
OI	• Stop process data communication • Stop mailbox communication
IB	• Start boot mode
BI	• Restart the slave

Master writes the target state to the slave AL (Application Layer) control register to change the slave state; reads the slave AL status register to get the actual state of slave, the related ESC registers are shown in following table.

Table 13-40 ESC application layer related register

Address	Length (B)	Description
0x0120:0x0121	2	Application layer control
0x0130:0x0131	2	Application layer status
0x0134:0x0135	2	Application layer status code

The host computer can display the ESM status and status codes. When the host computer shows 0X1C380000, it corresponds to the ESM status, and when it shows 0X1C390000, it corresponds to the ESM error code.

Table 13-41 The ESM status

Display value	Description
0x01	Init
0x02	Pre-Operational
0x03	Boot-Strap
0x04	Safe-Operational
0x08	Operational

Table 13-42 The ESM error code

Display value	Description
0x0011	The state transition request is invalid
0x0012	Unknown status request
0x0015	Incorrect email configuration
0x0016	Incorrect email configuration
0x001B	The process data watchdog has exceeded the time limit
0x001D	SM2 configuration error
0x001E	SM3 configuration error
0x0026	Inconsistent setting

0x002C	Synchronization error, SYNC0 signal interrupted
0x002D	The synchronization signal was not received during the SO conversion
0x0061	The ID selector conflicts with the alias in SII

**Note**

- When the slave ESM is in an error state, the upper 4 bits of the ESM status value are 1 (for example, 0x14, that is, ErrSafeOp), and the ESM status code indicates the specific error code.
- When a slave station responds to the ID request of the master station, the upper 4 bits of the ESM status value are 2, and the ESM status code represents the ID value of the slave station.

13.3.2 CoE protocol

Kinco's EtherCAT servo supports the DS 402 guild regulation (CoE) in the CANopen protocol, as shown in Figure 13-10. The CoE uses email to access the object dictionary for device configuration and aperiodic parameter reading and writing ; Use process data communication to periodically transmit instruction data and status data. The relevant terms are explained as follows——

- (1) Object dictionary: Use object to describe the full functionality of the CANopen device , it is a list of all parameters of the device.
- (2) SDO: Service Data Object is used for aperiodic mailbox communication.
- (3) PDO: Process Data Objects is used for periodic process data communication.
- (4) SM: Sync Manager is used to ensure the consistency and security of data exchange between master and slave.

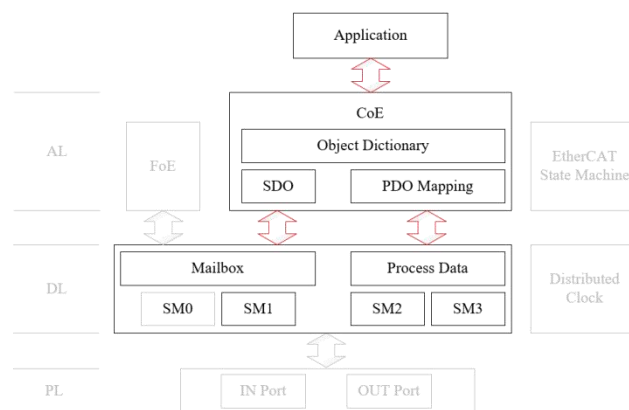


Figure 13-10 Coe based iSMD slave model

13.3.2.1 CoE object dictionary

CoE complies with the CANopen protocol and has the same definition in its object dictionary. Related communication objects 0x1C00 to 0x1C4F for EtherCAT communication are extended for setting SM channels and PDO allocation.

Table 13-43 CoE object dictionary definition

Index	Definition
0x0000:0x0FFF	Data type description
0x1000:0x1018	Device type and identifier
0x1600:0x17FF	RxPDO mapping Subindex 0: Number of object Subindex 1: The first output data object of the mapping : Subindex n: The last output data object of the mapping
0x1A00:0x1BFF	TxPDO mapping Subindex 0: Number of object Subindex 1: The first input data object of the mapping : Subindex n: The last input data object of the mapping
0x1C00	SM Channel type 0: Mailbox output, aperiodic data communication 1: Mailbox input, aperiodic data communication 2: Process data output, periodic data communication 3: Process data input, periodic data communication
0x1C10:0x1C2F	PDO assignment of SM channel during process data communication Subindex 0: Numbe of assigned PDO Subindex 1~n: PDO mapping object index
0x1C30:0x1C4F	SM channel parameter
0x2000:0x5FFF	Data object are defined by kinco
0x6000:0x9FFF	Data object are defined by DS 402 guild regulation

13.3.2.2 Process data communication

Real-time is the ability to complete a task in a certain amount of time. As a real-time industrial Ethernet, EtherCAT periodically transmits real-time data, known as process data communication. Process data is divided into output process data (RxPDO) and input process data (TxPDO). RxPDO contains the command data updated by the master station, and TxPDO contains the status data updated by the slave station.

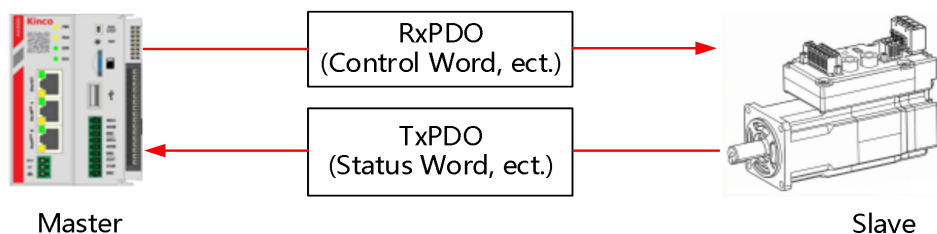


Figure 13-11 Process data communication

For the user, the actual data objects that need to be used are PDO entries, such as control word, target location, status word, actual location, etc. The data communication process is as follows:

- (1) Select the desired PDO entry.
- (2) Perform PDO mapping

(3) Perform PDO allocation, that is, configure the synchronization manager.

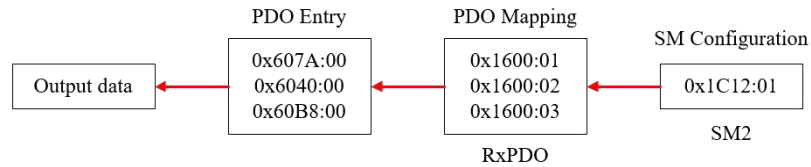


Figure 13-12 Configuring process data communication (using output data as an example)

For example, if you compare PDO entries to desired goods, then the data objects 0x1600 and 0x1A00 are trucks. Mapping PDO entries to 0x1600 and 0x1A00 is the equivalent of loading goods onto a truck. After the goods are loaded, roads are needed. The CoE builds one one-way street (SM2 channel and SM3 channel) from the primary station to the secondary station and from the secondary station to the primary station. Using data objects 0x1C12 and 0x1C13, PDO allocation is equivalent to putting a truck on the corresponding road.

iSMD supports variable PDO mapping and can be configured on demand via CoE communication.

Table 13-44 Default PDO mapping

Applicable operation mode	PDO mapping		PDO entry
CSP / CSV / CST	RxPDO	0x1600	Controlword[604000] Operation_Mode[606000] Target_Position[607A00] Target_Speed[60FF00] Target_Torque[607100] Touch_Probe_Function[60B800]
	TxPDO	0x1A00	Statusword[604100] Operation_Mode_Buff[606100] Error_Code[603F00] Pos_Actual[606400] Speed_Real[606C00] Actual_Torque[607700] Touch_Probe_Status[60B900]
PP / HM (Position mode/Homing mode)	RxPDO	0x1601	Controlword[604000] Operation_Mode[606000] Target_Position[607A00] Profile_Speed[608100] Profile_Acc[608300] Home_Offset[607C00] Homing_Method[609800]
	TxPDO	0x1A01	Statusword[604100] Operation_Mode_Buff[606100] Error_Code[603F00] Pos_Actual[606400] Speed_Real[606C00]

CoE uses data object 0x1C10:0x1C2F to define the PDO mapping object list of the SM channel. The output process data uses the SM2 channel and the PDO allocation is defined by data object 0x1C12. The input process data uses the SM3 channel and the PDO allocation is defined by the data object 0x1C13.

Table 13-45 PDO allocation

PDO allocation		PDO mapping index
SM	Index:Subindex	
SM2	0x1C12:01	0x1600 ~ 0x1607
SM3	0x1C13:01	0x1A00 ~ 0x1A07

After configuring process data communication in KincoServoPro PC software, click Config→Operation mode→EtherCAT, and you can see the configured PDO:

TPDO1	TPDO2	TPDO3	TPDO4	TPDO5	TPDO6	TPDO7	TPDO8
Index	Name	Current value	Mapping object	Mapping object value	Mapping Unit		
1A0000	Group_TX1_P...	7	D	-	-	-	
1A0001	TX1_PDO1	60640020	H	Pos_Actual	52	D	inc
1A0002	TX1_PDO2	60410010	H	Statusword	5231	H	
1A0003	TX1_PDO3	603F0010	H	Error_Code	0	H	
1A0004	TX1_PDO4	60610008	H	Operation_Mode_Buff	8	D	
1A0005	TX1_PDO5	60B90010	H	Touch_Probe_Status	0	H	
1A0006	TX1_PDO6	60BA0020	H	Touch_Probe_Rising1	0	D	
1A0007	TX1_PDO7	60FD0020	H	Digital_Inputs	180000	H	
1A0008	TX1_PDO8	0	H	-	-	-	

RPDO1	RPDO2	RPDO3	RPDO4	RPDO5	RPDO6	RPDO7	RPDO8
Index	Name	Current value	Mapping object	Mapping object value	Mapping Unit		
160000	Group_RX1_P...	3	D	-	-	-	
160001	RX1_PDO1	607A0020	H	Target_Position	0	D	inc
160002	RX1_PDO2	60400010	H	Controlword	6	H	
160003	RX1_PDO3	60880010	H	Touch_Probe_Function	0	H	
160004	RX1_PDO4	0	H	-	-	-	
160005	RX1_PDO5	0	H	-	-	-	
160006	RX1_PDO6	0	H	-	-	-	
160007	RX1_PDO7	0	H	-	-	-	
160008	RX1_PDO8	0	H	-	-	-	

13.3.2.3 Mailbox data communication

Mailbox communication uses SM0 and SM1 channels to access entries in the object dictionary through SDO for communication configuration, device configuration, and aperiodic parameter reading and writing.

SDO transfers are divided into download and upload. Download transfers are often used by the master to set the parameters of the slave , and upload transfers are often used by the master to read the performance parameters of the slave . Take SDO download as an example, the master sends the SDO download request to the slave SM0 channel, the slave reads the mailbox data, executes the corresponding processing, and writes the response data to the SM1 channel; The master then reads the SM1 channel to obtain the data and judge the execution result of the slave.

iSMD supports SDO request service and SDO response service.

13.3.2.4 DS402 guild regulation

The following figure shows the CoE servo driver structure based on DS402 guild regulation. The following briefly describes the device control state machine and periodic synchronization working mode of DS402 guild regulation.

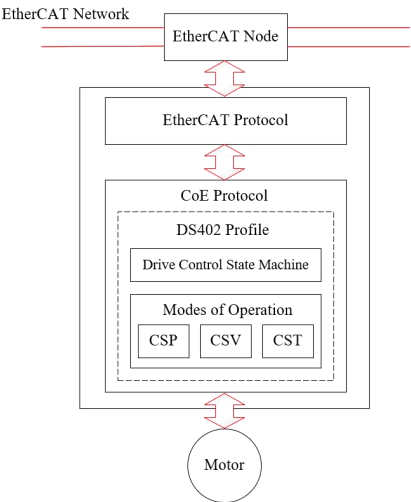


Figure 13-13 CoE servo drive structure

Device control state machine

The DS 402 guild regulation defines the device control state machine of the servo drive. The state of the servo must be changed according to certain rules. The master controls the operating state of the servo by modifying the control word [604000], and obtains the current state of the servo by reading the status word [604100].

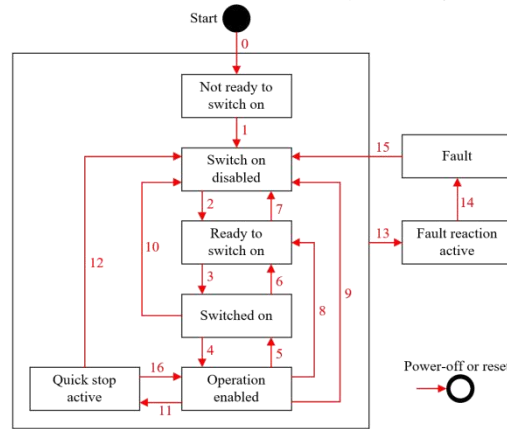


Figure 13-14 Device control state machine

Table 13-46 Device control state machine status and conversion description

Status description							
Status	Description	The bit of the statusword [604100]					
		Bit6	Bit5	Bit3	Bit2	Bit1	Bit0
Not ready to switch on	The control power is on, but initialization is not complete	0	×	0	0	0	0
Switch on disabled	Server initialization is complete	1	×	0	0	0	0
Not ready to switch on	The control power is on, the power switch is off, and the motor has no torque	0	1	0	0	0	1
Switched on	The power is on, the motor has no torque	0	1	0	0	1	1
Operation enabled	The servo driver controls the motor according to the configuration, and the motor has torque	0	1	0	1	1	1
Quick stop	Servo stops as set	0	0	0	1	1	1
Fault reaction	Servo error occurred, stop the machine according to the error setting, the motor has torque	0	×	1	1	1	1
Fault	Error status, the motor has no torque	0	×	1	0	0	0
Conversion description							
Conversion	Description	The bit of the controlword [604000]					
		Bit7	Bit3	Bit2	Bit1	Bit0	
0	Automatic transition after power-on or reset						
1	Automatic transfer						
2	Get the power cut command from the master	0	×	1	1	0	
3	Get the power on command from the master	0	×	1	1	1	
4	Get the enable servo run command from the master	0	1	1	1	1	
5	Get a stop servo run command from the master	0	0	1	1	1	
6	Get the power cut command from the master	0	×	1	1	0	
7	Get an quickly stop or power cut command from the master	0	×	×	0	×	
		0	×	0	1	×	

8	Get the power cut command from the master	0	×	1	1	0
9		0	×	×	0	×
10	Get an quickly stop or power cut command from the master	0	×	×	0	×
		0	×	0	1	×
11	Get the quickly stop command from the master	0	×	0	1	×
12	After the quickly stop function is executed, the power off command is obtained from the master	0	×	×	0	×
13	Fault					
14	Automatic transfer					
15	Get the error reset command from the master	Rising edge	×	×	×	×
16	Not recommended	0	1	1	1	1

Operation mode

The periodic synchronous operation mode is an extension of the CoE to the DS 402, including CSP (8 mode), CSV (9 mode), and CST (10 mode). The master sets the servo operating mode by writing the Operation_mode [606000], and the servo uses Operation_mode_Buff [606100] to represent the actual operating mode.

The periodic synchronous operation mode uses Bit12 of the Statusword [604100]

Table 13-47 Statusword Bit12 definition in the periodic synchronous operation mode

Bit	Value	Description
12	0	• CSP: Ignore target position • CSV: Ignore target speed • CST: Ignore target torque
	1	• CSP: Target position valid • CSV: Target speed valid • CST: Target torque valid

Cyclic synchronous position mode (CSP): The master sends periodically synchronized position command values to the servo, and the servo performs position control, speed control, and torque control. The servo can provide the actual position value, the actual speed value and the actual torque value to the master.

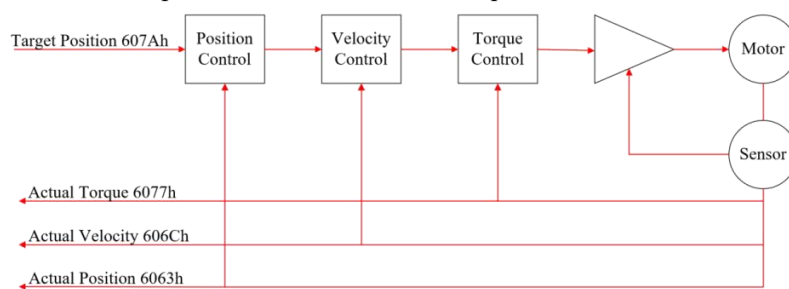


Figure 13-15 Cyclic synchronous position mode

Cyclic synchronous velocity mode (CSV): The master periodically sends target speed instructions to the servo, the servo performs speed control and torque control, and the position loop can be realized by the master. The servo can provide the actual position value, the actual speed value and the actual torque value to the master.

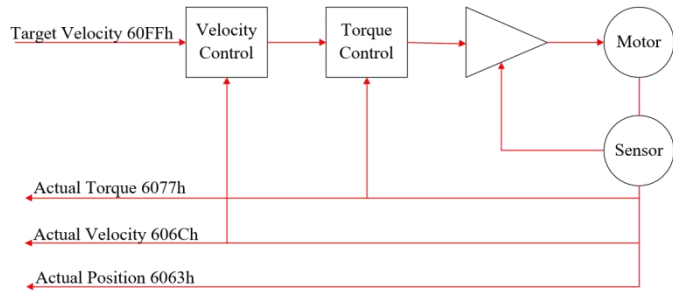


Figure 13-16 Cyclic synchronous velocity mode

Cyclic synchronous torque mode (CST): The main periodically sends target torque instructions to the servo, which performs torque control. The servo can provide the actual position value, the actual speed value and the actual torque value to the master. Cyclic synchronous torque mode can also be extended for torque limits and speed limits to limit dynamic values.

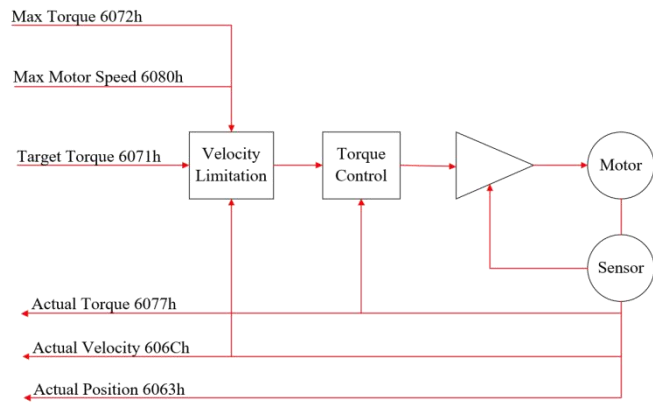


Figure 13-17 Cyclic synchronous torque mode

13.3.3 Application cases

13.3.3.1 Communication parameter settings

Slave station alias setting

iSMD supports the setting of slave station alias. The alias mechanism can ignore the unconnected slave stations in EtherCAT communication without affecting the normal communication of other stations. It also allows the connection sequence and configuration sequence of slave stations to be inconsistent as long as the aliases are set correctly. The setting can be done through the device Node_ID [100B00].

All Param [0] X

Node_ID

	☐	Index	Description	Preset value	Current value	Max	Min	Units	Effect mode
	☐	100B00	Node_ID		0	DEC	127	0	Save & Reboot

When the device Node_ID [100B00] is 0, the PLC/ controller is allowed to configure aliases.

When the device Node_ID [100B00] is non-0 (must be greater than 0), only the alias configured by the servo can be used, and aliases configured by the PLC/ controller are not allowed.

Default communication parameter

In order to enable iSMD to be connected to the EtherCAT fieldbus network more accurately and conveniently, the following factory default settings have been made for the relevant parameters.

- a. The digital input (DI) is not configured with any functions by default to prevent conflicts between certain functions (such as controlword) and EtherCAT control;

IO Config [0] X

Physical DIO

DIO Monitor

DIO Settings

Digital Input (DI)

PIN number	Function	Actual input	Simulation input	Polarity	Valid input
DIN1	-			f High	
DIN2	-			f High	
DIN3	-			f High	
DIN4	-			ī Low	
DIN5	-			ī Low	
DIN6	-			f High	
DIN7	-			f High	

- b. Keba[23400D] is set to 1 by default. The function of this parameter is shown in Table 13-48. The default Operation_mode [606000] is 8 (CSP mode), the ECAN_Sync_Clock [301102] is 1, enabling DC synchronization; the ECAN_Sync_Cycle [301101] is 3, 1ms; the Abort_Connection_Mode [600700] is 1, enabling communication error reporting.

Parameter monitoring			
Index	Description	Current value	Units
23400D	Keba	1	D
606000	Operation_Mode	8	D
301102	ECAN_Sync_Clock	1	D
301101	ECAN_Sync_Cycle	3	D
600700	Abort_Connection_Mode	1	D

Table 13-48 Keba parameter function description

Keba	Description
0	When the servo is not enabled, the Operation_Mode_Buff [606100] remains 0, and the Operation_Mode [606000] is immediately updated to the Operation_Mode_Buff [606100] after the servo is enabled.
1	Whether the servo is enabled or not, the Operation_Mode [606000] is immediately updated to the Operation_Mode_Buff [606100]

13.3.3.2 Kinco AX500 controller application

When AX500 controller communicates with a single drive, directly use the network cable to connect the EtherCAT interface of the controller and the IN interface of the servo drive; when the controller connects with multiple drives, connect the OUT interface of the previous drive to the IN interface of the next drive.

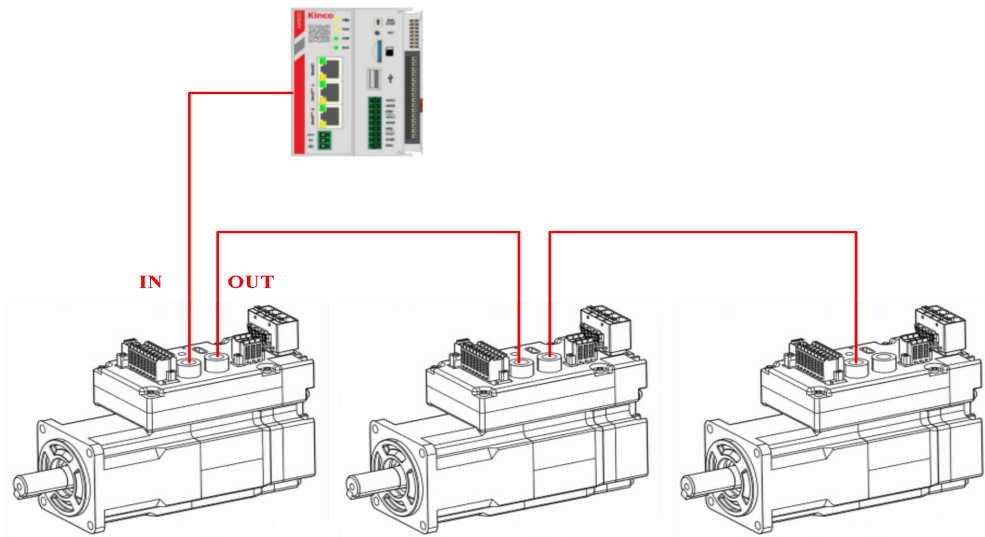
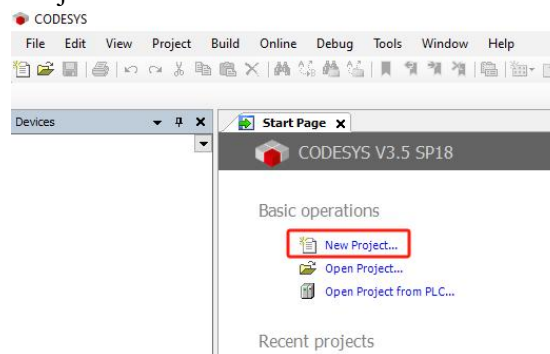


Figure 13-18 Kinco AX500 controller connecting drives

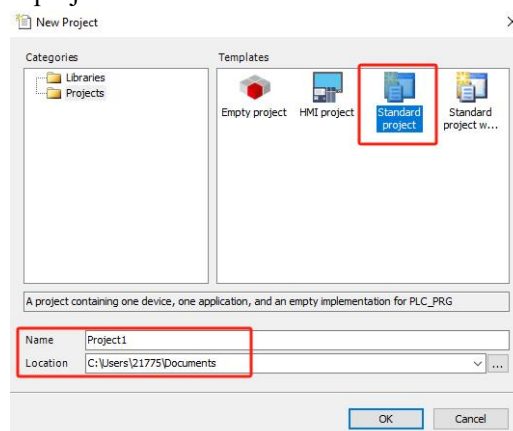
1. Create a new CODESYS project

(1) Start CODESYS → “New Project”

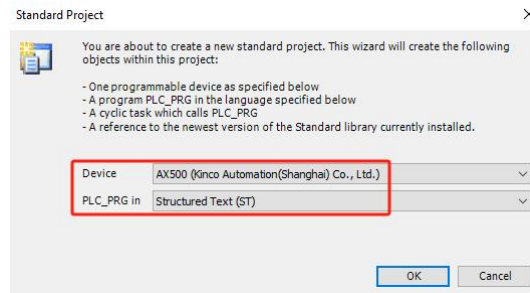


(2) The following options are available in “New Project”:

- Select "Standard project".
- Fill in the project name, such as “Project1”.
- Select the location to save the project.



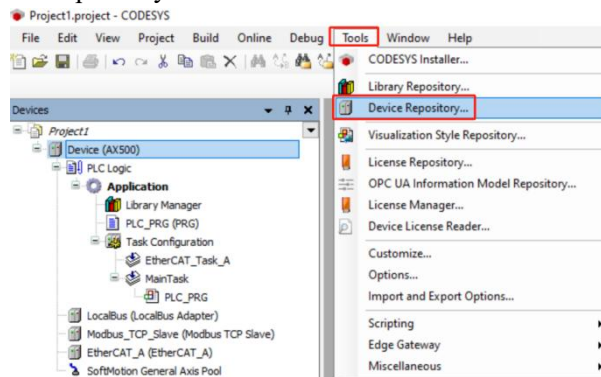
- Select the controller - AX500.
- Select a programming language, such as Structured Text (ST).



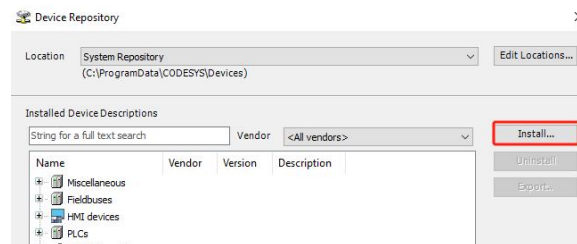
f. After adding the controller, the master "EtherCAT_A" of type EtherCAT Master SoftMotion is automatically added.

2. Install the servo XML file

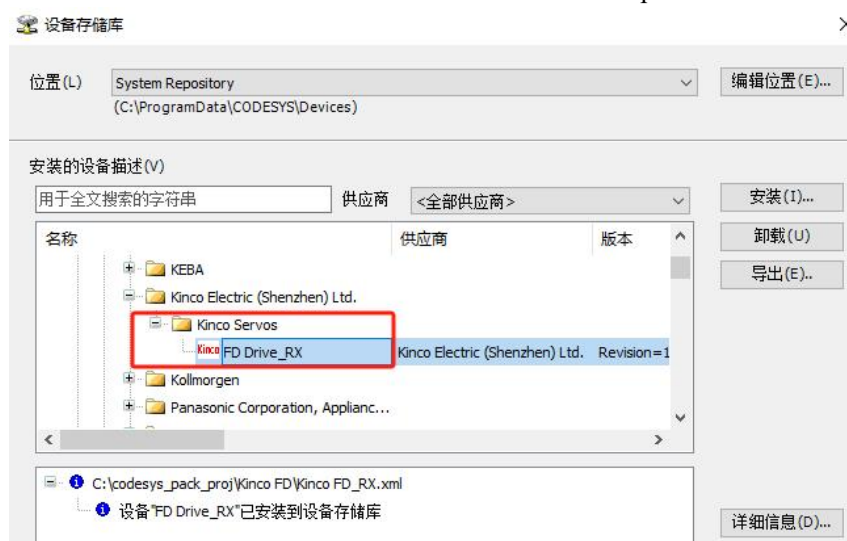
(1) Click “Tools” → “Device Repository...” in the menu bar.



(2) Click “Install...”.

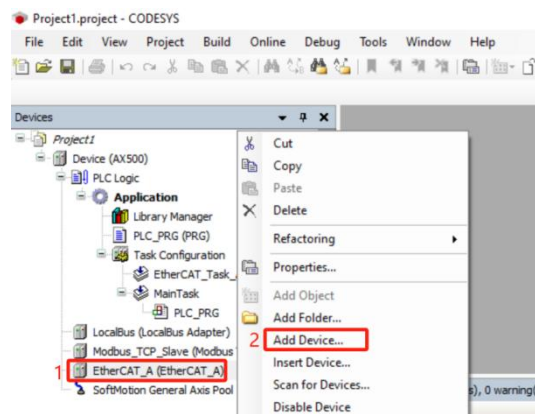


(3) Find the XML file for the drive and install it. The installation is completed as shown below.

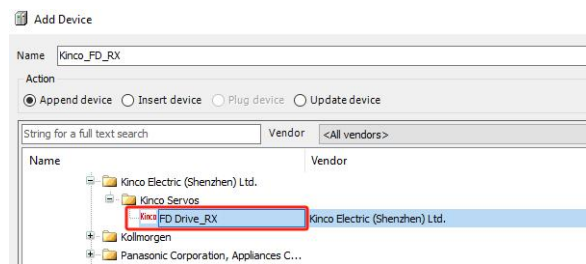


3. Add servo slave

(1) The EtherCAT master has been added automatically, right click on “EtherCAT_A (EtherCAT_A)” → “Add Device...”.



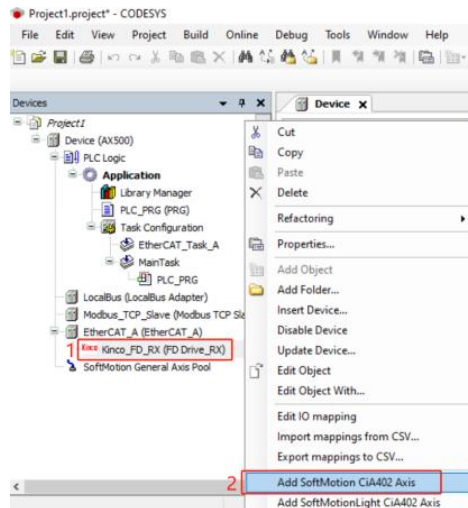
(2) Find the servo drive and double-click to add the slave.



Note

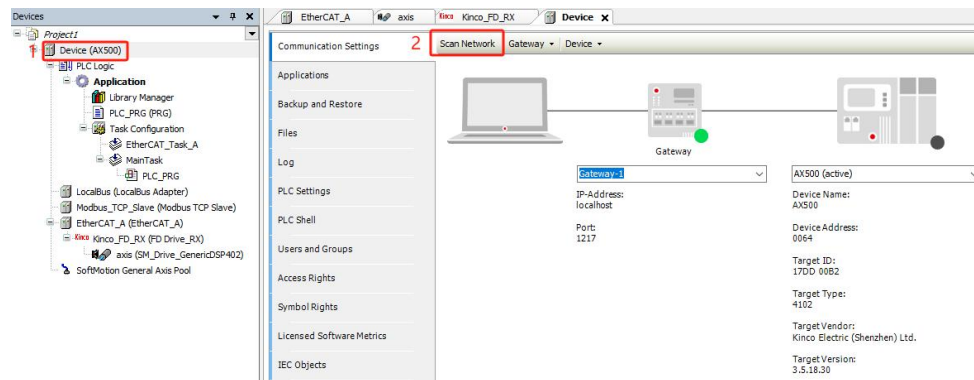
- Multiple slaves can be added here, the number added depends on the actual number of drives in the network.

(3) To add a CiA402 axis, right-click on the slave → “Add SoftMotion CiA402 Axis”.

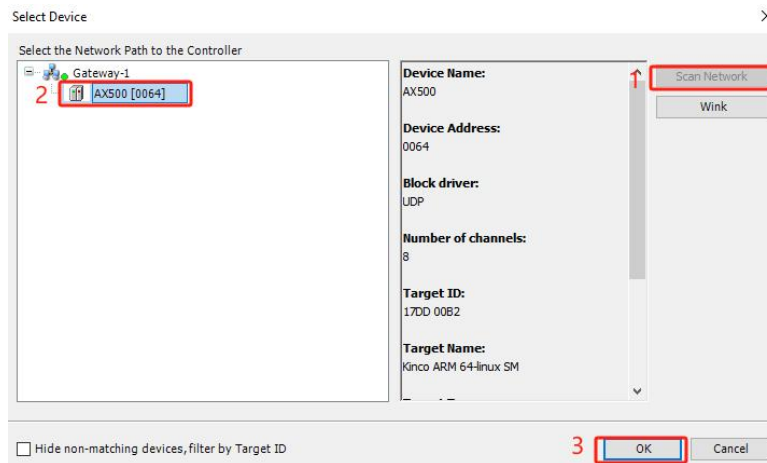


4. Connect the controller

(1) Double-click “Device (AX500)” → “Communication Settings” → “Scan Network”.



(2) Click on “Scan Network”, select “AX500”, and then OK.

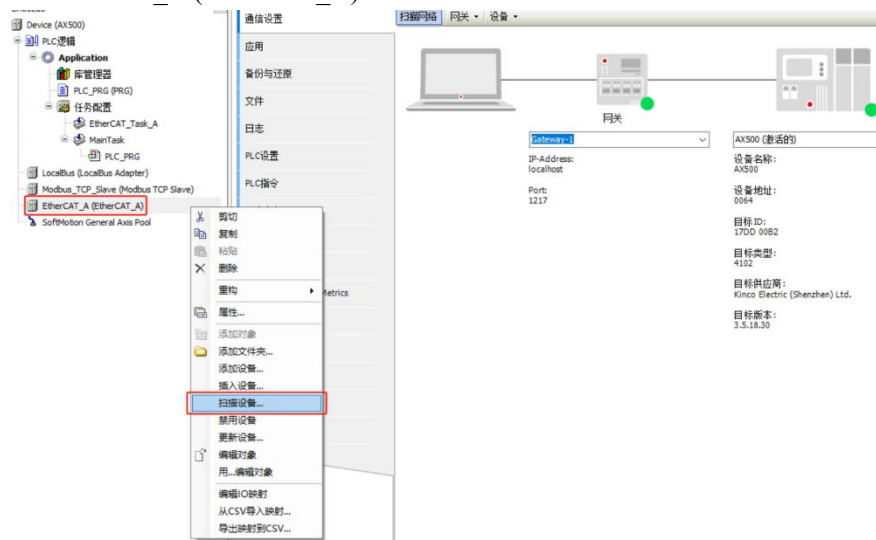


Note

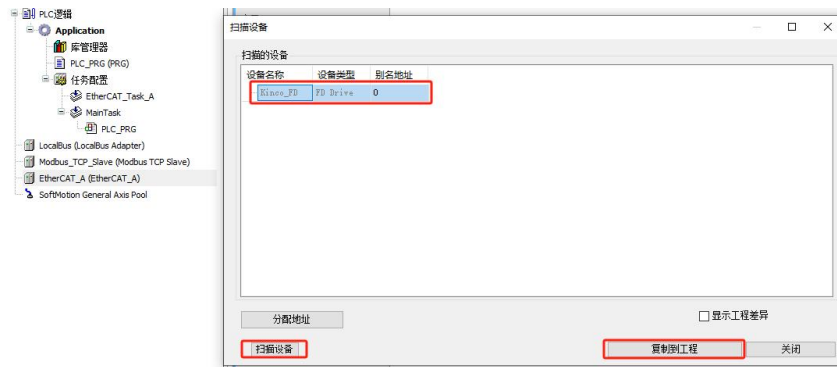
- Make sure that the IP of the controller is in the same network segment as the IP of the computer's network card.

(3) Slaves can also be added by online scanning once the controller is connected:

a. Right-click “EtherCAT_A (EtherCAT_A)” → “Scan for Devices...”.



b. Click “Scan Device”, select the scanned device, click “Copy to project”, the slave will be added to the device tree.

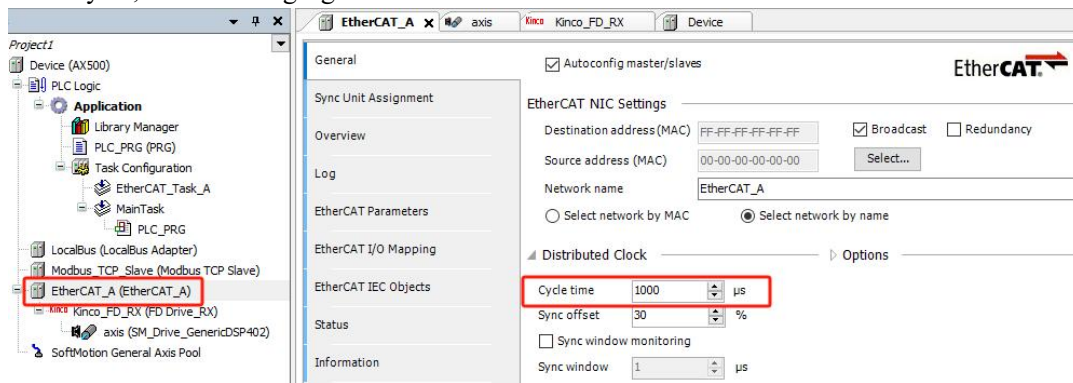


Note

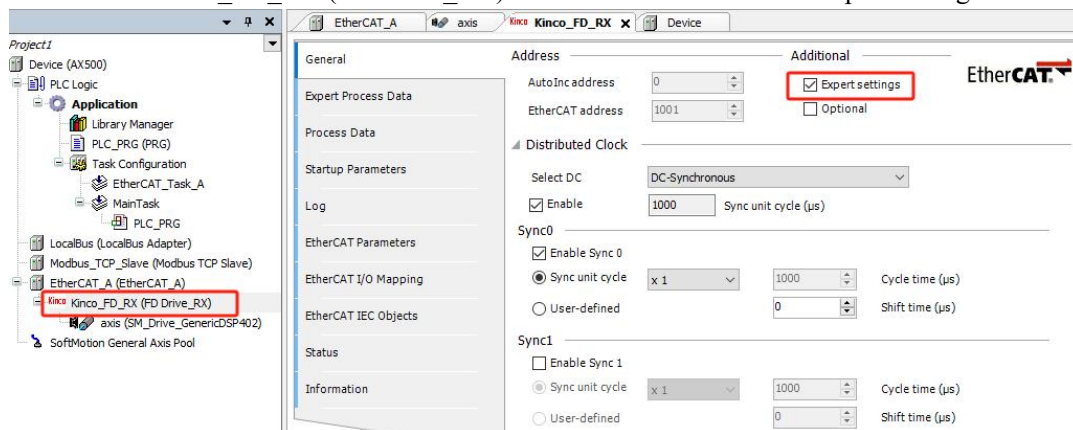
- This method of adding a slave is an alternative to the method in step 3.

5. Set the EtherCAT communication parameter

(1) Double click “EtherCAT_A(EtherCAT_A)” → “General” → “Distributed Clock”, you can set the synchronization cycle, the following figure is 1ms.

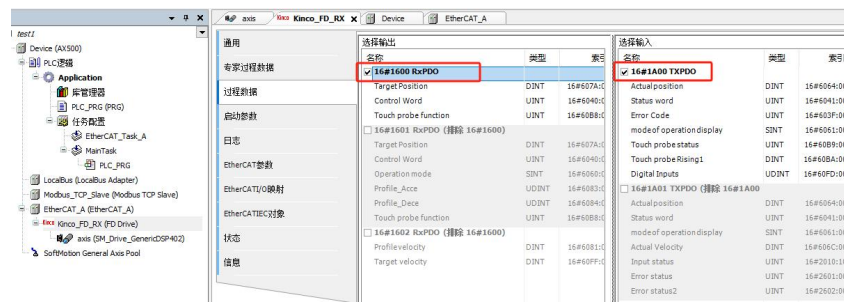


(2) Double-click “Kinco_FD_RX (FD Drive_RX)” → "General" to enable “Expert settings”.

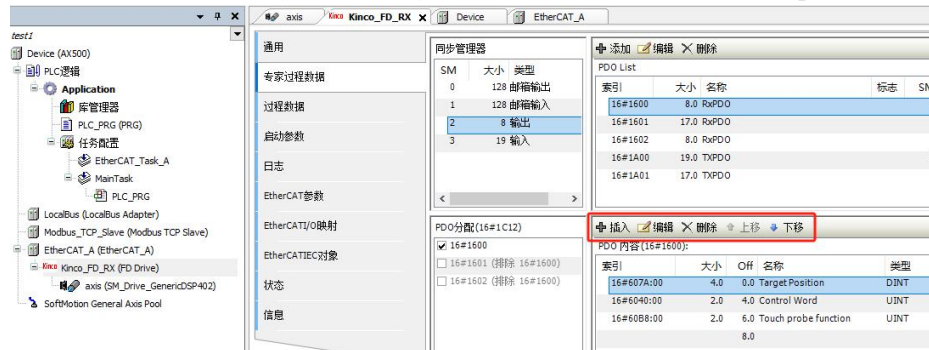


(3) Configure PDO

a. To add RxPDO and TxPDO, click "Process Data" to select RxPDO and TxPDO.

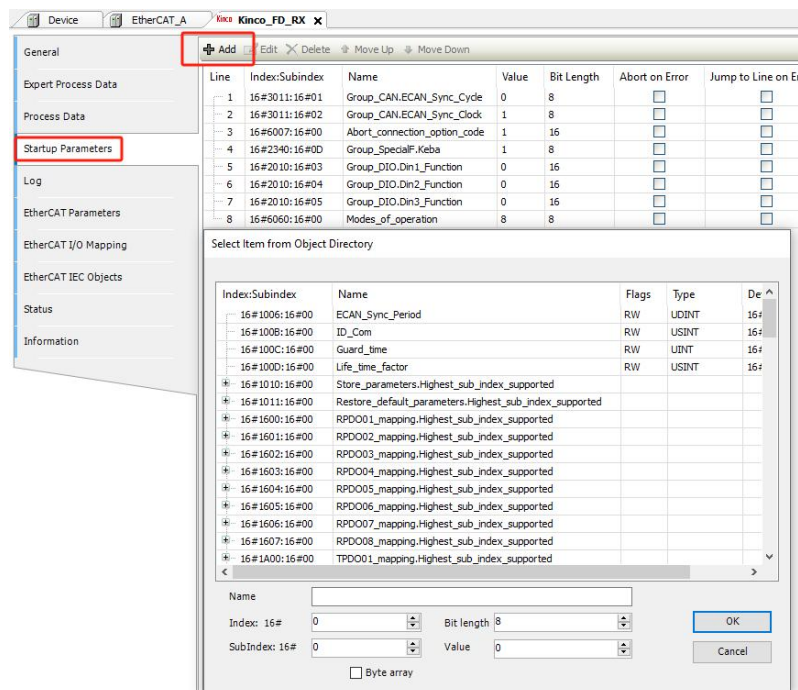


b. If the default PDO does not meet your needs, you can add or delete it in the “Expert Process Data” screen.



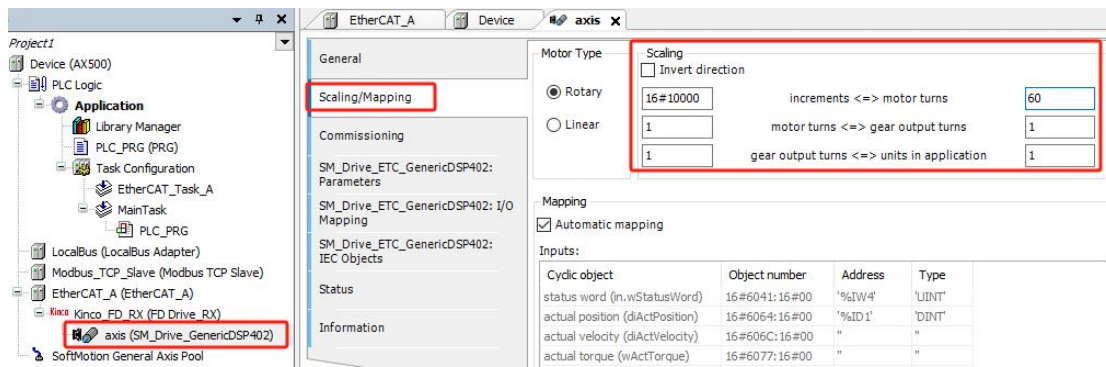
(4) Set startup parameter

Drive parameters can be set automatically by SDO at power-on. Click “Add” in the “Startup Parameters” interface to select the parameter to be set at power-on and set the parameter value.

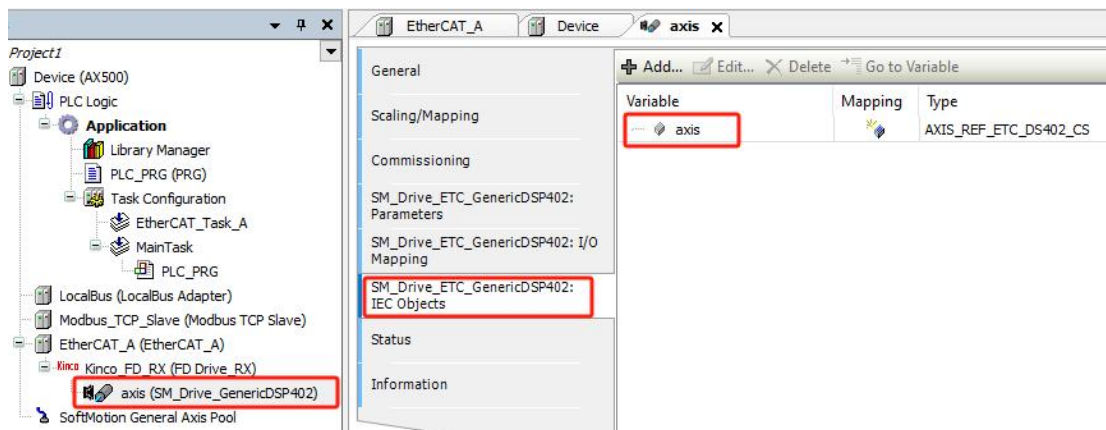


6. Set CiA402 axis parameter

(1) Set the scaling of the axis. Double-click the CiA402 axis and set it in “Scaling/Mapping”.

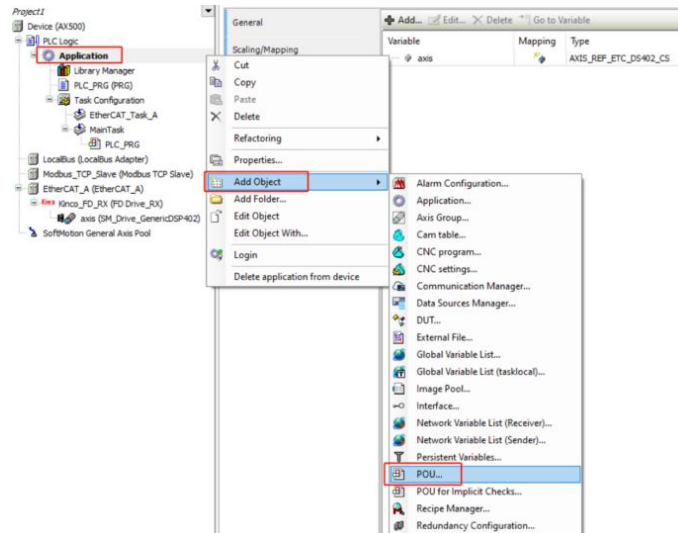


(2) Set the axis name. Set in “SM_Drive_ETC_GenericDSP402:IEC Objects”, the name of the object is the axis name in the MC instruction.

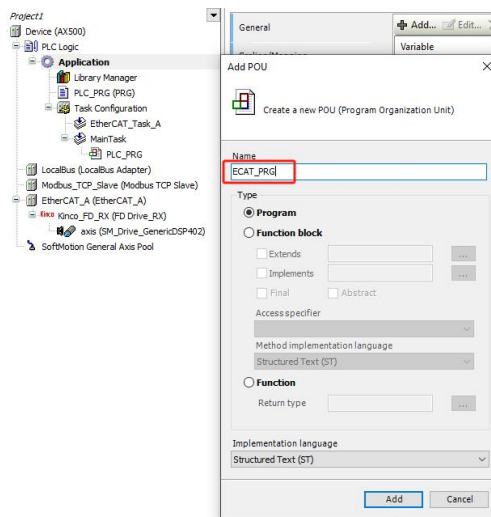


7. Write PLC program

(1) Create a new POU program as shown in the following figures, right-click “Application” → “Add Object” → “POU...”, and enter the name of the POU.

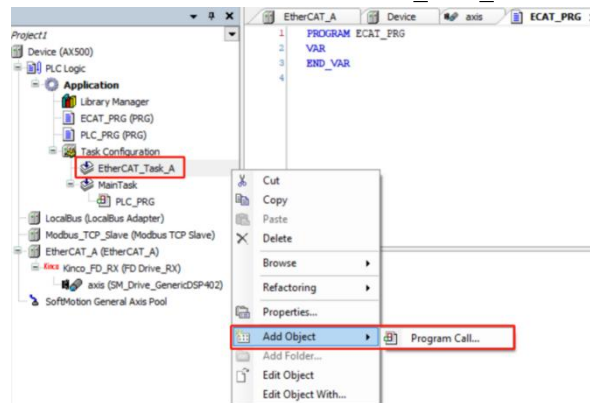


(a)

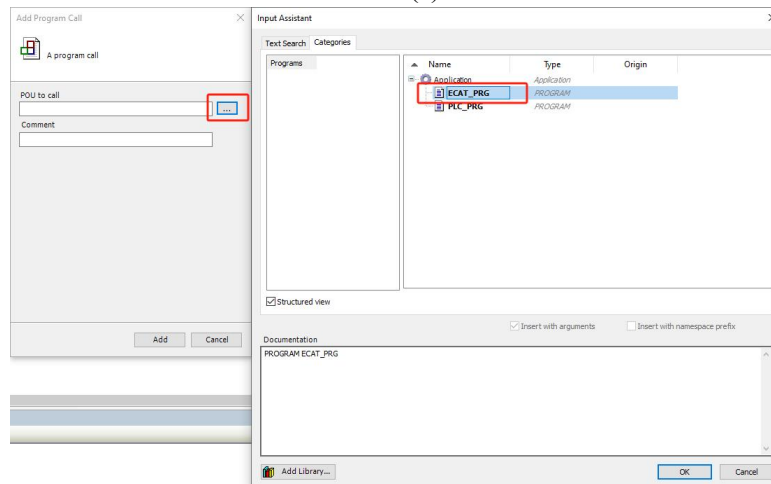


(b)

(2) The program “ECAT_PRG” is called in the task “EtherCAT_Task_A” as shown in the following figures.

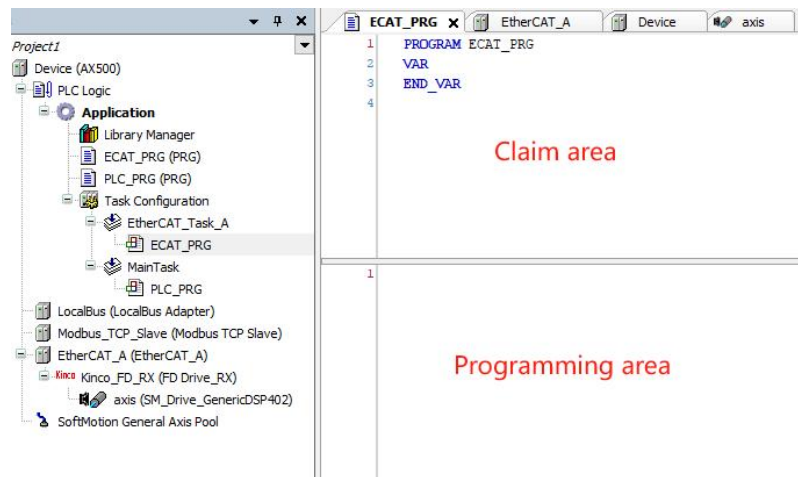


(a)



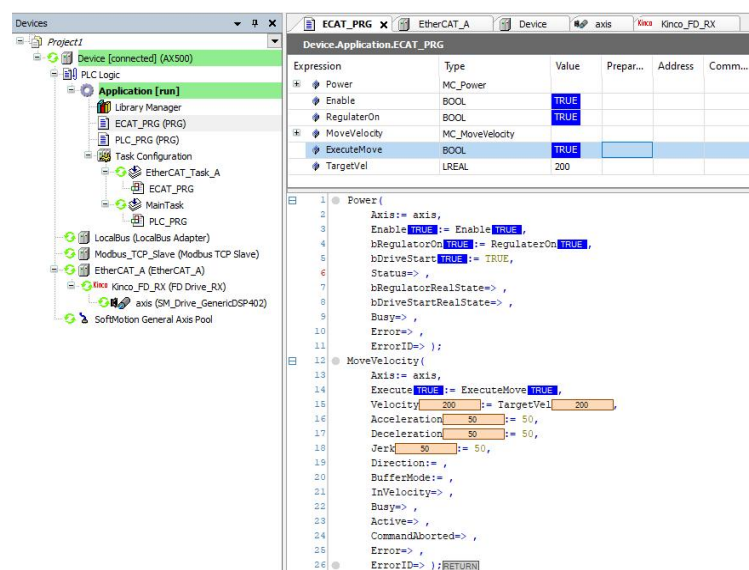
(b)

(3) Open “ECAT_PRG”, the upper part is the declaration area, the lower part is the programming area.



(4) Write a program to control the drive using motion control instructions.

Instruction	Role
MC_Power	Servo enable operation
MC_Jog	Servo point operation
MC_MoveVelocity	Servo operates at the specified speed.
MC_MoveRelative	Servo relative displacement
MC_MoveAdditive	The relative displacement of the superimposed servo target
MC_MoveAbsolute	Clear the servo alarm
MC_Halt	Servo stop running
MC_Stop	Servo stop running and lock
MC_Home	Servo homing command
MC_SetPosition	Set position offset
MC_SetControllerMode	Set servo control mode
MC_SetTorque	Set servo torque
MC_ReadStatus	Read servo axis status



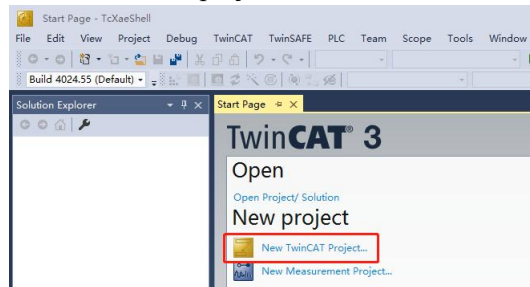
13.3.3.3 TwinCAT3 application

Direct servo control with TwinCAT

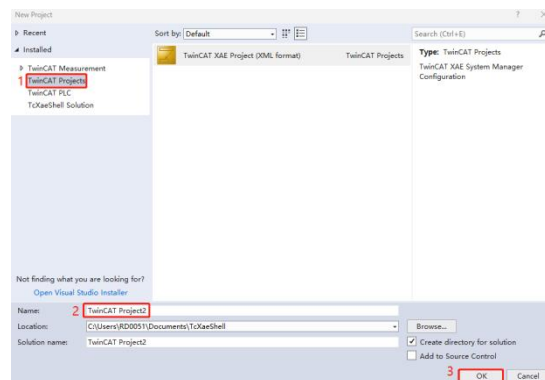
1. Create TwinCAT project

(1) Before starting TwinCAT, you need to copy the XML file of the drive to the installation directory of TwinCAT, the default path is C:\TwinCAT\3.1\Config\Io\EtherCAT.

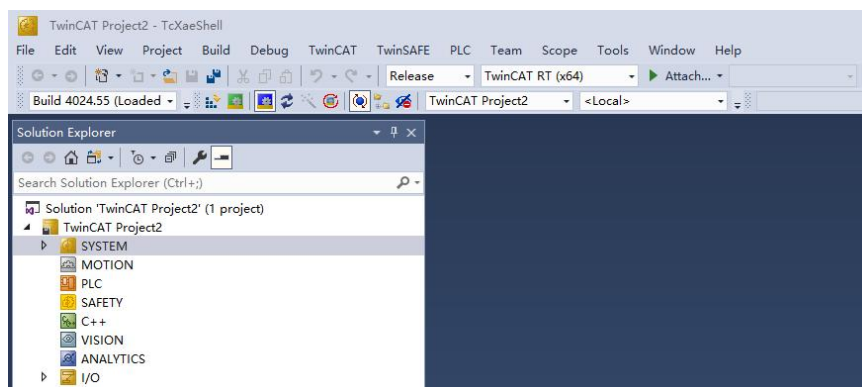
(2) Start TwinCAT and create a new TwinCAT project as shown in the following figure.



(a)



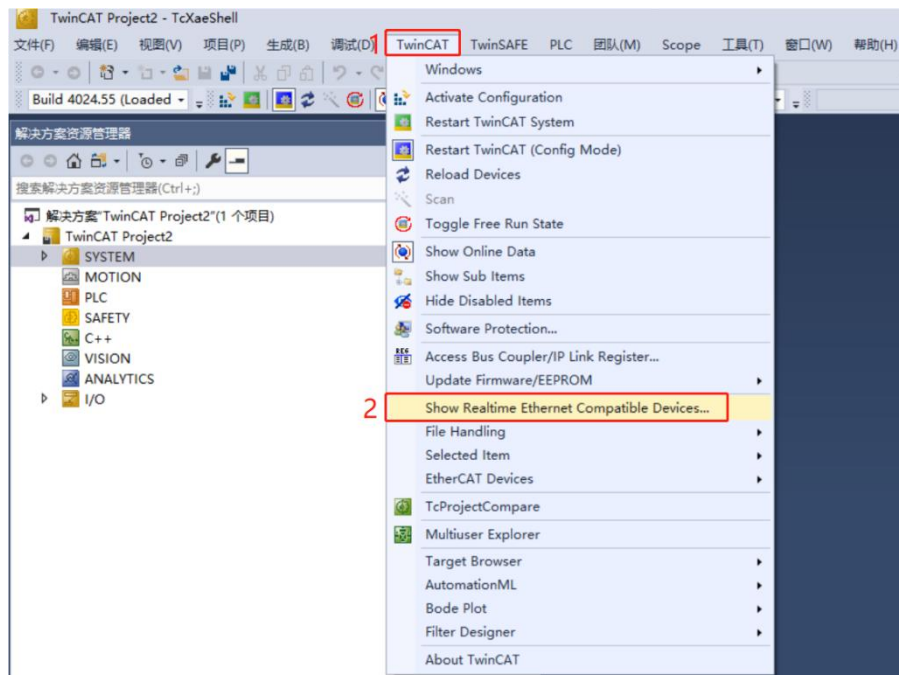
(b)



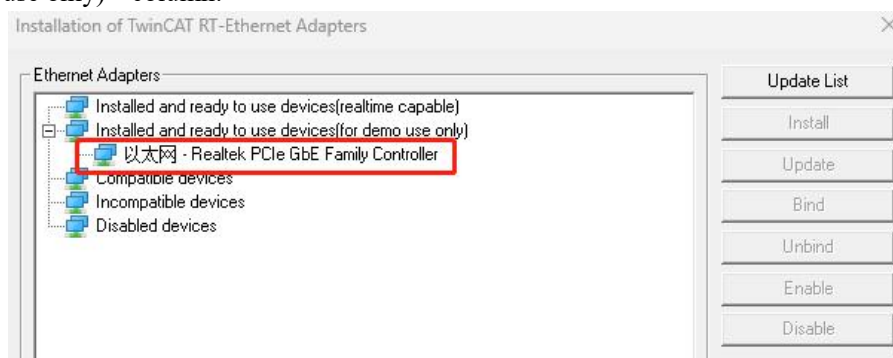
(c)

2. Install Ethernet real time drive

(1) TwinCAT can do NC control of servo directly without PLC, you need to install Ethernet real time driver before doing NC control.



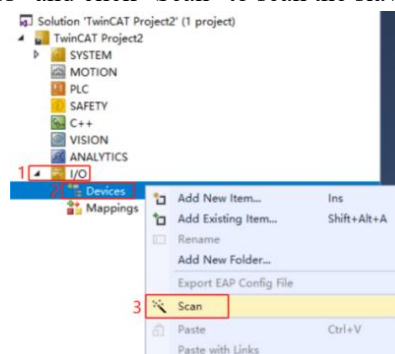
(2) As shown in the figure above, open the menu "TwinCAT" → "Show Realtime Ethernet Compatible Devices..." The following window is displayed. Select a local NIC (Network Interface Card) in the "Incompatible devices" column and click "Install". After successful installation, the NIC is displayed in "Installed and ready to use devices(for demo use only) " column:



(3) After the Ethernet driver is installed successfully, you can scan for slaves. According to the official instructions of BECKHOFF, the computer NIC should choose Intel 100 Mbit/s NIC, other NICs are not guaranteed to successfully install the driver.

3. Scan for slave

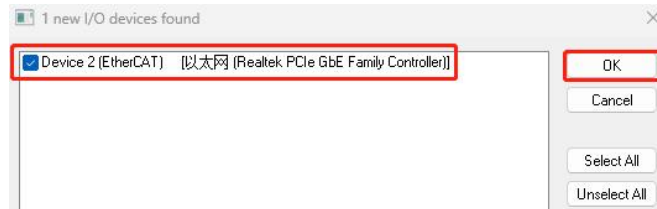
(1) Right click on "I/O" → "Devices" and click "Scan" to scan the slave.



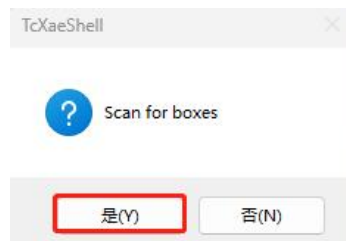
(2) As shown in the following figures, after scanning the slave, continue to click "Scan for boxes", and finally click "Append linked axis to NC - Configuration"



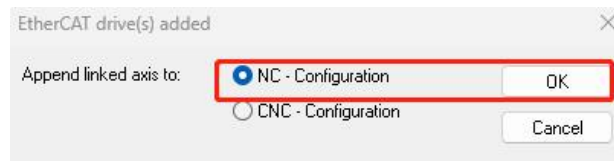
(a)



(b)



(c)



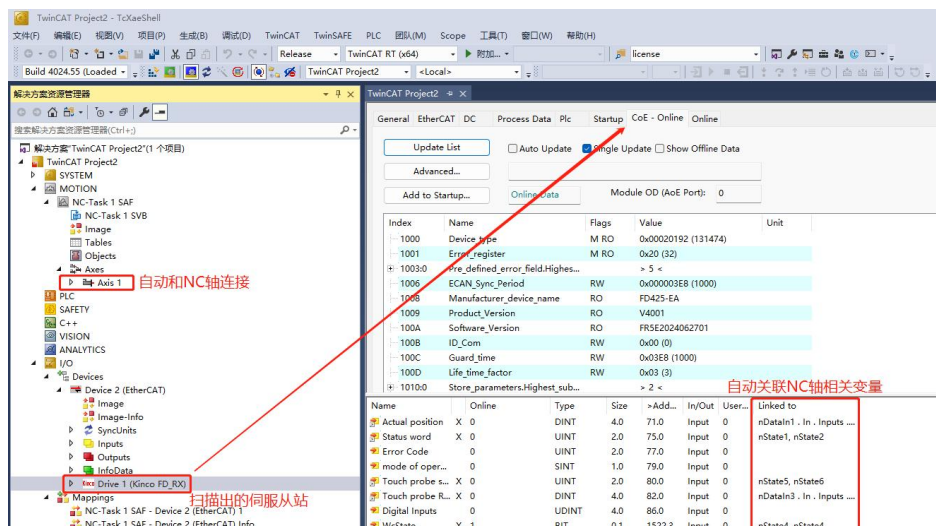
(d)

(3) After scanning the slave, you can see the slave information of "Drive1 (KincoFD_RX)" under "Devices", and it is automatically connected to the NC axis, TwinCAT automatically turns to the config mode .



If this blue gear does not appear, it will not be able to read and write SDO normally.

The scanning result is shown below.



4. Set drive parameters via COE-Online

(1) In the COE-Online interface you can set the DIN of the drive. The default functions of DIN1-DIN7 that must be canceled are Enable (value 1), Reset Errors (value 2) and Operate Mode Select (value 4). Others such as Position Limit (value 0x10 and 0x20) can be set as needed. For details, see the IO definition chapter of the drive manual.

2010:0	Group_DIO.Highest_sub_index...		> 41 <
2010:02	Group_DIO.Din_Simulate	RW	0x0000 (0)
2010:03	Group_DIO.Din1_Function	RW	0x0000 (0)
2010:04	Group_DIO.Din2_Function	RW	0x0000 (0)
2010:05	Group_DIO.Din3_Function	RW	0x0000 (0)
2010:06	Group_DIO.Din4_Function	RW	0x0000 (0)
2010:07	Group_DIO.Din5_Function	RW	0x0000 (0)
2010:08	Group_DIO.Din6_Function	RW	0x0000 (0)
2010:09	Group_DIO.Din7_Function	RW	0x0000 (0)

(2) Configure the appropriate motor for the drive, and fill in the object [641001] with the hexadecimal motor code. For details of the relevant motor codes, please refer to the motor chapter of the drive manual.

6410:0	Group_Motor		> 31 <
6410:01	Group_Motor.Motor_Num	RW	0x3542 (13634)

(3) Set the ECAN_Sync_Cycle [301101] of the drive, and enable the ECAN_Sync_Clock ([301102] value is 1), as below, the default has enabled the synchronization mode, and the ECAN_Sync_Cycle is 2ms ([301101] value is 1).

3011:0	Group_CAN.ECAN		> 8 <
3011:01	Group_CAN.ECAN_Sync_Cycle	RW	0x01 (1)
3011:02	Group_CAN.ECAN_Sync_Clock	RW	0x01 (1)
3011:03	Group_CAN.ECAN_Sync_Shift	RW	0x00 (0)
3011:04	Group_CAN.Sync_TPDO_Diff	RW	0

(4) As above, the drive parameters have been set. If it is the first time to set, you need to store the control parameters ([2FF001] is set to 1) and motor parameters ([2FF003] is set to 1), and it will take effect after restarting the drive with power off.

2FF0:02	Group_Panel.Store_Calibrate_D...	RW	0x00 (0)
2FF0:03	Group_Panel.Store_Motor_Data	RW	0x01 (1)

(5) After setting the drive parameters, let the drive work in 8 mode ([606000] is set to 8), NC control is possible.

6060	Modes_of_operation	RW P	8
------	--------------------	------	---

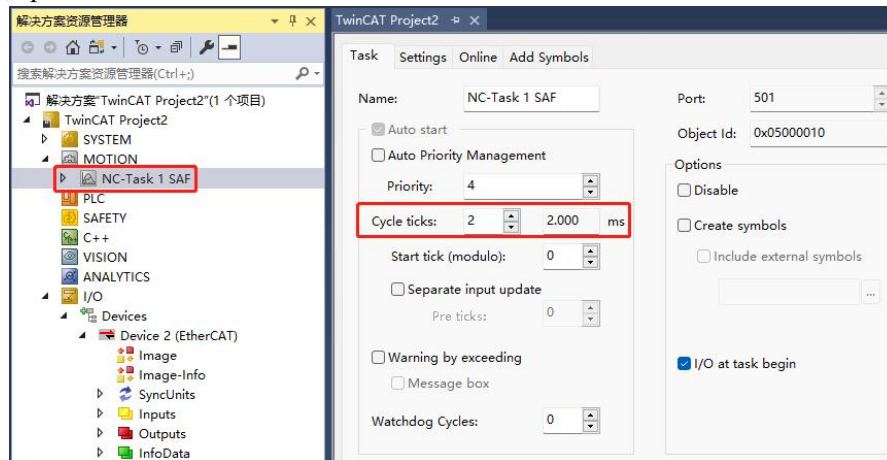
(6) If you need to adjust the drive PI and other performance parameters, you can set the speed loop (0x60F9 group) and position loop (0x60FB group), etc., for details, see the performance adjustment chapter of the drive manual. Note that the object units here are drive internal units, need to be converted, see the unit conversion chapter of the drive manual.

60F9:0	Group_Speed_Loop.Highest_sub_inde...		> 0 <
60FB:0	Group_Position_Loop.Highest_sub_ind...		> 1000 <

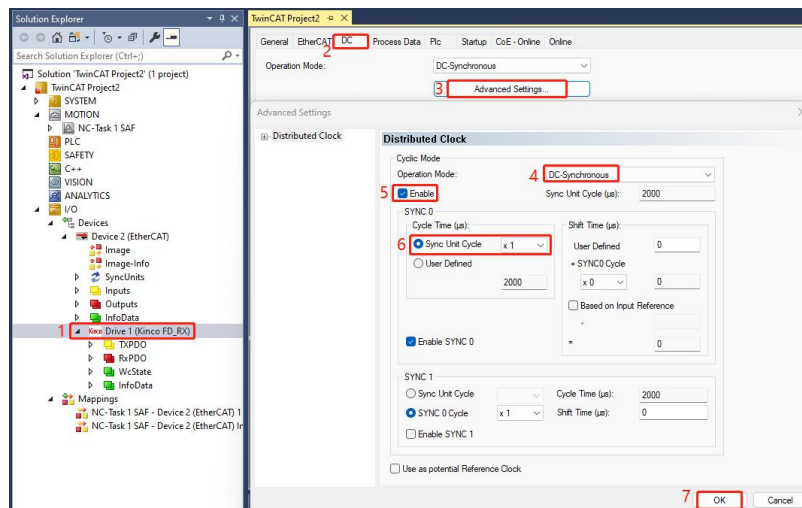
(7) If you can't find the object in COE-Online or want to monitor the drive more conveniently, please use the KincoServo software.

5. TwinCAT project setting

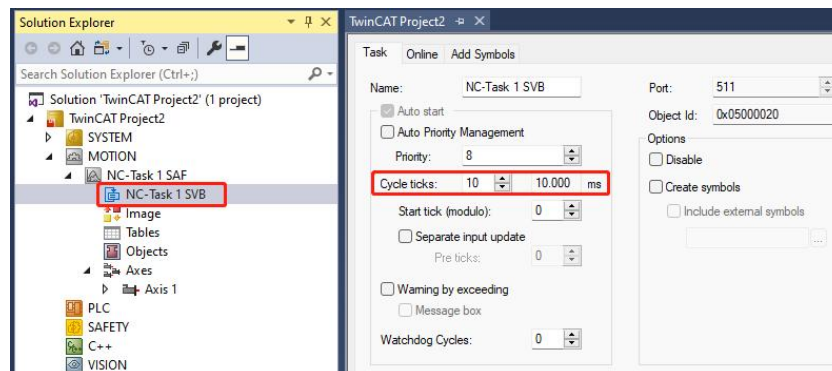
(1) Set the control cycle of the NC task, i.e. the cycle of the NC-Task SAF (the cycle in which the NC axis exchanges data with the drive), here set to 2 ms. In this task, TwinCAT NC completes the calculation of the set values for position, speed and acceleration.



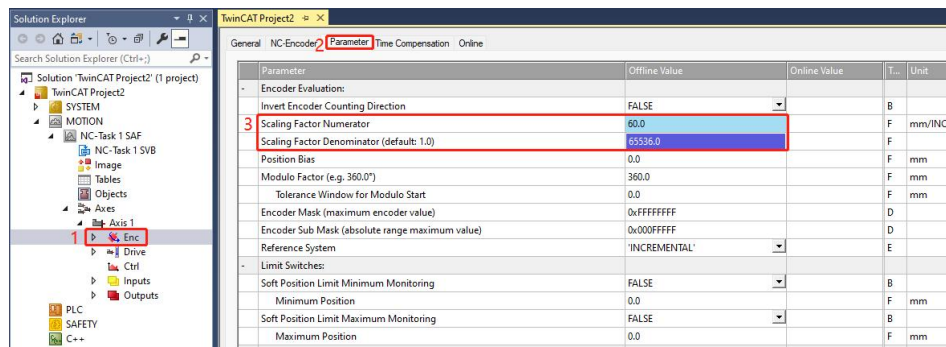
(2) Enable the distributed clock, note that the "Cycle Time" here should be the same as the ECAN_Sync_Cycle [301101] of the drive.



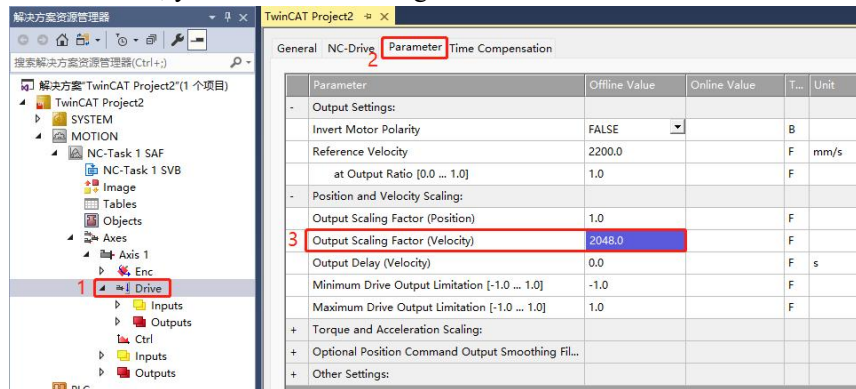
(3) The other NC cycle, the NC-Task SVB cycle, is the cycle in which the NC axis exchanges data with the PLC, with a typical value of 10ms.



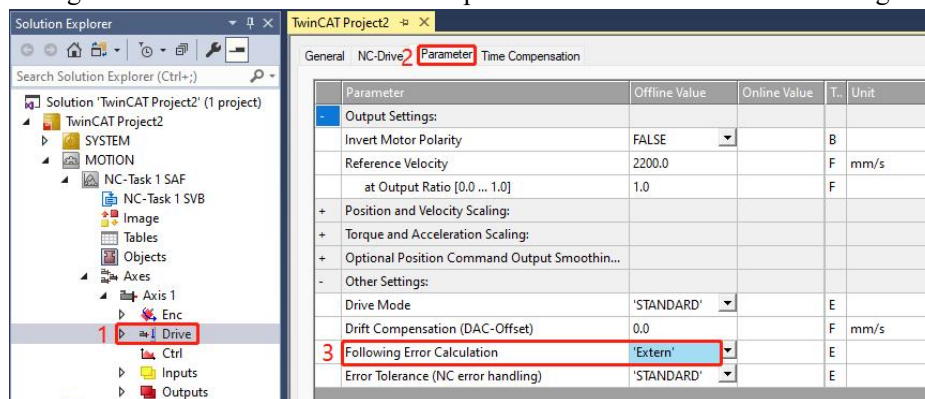
(4) In "Axis1" → "Enc", you can set the "Scaling Factor", i.e. the distance for each encoder pulse.



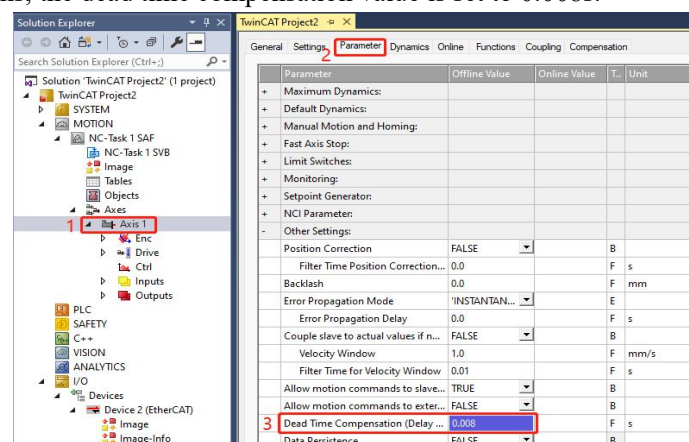
(5) In "Axis1" → "Drive", you can set the scaling factor of the set value.



(6) Set "Following Error Calculation" to "Extern" to prevent alarms of excessive following error.



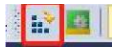
(7) Or increase the dead time compensation. Set the dead time compensation to an integer multiple of the synchronization cycle. 4 times the synchronization cycle is recommended. As shown in the following figure, if the synchronization cycle is 2ms, the dead time compensation value is set to 0.008s.



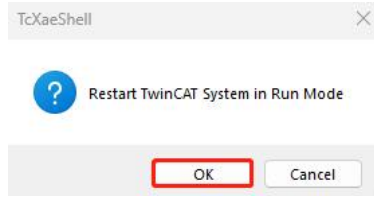
The above completes the basic settings of the drive in NC mode.

6. NC control

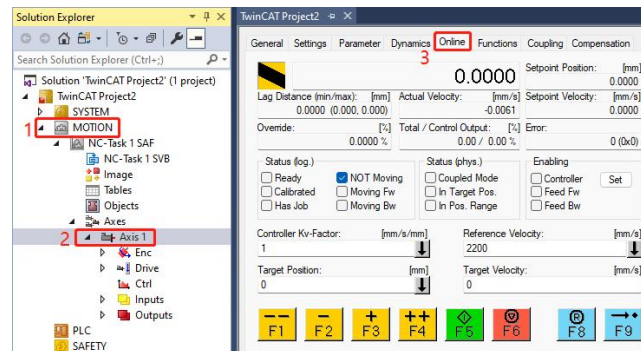
(1) After the above settings are completed, NC control can be carried out. First, Activate Configuration



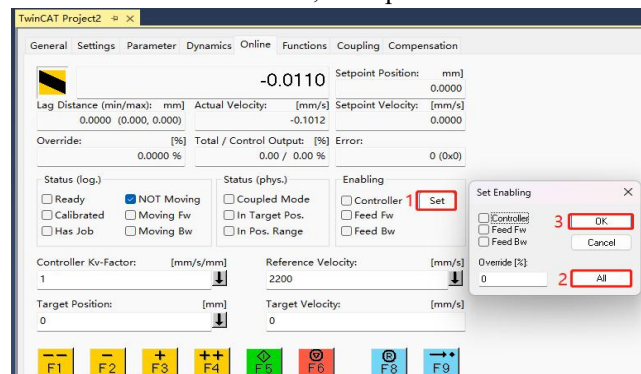
, click OK to turn to run mode, as shown in the following figure.



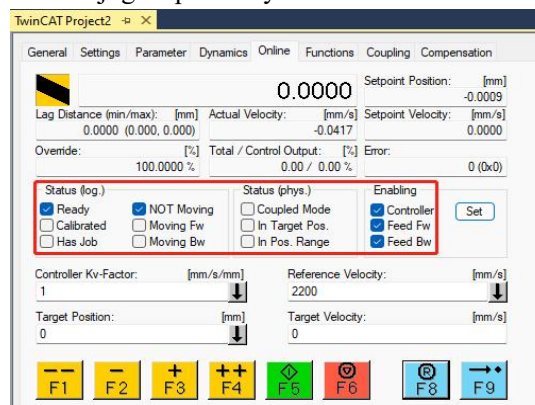
(2) Click on the "MOTION" → "Axis 1" → "Online" tab to commission the drive.



(3) Click "Set" in the current page, and then click "All" to enable the drive. The "Set Enabling" window allows you to set the axis enabling, forward and reverse rotation, and speed ratio.



(4) After enabling, the "Ready" status will be ticked, press F1 to F4 to perform the jog operation, the jog speed is set in "Manual Velocity" in "MOTION" → "Axis 1" → "Parameter" tab, the default speed is 100 mm/s and 600 mm/s, which corresponds to slow and fast jog respectively.



The above is an example of direct servo control with TwinCAT, more TwinCAT NC control methods see the official manual of BECKHOFF.

Servo control with PLC

When BECKHOFF PLC communicates with a single drive, use the network cable to connect the EtherCAT port of the PLC and the IN interface of the drive; when the PLC connects with multiple drives, connect the OUT interface of the previous drive to the IN interface of the next drive as shown in the figure below (take BECKHOFF CX5020 as an example).

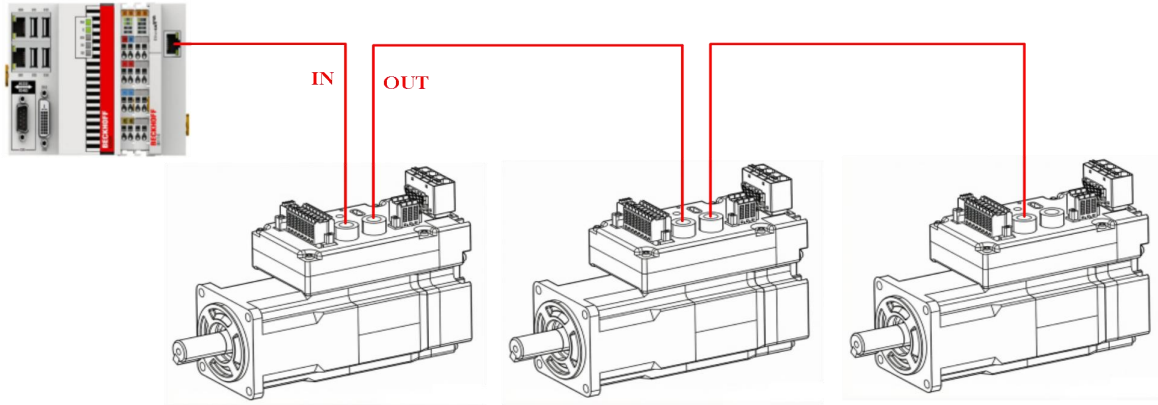
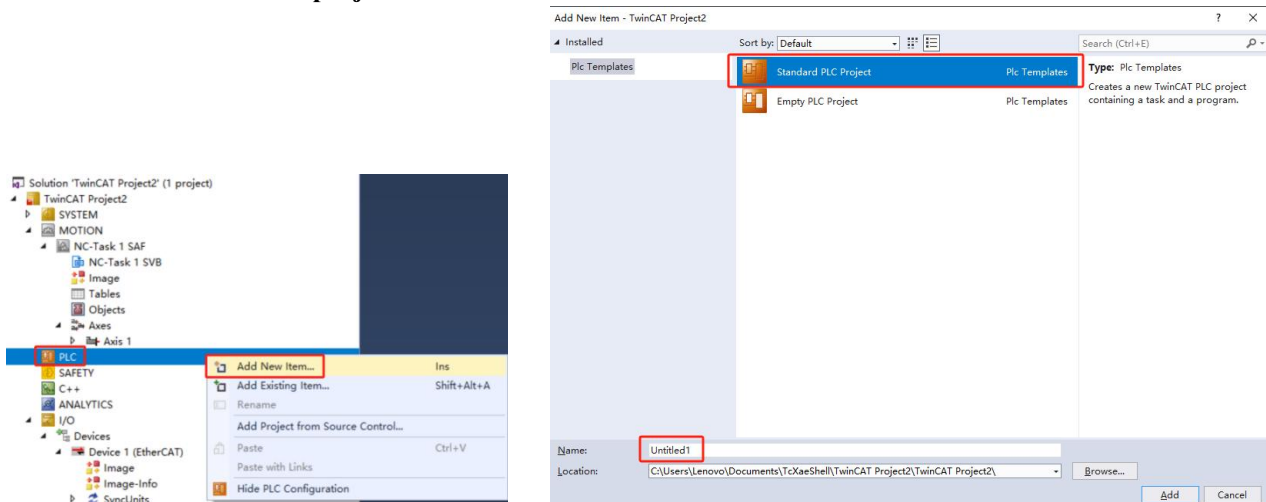
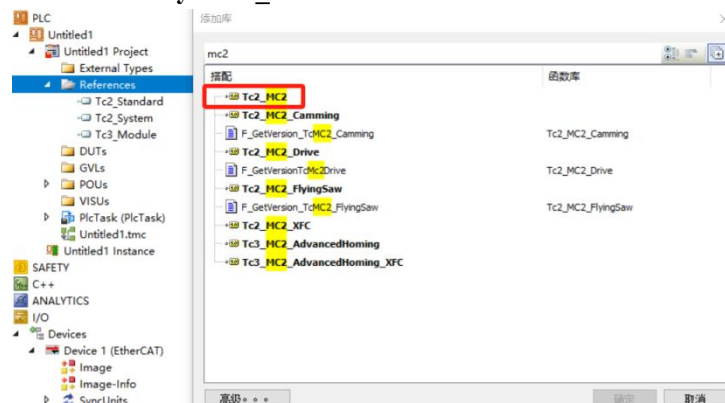


Figure 13-19 BECKHOFF CX5020 connecting drives

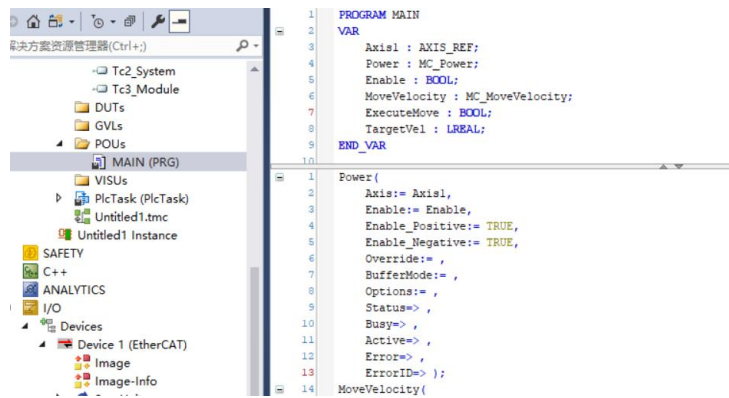
1. Create a new PLC project



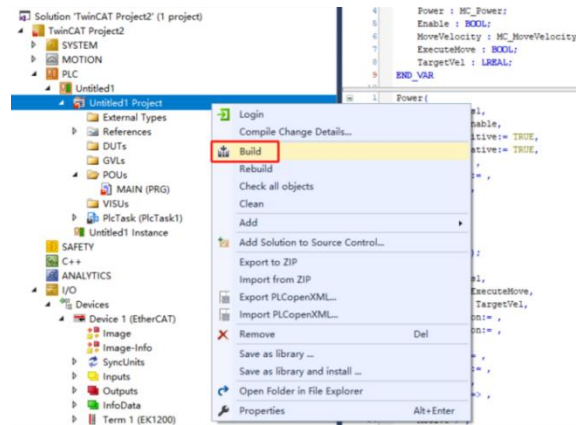
2. Add the motion control library "Tc2_MC2"



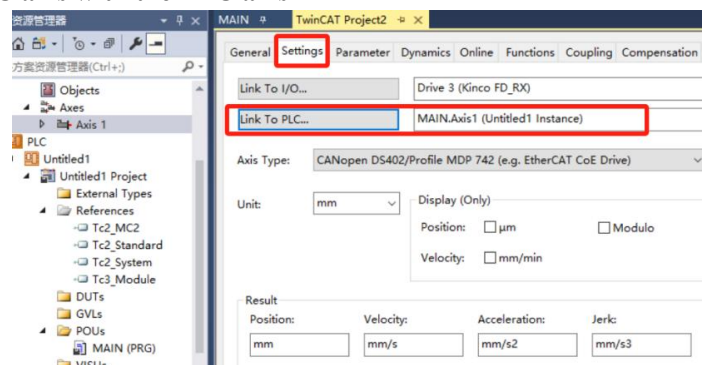
3. Call the function blocks and write a program



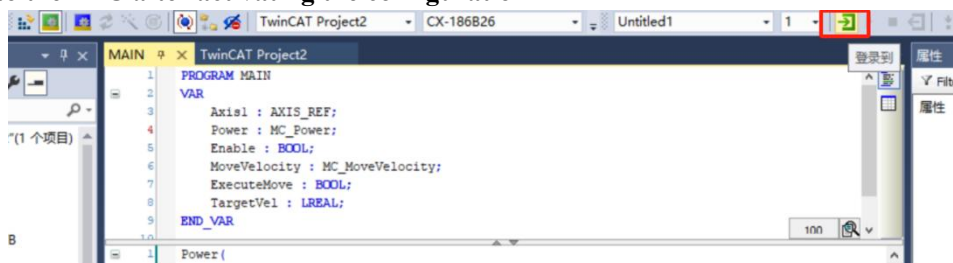
4. Compile the program



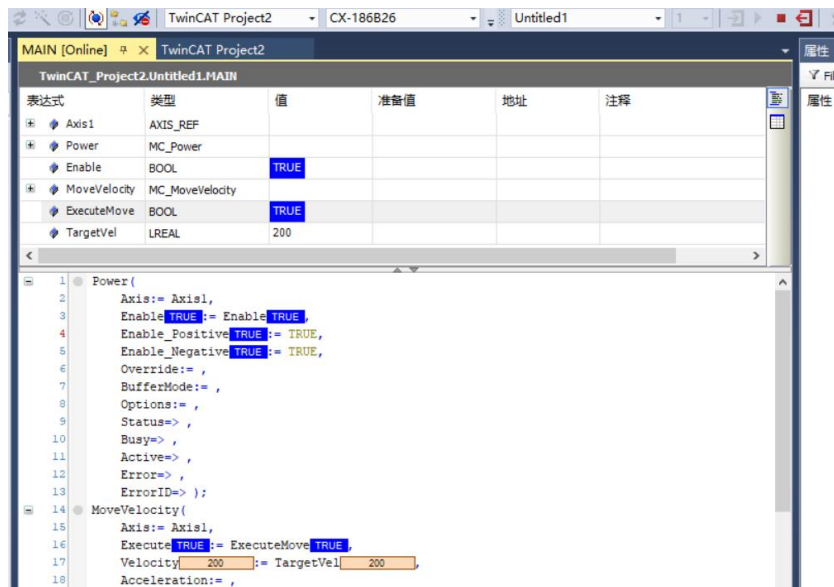
5. Associate the NC axis with the PLC axis



6. Log in to the PLC after activating the configuration



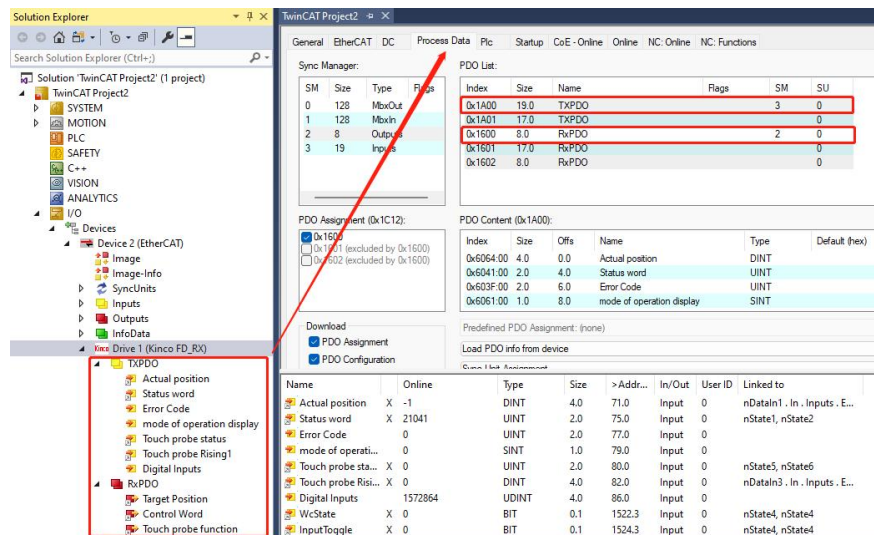
7. Run the program to control the servo motion



PDO configuration

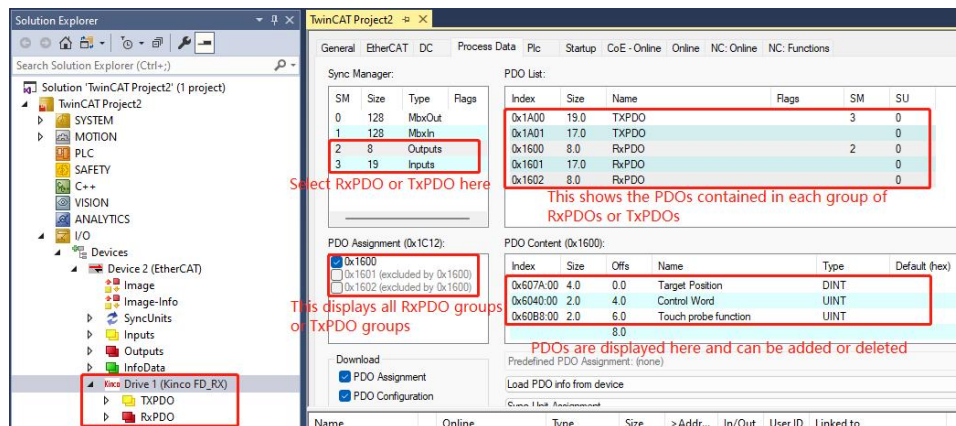
1. The following figure shows the default RxPDO group (0x1600) and TxPDO group (0x1A00) of the drive.

The default parameters are sufficient for NC control.

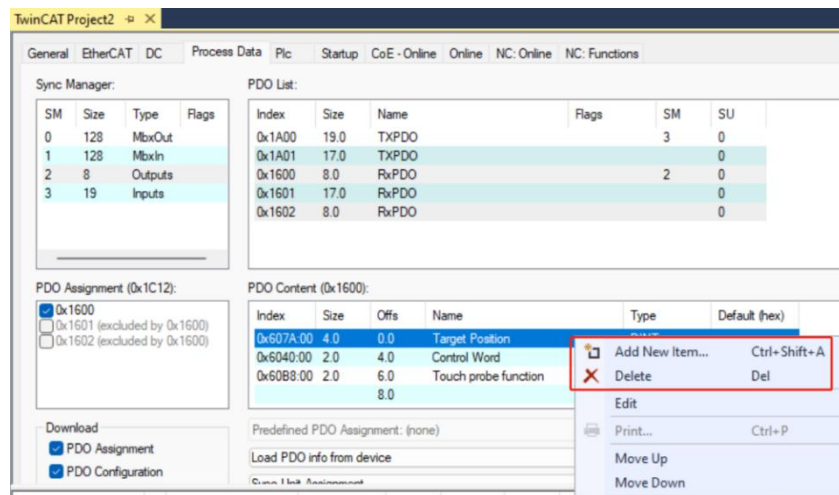


2. If the default PDO group does not meet the requirements, then refer to the following figure to select another

PDO group.

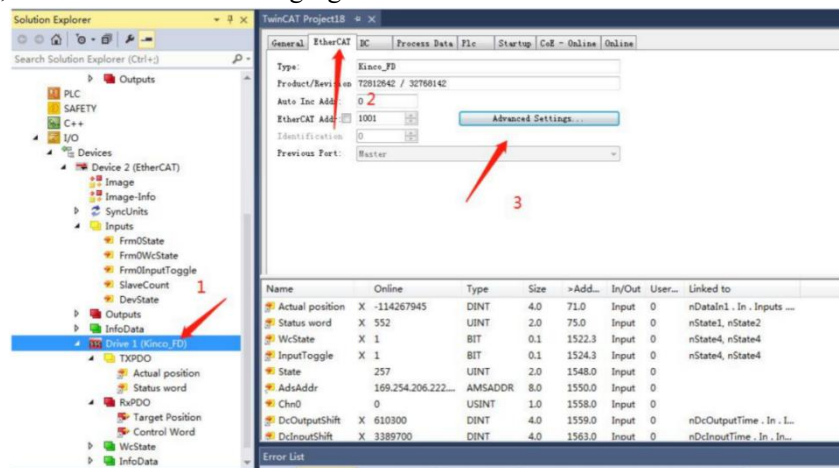


3. If you do not have the desired object in the PDO group, you can add or delete it. Right-click in the "PDO Content" window to remove the default PDO and add the required PDO.

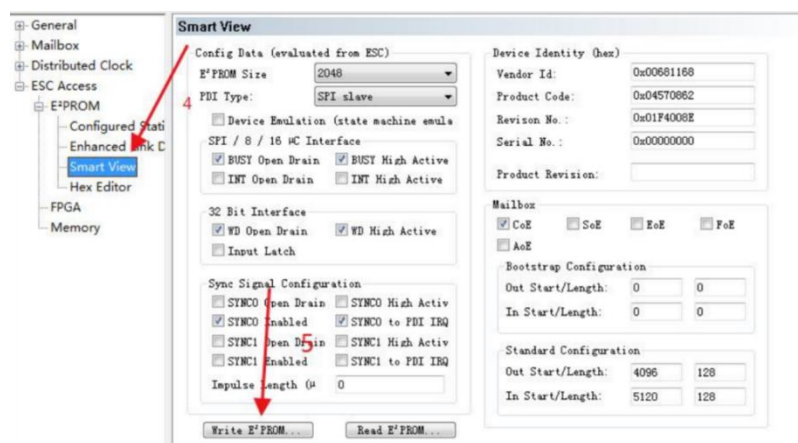


Write slave E2PROM

1. If TwinCAT has already scanned the slave, but the slave Drive1 (Kinco FD) has a question mark next to it or the PDO cannot be read or written due to a mismatch in the XML file, then you need to write the correct XML into the drive's E²PROM, as shown in the following figures.



(a)



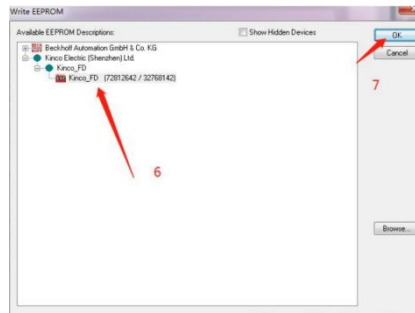
(b)

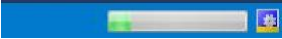


Note

- Before writing E2PROM, you need to copy the drive XML file to the TwinCAT installation directory, default path C:\TwinCAT3.1\Config\Io\EtherCAT.

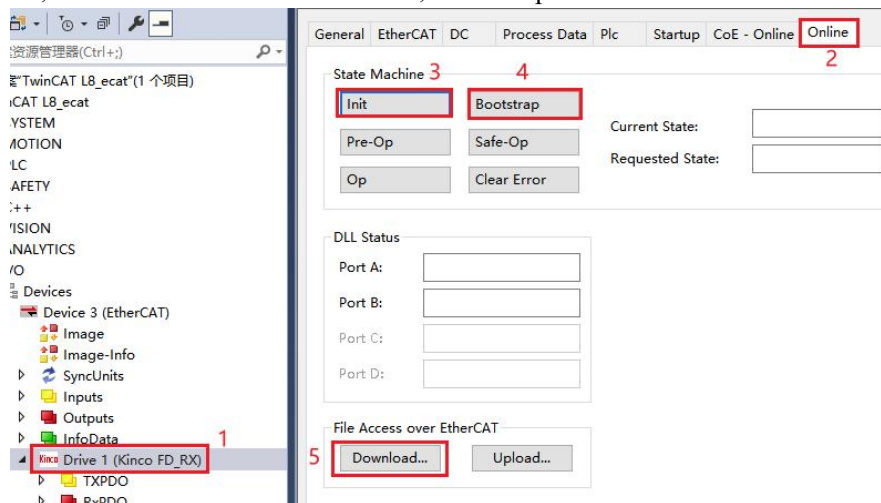
1. After clicking "Write E²PROM" the following window will pop up, in the window will appear in the list of XML has been added, select the correct XML, click OK to download.



2. At the same time, there is a progress bar  in the lower right corner of the TwinCAT window, please observe the status of the progress bar. After the operation is finished, please power off and restart the PLC and drive, and re-scan the slave to make the newly written XML take effect.

FoE upgrades the slave firmware

FoE supports the firmware download function from the master station to the slave station in the boot state via EtherCAT communication. Before downloading, the Kinco JD2 server needs to be switched to the Init state first, then to the Boot state, and then click Download, and select the firmware in the .elf format for downloading. After the download is complete, switch the server to the Init state, and then power off and restart the server.



13.3.3.4 OMRON NJ series controller application

When Omron NJ series PLC controller communicates with a single drive, directly use the network cable to connect the EtherCAT interface of the controller and the IN interface of the drive; when the controller connects with multiple drives, connect the OUT interface of the previous drive to the IN interface of the next drive.

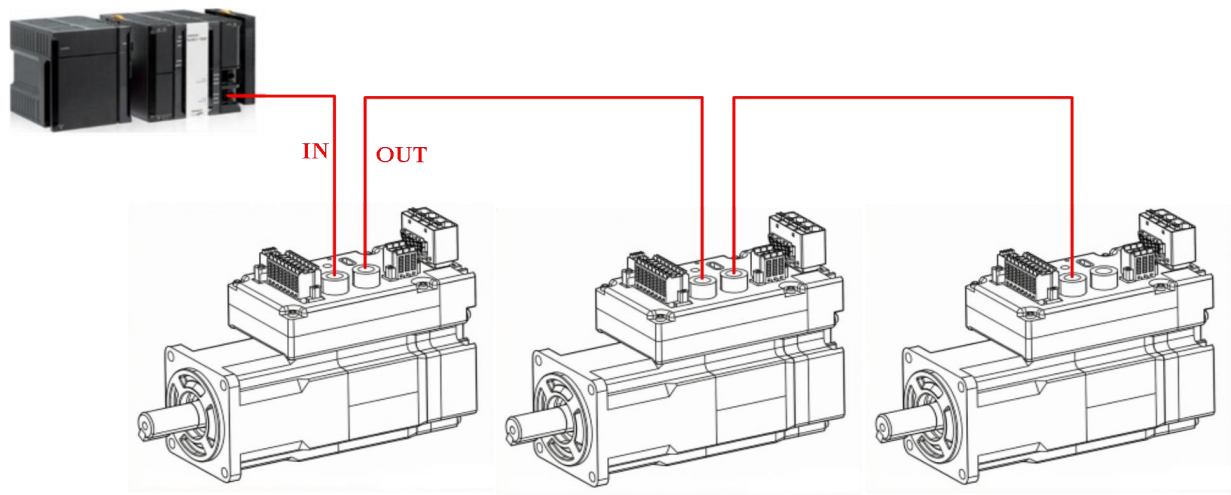
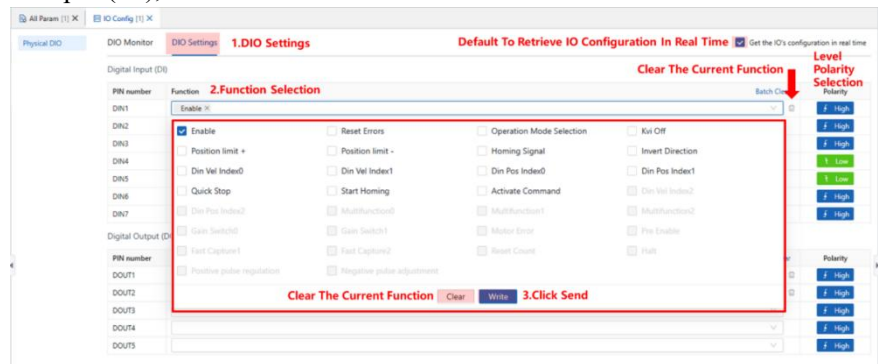


Figure 13-20 Omron NJ controller connecting drives

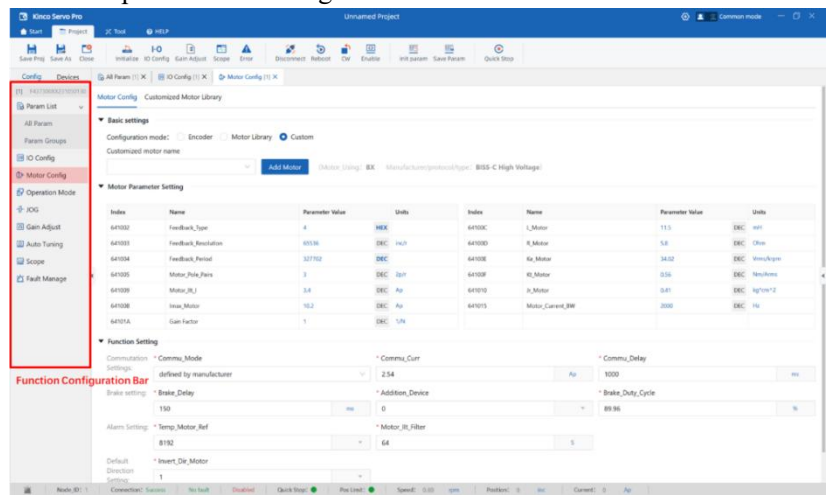
Drive parameter setting

Use KincoServoPRO software to set the drive parameters, refer to the drive user manual for details, and only the parameters that must be set and the function description of commonly used parameters are listed below.

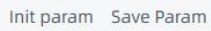
- 1. In the digital input (DI), functions such as limit functions can be set as needed.



- 2. Configure the corresponding motor. In the function configuration section, find the **Motor Config** , as shown in the following figure. iSMD offers three configuration methods. It can read the parameters stored internally in the encoder or read the parameters stored in the motor's inventory; for third-party motors, custom configuration can be done based on the motor parameter drawings.



3. When using Omron controller, the EtherCAT communication station number must be set through the Node_ID [100B00]. It is recommended to set it according to the actual physical connection sequence to facilitate management and configuration.

4. After the servo parameters are set, it is necessary to save the control parameters and motor parameters. Click the [Init param] and [Save param] buttons  in the project directory.

5. If it is necessary to adjust parameters such as the servo PI, set the parameters of the speed loop and position loop.

Controller parameter setting

1. The drive XML file can be downloaded from the official website of Kinco.

2. Copy the drive XML file to the NJ series programming software Sysmac Studio installation directory, such as:

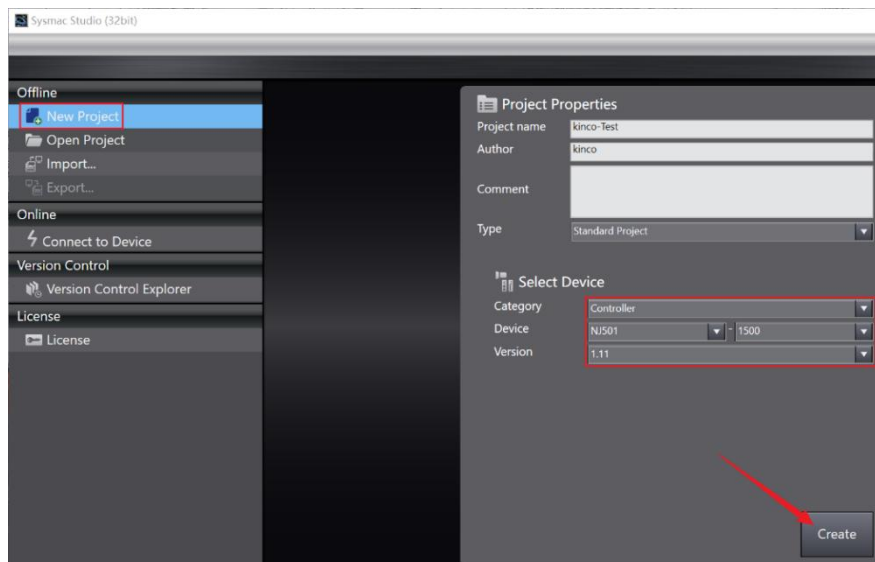
C:\Program Files\OMRON\Sysmac Studio\IODeviceProfiles\EsiFiles\SystemEsiFiles

When the right toolbar can not find the drive slave, you can add the XML file to:

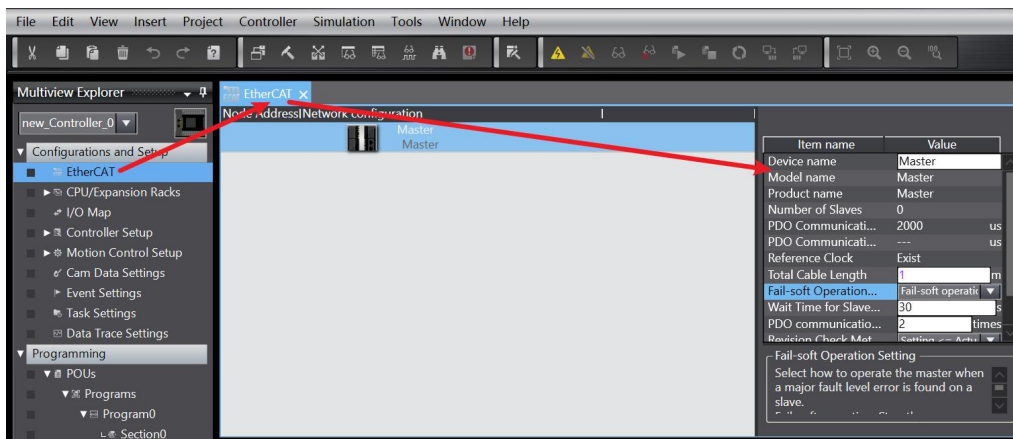
C:\Program Files\OMRON\Sysmac Studio\IODeviceProfiles\EsiFiles\UserEsiFiles

If you need to add or change XML files, please exit Sysmac Studio first. The XML files will not take effect until Sysmac Studio is restarted.

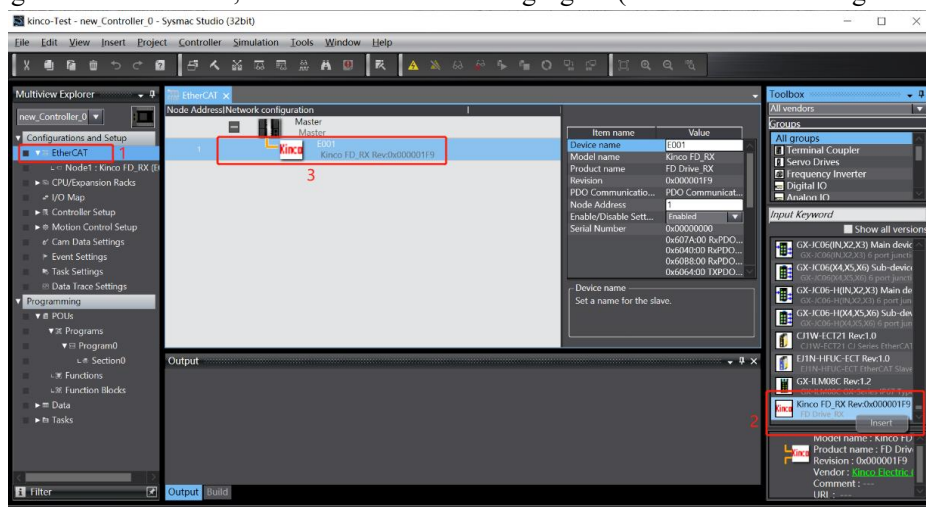
3. Connect the EtherNet port of the computer and the controller with a network cable, open Sysmac Studio, click New Project, and select the device and version information of the connected controller (found in the Controller product TAB).



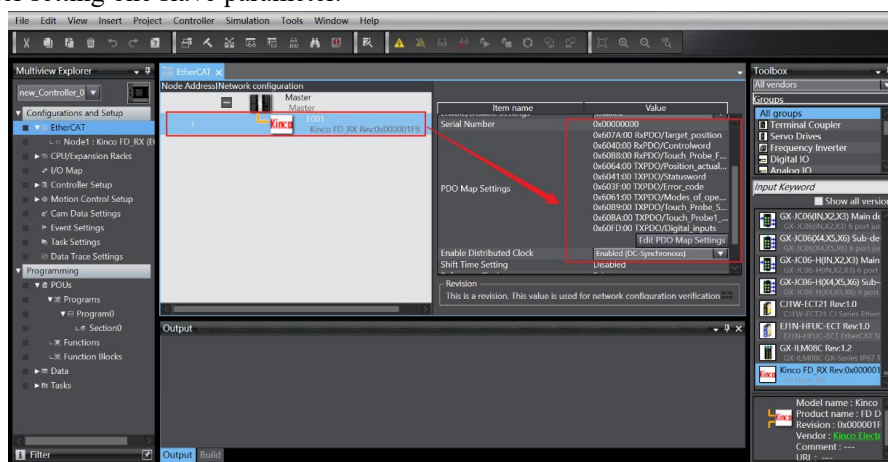
4. After creating the project, double-click "EtherCAT" in "Configurations and Setup" on the left, and then bring up the master view. Then set the parameters of the right master station (generally, the parameters can be set by default).



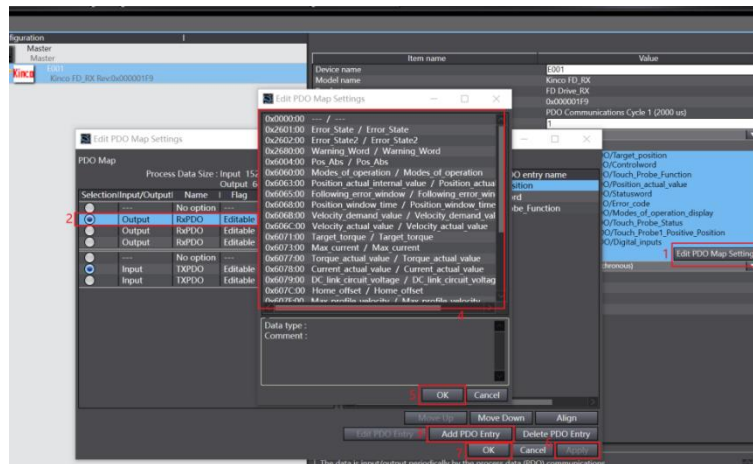
5. Find the kinco slave at the bottom of the right toolbox, right click Insert, add the connected slave (insert a few of the actual connections), after adding the slave, the main view area and the left configuration area will display the slave and assign the node address, as shown in the following figure (the test takes the single axis as an example).



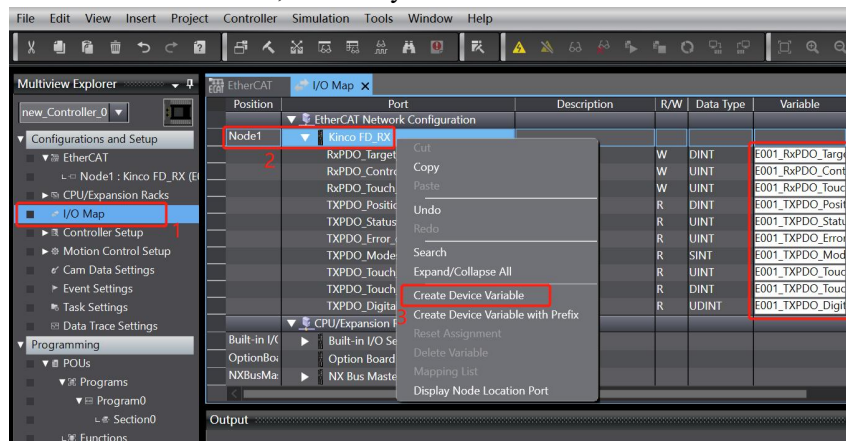
6. Click the slave to set the slave parameters. Generally, the default parameters in the box as shown below can meet the basic motion control requirements. If the parameters of multiple slave stations are set, you can right-click to copy → paste after setting one slave parameter.



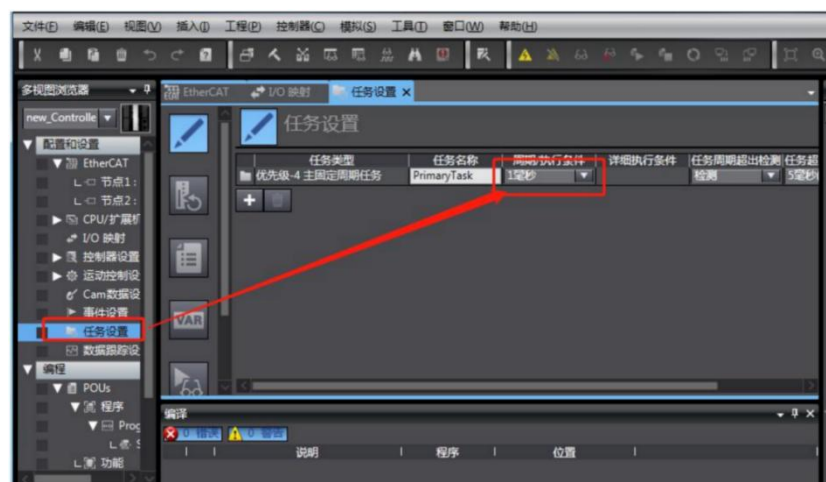
If there are other control requirements (for example, if there is no motion control, most of the PDOs need to be configured only as an ordinary EtherCAT slave), you need to add more PDOs. Click **Edit PDO Map Settings**, select output RPDO channel or input TPDO channel, click Add PDO Entry, and select the PDO to be added in the PDO menu bar that pops up. Then click Apply and OK.



7. Associate the drive PDO to the controller local IO. In the left menu bar Configuration and Setup, double-click the I/O Map, then select the slave (note that only one slave can be set at a time, all slaves need to be set), right-click and select Create Device Variable, and the system will allocate the local IO variables.



8. Sets the period in the task settings. In the Configurations and Setup area, double-click Task Settings to set the period of the main fixed-cycle task. Note that this task cannot be deleted and the period must correspond to the sync cycle set by the drive. Otherwise, it will cause the loss of sync messages and cause the motor operation oscillation!

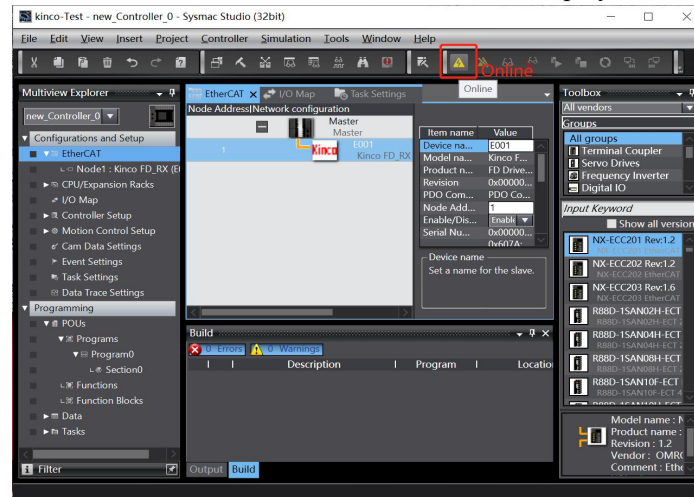


9. Select Project → Check All Programs for errors.

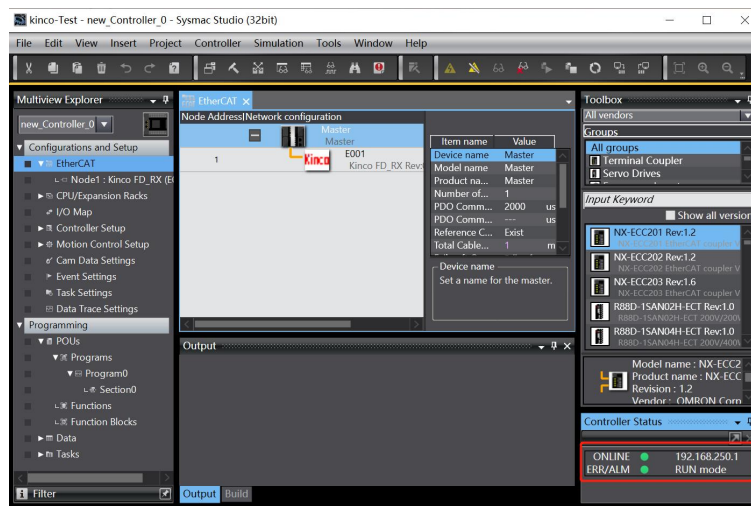
10. The basic communication parameters are set, and the program can be downloaded and debugged:

(1) Select Controller → Communication Setup, select the programming port used (such as network port).

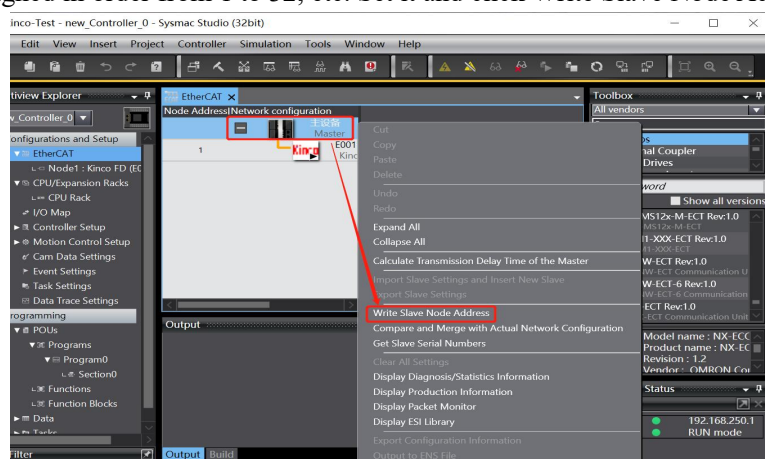
(2) Select Controller → Online. After online, the controller status will be displayed in the bottom right corner.



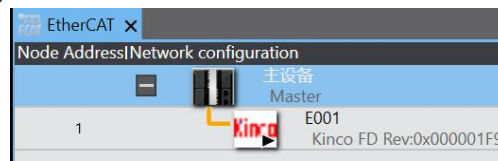
(3) Select Controller → Synchronize. After synchronization, the comparison between the controller and the current local engineering program and parameters will be displayed. Select transfer to the controller, select "Yes" in the prompt box to enter the programming mode, and click "Yes" after the transfer is completed to enter the running mode.



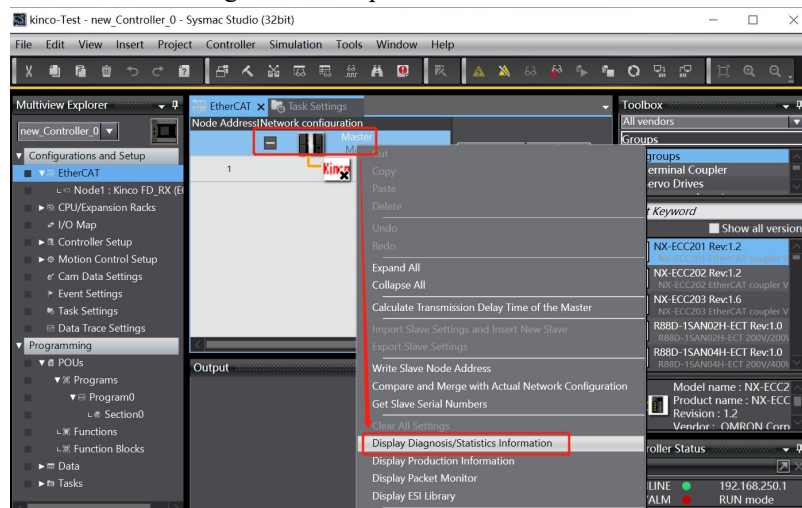
11. If more than one drive is used, you can go back to the EtherCAT settings page, select the master, right-click to reallocate the slave node address, and assign the slave node address in the settings window that appears. Generally, the slave address is assigned in order from 1 to 32, etc. Set it and click Write Slave Node Address.



12. Power off and reboot the controller and drive, power on and online again. The monitor shows the slave station has been connected normally.



If the connection is wrong, the box below shows a cross instead of a triangular arrow. Click the Master, right click to display diagnosis/statistics information, can monitor the network status. If there is an error, you can select the Toolbar → Tools → Troubleshooting to see the specific error information.



13. The corresponding slave data can be viewed and read in the IO map, indicating successful communication.

位置	端口	说明	R/W	数据类型	值	变量
节点1	Kinco FD					
		RxPDO_Target_position_607A_00	W	DINT	0	
		RxPDO_Controlword_6040_00	W	UINT	0	
		RxPDO_Touch_Probe_Function_60B8_00	W	UINT	0	
		TXPDO_Position_actual_value_6064_00	R	DINT	1188971479	
		TXPDO_Statusword_6041_00	R	UINT	536	
		TXPDO_Error_code_603F_00	R	UINT	29472	
		TXPDO_Modes_of_operation_display_6061_00	R	SINT	0	
		TXPDO_Touch_Probe_Status_60B9_00	R	UINT	0	
		TXPDO_Touch_Probe1_Positive_Posit_60BA_00	R	DINT	0	
		TXPDO_Digital_inputs_60FD_00	R	UDINT	0	

Can also view the three indicators of the EtherCAT port in the lower right corner of the controller body. The meaning of the status of each indicator is as follows:

Indicator	Color	Status	Meaning
NET RUN	Green	Lit.	EtherCAT communications are in progress. • Inputs and outputs for I/O data are in operation.
		Flashing	EtherCAT communications are established. This indicator shows either of the following conditions. • Only message communications are in operation. • Only message communications and I/O data inputs are in operation.
		Not lit.	EtherCAT communications are stopped. • The power supply is OFF or the CPU Unit was reset. • A MAC address error, communications Controller error, or other error occurred.
NET ERR	Red	Lit.	A hardware error or unrecoverable error occurred, such as for exception processing.
		Flashing	A recoverable error occurred.
		Not lit.	There are no errors.
LINK/ACT	Yellow	Lit.	A link is established.
		Flashing	Data communications are in progress after establishing link. Flashes every time data is sent or received.
		Not lit.	The link was not established.

At this time, in the KincoServoPro software, click Operation mode → EtherCAT, you can see the PDO

information of successful configuration, as shown below. You can also see whether there is sync data in the "Others" (the ECAN_Sync is not 0, which means there is sync data and has been synchronized).

Index	Name	Current value	Mapping object	Mapping object value	Mapping Unit
1A0000	Group_TX1_P...	7	-	-	-
1A0001	TX1_PDO1	60640020	Pos_Actual	52	D inc
1A0002	TX1_PDO2	60410010	Statusword	5231	H
1A0003	TX1_PDO3	603F0010	Error_Code	0	H
1A0004	TX1_PDO4	60610008	Operation_Mode_Buff	8	D
1A0005	TX1_PDO5	60890010	Touch_Probe_Status	0	H
1A0006	TX1_PDO6	608A0020	Touch_Probe_Rising1	0	D
1A0007	TX1_PDO7	60FD0020	Digital_Inputs	180000	H
1A0008	TX1_PDO8	0	-	-	-

Index	Name	Current value	Mapping object	Mapping object value	Mapping Unit
160000	Group_RX1_P...	3	-	-	-
160001	RX1_PDO1	607A0020	Target_Position	0	D inc
160002	RX1_PDO2	60400010	Controlword	6	H
160003	RX1_PDO3	60880010	Touch_Probe_Function	0	H
160004	RX1_PDO4	0	-	-	-
160005	RX1_PDO5	0	-	-	-
160006	RX1_PDO6	0	-	-	-
160007	RX1_PDO7	0	-	-	-
160008	RX1_PDO8	0	-	-	-

ECAN Configuration

ECAN_Sync_Cycle: 1ms

ECAN_Sync_Shift: 0

ECAN_Sync: 500E

ECAN_Sync_Clock: clock sync

Sync_TPDO_Diff: 0

According to the above information, it means that the controller has successfully communicated with the EtherCAT of the drive.

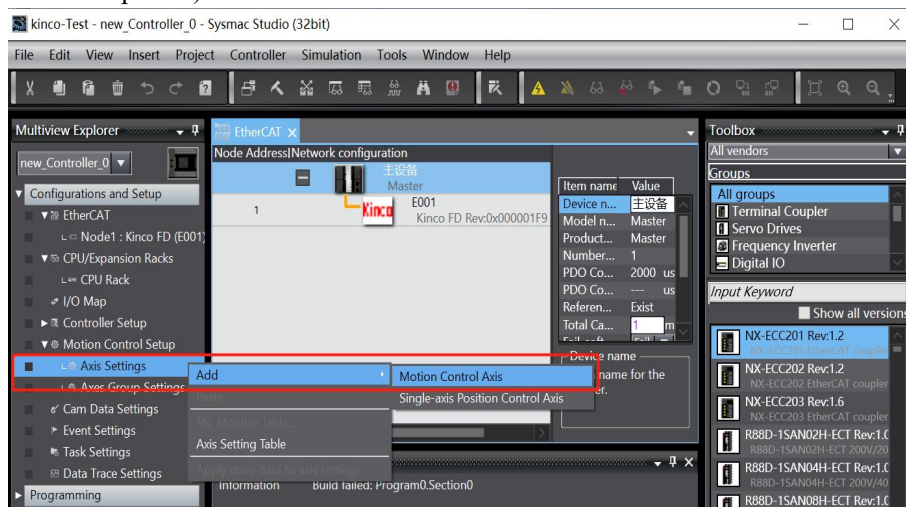
Programming control axis motion

1. Point to point control

After the communication is established, the variables mapped to the controller by IO in the program can be assigned to each object of the drive, to control the drive motion. For example, a value can be assigned to the target velocity, the control word is assigned to 0xF, and the operation mode is set to 3 to run the velocity mode. See the drive user manual and controller programming information for details.

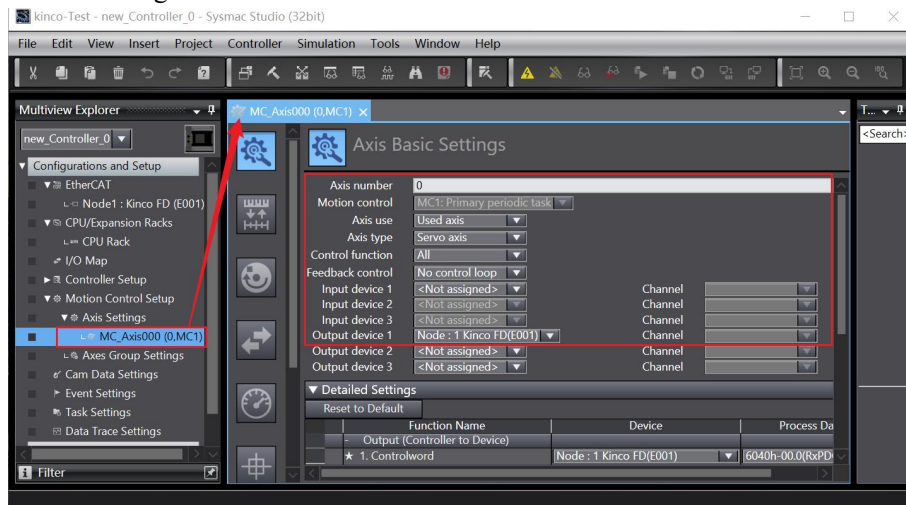
2. Motion control axis configuration

(1) Add axis. In Configurations and Setup, double-click Motion Control Setup, select Axis Settings, click Add → Motion Control Axis, the project needs to control a few axes to add a few axes (must actually connect these axes, otherwise an error will be reported).



(2) Associate PDO to axis control variables. Double-click the axis you added to configure the axis. With the motion

control axis, the point-to-point normal control can no longer be used, and the IO mapping will be automatically removed when the axis is configured.

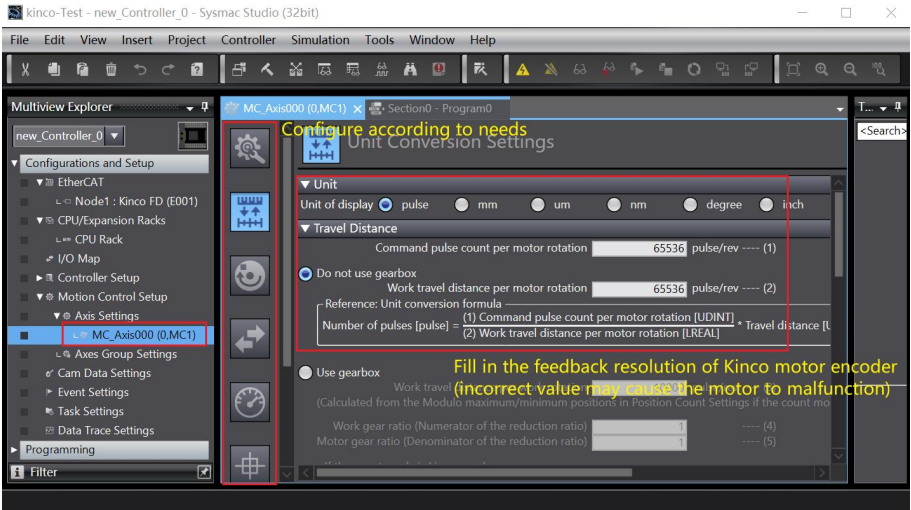


To use the instructions of the MC function module for the axis, the following objects must be set (if the corresponding PDO map is not selected, please refer to "[Edit PDO Settings](#)" to add it).

	Function Name	Device	Process Data
	- Output (Controller to Device)		
★ 1.	Controlword	Node : 1 Kinco FD(E001)	6040h-00.0(RxPDO_Co
★ 3.	Target position	Node : 1 Kinco FD(E001)	607Ah-00.0(RxPDO_Tar
5.	Target velocity	<Not assigned>	<Not assigned>
7.	Target torque	<Not assigned>	<Not assigned>
9.	Max profile Velocity	<Not assigned>	<Not assigned>
11.	Modes of operation	Node : 1 Kinco FD(E001)	6060h-00.0(RxPDO_Mc
15.	Positive torque limit value	<Not assigned>	<Not assigned>
16.	Negative torque limit value	<Not assigned>	<Not assigned>
21.	Touch probe function	<Not assigned>	<Not assigned>
44.	Software Switch of Encoder's Input	<Not assigned>	<Not assigned>
	- Input (Device to Controller)		
★ 22.	Statusword	Node : 1 Kinco FD(E001)	6041h-00.0(TXPDO_Sta
★ 23.	Position actual value	Node : 1 Kinco FD(E001)	6064h-00.0(TXPDO_Po
24.	Velocity actual value	<Not assigned>	<Not assigned>
25.	Torque actual value	<Not assigned>	<Not assigned>
27.	Modes of operation display	Node : 1 Kinco FD(E001)	6061h-00.0(TXPDO_Mc
40.	Touch probe status	<Not assigned>	<Not assigned>
41.	Touch probe pos1 pos value	<Not assigned>	<Not assigned>
42.	Touch probe pos2 pos value	<Not assigned>	<Not assigned>

For configuration of other input and output PDOs, refer to the controller manual. If PDO is not necessary, it is not recommended to configure. Too much configuration will occupy the EtherCAT bus resources and affect the transmission efficiency.

(3) Set axis parameters such as encoder resolution, maximum velocity, home definition, position limit, etc. Please refer to the controller manual for details.



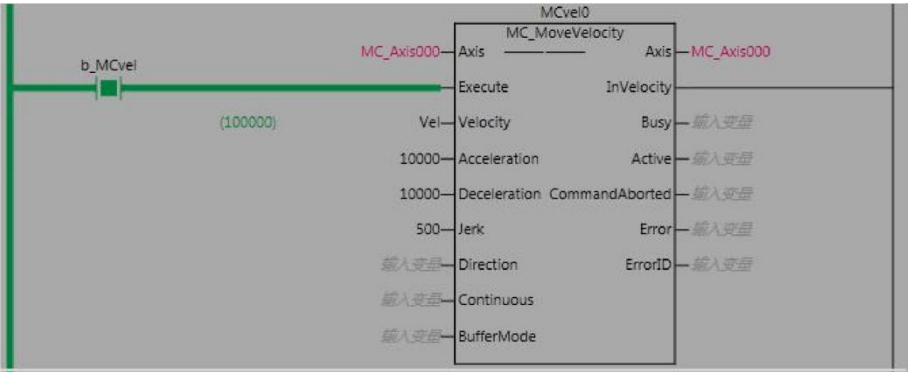
NJ series motion control instructions conform to PLCopen standard. Kinco drive is the standard EtherCAT slave. The following is a brief description of CSP mode, CSV mode, and the MC_Home homing instruction. See the controller manual for more MC instructions.

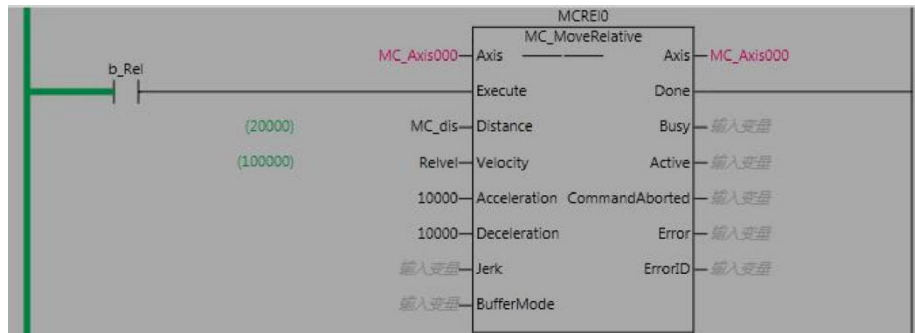
3. Cyclic Synchronous Position mode (CSP) instruction

This mode can meet the requirements of most working conditions. For specific application instructions, please refer to the controller manual. The following PDOs need to be configured for this mode.

恢复默认值	功能名称	设备	过程数据
	输出(控制器到设备)		
★	1. Controlword	节点1 Kinco FD(E001)	6040h-00.0(RxPDO_Co)
★	3. Target position	节点1 Kinco FD(E001)	607Ah-00.0(RxPDO_Ta)
	5. Target velocity	<未分配>	<未分配>
	7. Target torque	<未分配>	<未分配>
	9. Max profile Velocity	<未分配>	<未分配>
	11. Modes of operation	节点1 Kinco FD(E001)	6060h-00.0(RxPDO_Mc)
	15. Positive torque limit value	<未分配>	<未分配>
	16. Negative torque limit value	<未分配>	<未分配>
	21. Touch probe function	<未分配>	<未分配>
	14. Software Switch of Encoder's Input	<未分配>	<未分配>
	输入(设备到控制器)		
★	22. Statusword	节点1 Kinco FD(E001)	6041h-00.0(TXPDO_St)
★	23. Position actual value	节点1 Kinco FD(E001)	6064h-00.0(TXPDO_Po)
	24. Velocity actual value	<未分配>	<未分配>
	25. Torque actual value	<未分配>	<未分配>
	27. Modes of operation display	<未分配>	<未分配>

MC_MoveVelocity instruction



MC_MoveRelative instruction**Note**

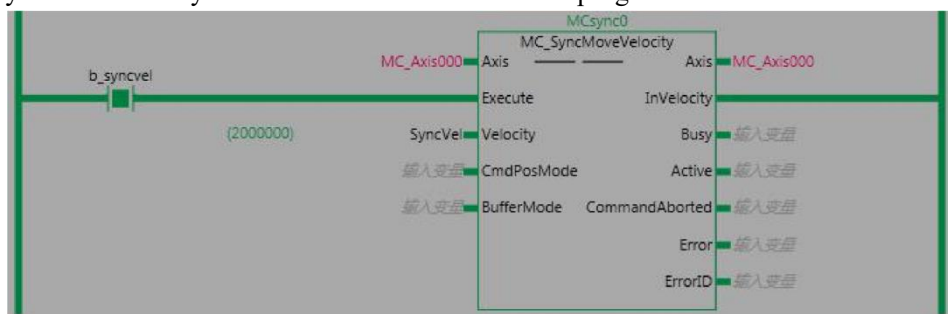
- Calculate the actual velocity (RPM) of the motor according to the feedback resolution of the motor encoder. Because the instruction is CSP mode instruction, the actual velocity of the motor is about $100000/65536 * 60 = 92$ RPM. (65536 is the feedback resolution of the kinco motor encoder)

4. Cyclic Synchronous Velocity mode (CSV) instruction

To use this instruction, the target velocity (0x60FF) must be mapped in the controller's axis settings. Here are the PDOs that need to be configured for this mode.

- Output (Controller to Device)			
★ 1. Controlword	Node : 1 Kinco FD(E001)	▼	6040h-00.0(RxPDO_Co ▼)
★ 3. Target position	Node : 1 Kinco FD(E001)	▼	607Ah-00.0(RxPDO_Tar ▼)
5. Target velocity	Node : 1 Kinco FD(E001)	▼	60FFh-00.0(RxPDO_Tar ▼)
7. Target torque	<Not assigned>	▼	<Not assigned>
9. Max profile Velocity	<Not assigned>	▼	<Not assigned>
11. Modes of operation	Node : 1 Kinco FD(E001)	▼	6060h-00.0(RxPDO_Mc ▼)
15. Positive torque limit value	<Not assigned>	▼	<Not assigned>
16. Negative torque limit value	<Not assigned>	▼	<Not assigned>
21. Touch probe function	<Not assigned>	▼	<Not assigned>
44. Software Switch of Encoder's Input	<Not assigned>	▼	<Not assigned>
- Input (Device to Controller)			
★ 22. Statusword	Node : 1 Kinco FD(E001)	▼	6041h-00.0(TXPDO_Sta ▼)
★ 23. Position actual value	Node : 1 Kinco FD(E001)	▼	6064h-00.0(TXPDO_Po ▼)
24. Velocity actual value	<Not assigned>	▼	<Not assigned>
25. Torque actual value	<Not assigned>	▼	<Not assigned>
27. Modes of operation display	Node : 1 Kinco FD(E001)	▼	6061h-00.0(TXPDO_Mc ▼)
40. Touch probe status	<Not assigned>	▼	<Not assigned>
41. Touch probe pos1 pos value	<Not assigned>	▼	<Not assigned>
42. Touch probe pos2 pos value	<Not assigned>	▼	<Not assigned>

The MC_SyncMoveVelocity instruction block is called in the program to run.



**Note**

- Calculate the actual velocity (RPM) of the motor according to the feedback resolution of the motor encoder. Since the instruction is in CSV mode, the actual velocity of the motor is about $2000000/65536/16.384 * 60 = 111$ RPM. (65536 is the feedback resolution of the kinco motor encoder, 16.384 is the basic unit coefficient of the drive)

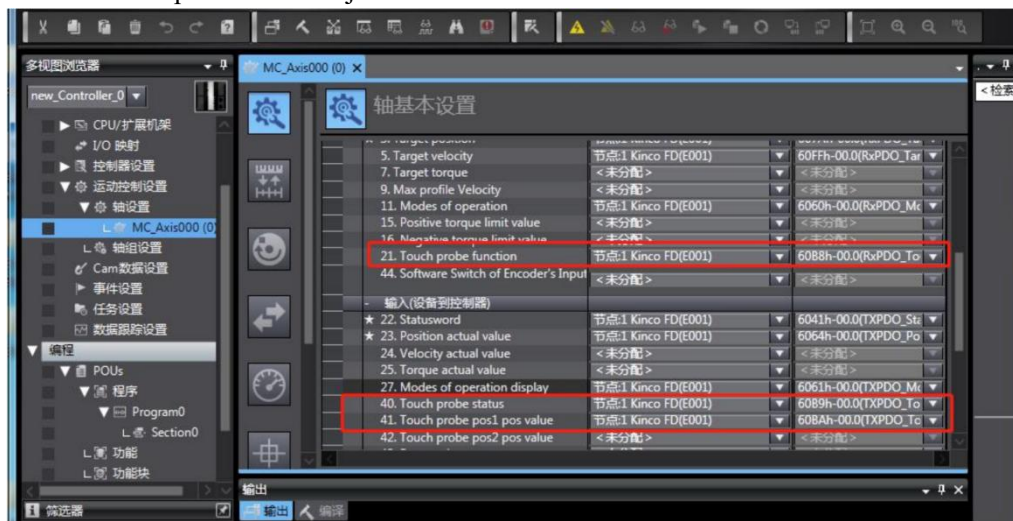
5. Use the probe for homing

(1) The drive DIN port defines the fast capture function, as shown in the following figure, DIN2 must be defined as Fast Capture 1, DIN3 as Fast Capture 2, so that the input state 0x60FD00 bit17 and bit18 exactly correspond to Fast Capture 1 and Fast Capture 2.



When using the Z phase signal as the trigger signal, because the Z signal is small, the drive starts the Z signal capture function after the fast capture 1 rising edge is triggered. That is, the drive will capture the position where the first Z signal is encountered after the fast capture 1 signal is set, so attention should be paid to reset the fast capture 1 signal when capturing continuously. The reset of the fast capture signal in DIN does not necessarily require the digital input port connection level for signal trigger, but can also be triggered by DIN simulation (0x201002).

(2) Add output and input mapping objects to the axis basic settings. When using the Omron MC_Home home reset instruction, three objects 0x60B8, 0x60B9 and 0x60BA must be mapped. If these settings are not made, the controller will alarm the set process data object insufficient error 0x3461.



Index	Description	Current value	Units
60B800	Touch_Probe_Function	0	H
60B900	Touch_Probe_Status	0	H
60BA00	Touch_Probe_Rising1	0	D

Set the capture function (0x60B800) according to the application requirements, observe the capture state (0x60B900) and capture location. The capture function and the capture state are described as follows:

Capture function 0x60FD00 description:

0	0	Negative position limit signal input is valid
	1	Negative position limit signal input is invalid
1	0	Positive position limit signal input is valid
	1	Positive position limit signal input is invalid
2	0	Home signal input is valid
	1	Home signal input is invalid
16	0	Fast capture 1 signal input is valid
	1	Fast capture 1 signal input is invalid
17	0	Fast capture 2 signal input is valid
	1	Fast capture 2 signal input is invalid
0	0	Negative position limit signal input is valid

Capture function 0x60B800 description:

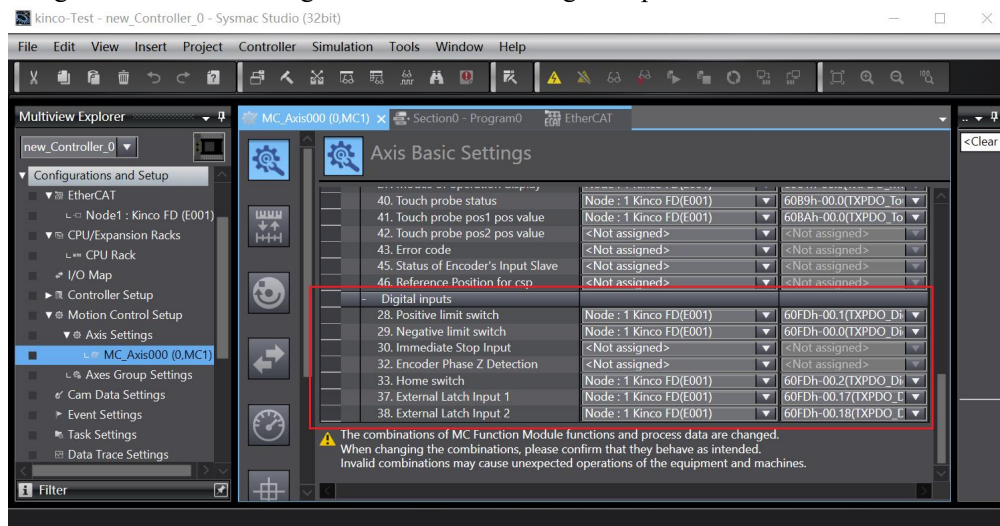
Bit	Value	Description
0	0	Fast capture 1 is invalid
	1	Fast capture 1 is valid
1	0	Single capture, only a single capture of the position. If you need to trigger fast capture 1 again to capture the position, you need to reset → set bit 0.
	1	Continuous capture, the position is continuously captured, and the position is captured every time the fast capture signal 1 is triggered, without the reset → set bit 0.
2	0	Fast capture 1 triggered by an external signal
	1	Fast capture 1 triggered by Z signal
3	-	None
4	0	Fast capture 1 rising edge is invalid
	1	Fast capture 1 rising edge is valid
5	0	Fast capture 1 falling edge is invalid
	1	Fast capture 1 falling edge is valid
6,7	-	None
8	0	Fast capture 2 is invalid
	1	Fast capture 2 is valid
9	0	Single capture, only a single capture of the position. If you need to trigger fast capture 2 again to capture the position, you need to reset → set bit 0.
	1	Continuous capture, the position is continuously captured, and the position is captured every time the fast capture signal 2 is triggered, without the reset → set bit 0.
10	0	Fast capture 2 triggered by an external signal
	1	Fast capture 2 triggered by Z signal

11	-	None
12	0	Fast capture 2 rising edge is invalid
	1	Fast capture 2 rising edge is valid
13	0	Fast capture 2 falling edge is invalid
	1	Fast capture 2 falling edge is valid
14,15	-	None

Capture function 0x60B900 description:

Bit	Value	Description
0	0	Fast capture 1 is invalid
	1	Fast capture 1 is valid
1	0	Fast capture 1 or Z signal rising edge capture is not completed
	1	Fast capture 1 or Z signal rising edge capture is completed
2	0	Fast capture 1 or Z signal falling edge capture is not completed
	1	Fast capture 1 or Z signal falling edge capture is completed
3,4,5,6,7	-	None
8	0	Fast capture 2 is invalid
	1	Fast capture 2 is valid
9	0	Fast capture 2 or Z signal rising edge capture is not completed
	1	Fast capture 2 or Z signal rising edge capture is completed
10	0	Fast capture 2 or Z signal falling edge capture is not completed
	1	Fast capture 2 or Z signal falling edge capture is completed
11,12,13,14,15	-	None

(3) In addition to the above three objects, we also need to add a mapping of the digital input to transmit the relevant signal through 0x60FD. See the figure below to set the digital input.



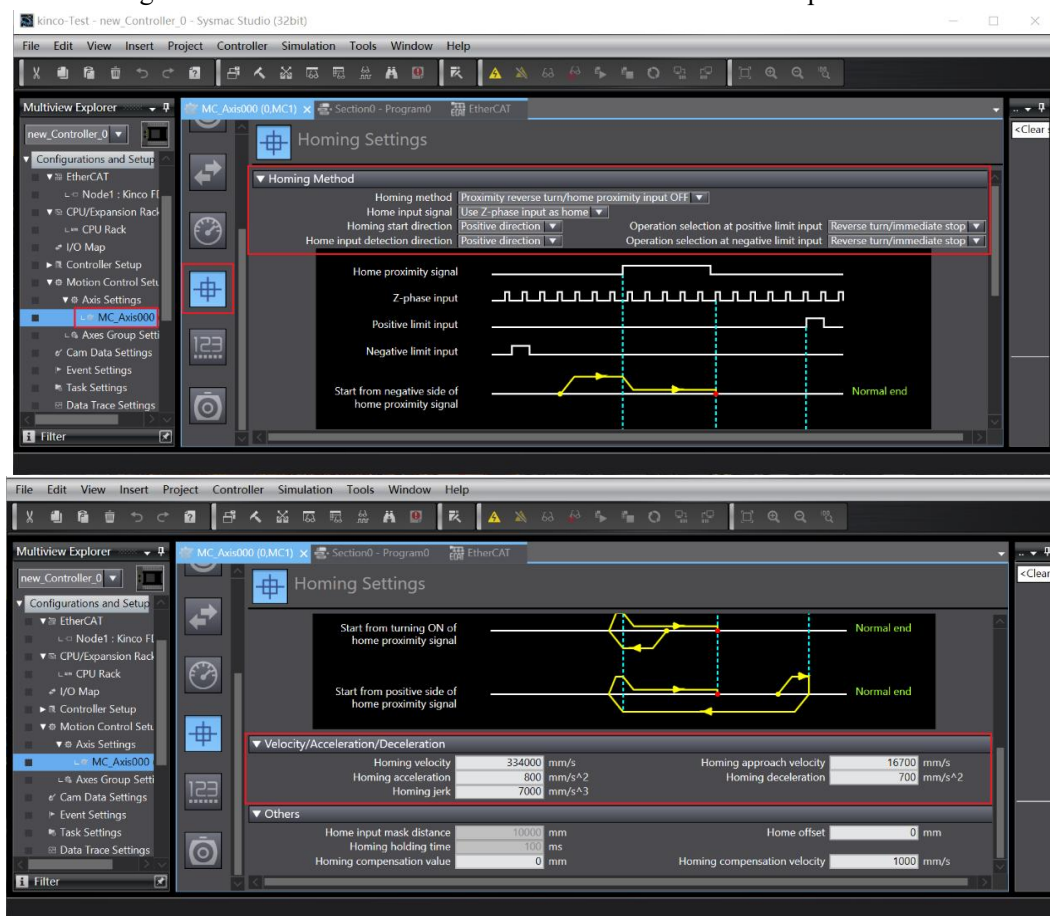
In the above figure, the Home switch is the home proximity input signal of the Omron controller, which corresponds to the home signal of the drive.

External Latch Input 1 is the home proximity input signal, which corresponds to the fast capture 1 of the servo.

Positive limit switch is a positive position limit signal, corresponding to the positive limit of the drive.

Negative limit switch is the negative position limit signal, corresponding to the negative limit of the drive.

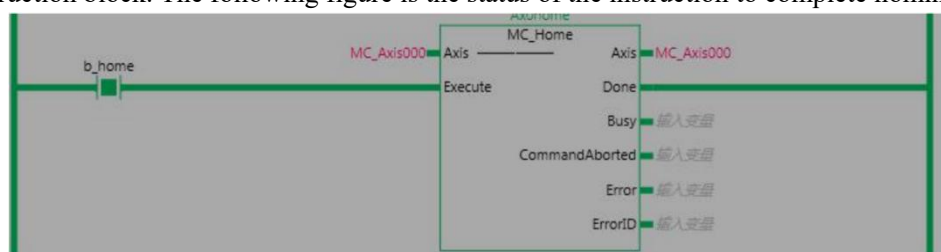
(4) There are 10 homing methods for selection according to the actual application. Through the homing setting interface, set the homing method. Please refer to the controller manual for detailed parameter information.



Note

- Please refer to the controller manual for the specific setting of the velocity / acceleration / deceleration of the drive to return the home. Since the homing instruction is CSP mode instruction, the return velocity of the motor is about $334000/65536 * 60 = 306$ RPM

(5) After enabling the drive via MC_Power, the execution is triggered by giving a rising edge signal to the MC_Home instruction block. The following figure is the status of the instruction to complete homing.



(6) The homing method in this example is to use the home proximity input OFF, that is, the motor runs at the home return velocity after triggering the home instruction, switches to the home return velocity after touching the rising edge of the home signal, and triggers the fast capture 1 after meeting the falling edge of the home signal to return the home. The overall definition of DIN input of the drive is as follows:

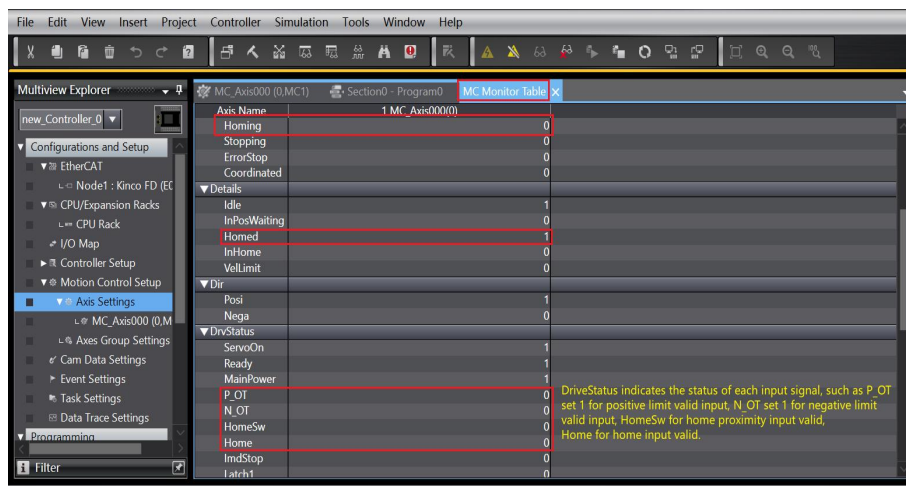
DIO Monitor

DIO Settings

Digital Input (DI)

PIN number	Function	Actual input	Simulation input	Polarity	Valid input
DIN1	-			f High	
DIN2	Fast Capture1			f High	
DIN3	Fast Capture2			f High	
DIN4	-			z Low	
DIN5	-			z Low	
DIN6	-			f High	
DIN7	-			f High	

(7) Through the Controller → MC Monitor Table in the Sysmac Studio software toolbar, the running status, current operation mode, actual position, and actual velocity of the drive can be monitored. The final homing status of the axis can be monitored in the MC monitor table. If the homing cannot be successfully achieved, please refer to step (3) to check whether the digital input setting is correct.



13.4 Profinet communication

13.4.1 Profinet introduction

Profinet (Process Field Net) is a real-time industrial Ethernet technology proposed by the PROFIBUS International Organization in 2000 and became an international standard in 2005. It has the following characteristics:

- (1) Based on Fast Ethernet: The transmission rate reaches 2×100 Mbit/s, and standard Ethernet devices such as switches can be used. It supports various topological structures including linear and ring. Use standard IP addresses for convenient remote access and diagnosis.
- (2) Divide multiple communication channels: Support standard Ethernet channels (TCP/IP, for non-real-time data such as parameter settings), real-time channels (RT, for process data), and isochronous real-time channels (IRT, for motion control), with priority given to motion control data transmission.
- (3) Support for real-time motion control: Motion control data is transmitted via RT or IRT channels, with

dedicated hardware and clock synchronization mechanisms ensuring jitter below 1 μs; the PROFIdrive profile defines process data objects for real-time motion control, known as messages.

- (4) Flexibly meeting user needs: Different real-time levels and application categories are divided according to different communication functions. Users can use the necessary communication functions based on their own needs, saving time and technical costs.
- (5) Emphasize diagnosis and safety: Provide a complete hierarchical diagnosis mechanism from channel level, device level to network level to facilitate rapid fault location. Support redundant communication mechanisms to enhance network reliability.
- (6) Ecological maturity and compatibility: It features mature control platforms and toolchains, such as TIA Portal and PRONETA, which simplify configuration through a unified device description file (GSDML) and offer extensive device compatibility.

As a PROFINET IO device, iSMD supports the PROFIdrive standard, non-periodic communication, real-time periodic communication, and equal-time synchronization mode.It also supports motion control messages such as No. 1, No. 3, No. 102, No. 105, No. 111, and No. 750, which can meet various application scenarios ranging from basic open-loop speed control to high-end synchronous position control. Specific message descriptions can be found in Section 13.4.1.1-13.4.1.3.

13.4.1.1 Supported message

iSMD support telegram 1, 3, 5, 9, 102, 105, 111, and additional message 750, covering AC (Application Class) 1, 3, and 4, as shown in the following table

Table 13-49 Available message

Application Class	Message number
AC1	1
AC3	9,111
AC4	3,5,102,105



Note

- Different versions of PN firmware support different messages, as shown in Table 13-50. The PN firmware version can be viewed through [308001]. For details, please refer to Table 13-70.

The structure of each message are as follow:

Table 13-50 Each PN version supported message

PN firmware version ^a	Supported message
xxxx xxx4	1, 3, 111
xxxx nxx5	1, 3, 5, 9, 102, 105, 111, 750 ^b
^a PN4 firmware applicable to GSDML-V2.33-Kinco-MD60-20210507 PN5 firmware applicable to GSDML-V2.43-Kinco-PA5-20240328 ^b When ‘ n ’ is 0, 750 message is not supported.	

Table 13-51 Message 1 (Application Class 1)

I/O data number	Message 1	
	Set value	Actual value
1	STW1	ZSW1
2	NSOLL_A	NIST_A

Table 13-52 Message 9, 111 (Application Class 3)

I/O data number	Message 9		Message 111	
	Set value	Actual value	Set value	Actual value
1	STW1	ZSW1	STW1	ZSW1
2	SATZANW	AKTSATZ	POS_STW1	POS_ZSW1
3	STW2	ZSW2	POS_STW2	POS_ZSW2
4	MDI_TARPOS	XIST_A	STW2	ZSW2
5			VERRIDE	MELDW
6	MDI_VELOCITY		MDI_TARPOS	XIST_A
7				
8	MDI_ACC		MDI_VELOCITY	NIST_B
9	MDI_DEC			
10	MDI_MOD		MDI_ACC	FAULT_CODE
11			MDI_DEC	WARN_CODE
12			user	user

Table 13-53 Message 3, 5, 102, 105 (Application Class 4)

I/O data number	Message 3		Message 102		Message 5		Message 105	
	Set value	Actual value	Set value	Actual value	Set value	Actual value	Set value	Actual value
1	STW1	ZSW1	STW1	ZSW1	STW1	ZSW1	STW1	ZSW1
2	NSOLL_B	NIST_B	NSOLL_B	NIST_B	NSOLL_B	NIST_B	NSOLL_B	NIST_B
3								
4	STW2	ZSW2	STW2	ZSW2	STW2	ZSW2	STW2	ZSW2
5	G1_STW	G1_ZSW	MOMRED	MELDW	G1_STW	G1_ZSW	MOMRED	MELDW
6		G1_XIST1	G1_STW	G1_ZSW	XERR	G1_XIST1	G1_STW	G1_ZSW
7				G1_XIST1			XERR	G1_XIST1
8		G1_XIST2		G1_XIST2	KPC	G1_XIST2	KPC	G1_XIST2
9								
10								

Table 13-54 Additional message 750

I/O data number	Additional message 750	
	Set value	Actual value
1	M_ADD1	M_ACT
2	M_LIMIT_POS	—
3	M_LIMIT_NEG	—

13.4.1.2 I/O data signal

I/O data signal (set value and actual value) are used to compose the telegram, and the following table provides all signals for the I/O data.

Table 13-55 Soignal table

Singal	Abbreviation	Data type	Description
Control word 1	STW1	U16	See section13.4.1.3.1
Status word 1	ZSW1	U16	
Control word 2	STW2	U16	See section13.4.1.3.2
Status word 2	ZSW2	U16	
Speed set value A	NSOLL_A	I16	4000h corresponds to the reference speed ^a
Speed actual value A	NIST_A	I16	
Speed set value B	NSOLL_B	I32	40000000h corresponds to the reference speed
Speed actual value B	NIST_B ^b	I32	
Encoder 1 control word	G1_STW	U16	See section 13.4.1.3.5
Encoder 1 status word	G1_ZSW	U16	
Message status word	MELDW	U16	
Encoder 1 position acutal value 1	G1_XIST1	I32	Equal to actual position [606300] plus position shift [60FB07]. Unit DEC
Encoder 1 position acutal value 2	G1_XIST2	I32	—
Position deviation value	XERR	I32	—
Position control factor	KPC	I32	—
Position acutal value A	XIST_A	I32	Equal to actual position [606300]. Unit DEC
Position table control word	SATZANW	U16	See section13.4.1.3.3
Position table status word	AKTSATZ	U16	
MDI mode control word	MDI_MOD	U16	
Position control word 1	POS_STW1	U16	See section13.4.1.3.4
Position status word 1	POS_ZSW1	U16	
Position control word 2	POS_STW2	U16	
Position status word 2	POS_ZSW2	U16	
MDI target position	MDI_TARPOS	I32	Equal to target position [607A00]. Unit DEC
MDI velocity	MDI_VELOCITY	U32	Unit DEC, DEC = (RPM * feedback_resolution [641003] * 512) / 1875
MDI acceleration percentage	MDI_ACC	U16	0~4000h corresponds to 0~100%, 2000h means the acceleration/deceleration is reduced to 1/2 of the drive setpoint
MDI deceleration percentage	MDI_DEC	U16	
MDI velocity percentage	OVERRIDE	U16	0~4000h corresponds to 0~100%, maximum 199%, 2000h means to reduce the speed to 1/2 of the drive setpoint
Torque limit value	MOMRED	I16	0~4000h corresponds to 100%~0, 4000h means the target current (torque) is limited to 0
Fault code	FAULT_CODE	U16	—
Warn code	WARN_CODE	U16	—
Message 111 custom receive word	user	I16	See section 13.4.5.4.1,article 7
Message 111 custom send word	user	I16	
Additional message 750 target torque	M_ADD1	I16	Equal to the Target_Torque %[607100], unit DEC

Singal	Abbreviation	Data type	Description
Additional message 750 torque limit	M_LIMIT_POS	U16	Equal to the Max_Torque [607200], unit DEC
Additional message 750 actual torque	M_ACT	I16	Equal to the Actual_Torque [607700], unit DEC
^a Reference speed = 1/2 of the maximum speed [607F00]			
^b When using message 111, the speed actual value NIST_B is equal to the actual speed [606C00] in DEC.			

13.4.1.3 Control word and status word

13.4.1.3.1 STW1 and ZSW1

Control word 1 (STW1)

Table 13-56 STW1 each bit value meaning

Bit	Meaning	
	Message 1, 3, 5, 102, 105 (Application Class 1, 4)	Message 9,111 (Application Class 3)
0	On / Off (ramp stop ^a)	
1	No coast stop / Coast stop	
2	No quick stop / Quick stop	
3	Operation enable / Operation disable	
4	Enable RFG ^b / Disable RFG (RFG output is 0)	Do not reject traversing task / Reject traversing task
5	Unfreeze RFG / Freeze RFG (RFG input is not updated)	No intermediate stop / Intermediate stop
6	Enable setpoint / Disable setpoint (RFG input is 0)	Activate traversing task (0→1)
7	Fault acknowledge (0→1)	
8	Reserved (Application Class 1, 4 do not support jog)	Start reverse jog / Stop reverse jog
9	Reserved	Start forward jog / Stop forward jog
10	Control by PLC / No control by PLC	
11	Setpoint is reversed / Setpoint is not reversed	Start homing procedure / Stop homing procedure
12~13	Reserve	
14	Use torque mode/Do not use torque mode	
15	Reserve	
^a When bit 4 of STW1 is 1, ramp stop corresponds to halt mode [605D00], and disable operation after stop. When bit 4 of STW1 is 0, ramp stop corresponds to shutdown stop mode [605B00].		
^b RFG = Ramp Function Generator (Trapezoidal curve velocity generator) .When using Application Class 4, bit 5 does not work, and either bit 4 or bit 6 is 0, resulting in an RFG output of 0.		
Note: the significance for bit value = 1 is left of the slash ; bit value = 0 is right of the slash .		

Status word 1 (ZSW1)

Table 13-57 ZSW1 each bit value meaning

Bit	Meaning	
	Message 1,3,5,102,105 (Application Class 1, 4)	Message 9, 111 (Application Class 3)
0	Ready to switch on / Not ready to switch on	
1	Ready to operate / Not ready to Operate	
2	Operation enabled / Operation disabled	
3	Fault present / No fault	
4	Coast stop not activated / Coast stop activated	
5	Quick stop not activated / Quick stop activated	

6	Switching on inhibited / Switching on not inhibited	
7	Warn present / No warning	
8	Speed error within tolerance range / Speed error out of tolerance range	Following error within tolerance range / Following error out of tolerance range
9	Control requested / No control requested	
10	Speed reach or exceed reference speed/ Speed does not reach reference speed	Target position reached / Not at target position
11	Reserved	Reference point set / Reference point not yet set
12	Reserved	Traversing task acknowledgement (0→1)
13	Reserved	Drive stopped / Drive moving
14	Forward movement/Reverse movement	Accelerating / Not accelerating
15	Reserved	Decelerating / Not decelerating
Note: the significance for bit value = 1 is left of the slash ; bit value = 0 is right of the slash .		

13.4.1.3.2 STW2 and ZSW2

Control word 2(STW2)

For message 3, 5, 102, and 105, valid only in IRT mode.

Table 13-58 STW2 each bit value meaning

Bit	Meaning
0~11	Reserved
12~15	PLC heartbeat signal

**Note**

- If there is a PLC heartbeat signal error or other abnormal communication, the drive will generate a "CAN bus fault (alarm code 100.0)" alarm. The user can set Abort_Connection_Mode [600700] to 1: error ; set to 0: no process.

Index	Description	Preset value
600700	Abort_Connection_Mode	0:no process 1:error

- The alarm time threshold is determined by the product of Guard_Time [100C00](in ms) and Life_Time_Factor [100D00], and the default value is 1000ms * 3 = 3000 ms.

Failure supervision

Guard_Time

1000

D

Life_Time_Factor

3

D

Status word 2 (ZSW2)

For message 3, 5, 102, and 105, valid only in IRT mode.

Table 13-59 ZSW2 each bit value meaning

Bit	Meaning
0~11	Reserved
12~15	Drive heartbeat signal

13.4.1.3.3 SATZANW, AKTSATZ, MDI_MOD

Position table control word (SATZANW)

Used for message 9

Table 13-60 SATZANW each bit value meaning

Bit	Meaning
0~2	Index of the position task to be started in the position table (value range: 0~7) Bit 0~2 are valid only in the position table mode
3~14	Reserved
15	Operating mode selection = 1 MDI mode = 0 Position table mode

Position table status word (AKTSATZ)

Used for message 9

Table 13-61 AKTSATZ each bit value meaning

Bit	Meaning
0~4	The index of the actual effective positioning task (value range: 0~31) Bit 0~4 are valid only in the position table mode
5~14	Reserved
15	Operating mode = 1 MDI mode activated = 0 Position table mode activated

MDI mode control word (MDI_MOD)

Used for message 9

Table 13-62 MDI_MOD each bit value meaning

Bit	Meaning
0	Select absolute/relative positioning mode = 1 Absolute positioning mode = 0 Relative positioning mode
1~15	Reserved

13.4.1.3.4 POS_STW1, POS_STW2, POS_ZSW1, POS_ZSW2

Position control word 1 (POS_STW1)

Used for message 111

Table 13-63 POS_STW1 each bit value meaning

Bit	Meaning
0~2	Index of the position task to be started in the position table (value range: 0~7) Bit 0~2 are valid only in the position table mode
3~7	Reserved
8	Select absolute/relative positioning mode = 1 Absolute positioning mode = 0 Relative positioning mode

Bit	Meaning
9~11	Reserved
12	Absolute positioning mode = 1 Execute the absolute positioning command immediately according to the target position change = 0 The absolute positioning command will not be executed immediately according to the target position change.
13~14	Reserved
15	Operating mode selection = 1 MDI mode = 0 Position table mode

Position control word 2 (POS_STW2)

Used for message 111

Table 13-64 POS_STW2 each bit value meaning

Bit	Meaning
0	Reserved
1	= 1 Set the current position as the homing
2	= 1 Activate homing switch signal
3~13	Reserved
14	= 1 Activate software limit switch signal
15	= 1 Activate hardware limit switch signal

Position status word 1 (POS_ZSW1)

Used for message 111

Table 13-65 POS_ZSW1 each bit value meaning

Bit	Meaning
0~4	The index of the actual effective positioning task (value range: 0~31) Bit 0~4 are valid only in the position table mode
5~7	Reserved
8	= 1 Negative hardware limit switch signal active
9	= 1 Positive hardware limit switch signal active
10	= 1 Jog mode activated
11~12	Reserved
13	= 1 Position table mode activated
14	Reserved
15	= 1 MDI mode activated = 0 MDI mode not activated

Position status word 2 (POS_ZSW2)

Used for message 111

Table 13-66 POS_ZSW2 each bit value meaning

Bit	Meaning
0~3	Reserved
4	= 1 Moving forward = 0 Not moving forward
5	= 1 Moving reversely = 0 Not moving reversely

Bit	Meaning
6	= 1 Negative software limit switch reached
7	= 1 Positive software limit switch reached
8~15	Reserved

13.4.1.3.5 G1_STW, G1_ZSW , MELDW

Encoder 1 control word (G1_STW)

Used for message 3, 5, 102, 105.

Table 13-67 G1_STW each bit value meaning

Bit	Value	Meaning
0	1	Function 1 (Bit7=0): Request to search sensor index signal position
1~3	—	Reserved
4~6	1~3	Command: 0: — 1: Activate function 1 ^a 2: Read sensor index signal position and put it into G1_XIST2 3: Cancel function 1 4~7: —
7	0/1	Mode: Bit7=0: Sensor index signal position search Bit7=1: Reserved
8~12	—	Reserved
13	1	Request to transmit absolute actual position in G1_XIST2
14	1	Request to switch off the actual value measurements in the drive
15	1	Request to reset a sensor error
a Selection of function 1 (bit 0) and command (bit4~bit7) shall be set simultaneously.		

Encoder 1 status word (G1_ZSW)

Used for message 3, 5, 102, 105.

Table 13-68 G1_ZSW each bit value meaning

Bit	Value	Meaning
0	1	Status: Function 1 activated
1~3	—	Reserved
4	1	Status: Sensor index signal position available
5~10	—	Reserved
11	1	Error acknowledgement in process
12	—	Reserved
13	1	Indication of the transmission of absolute actual position in G1_XIST2
14	1	Acknowledgement for indication of invalid G1_XIST1
15	1	Sensor fault present

Message status word (MELDW)

Used for message 102, 105.

Table 13-69 MELDW each bit value meaning

Bit	Meaning
0	Reserved
1	= 1 Torque does not reach the limit value = 0 Torque has reached the limit value
2~15	Reserved

13.4.2 Application description

The PROFINET communication application in Chapters 13.4.3 to 13.4.6 is based on the following conditions:

1. TIA V17 PLC S7-1500(CPU 1511T-1 PN 6ES7 511-1TK01-0AB0 Firmware V2.8)
2. GSD file

PN firmware version	Applicable GSD file
xxxx xxx5	GSDML-V2.43-Kinco-PA5-20240328 ^a
^a If PLC does not support V2.43 GSD, use GSDML-V2.33-Kinco-PA5-20240328	

Chapter 13.4.7 introduces the method of using S7-200 SMART.

Chapter 13.4.8 introduces the usage method of the additional message 750.

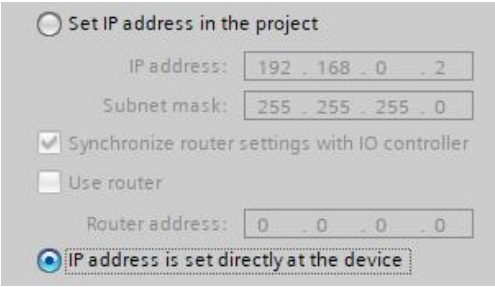
Table 13-70 PROFINET communication parameter

Parameter index	Parameter name	Description
308001	PN firmware version	Applicable PN firmware version for this manual is xxxx xxx5
308003	PN device name	PROFINET device name (read-only)
308004~308007	PN device name 1~4	Used to modify PROFINET device name
308008	IP address	Drive IP address (hex, read-only)
308009	Subnet mask	Drive subnet mask (hex, read-only)
30800A	Default router	Drive default router (hex, read-only)
30800B	PN enable	= 1 PN telegram enabled, using message (e. g. message 105) control = 0 PN telegram disabled, using RS232 or acyclic communication control
30800C	Enable software limit	See section 13.4.4.1, Article 5
30800D	PN user receive word	See section 13.4.4.1, Article 8
30800E	PN user send word	
30800F~308010	MAC address 1~2	Drive MAC address (hex, read-only)
301101	ECAN synchronization cycle	Optional: 1ms, 2ms, 4ms, 8ms Need to be consistent with the update time of TIA portal configuration
301102	ECAN synchronization mode	= 1 Use isochronous synchronization (IRT) mode = 0 Do not use isochronous synchronization (IRT) mode
301103	ECAN synchronization shift	—
600700	Communication abort mode	See section 13.4.1.3.2
100C00	Node guard time	
100D00	Node guard time factor	

13.4.3 Acyclic communication

13.4.3.1 Drive IP and device name setting

The PROFINET device name of the drive can be set using the KincoServo PRO software, TIA portal and PRONETA software; the IP address of the drive can only be set using TIA portal and PRONETA software. Please note that if the IP address is to be set using the method described below, the "IP address is set directly at the device" option should be selected in TIA portal configuration:



To set the device name in the KincoServo PRO software, proceed as follows:

1. Disconnect the drive from the PLC and connect the drive to the KincoServo PRO software via RS232.
2. Enter the device name in PN_Device_Name 1~4 [308004~308007], the length of each PN_Device_Name is limited to 4 characters:

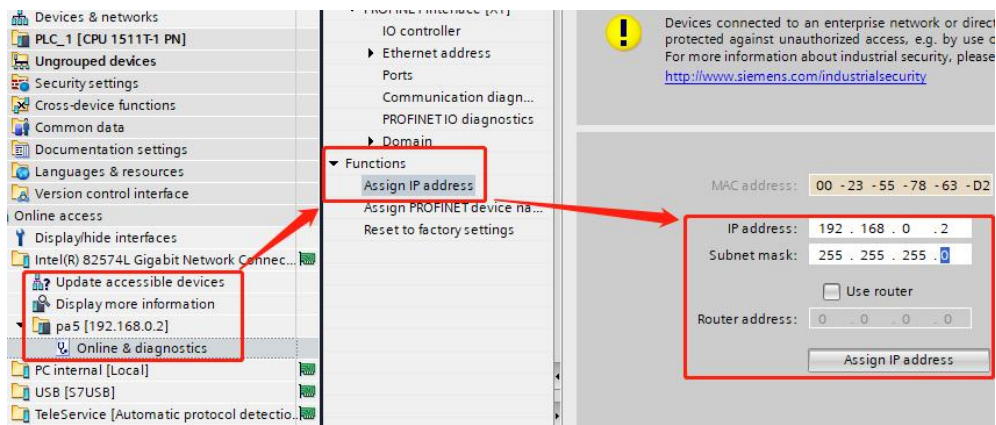
Index	Description	Preset value		Current value	
308004	PN_Device_Name1	fd42	ASCII	fd42	ASCII
308005	PN_Device_Name2	5-1	ASCII	5-1	ASCII

3. When the setting is completed, you can see that the PN_Device_Name [308003] has been modified:

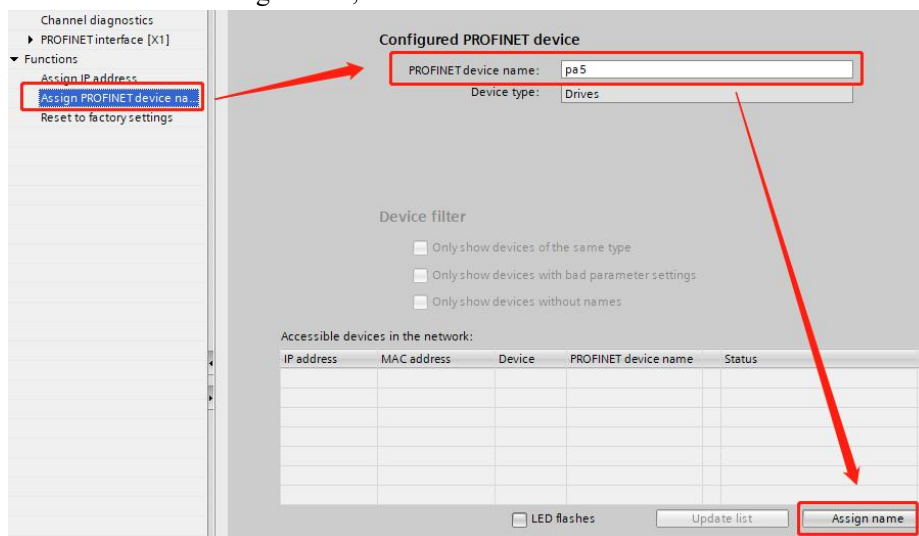
Index	Description	Preset value		Current value	
308003	PN_Device_Name	Write Prohibition		fd425-1	String

To set the drive IP and device name in TIA portal, proceed as follows:

1. Connect the drive to the computer with a network cable.
2. Open TIA portal, expand the "Online access" drop-down list, select the corresponding network card, and click "Update accessible devices". Once the software has scanned the drive, double click "Online & Diagnostics".
3. Double-click "Assign IP address" in the "Functions" drop-down list. Fill in the IP address and subnet mask, and then click the "Assign IP address" button. If you select "Set IP address in the project" in the configuration, this step can be omitted and you can go directly to step 4 to set the PROFINET device name.



- Follow step 3, double click "Assign PROFINET device name" in the "Functions" drop-down list. Fill in the PROFINET device name and click "Assign name" button, the PROFINET device name must be the same as the name in the configuration, otherwise the drive can not communicate with the PLC.

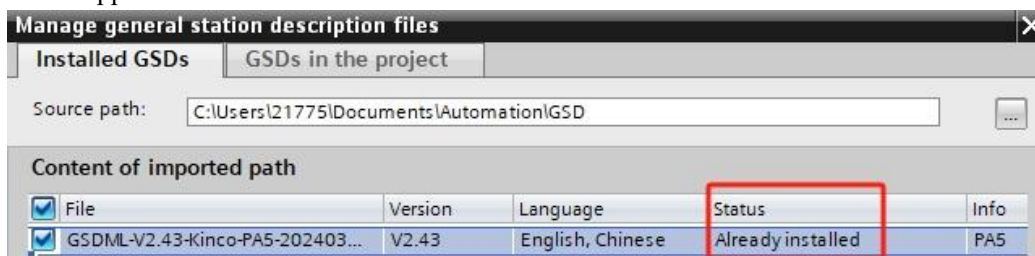


- When the setting is complete, click "Update accessible devices" again to check whether the IP address and device name have been modified successfully.

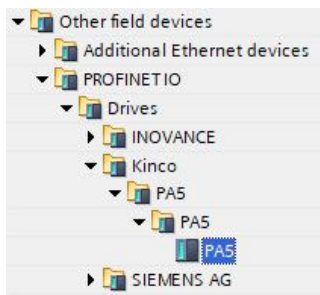
Alternatively, the drive IP address and device name can be set using the PRONETA software, which can be downloaded from the Siemens website: [PRONETA - Siemens China](http://www.siemens.com/PRONETA).

13.4.3.2 Project configuration

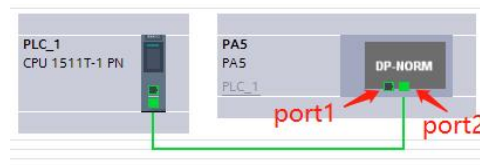
- Click menu "Options → Manage general station description files" to install GSD, please take the GSDML-V2.43-Kinco-PA5-20240328 as the standard. In addition, the GSD to which the PN firmware 00000004 applies is GSDML-V2.33-Kinco-MD60-20210507.



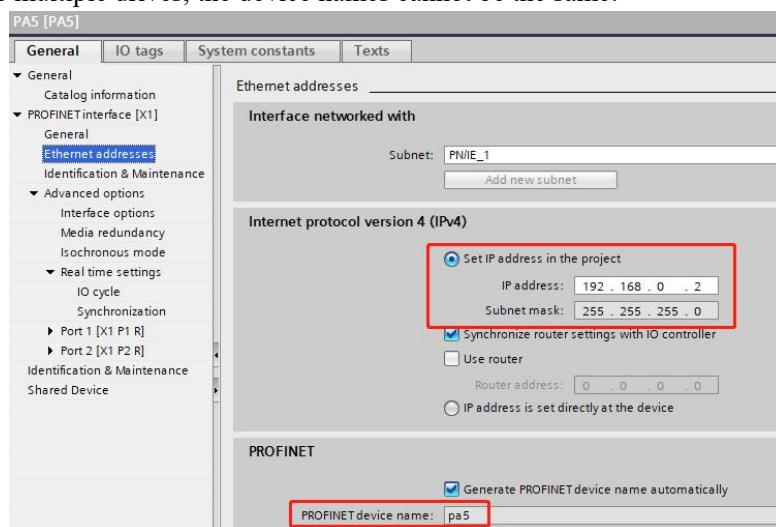
- In the network view, click "Hardware catalog → Other field devices → PROFINET IO → Drives → Kinco → PA5" to add a drive and assign it to the PLC:



3. If PLC is required to automatically configure the device name of drive (the premise of automatic allocation is that the name of the drive is empty), the topology diagram must be connected. In the topology view, connect the ports according to the actual situation. The IN port of iSMD corresponds to port 1 in the topology diagram



4. Double-click the PA5 module, click "Ethernet addresses" in "General", and set the configured IP address and PROFINET device name. The device names need to be consistent with the actual device names of the drives. When there are multiple drives, the device names cannot be the same:



5. Click on the "System constants" column and note down the HW_Interface, which will be used later in the read-write parameters module:

PA5 [PA5]				
General IO tags System constants Texts				
Show hardware system constants				
Name	Type	Hardware identi.	Used by	
PA5-Interface-端口_1	Hw_Interface	260	PLC_1	
PA5-Interface-端口_2	Hw_Interface	261	PLC_1	
PA5-Interface	Hw_Interface	259	PLC_1	
PA5-Proxy	Hw_SubModule	258	PLC_1	
PA5-Head	Hw_SubModule	262	PLC_1	

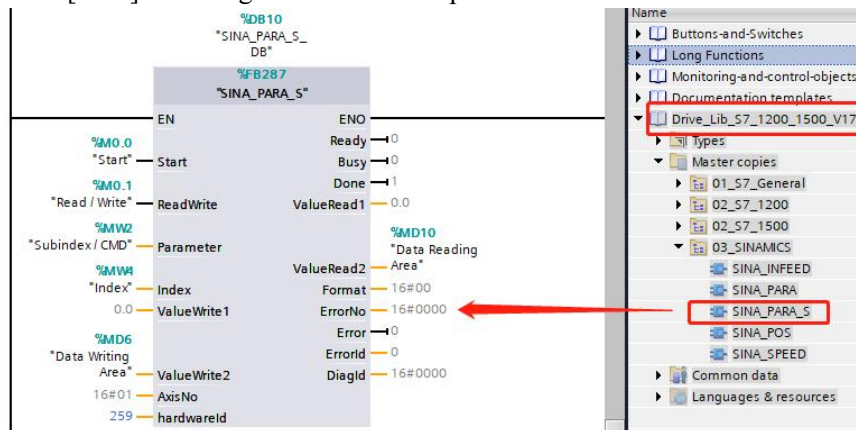
13.4.3.3 SINA_PARA_S (FB287) introduction

FB287 can realize acyclic communication between PLC and drive, and realize servo parameters read-write function. If you cannot find the corresponding function block (FB), you can download the library file Drive_Lib_S7_1200_1500_V17 from Siemens official website:

<https://support.industry.siemens.com/cs/document/109475044/sinamics-communication-blocks-drivelib-for-reading-and-writing-drive-data-within-tia-portal-context?dti=0&lc=en-WW>

After the download is completed, click menu "Options → Global libraries → Open library", and install the corresponding library file according to the prompt.

Add FB287 to Main[OB1] and assign variables to its pins as shown below.



FB287 pin definition are as follow:

Table 13-71 FB287 pin definition

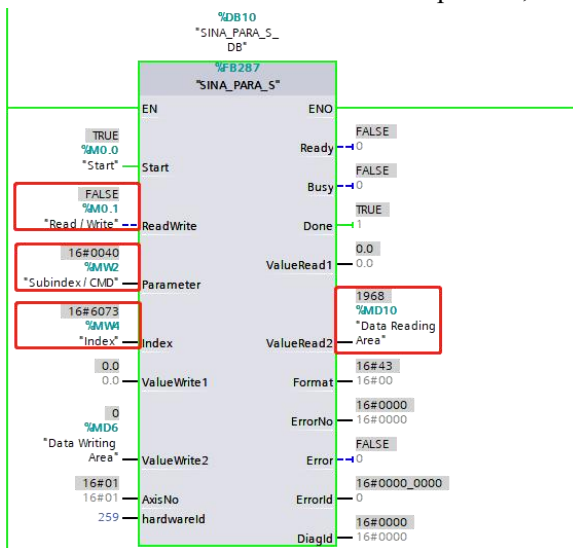
Pin name	Data type	Description
Start	Bool	The rising edge triggers data transmission
ReadWrite	Bool	= 0 Read = 1 Write
Parameter	Int	Bit8~15: Parameter subindex Bit0~7: Command word
Index	Int	Parameter index
ValueWrite1	Real	Reserved
ValueWrite2	DInt	Data writing area, can write 1 to 4 bytes
AxisNo	Byte	Fixed to 1
hardwareId	HW_IO	See Section 13.4.3.2, Article 5, by which the parameter is used to distinguish between different axes
ValueRead1	Real	Reserved
ValueRead2	DInt	Data reading area
Error	Bool	Indicates read-write parameters error, when the responding command word in Kinco communication protocol is 0x80, this bit is set to 1

Note: Command word: read all 40H; Write 1 byte 2FH, write 2 bytes 2BH, and write 4 bytes 23H.

13.4.3.4 FB287 read-write parameter

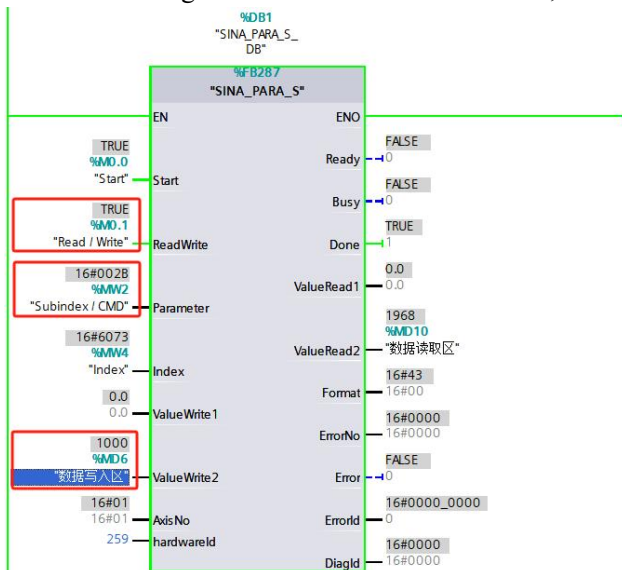
Take the read-write CMD_q_Max [607300] as an example: the parameter index is 0x6073, the subindex is 0x00, and the data type is uint16, so the command word of writing parameter is 0x2B, and that of reading parameter is uniformly 0x40.

1. Read CMD_q_Max: When the "Start" pin changes from 0 to 1, the pin "Data Reading Area" automatically becomes 1968 and "ERROR" pin is 0, the data reading is successful:



Index	Description	Preset value	Current value
606100	Operation_Mod...	Write Prohibition	0 DEC
606300	Pos_Actual	Write Prohibition	0 DEC
606C00	Speed_Real	Write Prohibition	0 DEC
607100	Target_Torque%		0 DEC
607300	CMD_q_Max		1968 DEC
607800	I_q	Write Prohibition	0 DEC
607A00	Target_Position		0 DEC
608100	Profile_Speed		0 DEC
608300	Profile_Acc		100 DEC
608400	Profile_Dec		100 DEC
60FF00	Target_Speed		0 DEC

2. Write CMD_q_Max: Verify that PN_Enable [30800B] is 0. "ReadWrite" pin is switched to 1, "Parameter" pin is 0x002B. "Data Writing Area" is written to 1000. Then trigger the "Start" pin to write the data. Observing the KincoServo PRO software, CMD_q_Max is changed to 1000 DEC.



Index	Description	Preset value	Current value
606100	Operation_Mod...	Write Prohibition	0 DEC
606300	Pos_Actual	Write Prohibition	0 DEC
606C00	Speed_Real	Write Prohibition	0 DEC
607100	Target_Torque%		0 DEC
607300	CMD_q_Max		1000 DEC
607800	I_q	Write Prohibition	0 DEC
607A00	Target_Position		0 DEC
608100	Profile_Speed		0 DEC
608300	Profile_Acc		100 DEC
608400	Profile_Dec		100 DEC
60FF00	Target_Speed		0 DEC

13.4.3.5 Read-write parameters using WRREC and RDREC

The following describes how to read/write parameters using the WRREC / RDREC blocks, which can be found in "Instructions → Extended instructions → Distributed I/O".

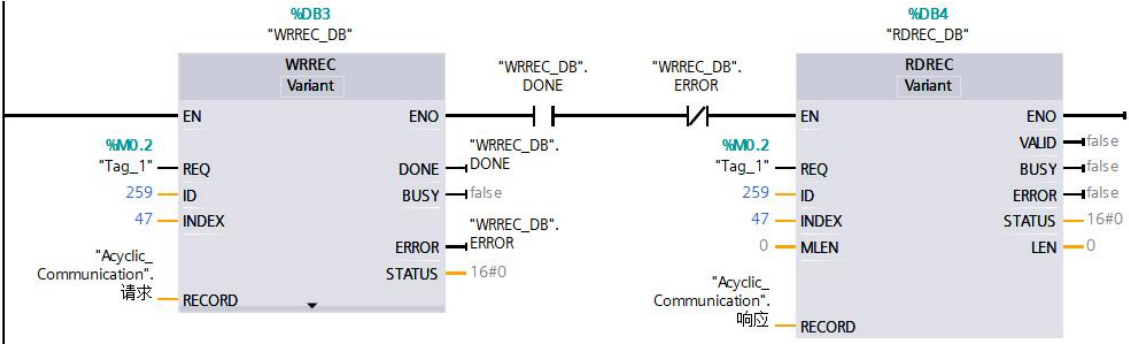
1. First, create a new data block (DB) called Acyclic_Communication. The data in this block is defined as follows:

Acyclic_Communication				
	Name	Data type	Start value	Monitor value
1	Static			
2	Request	Struct		
3	ReqRef	Byte	16#01	16#01
4	ReqID	Byte	16#01	16#01
5	AxisID	Byte	16#01	16#01
6	ParaNo	Byte	16#01	16#01
7	Attribute	Byte	16#10	16#10
8	Number of Element	Byte	16#01	16#01
9	Subindex / CMD	Word	16#002B	16#002B
10	Index	Word	16#6073	16#6073
11	Write Format	Byte	16#43	16#43
12	Write Values numb...	Byte	16#01	16#01
13	Write Value	Dint	0	0
14	Response	Struct		
15	RespRef	Byte	16#0	16#00
16	RespID	Byte	16#0	16#00
17	AxisID	Byte	16#0	16#00
18	ParaNo	Byte	16#0	16#00
19	Format	Byte	16#0	16#00
20	CMD	Byte	16#0	16#00
21	Value	Dint	0	0

The data block Acyclic_Communication is described as follows:

Request	Response
ReqID: 0x01: read-only 0x02: readable and writable Other: Reserved	CMD : Responing command word
Subindex / CMD: Bit 8~15: parameter subindex Bit 0~7: sending command word	
Index: Parameter index	Value : Data reading area
Write Value: Data writing area	
Other data are in principle consistent with the above figure.	

2.
- Establish the connection of WRREC / RDREC block according to the following figure. The transmission starts when the "REQ" pin is set to 1. "ID" is the HW_Interface and can be used to distinguish between different axes. "INDEX" is fixed to 47 and "MLen" is 0.



3.
- The following utilizes the WRREC / RDREC block to read/write CMD_q_Max [607300]. The parameter index is 0x6073, subindex is 0x00, and data type is uint16, so the sending command word for writing parameter is 0x2B, and for reading parameter is uniformly 0x40.
- a.
- Read CMD_q_Max:

Use KincoServo PRO software to check CMD_q_Max is 1000 DEC. The "ReqID" of the Acyclic_Communication data block is assigned 0x02, "Subindex / CMD" is assigned 0x0040, and "Index" is assigned 0x6073. Trigger "REQ" pin to complete the data reading, and the reading data is displayed in the "Value" of "Response":

Acyclic_Communication				
Name	Data type	Start value	Monitor value	
Static				
Request	Struct			
ReqRef	Byte	16#01	16#01	
ReqID	Byte	16#01	16#02	
AxisID	Byte	16#01	16#01	
ParaNo	Byte	16#01	16#01	
Attribute	Byte	16#10	16#10	
Number of Element	Byte	16#01	16#01	
Subindex / CMD	Word	16#002B	16#0040	
Index	Word	16#6073	16#6073	
Write Format	Byte	16#43	16#43	
Write Values numb...	Byte	16#01	16#01	
Write Value	DInt	0	0	
Response	Struct			
RespRef	Byte	16#0	16#01	
RespID	Byte	16#0	16#02	
AxisID	Byte	16#0	16#01	
ParaNo	Byte	16#0	16#01	
Format	Byte	16#0	16#43	
CMD	Byte	16#0	16#4B	
Value	DInt	0	1000	

Index	Description	Preset value	Current value	
606100	Operation_Mod...	Write Prohibition	0	DEC
606300	Pos_Actual	Write Prohibition	0	DEC
606C00	Speed_Real	Write Prohibition	0	DEC
607100	Target_Torque%		0	DEC
607300	CMD_q_Max		1000	DEC
607800	I_q	Write Prohibition	0	DEC
607A00	Target_Position		0	DEC
608100	Profile_Speed		0	DEC
608300	Profile_Acc		100	DEC
608400	Profile_Dec		100	DEC
60FF00	Target_Speed		0	DEC

b. Write CMD_q_Max 1968DEC:

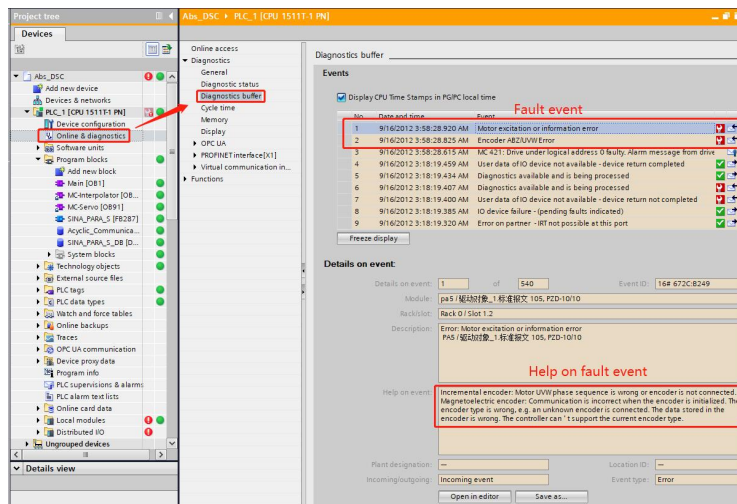
Verify that PN_Enable [30800B] is 0. "ReqID" is assigned 0x02, "Subindex / CMD" is assigned 0x002B, "Index" is assigned 0x6073, "Write Value" is assigned 1968. Triggers "REQ" pin to complete the data writing. See the figure below, "CMD" of "Response" is 0x60, indicating that the writing is successful. Use KincoServo PRO software to view CMD_q_Max becomes 1968 DEC.

Acyclic_Communication				
Name	Data type	Start value	Monitor value	
Static				
Request	Struct			
ReqRef	Byte	16#01	16#01	
ReqID	Byte	16#01	16#02	
AxisID	Byte	16#01	16#01	
ParaNo	Byte	16#01	16#01	
Attribute	Byte	16#10	16#10	
Number of Element	Byte	16#01	16#01	
Subindex / CMD	Word	16#002B	16#002B	
Index	Word	16#6073	16#6073	
Write Format	Byte	16#43	16#43	
Write Values numb...	Byte	16#01	16#01	
Write Value	DInt	0	1968	
Response	Struct			
RespRef	Byte	16#0	16#01	
RespID	Byte	16#0	16#02	
AxisID	Byte	16#0	16#01	
ParaNo	Byte	16#0	16#01	
Format	Byte	16#0	16#43	
CMD	Byte	16#0	16#60	
Value	DInt	0	1968	

Index	Description	Preset value	Current value	
606100	Operation_Mod...	Write Prohibition	0	DEC
606300	Pos_Actual	Write Prohibition	0	DEC
606C00	Speed_Real	Write Prohibition	0	DEC
607100	Target_Torque%		0	DEC
607300	CMD_q_Max		1968	DEC
607800	I_q	Write Prohibition	0	DEC
607A00	Target_Position		0	DEC
608100	Profile_Speed		0	DEC
608300	Profile_Acc		100	DEC
608400	Profile_Dec		100	DEC
60FF00	Target_Speed		0	DEC

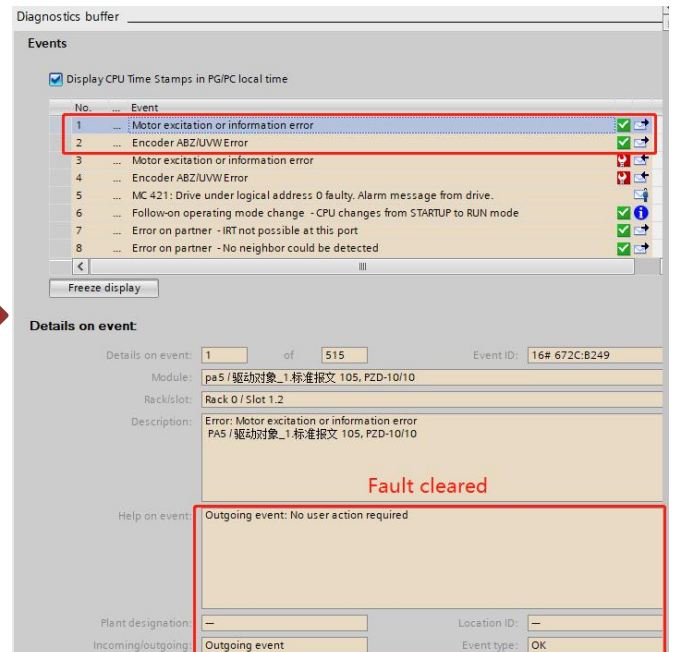
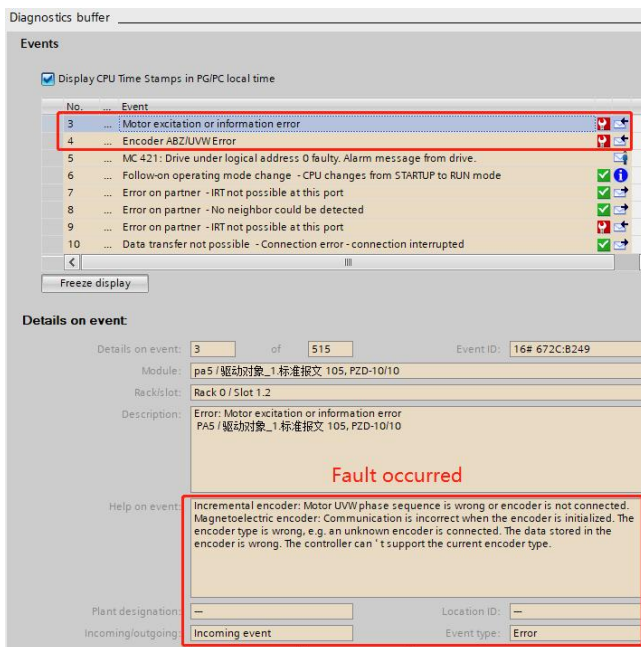
13.4.3.6 Fault diagnosis function

Through the online diagnosis interface of TIA portal, the fault information of the drive can be viewed. Double-click "Online & diagnosis" under PLC_1, click "Diagnosis buffer" in the "Diagnosis" drop-down list, and the drive fault event can be seen, which is consistent with the error display of KincoServo PRO software. The "Details on event" section below provides the cause of the alarm and what to do about it:



错误状态字1			400.3	错误状态字2			100.0
编号	状态	错误描述		编号	状态	错误描述	
0001	●	扩展错误		0001	●	电流传感器故障	
0002	●	编码器ABZ故障或通信故障		0002	●	看门狗故障	
0004	●	编码器UVW故障或内部故障		0004	●	异常中断	
0008	●	编码器计数错误或CRC错误		0008	●	MCU故障	
0010	●	驱动器温度过高		0010	●	电机配置错误	
0020	●	驱动器总线电压过高		0020	●	逻辑输出故障	
0040	●	驱动器总线电压过低		0040	●	STO1	
0080	●	驱动器输出短路		0080	●	STO2	
0100	●	驱动器制动电阻异常		0100	●	扩展能报警	
0200	●	实际跟踪误差超过允许值		0200	●	正限位报警	
0400	●	逻辑电压		0400	●	负限位报警	
0800	●	电机或驱动器 II 故障		0800	●	SPI故障	
1000	●	输入脉冲频率过高		1000	●	CAN总线故障	
2000	●	电机温度过高		2000	●	全闭环故障	
4000	●	电机UVW故障或编码器信息错误		4000	●	主编码器ABZ故障	
8000	●	EEPROM数据错误		8000	●	主编码器计数错误	

When the drive fault has been eliminated and confirmed, the diagnostics buffer will show that the drive fault event has been cleared:



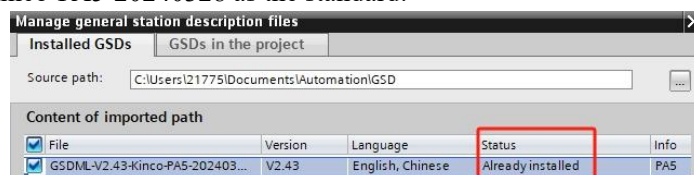
13.4.4 Application class1

13.4.4.1 Overview

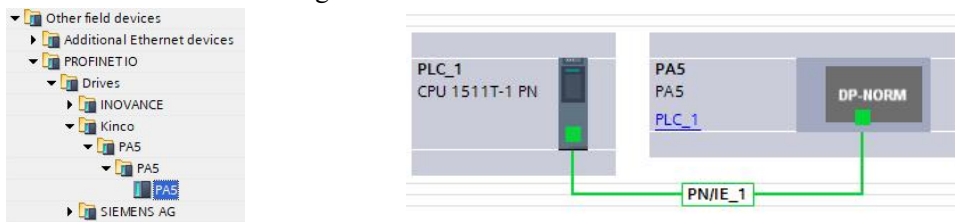
iSMD can use telegram 1 in Application Class 1. PLC can control the drive through FB285 to achieve speed control. Isochronous synchronization (IRT) operation is not supported.

13.4.4.2 Project configuration

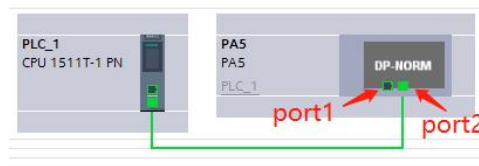
1. See Section 13.4.3.1 for the modification of the drive IP address and device name.
2. Click menu "Options → Manage general station description files" to install GSD, please take the GSDML-V2.43-Kinco-PA5-20240328 as the standard.



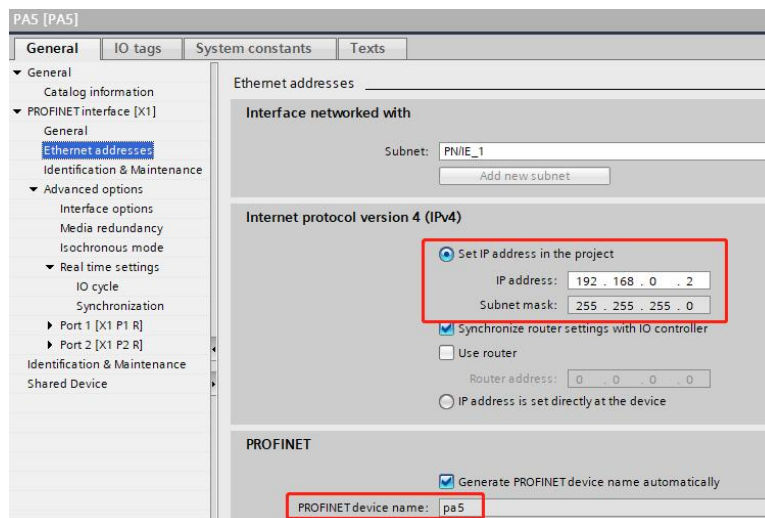
3. In the network view, click "Hardware catalog → Other field devices → PROFINET IO → Drives → Kinco → PA5" to add a drive and assign it to the PLC:



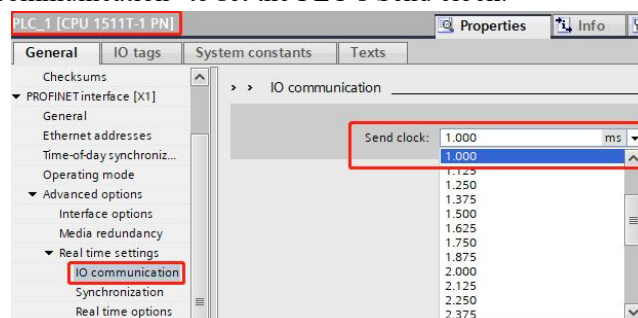
4. If PLC is required to automatically configure the device name of drive (the premise of automatic allocation is that the name of the drive is empty), the topology diagram must be connected. In the topology view, connect the ports according to the actual situation. The IN port of iSMD corresponds to port 1 in the topology diagram



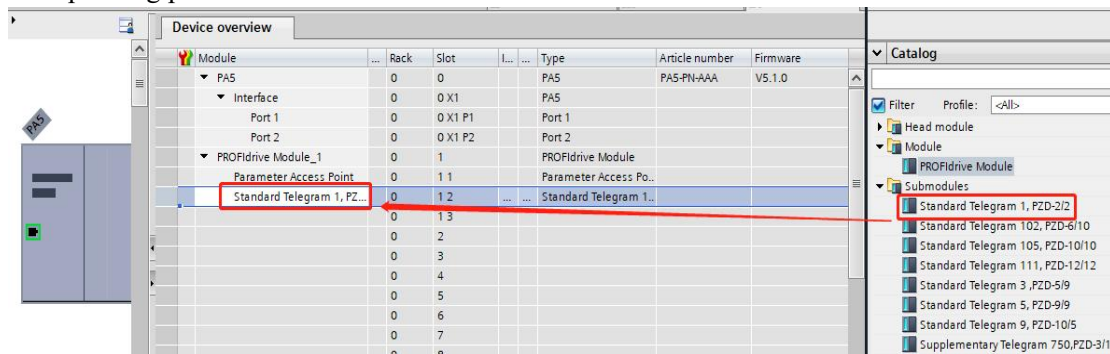
5. Double-click the PA5 module, click "Ethernet addresses" in "General", and set the configured IP address and PROFINET device name. The device names need to be consistent with the actual device names of the drives. When there are multiple drives, the device names cannot be the same:



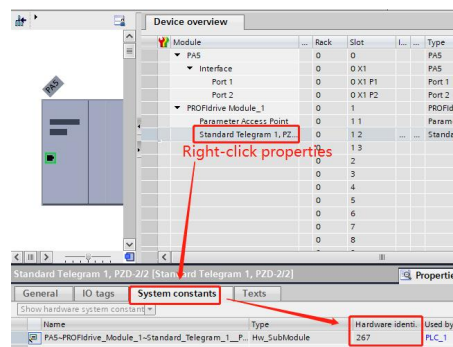
6. In the device view, select PLC_1, click "general → PROFINET interface → Advanced options → Real time settings → IO communication" to set the PLC's Send clock:



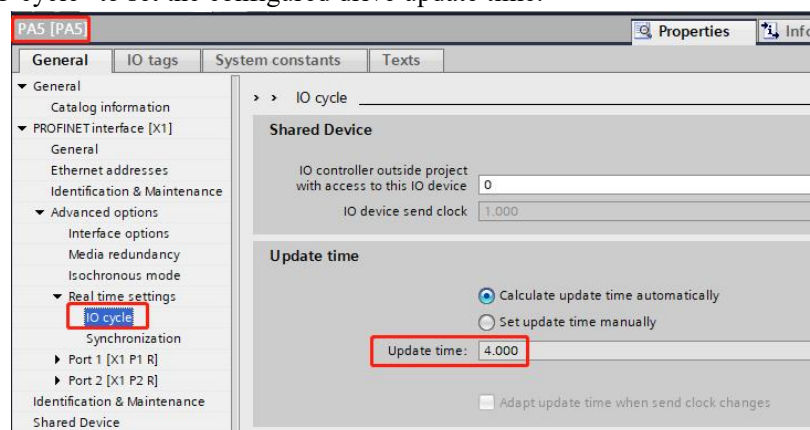
7. In the device view, select PA5, click "Hardware catalog → Module → PROFIdrive Module → Submodules", delete telegram 3 under "Device overview" first, and then add telegram 1 to the corresponding position:



8. Remember the hardware identifier: select telegram 1 → right-click Properties → System constants → Hardware identifier:



9. In the device view, select PA5, click "General → PROFINET interface → Advanced options → Real time settings → IO cycle" to set the configured drive update time:

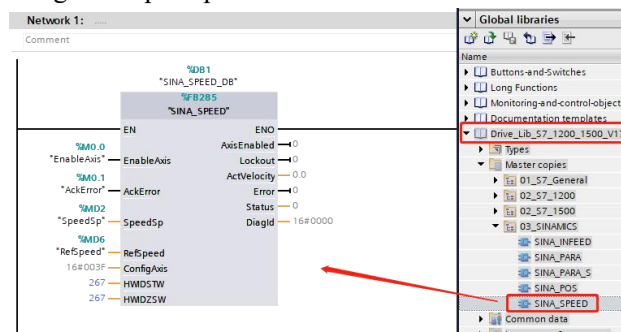


10. Add FB285 to Main[OB1] and assign variables to its pins.

If you cannot find the corresponding function block (FB), you can download the library file Drive_Lib_S7_1200_1500_V17 from Siemens official website:

<https://support.industry.siemens.com/cs/document/109475044/sinamics-communication-blocks-drivelib-for-reading-and-writing-drive-data-within-tia-portal-context?dti=0&lc=en-WW>

After the download is completed, click menu "Options → Global libraries → Open library", and install the corresponding library file according to the prompt.



13.4.4.3 SINA_SPEED (FB285) introduction

FB285 pin definition are as follow:

Table 13-72 FB285 pin definition

Pin name	Data type	Description																		
EnableAxis	Bool	Enable axis: =0 Disable, control word is 0xE =1 Enable, control word is 0xF																		
AckError	Bool	Reset fault, rising edge valid																		
SpeedSp	Real	Speed setpoint in RPM																		
RefSpeed	Real	Reference speed, must be 1/2 of the maximum speed [607F00] in RPM																		
ConfigAxis	Word	The default is 16#3F, and the corresponding relationship with STW1 is as follows:<table><tr><th>Bit in ConfigAxis</th><th>Bit in STW1</th></tr><tr><td>Bit 0</td><td>Bit 1</td></tr><tr><td>Bit 1</td><td>Bit 2</td></tr><tr><td>Bit 2</td><td>Bit 3</td></tr><tr><td>Bit 3</td><td>Bit 4</td></tr><tr><td>Bit 4</td><td>Bit 5</td></tr><tr><td>Bit 5</td><td>Bit 6</td></tr><tr><td>Bit 6</td><td>Bit 11</td></tr><tr><td>Bit 7~15: Reserved</td><td></td></tr></table> See Table 13-55 for the meaning of each bit in STW1	Bit in ConfigAxis	Bit in STW1	Bit 0	Bit 1	Bit 1	Bit 2	Bit 2	Bit 3	Bit 3	Bit 4	Bit 4	Bit 5	Bit 5	Bit 6	Bit 6	Bit 11	Bit 7~15: Reserved	
Bit in ConfigAxis	Bit in STW1																			
Bit 0	Bit 1																			
Bit 1	Bit 2																			
Bit 2	Bit 3																			
Bit 3	Bit 4																			
Bit 4	Bit 5																			
Bit 5	Bit 6																			
Bit 6	Bit 11																			
Bit 7~15: Reserved																				
HWIDSTW	HW_IO	See Section 13.4.4.2, Article 8, by which the parameter is used to distinguish between different axes																		
HWIDZSW	HW_IO																			
AxisEnabled	Bool	=1 Axis is enabled																		
Lockout	Bool	=1 Switching on inhibited																		
ActVelocity	Real	Speed actual value in RPM																		
Error	Bool	=1 Fault present																		
Status	Int	Status indication: 16#7002: No error 16#8401: Drive fault 16#8402: Switching on inhibited 16#8600: DPRD_DAT error 16#8601: DPWR_DAT error																		
DiagId	Word	Extended communication error																		

13.4.5 Application class3

13.4.5.1 Overview

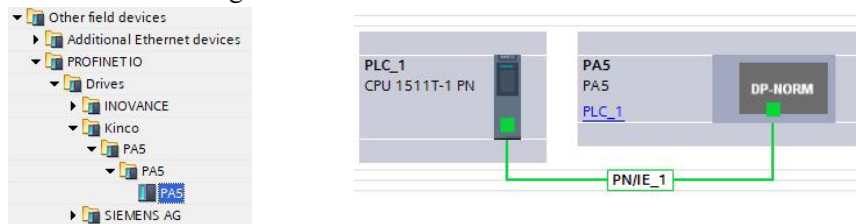
iSMD can select telegram 9 and 111 in Application Class 3. When using telegram 111, PLC can control the drive through FB284 to realize the basic positioning (EPOS) function. The operation modes are Jog, Homing, MDI, position table, etc. Isochronous synchronization (IRT) operation is not supported.

13.4.5.2 Project configuration

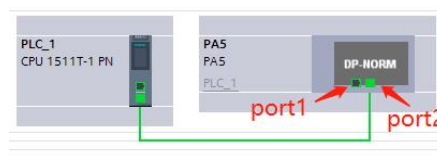
1. See Section 13.4.3.1 for the modification of the drive IP address and device name.
2. Click menu "Options → Manage general station description files" to install GSD, please take the GSDML-V2.43-Kinco-PA5-20240328 as the standard.



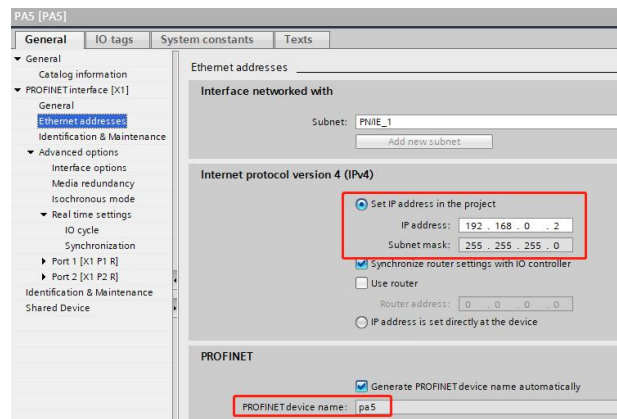
3. In the network view, click "Hardware catalog → Other field devices → PROFINET IO → Drives → Kinco → PA5" to add a drive and assign it to the PLC:



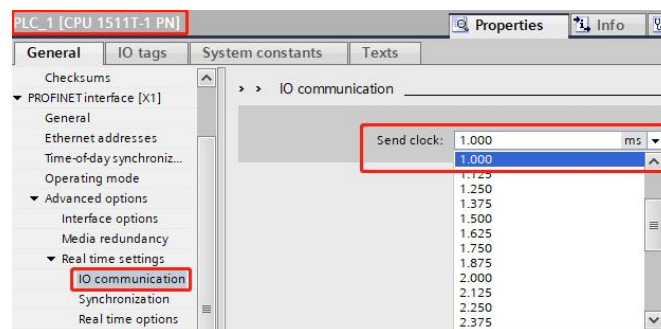
4. If PLC is required to automatically configure the device name of drive (the premise of automatic allocation is that the name of the drive is empty), the topology diagram must be connected. In the topology view, connect the ports according to the actual situation. The IN port of iSMD corresponds to port 1 in the topology diagram:



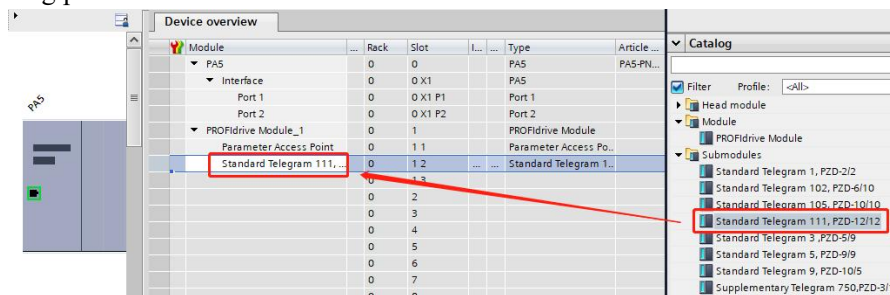
5. Double-click the PA5 module, click "Ethernet addresses" in "General", and set the configured IP address and PROFINET device name. The device names need to be consistent with the actual device names of the drives. When there are multiple drives, the device names cannot be the same:



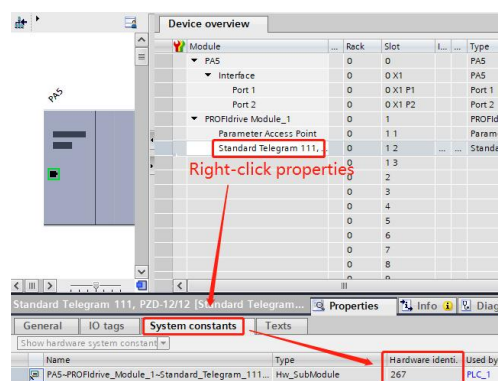
6. In the device view, select PLC_1, click "general → PROFINET interface → Advanced options → Real time settings → IO communication" to set the PLC's Send clock:



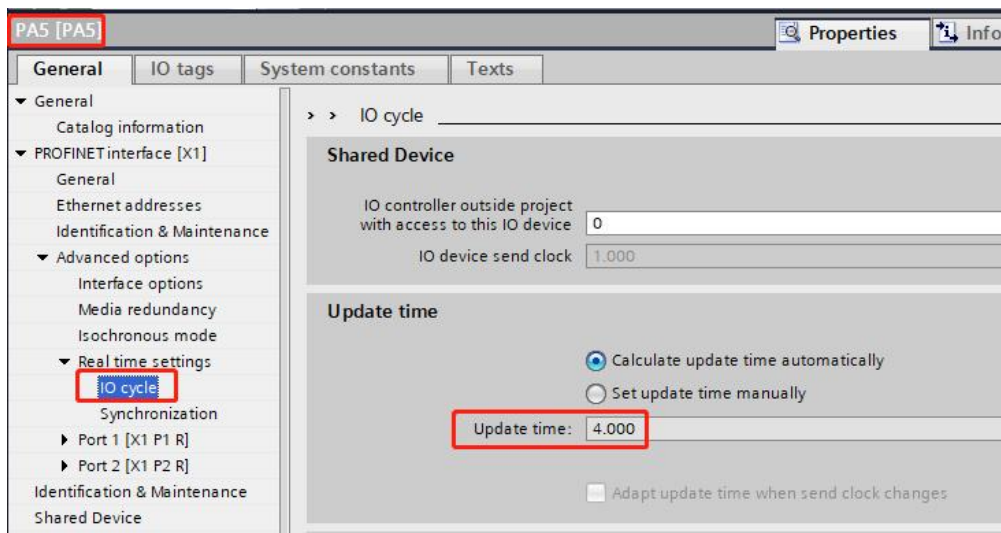
7. In the device view, select PA5, click "Hardware catalog → Module → PROFIdrive Module → Submodules", delete telegram 3 under "Device overview" first, and then add telegram 111 to the corresponding position:



8. Remember the hardware identifier: select telegram 111 → right-click Properties → System constants → Hardware identifier:



9. In the device view, select PA5, click "General → PROFINET interface → Advanced options → Real time settings → IO cycle" to set the configured drive update time:

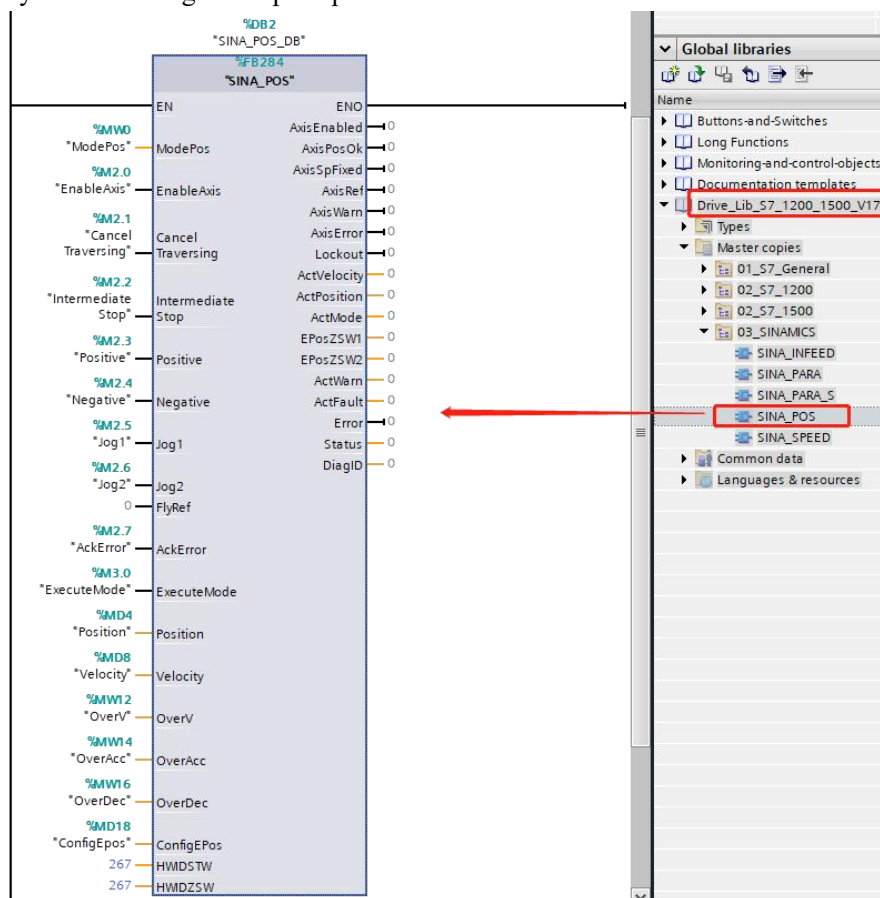


10. Add FB284 to Main[OB1] and assign variables to its pins.

If you cannot find the corresponding function block (FB), you can download the library file Drive_Lib_S7_1200_1500_V17 from Siemens official website:

<https://support.industry.siemens.com/cs/document/109475044/sinamics-communication-blocks-drivelib-for-reading-and-writing-drive-data-within-tia-portal-context?dti=0&lc=en-WW>

After the download is completed, click menu "Options → Global libraries → Open library", and install the corresponding library file according to the prompt.



13.4.5.3 SINA_POS (FB284) introduction

FB284 pin definition are as follow:

Table 13-73 FB284 pin definition

Pin name	Data type	Description																
ModePos	Int	Operating mode: =0 — =1 Relative positioning =2 Absolute positioning =3 — =4 Active homing =5 Direct homing =6 Position table (not supported) =7 Jog at the specified speed =8 —																
EnableAxis	Bool	=0 Disable, control word is 0xE =1 Enable, control word is 0xF																
CancelTraversing	Bool	=1 Do not reject traversing task =0 Reject traversing task																
IntermediateStop	Bool	=1 No intermediate stop =0 Intermediate stop																
Positive	Bool	Positive direction																
Negative	Bool	Negative direction																
Jog1	Bool	Jog reversely																
Jog2	Bool	Jog forward																
FlyRef	Bool	Invalid, need to be 0																
AckError	Bool	Reset fault, rising edge valid																
ExecuteMode	Bool	Activate operating mode, rising edge valid																
Position	DInt	Position set point for operating mode 1, 2 Position table start/New task index for operating mode 6. The value ranges from 0 to 7																
Velocity	DInt	Speed set point for operating mode 1, 2, 7																
ConfigEPos	DWord	The default is 16#00000003. The corresponding relationship with relevant control word is as follows:<table><tr><th>Bit in ConfigEPos</th><th>Bit in control word</th></tr><tr><td>Bit 0</td><td>STW1 Bit 1</td></tr><tr><td>Bit 1</td><td>STW1 Bit 2</td></tr><tr><td>Bit 2</td><td>POS_STW2 Bit 14</td></tr><tr><td>Bit 3</td><td>POS_STW2 Bit 15</td></tr><tr><td>Bit 6</td><td>POS_STW2 Bit 2</td></tr><tr><td>Bit 8</td><td>POS_STW1 Bit 12</td></tr><tr><td>Other Bits: Reserved</td><td></td></tr></table> See Table 13-56 for the meaning of each bit in STW1 See Table 13-63 for the meaning of each bit in the POS_STW1 See Table 13-64for the meaning of each bit in the POS_STW2	Bit in ConfigEPos	Bit in control word	Bit 0	STW1 Bit 1	Bit 1	STW1 Bit 2	Bit 2	POS_STW2 Bit 14	Bit 3	POS_STW2 Bit 15	Bit 6	POS_STW2 Bit 2	Bit 8	POS_STW1 Bit 12	Other Bits: Reserved	
Bit in ConfigEPos	Bit in control word																	
Bit 0	STW1 Bit 1																	
Bit 1	STW1 Bit 2																	
Bit 2	POS_STW2 Bit 14																	
Bit 3	POS_STW2 Bit 15																	
Bit 6	POS_STW2 Bit 2																	
Bit 8	POS_STW1 Bit 12																	
Other Bits: Reserved																		
OverV	Int	The percentage of the speed set point when the operating mode is 1, 2, and 7, unit : %, Range 0 to 199%																
OverAcc	Int	The percentage of the acceleration when the operating mode is 1, 2, and 7, unit : %, Range 0 to 100%																
OverDec	Int	The percentage of the deceleration when the operating mode is 1, 2, and 7, unit : %, Range 0 to 100%																
HWIDSTW	HW_IO	See Section13.4. 5.2, Step 8, for this parameter to distinguish between different axes																
HWIDZSW	HW_IO																	
AxisEnabled	Bool	=1 Axis is enabled																
AxisPosOk	Bool	=1 Target position reached																
AxisSpFixed	Bool	=1 Setpoint position reached																

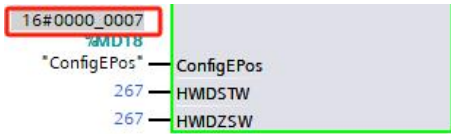
Pin name	Data type	Description
AxisRef	Bool	=1 Reference point set
AxisWarn	Bool	=1 Warn present
AxisError	Bool	=1 Fault present
Lockout	Bool	=1 Switching on inhibited
ActVelocity	DInt	Speed actual value
ActPosition	DInt	Position actual value
ActMode	Int	Current operating mode
EPosZSW1	Word	POS_ZSW1 current value, see Table 13-65
EPosZSW2	Word	POS_ZSW2 current value, see Table 13-66
ActWarn	Word	Servo alarm code
ActFault	Word	Drive error code [603F00]
Error	Bool	=1 Fault present
Status	Word	Status indication: 16#7002: No error 16#8401: Drive fault 16#8402: Switching on inhibited 16#8403: Homing procedure forbidden during positioning 16#8600: DPRD DAT error 16#8601: DPWR DAT error 16#8202: Operating mode is selected incorrectly 16#8203: Setpoint is incorrect 16#8204: Position table task index error
DiagID	Word	Extended communication error

13.4.5.4 FB284 function description

13.4.5.4.1 Operation condition and setting

1. The initial value of the input pin "ConfigEPos" should be 3 (Bit0 and Bit1 set to 1).
2. The input pins "CancelTraversing" and "IntermediateStop" are valid for all modes except jog mode, and should be set to 1 during operation, as follows:
 - a. When "CancelTraversing" = 0, it means that the current task is canceled, and the deceleration corresponds to the halt mode [605D00]. The operating mode can be switched after the axis stops.
 - b. When "IntermediateStop" = 0, it means to pause the current task, and the deceleration corresponds to the halt mode [605D00]. The axis will continue to run after resetting "IntermediateStop" = 1. The operating mode can be switched after the axis stops.
3. The input pin "ModePos" is used to select the operating mode; the rising edge of "ExecuteMode" triggers the positioning movement.
4. Activate software limit switch

After setting the reference point, if you need to use a software limit switch, you need to set bit 2 of the input pin "ConfigEPos" to 1 ("ConfigEPos" = 16#00000007) or set the En_SoftWare_Limit [30800C] to 1.



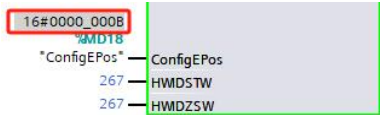
Activate the software limit function, set the soft positive limit [607D01] and the soft negative limit [607D02] for the drive.

30800C	En_SoftWare_Limit	1
607D01	Soft_Positive_Limit	10000000
607D02	Soft_Negative_Limit	-10000000

After you set soft positive limit more than soft negative limit and set the reference point, the software limit function is enabled.

5. Activate hardware limit switch

If you need to use a hardware limit switch, you need to set bit 3 of the input pin "ConfigEPos" to 1 ("ConfigEPos" = 16#0000000B):



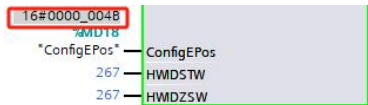
Use KincoServo PRO software to define the positive limit and negative limit for the DIN of the drive. The drive can only be operated when the hardware limit switch signal is high.

PIN number	Function	Actual input	Simulation input	Polarity	Valid input
DIN1	Homing Signal	☐	☐	⌘ High	☐
DIN2	Position limit +	☐	☐	⌘ High	☐
DIN3	Position limit -	☐	☐	⌘ High	☐

Please note that only positive and negative limits are configured in the KincoServo PRO software, but bit 3 of "ConfigEPos" is not set to 1, then the hardware limit function will not take effect.

6. Activate homing switch

The homing switch signal can be activated by setting bit 6 of the input pin "ConfigEPos" to 1 ("ConfigEPos" = 16#0000004B):



The same purpose can be achieved by configuring the homing signal for the drive's DIN on the KincoServo PRO software:

PIN number	Function	Actual input	Simulation input	Polarity	Valid input
DIN1	Homing Signal	☐	☐	⌘ High	☐
DIN2	Position limit +	☐	☐	⌘ High	☐
DIN3	Position limit -	☐	☐	⌘ High	☐

7. Setting and reading servo parameters using telegram 111

The PLC allocates 24-byte receiving and sending areas for telegram 111, which can be viewed in the FB284 DB block. The "Reserve" variables under the structures "sxSendBuf" and "sxRecvBuf" can be used to customize the settings and read servo parameters:

SINA_POS_DB			
	Name	Data type	Start value
37	▼ sxSendBuf	Struct	
38	STW1	Word	WORD# 16#0000
39	EPosSTW1	Word	WORD# 16#0000
40	EPosSTW2	Word	WORD# 16#0000
41	STW2	Word	WORD# 16#0000
42	OverrideV	Word	WORD# 16#0000
43	Position	DWord	DWORD# 16#0000
44	Velocity	DWord	DWORD# 16#0000
45	OverrideA	Word	WORD# 16#0000
46	OverrideD	Word	WORD# 16#0000
47	Reserve	Word	WORD# 16#0000
48	▼ sxRecvBuf	Struct	
49	ZSW1	Word	WORD# 16#0000
50	EPosZSW1	Word	WORD# 16#0000
51	EPosZSW2	Word	WORD# 16#0000
52	ZSW2	Word	WORD# 16#0000
53	NotUsed	Word	WORD# 16#0000
54	Position	DWord	DWORD# 16#0000
55	Velocity	DWord	DWORD# 16#0000
56	ErrNr	Word	WORD# 16#0000
57	WarnNr	Word	WORD# 16#0000
58	Reserve	Word	WORD# 16#0000

The relevant servo parameters are the PN_User_Receive [30800D] and the PN_User_Send [30800E]:

30800D	PN_User_Receive	0
30800E	PN_User_Send	0

The explanation is as follow:

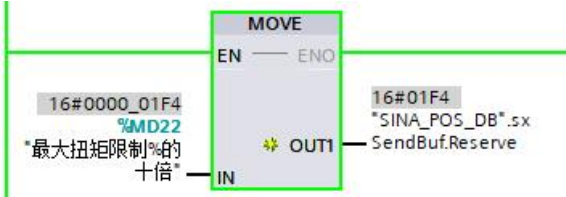
PN_User_Receive	=0 non-functional =1 Enable torque limiting and transmit the torque limit value in "Reserve" under "sxSendBuf"
PN_User_Send	=0 non-functional =2 Transfer the actual current in "Reserve" under "sxRecvBuf"

a. Torque limiting function

When using the torque limiting function, please set the appropriate torque limit value; if the limit value is too small, it may cause the drive's following error to be too large.

The "Reserve" variable of "sxSendBuf" in the FB284 DB block corresponds to ten times the Max_Torque%[607200].

First set "Reserve" to 16#1F4 = 500, then the Max_Torque% should be 50%:



Then set the PN_User_Receive [30800D] to 1: torque as follows:

Index	Description	Preset value	Current value
30800D	PN_User_Receive	0:non 1:torque	0:non

Max_Torque% has become 50%:

Index	Description	Preset value ▼	Current value
607200	Max_Torque%		50

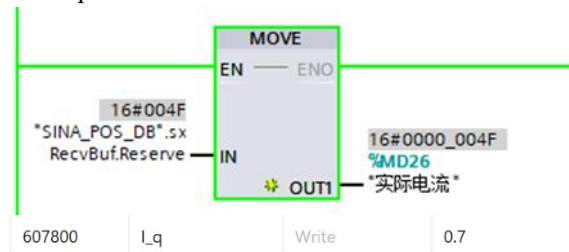
Please note that once [30800D] is set to 1, the torque limit becomes effective, and after setting [30800D] to 0 again, the torque limit value does not revert to the initial value, but it is possible to set the torque limit [607200] directly on the KincoServo PRO software.

b. Read actual current

First set the PN_User_Send [30800E] to 2: I_q as follows:

Index	Description	Preset value ▼	Current value ▼
30800E	PN_User_Send	0:non 1:actual torque 2:I_q	0:non DEC

In FB284 DB block, "sxRecvBuf" → "Reserve" variable reads drive actual current, unit DEC, which needs to be converted, such as the reading value converted to Ap: $16 \# 4F = 79, 79 / 2048 * 18A_p$ (I_{Max} [651003]) = 0.7Ap:



13.4.5.4.2 Operation mode 1(Relative position)

Requirement:

1. Operating mode "ModePos" = 1
2. "ConfigEPos" = 16#00000003
3. "EnableAxis" = 1
4. "CancelTraversing" and "IntermediateStop" must be 1, and "Jog1" and "Jog2" must be 0.

Step:

1. Specify the target position and target speed through the input pin "Position" and "Velocity". The unit of position and velocity is DEC. Refer to the drive manual for conversion, and the unit of velocity is transformed into: $DEC = RPM * 512 * feedback_resolution [641003] / 1875$.
2. Use "OverV", "OverAcc" and "OverDec" to scale the percentage of the target speed and profile acceleration and deceleration.
3. The direction of movement is determined by the positive or negative value set in "Position".
4. The movement is triggered by the rising edge of "ExecuteMode", and "AxisPosOk" is set to 1 after reaching the target position. If there is an error locating, "Error" is set to 1.

5. The currently running command can be replaced with a new command via the "ExecuteMode" rising edge. At any time, it can switch between operating mode 1 and 2. If you need to switch to another operating mode, the axis must be stationary.

13.4.5.4.3 Operation mode 2(Absolute position)

Requirement:

1. Operating mode "ModePos" = 2
2. "ConfigEPos" = 16#00000003
3. "EnableAxis" = 1
4. "CancelTraversing" and "IntermediateStop" must be 1, and "Jog1" and "Jog2" must be 0.

Steps:

1. Specify the target position and target speed through the input pin "Position" and "Velocity". The unit of position and velocity is DEC. Refer to the drive manual for conversion, and the unit of velocity is transformed into: $DEC = RPM * 512 * \text{feedback resolution} [641003] / 1875$.
2. Use "OverV", "OverAcc" and "OverDec" to scale the percentage of the target speed and profile acceleration and deceleration.
3. When running, the drive will follow the shortest path to the target position, and "Positive" and "Negative" must be 0.
4. The movement is triggered by the rising edge of "ExecuteMode", and "AxisPosOk" is set to 1 after reaching the target position. If there is an error locating, "Error" is set to 1.
5. The currently running command can be replaced with a new command via the "ExecuteMode" rising edge. At any time, it can switch between operating mode 1 and 2. If you need to switch to another operating mode, the axis must be stationary.
6. By setting bit 8 of "ConfigEPos" ("ConfigEPos" = 16#00000103), after updating "Position" on the PLC, there is no need to trigger "ExecuteMode", the new setting value will take effect immediately, and the servo will immediately execute the absolute positioning instruction according to the target position change.

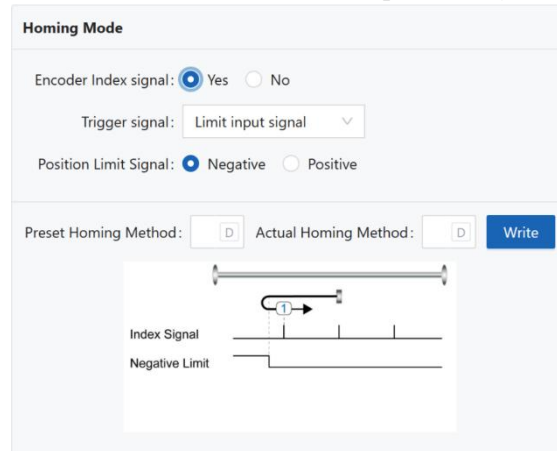
13.4.5.4.4 Operation mode 4(Active homing)

Requirement:

1. Operating mode "ModePos" = 4
2. "ConfigEPos" = 16#00000003. If hardware limits are used, "ConfigEPos" = 16#0000000B.
3. "EnableAxis" = 1
4. "CancelTraversing" and "IntermediateStop" must be 1, and "Jog1" and "Jog2" must be 0.

Steps:

1. The way and the speed of returning to the homing should be configured by the KincoServo PRO software (PLC only sends the command, and the planning of returning to the homing is set by the servo parameters). Please refer to the drive manual for the specific way of returning to the homing:



2. A return to the homing motion is triggered by the rising edge of "ExecuteMode". After the motion, "AxisRef" is set to 1, if there is an error during the operation, "Error" is set to 1.

13.4.5.4.5 Operation mode 5(Direct homing)

Requirements:

1. Operating mode "ModePos" = 5
2. "ConfigEPos" = 16#00000003
3. "EnableAxis" = 1
4. The axis must be stationary.

Steps:

The homing of the axis is set by the rising edge of the "ExecuteMode". When the homing operation has been done, the output pin "AxisRef" turns to be 1.

13.4.5.4.6 Operation mode 7(Jog at the specified speed)

Requirements:

1. Operating mode "ModePos" = 7
2. "ConfigEPos" = 16#00000003
3. "EnableAxis" = 1
4. The axis must be stationary.

Steps:

1. Specify the jog speed through the input pin "Velocity", which must be a positive value in unit DEC.
2. Use "OverV", "OverAcc" and "OverDec" to scale the percentage of the jog speed and profile acceleration and deceleration.
3. "CancelTraversing" and "IntermediateStop" have nothing to do with the jog mode. The running

direction is independent of "Positive" and "Negative".

- There is no need to trigger "ExecuteMode". Reverse jogging when "Jog1" = 1, forward jogging when "Jog2" = 1. "Jog1" and "Jog2" can only be triggered separately. When "Jog1" and "Jog2" are 0 or 1 at the same time, the axis stops.

13.4.6 Application class4

13.4.6.1 Overview

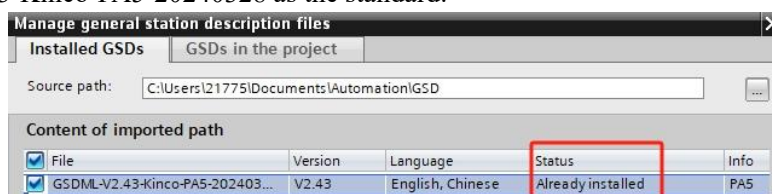
iSMD can select telegram 3, 5, 102, 105 in Application Class 4. The technology object is configured in PLC and controlled using PLC_open function blocks such as MC_Power and MC_MoveAbsolute, where the position loop calculation of the servo three-loop control is completed in PLC, and the drive is in speed mode. The functional differences between the different telegrams are shown in the following table.

Table 13-74 Application Class 4 telegram function difference

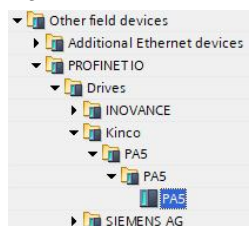
Telegram	IRT supported	Torque limiting supported	DSC supported
105	Yes	Yes	Yes
102	Yes	Yes	No
5	Yes	No	Yes
3	Yes	No	No

13.4.6.2 Project configuration

- See Section 13.4.3.1 for the modification of the drive IP address and device name.
- Click menu "Options → Manage general station description files" to install GSD, please take the GSDML-V2.43-Kinco-PA5-20240328 as the standard.



- In the network view, click "Hardware catalog → Other field devices → PROFINET IO → Drives → Kinco → PA5" to add a drive and assign it to the PLC:

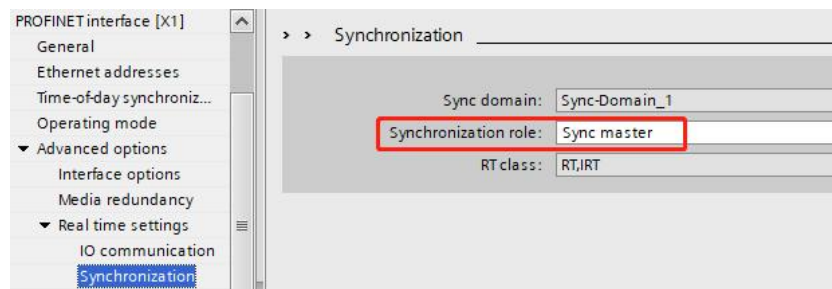
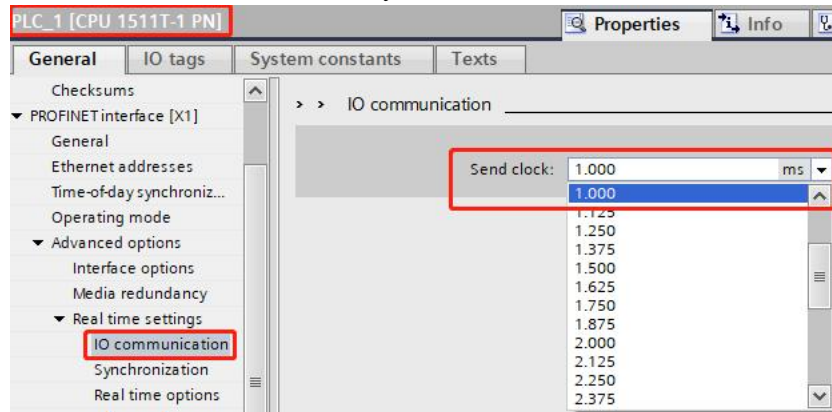


- If PLC is required to automatically configure the device name of drive (the premise of automatic allocation is that the name of the drive is

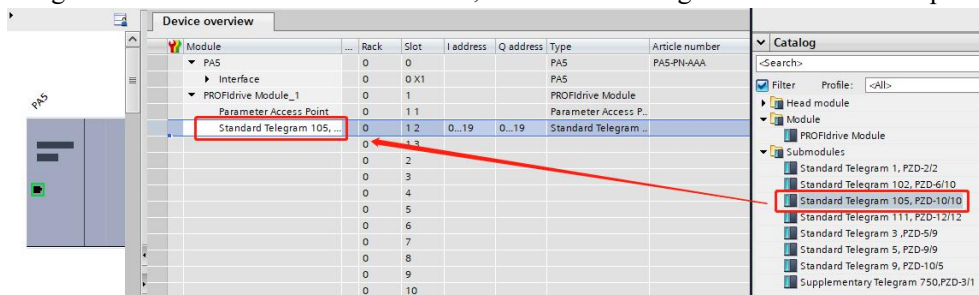
empty), the topology diagram must be connected. In the topology view, connect the ports according to the actual situation. The IN port of iSMD corresponds to port 1 in the topology diagram:



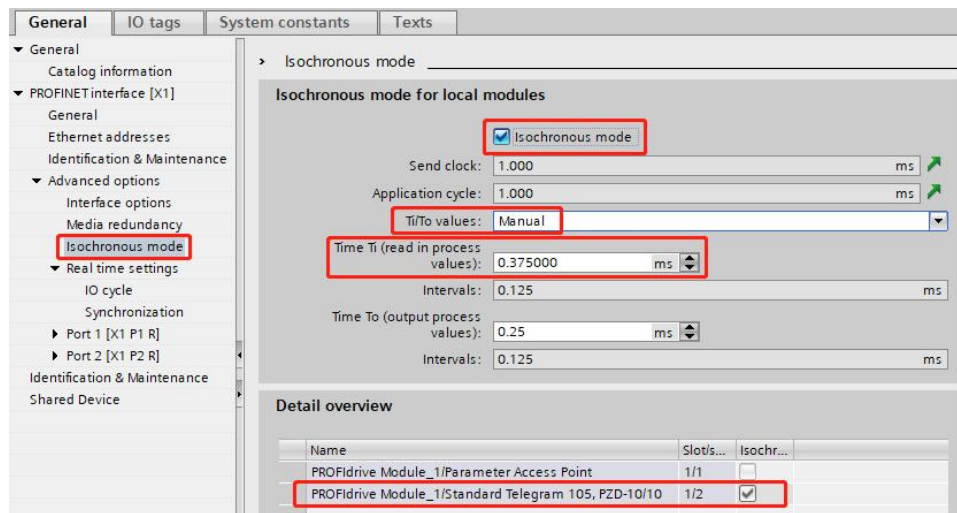
- In the device view, select PLC_1, click "General → PROFINET interface → Advanced options → Real time settings" to set the send clock and enable the synchronization mode:



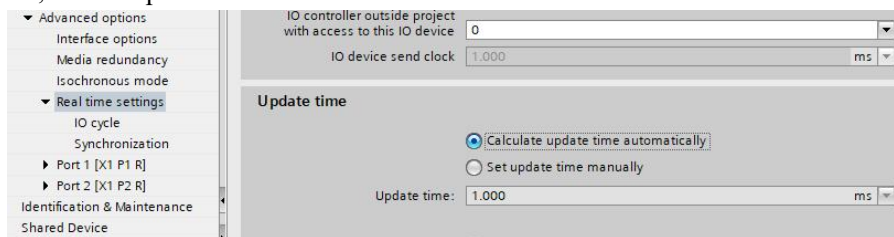
- In the device view, select PA5, click "Hardware catalog → Module → PROFIdrive Module → Submodules", delete telegram 3 under "Device overview" first, and then add telegram 105 to the corresponding position:



- Click "General → PROFINET interface → Advanced options → Isochronous mode", then check telegram 105 below to enable the synchronization mode, and set Ti to 0.375ms:



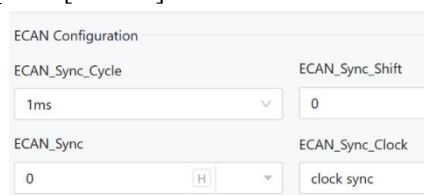
8. In IRT mode, device update time = PLC send clock:



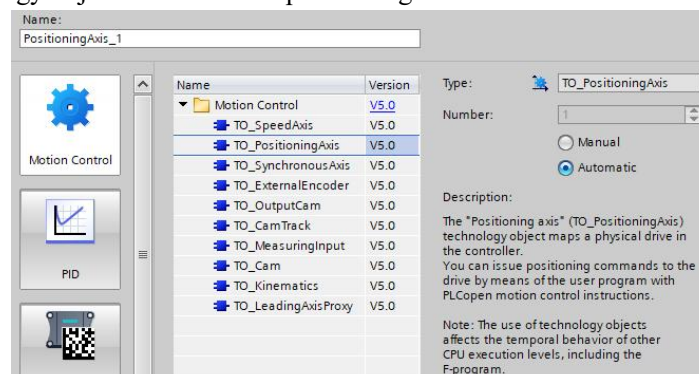
The update time of the configuration needs to be consistent with the ECAN_Sync_Cycle [301101], set to 3 for 1ms:

ECAN_Sync_Cycle [301101] setting value	Synchronous period
3	1ms
4	2ms
5	4ms
6	8ms

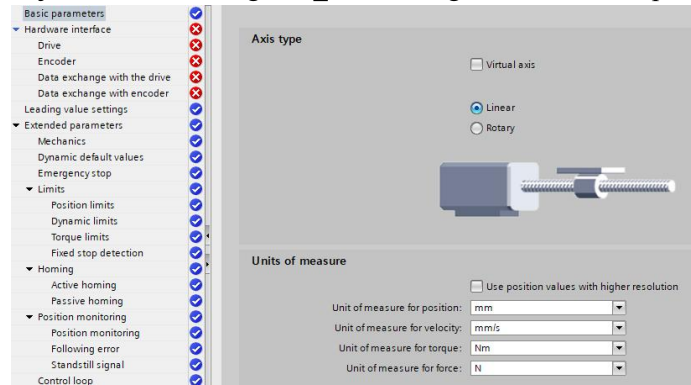
9. Enable isochronous synchronization mode on the drive side by setting ECAN_Sync_Clock [301102] to 1 and also setting ECAN_Sync_Shift [301103] to 0.



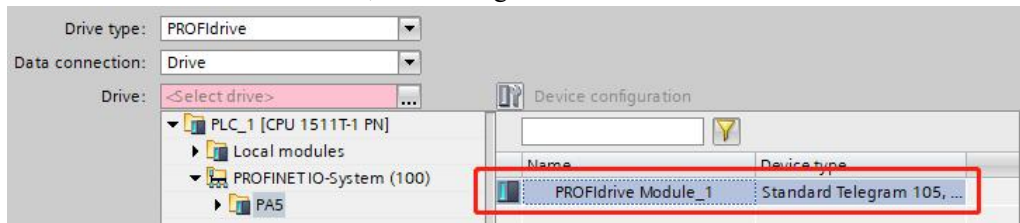
10. After the device is configured, click "Technology objects" in the drop-down list of PLC_1 on the left side to add a new technology object and select the positioning axis:



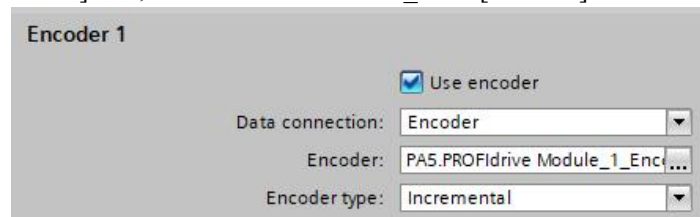
11. Click "Technology objects → PositioningAxis_1 → Configuration → Basic parameters", select Axis type:



12. Click "Hardware interface → Drive", select telegram 105:



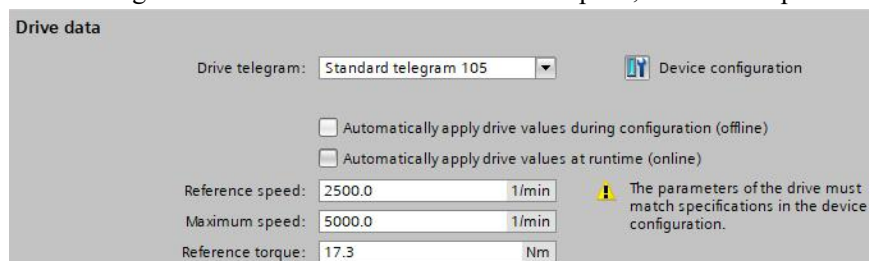
13. iSMD support incremental and absolute encoders. Click "Hardware interface → Encoder"
- a. If Kinco single-turn encoder motor is used, select "Incremental". When activating the DSC function, set the Store_Position [60FB06] to 1, then write 0 to the Pos_Shift [60FB07] and store the control parameters.



- b. Select "Cyclic absolute" if a Kinco multi-turn absolute encoder motor is used:



14. Click on "Data exchange with the drive" to set the Reference speed, Maximum speed and Reference torque:



The Reference speed is 1/2 of the Max_Speed [607F00]:

Index	Description	Preset value	Current value
607F00	Max_Speed		5000

CMD_q_Max [607300] and Max_Torque% [607200] of the drive are related, and the Reference torque can be taken as the initial default value of CMD_q_Max [607300], with the unit of Ap, which is generally the lesser of the two values of the drive peak current and the motor peak current:

Index	Description	Preset value	Current value
607300	CMD_q_Max		17.3

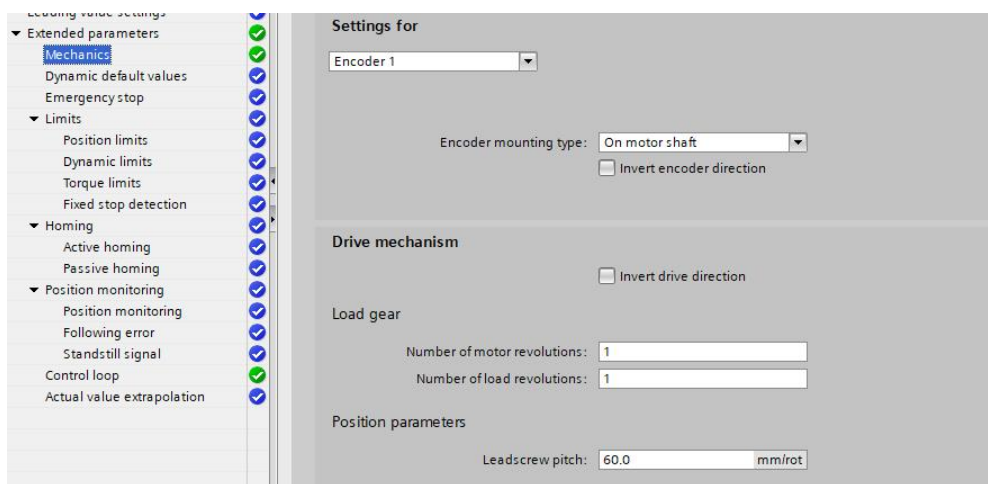
15. Click on "Data exchange with encoder". If a Kinco single-turn encoder motor is used, the Increments per revolution is Feedback_Resolution [641003]. For Fine resolution, select Gx_XIST1 bit 0:

Index	Description	Preset value	Current value
641003	Feedback_Resolution		65536

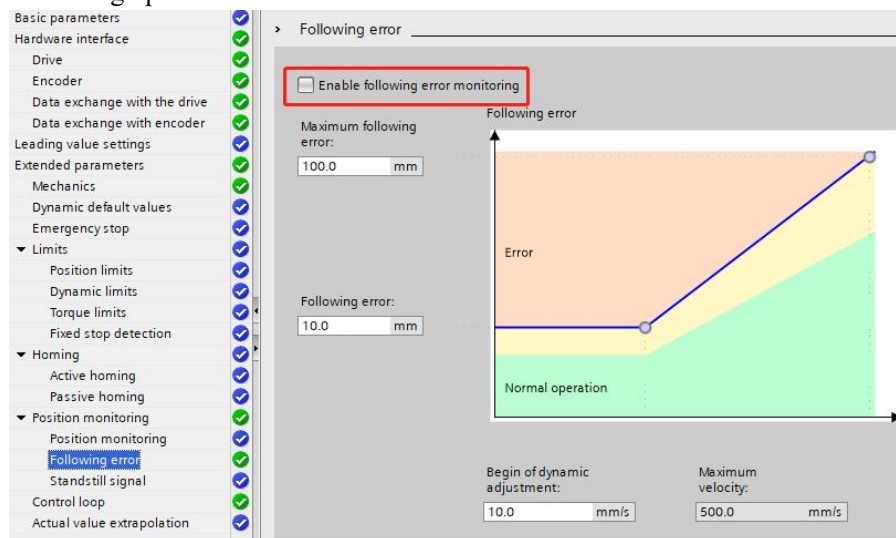
If a Kinco multi-turn absolute encoder motor is used, the Increments per revolution is Feedback_Resolution [641003], e.g. 65536, and the Number of revolutions is also set to 65536, with Fine resolution selections G1_XIST1 bit 0 and G1_XIST2 bit 0:

Index	Description	Preset value	Current value
641003	Feedback_Resolution		65536

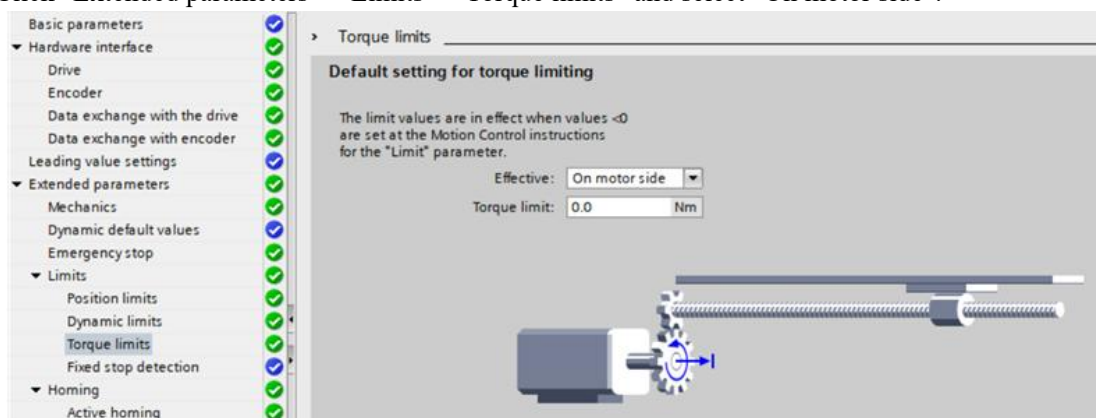
16. Click "Extended parameters → Mechanics" to set the position parameters :



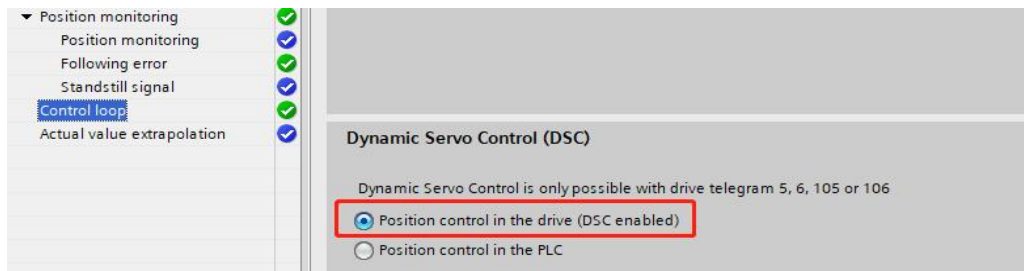
17. Click "Extended parameters → Position monitoring → Following error" to disable the following error monitoring. When monitoring is enabled and the following error setting is small, the PLC will report error easily after enabling operation:



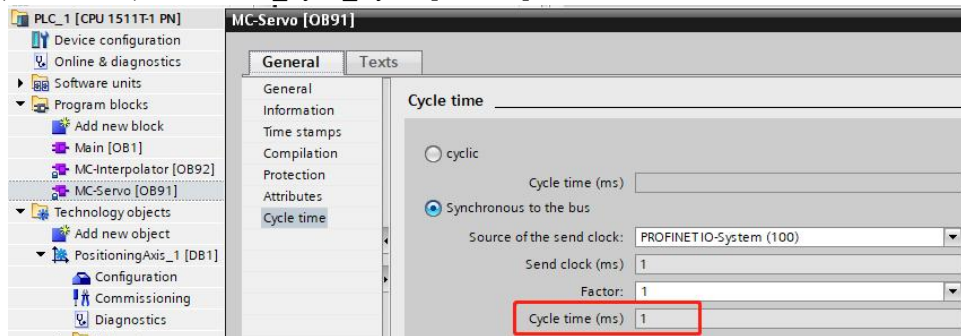
18. Click "Extended parameters → Limits → Torque limits" and select "On motor side":



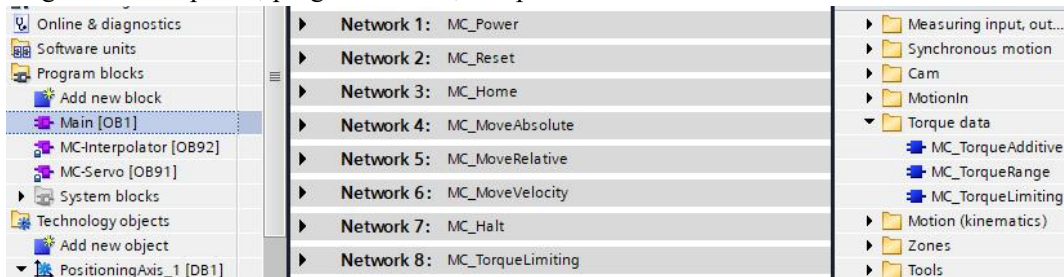
19. DSC can be enabled by clicking "Extended parameters → Control loop", see section 13.4.6.5 for details on how to use it.



20. After the technology object has been configured, click "Program blocks → MC-Servo [OB91]" in the drop-down list of PLC_1 on the left side, and then right-click to open the properties to set the PLC application cycle. Please note that when using the DSC function, the PLC application cycle, device update cycle (see Article 8) and ECAN_Sync_Cycle [301101] must be the same.

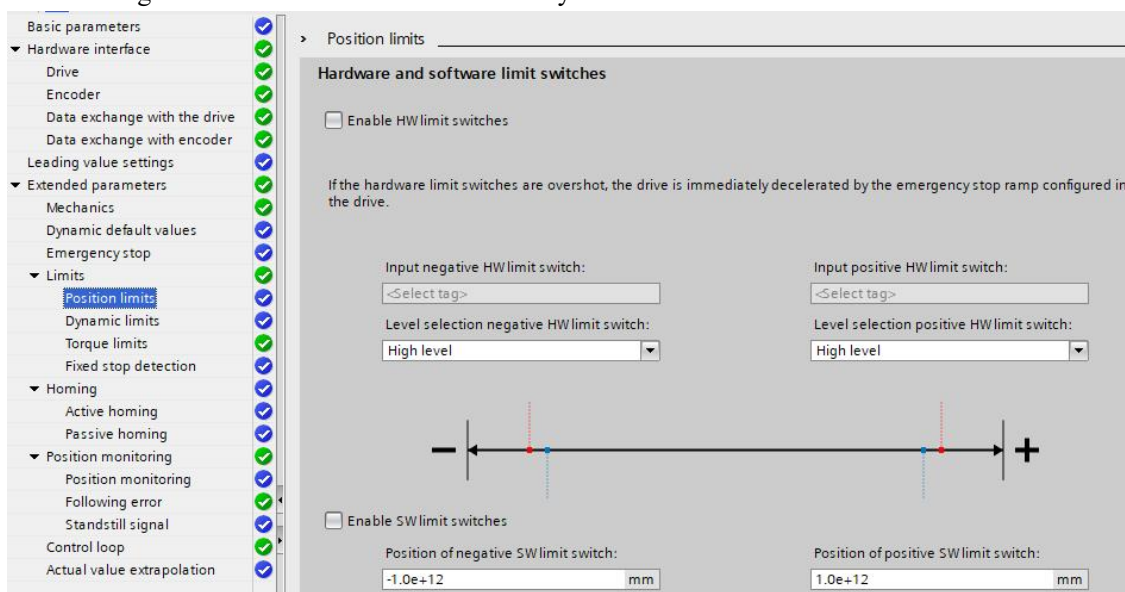


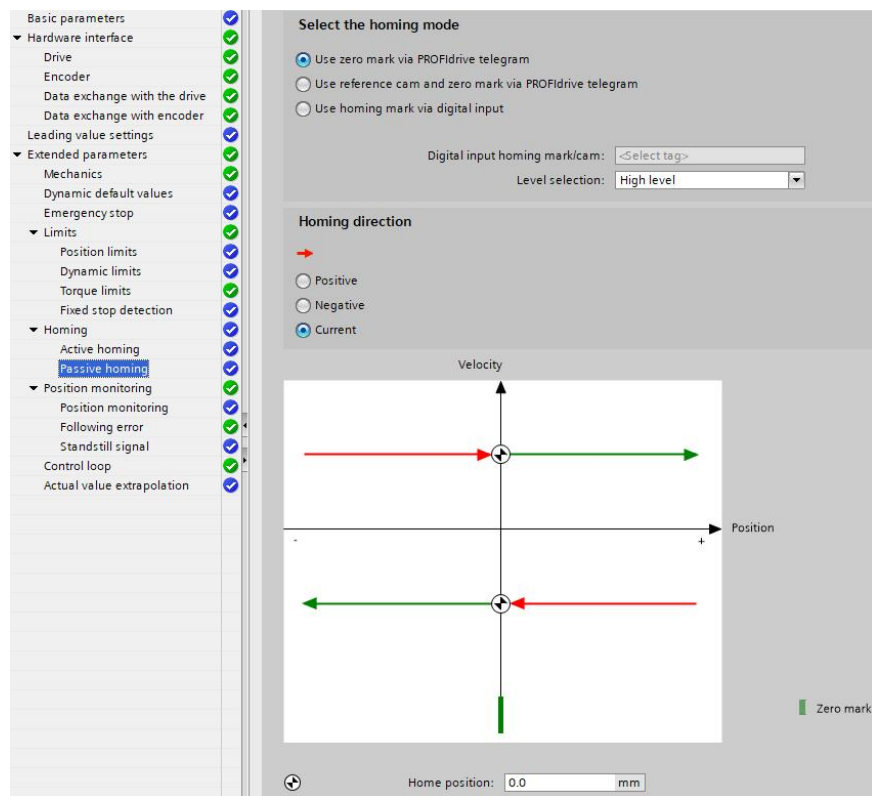
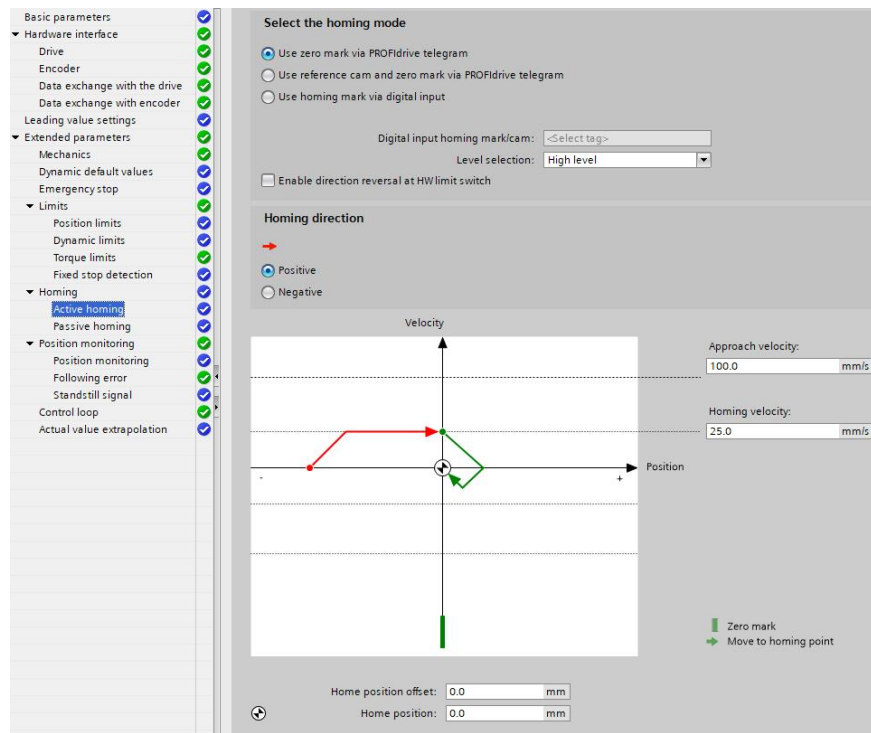
21. Configuration completed, program written, compiled and downloaded to PLC:



13.4.6.3 MC_Home introduction

The drive must complete homing motion before absolute positioning. The homing trajectory is generated by the PLC, and the homing switch and limit switches are usually connected to the PLC:





MC_Home pin definition are as follow:

Table 13-75 MC_Home pin definition

Pin name	Data type	Description
Axis	TO_PositioningAxis TO_SynchronousAxis TO_ExternalEncoder	Technology objects
Execute	Bool	Start job with a rising edge

Pin name	Data type	Description
Position	LReal	The specified value is used according to the selected "Mode"
Mode	Int	Homing modes: = 0 Direct homing (absolute), the current position of the technology object is set to the value of parameter "Position". = 1 Direct homing (relative), the current position of the technology object is shifted by the value of parameter "Position" = 2 Passive homing, the current position of the technology object is set to the value of parameter "Position". = 3 Active homing, the current position of the technology object is set to the value of parameter "Position". = 6 Absolute encoder adjustment (relative), the current position is shifted by the value of parameter "Position". = 7 Absolute encoder adjustment (absolute), the current position is set to the value of parameter "Position".
ReferenceMarkPosition	LReal	Display of the position at which the technology object was homed (valid when "Done" = TRUE)
Done	Bool	= 1 Job is completed
Busy	Bool	= 1 The job is being processed
CommandAborted	Bool	=1 The job was aborted by another job during execution
Error	Bool	=1 Error, see "ErrorID"
ErrorID	Word	Please consult the information system of TIA portal for help

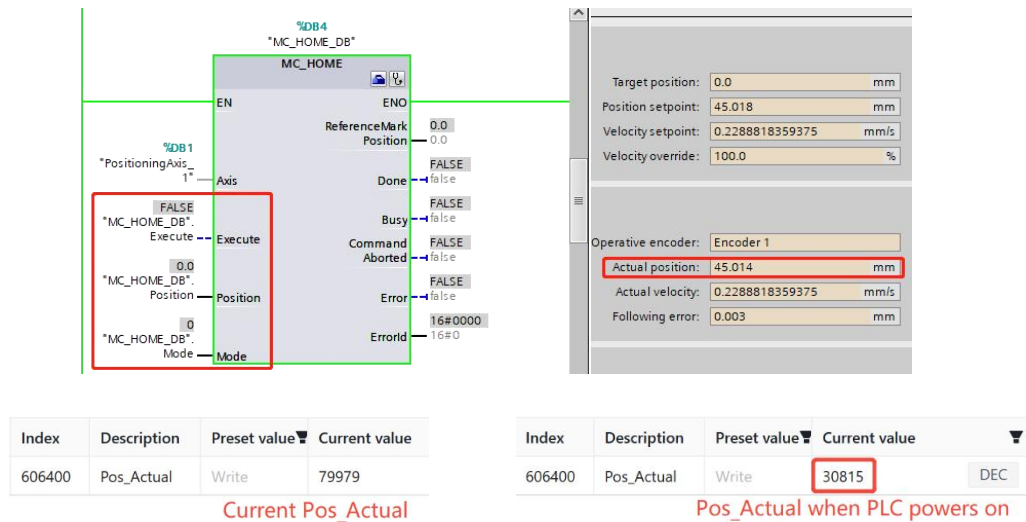
The homing mode supported by different encoders are shown in the following table.

Table 13-76 Supported homing mode

Homing mode	Incremental encoder (Kinco single-turn encoder)	Absolute encoder (Kinco multi-turn absolute encoder)
Active homing ("Mode"= 3)	√	—
Passive homing ("Mode"= 2)	√	—
Set actual position ("Mode"= 0)	√	√
Relative offset to the actual position ("Mode"= 1)	√	√
Absolute encoder adjustment ("Mode"= 6,7)	—	√

13.4.6.3.1 Homing mode 0 (Absolute direct homing)

The axis does not move after the MC_Home is executed. The result of executing the instruction is that the actual position of the axis is directly changed to "Position specified in MC_Home". In the following example, "Position" = 0 mm, then the actual position of the axis becomes 0 mm after homing. The coordinate value belongs to the "absolute" coordinate value, that is, the axis has established an absolute coordinate system and can be absolutely positioned.



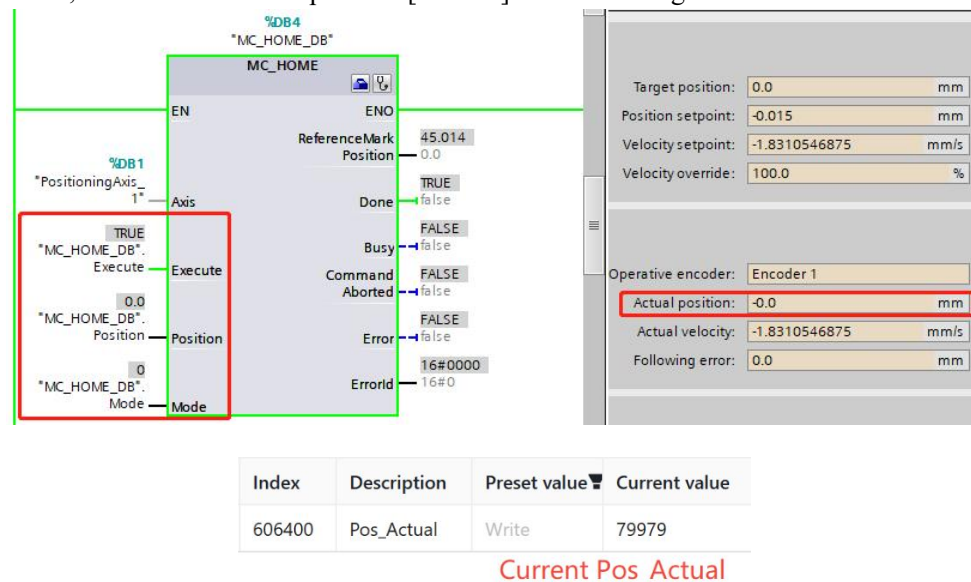
The above figure shows the state before homing, the current position of the axis is 45 mm and the drive actual position [606300] is 79979 DEC. In Section 13.4. 6.2, Article 16, the leadscrew pitch is 60 mm/rot, the conversion relation is as follows:

Actual position of PLC technology object axis (mm) = (Current servo actual position [606300] - Servo actual position [606300] at PLC power-up) / Feedback_Resolution [641003] * Leadscrew pitch

Or

Actual position change of PLC technology object axis (mm) = Servo actual position [606300] change / Feedback_Resolution [641003] * Leadscrew pitch

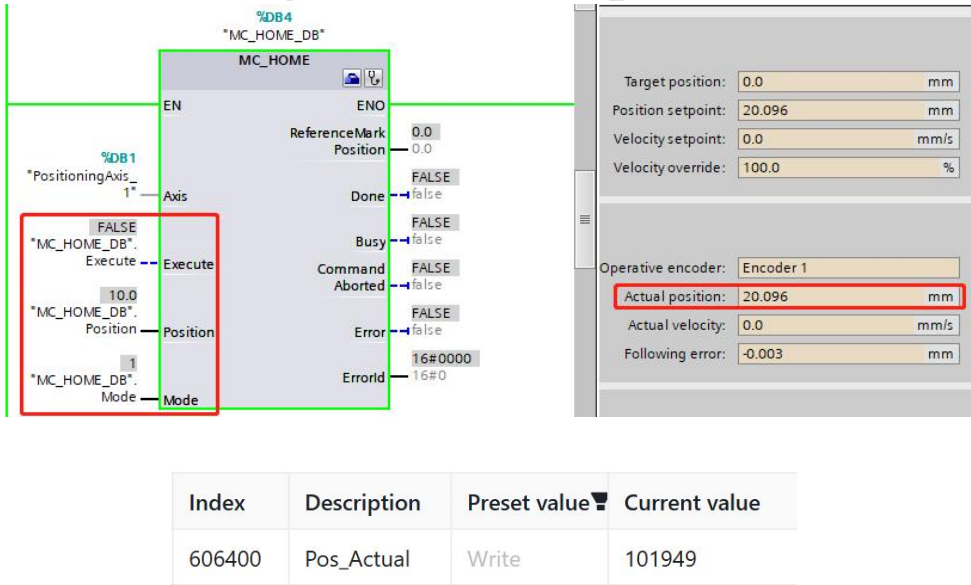
The following figure shows the state after homing, the actual position of the axis = "Position specified in MC_Home" = 0 mm, and the drive actual position [606300] does not change and remains at 79979 DEC.



After the homing motion is completed, no matter what position the axis moves to, as long as the PLC is powered off, the actual position data of the axis will be lost and become 0, instead of being converted over with the current encoder value of the drive.

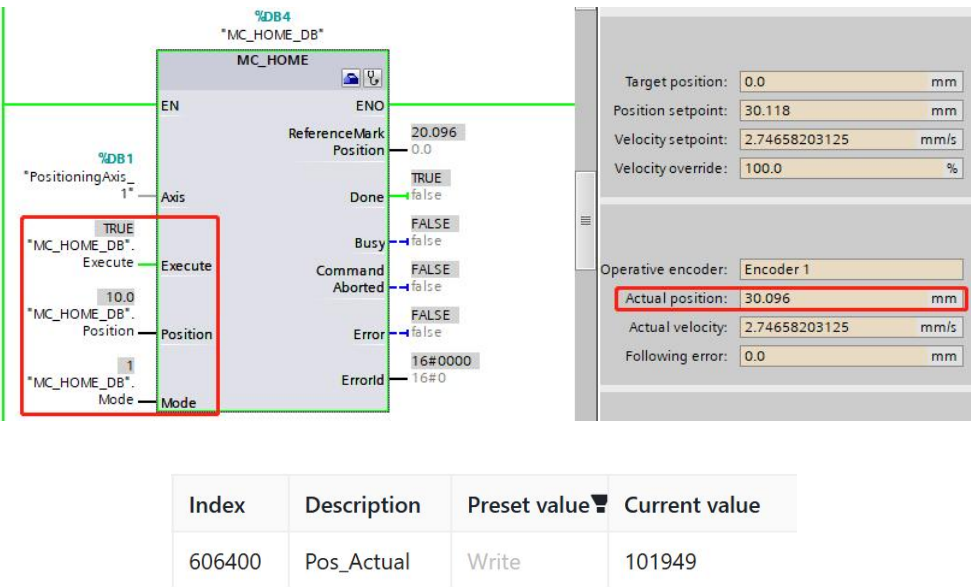
13.4.6.3.2 Homing mode 1 (Relative direct homing)

In the "Mode" = 1 , the axis will not move after triggering the MC_Home command.The result of executing the instruction is that the actual position of the axis is directly changed to "current position + Position specified in MC_Home". In the following example, "Position" = 10 mm, the current position of the axis was 20 mm before MC_Home was executed, and the actual position of the axis after MC_Home was executed became 30 mm.



The above figure shows the state before homing, the current position of the axis is 20 mm and the drive actual position [606300] is 101949 DEC.

The following figure shows the state after homing, the actual position of the axis = "current position + Position specified in MC_Home" = 30 mm, and the drive actual position [606300] does not change and remains at 101949 DEC.



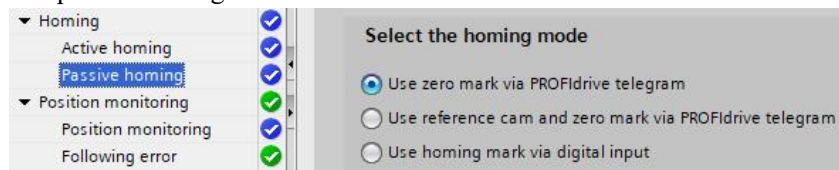
After the homing motion is completed, no matter what position the axis moves to, as long as the PLC is powered off, the actual position data of the axis will be lost and become 0, instead of being converted over with the current encoder value of the drive.

13.4.6.3.3 Homing mode 2 (Passive homing)

In the "Mode" = 2, when only MC_Home is triggered, the axis will not move, and other motion control instructions (e.g. MC_MoveRelative) are needed to make the axis run, and when the axis reaches the homing switch, the current position will be changed to "Position specified in MC_Home".

Passive homing requires the MC_Home command to be used in combination with the MC_MoveAbsolute or MC_MoveRelative or MC_MoveVelocity command, and the axis is completed to the homing in the process of executing other motion commands. After PLC power off and restart, the actual position data of the axis will be lost.

Passive homing needs to be configured in the process object configuration. As shown in the following figure, there are three modes for passive homing:



1. Use zero mark via PROFIdrive telegram (i.e. detecting zero mark)

Motion sequence:

- a. Start passive homing via the "MC_Home" instruction.
- b. Move via other motion control instruction (e.g. MC_MoveRelative). The detection of the homing mark is enabled when the actual position value of the axis moves in the assigned homing direction.
- c. When zero mark is detected, the actual position is changed to "Position specified in MC_Home".

Or

- a. Move in the assigned homing direction via other motion control instruction (e.g. MC_MoveRelative).
- b. Start passive homing via the "MC_Home" instruction.
- c. When zero mark is detected, the actual position is changed to "Position specified in MC_Home".

2. Use reference cam and zero mark via PROFIdrive telegram (i.e. detecting zero mark after digital input is detected.)

Motion sequence:

- a. Start passive homing via the "MC_Home" instruction.
- b. Move via other motion control instruction (e.g. MC_MoveRelative) and wait for the digital input.
- c. Detection of zero mark will begin as soon as the digital input is detected and disengaged.
- d. When zero mark is detected, the actual position is changed to "Position specified in MC_Home".

Or

- a. Move via other motion control instruction (e.g. MC_MoveRelative).
- b. Start passive homing via the "MC_Home" instruction and wait for the digital input.
- c. Detection of zero mark will begin as soon as the digital input is detected and disengaged.
- d. When zero mark is detected, the actual position is changed to "Position specified in MC_Home".

3. Use homing mark via digital input (i.e. detecting digital input)

Motion sequence:

- Start passive homing via the "MC_Home" instruction.
- Move via other motion control instruction (e.g. MC_MoveRelative) and wait for the digital input.
- When the digital input is detected, the actual position is changed to the "Position specified in MC_Home".

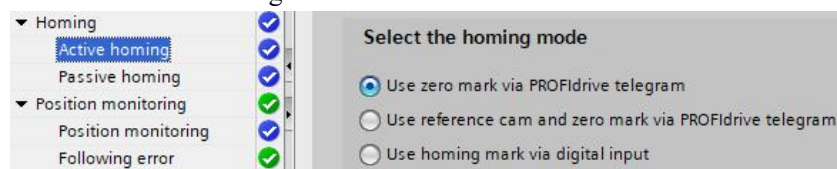
Or

- Move via other motion control instruction (e.g. MC_MoveRelative).
- Start passive homing via the "MC_Home" instruction and wait for the digital input.
- When the digital input is detected, the actual position is changed to the "Position specified in MC_Home".

13.4.6.3.4 Homing mode 3 (Active homing)

In the "Mode" = 3 mode, after triggering the MC_Home command, the axis will automatically run to complete the return to homing action.

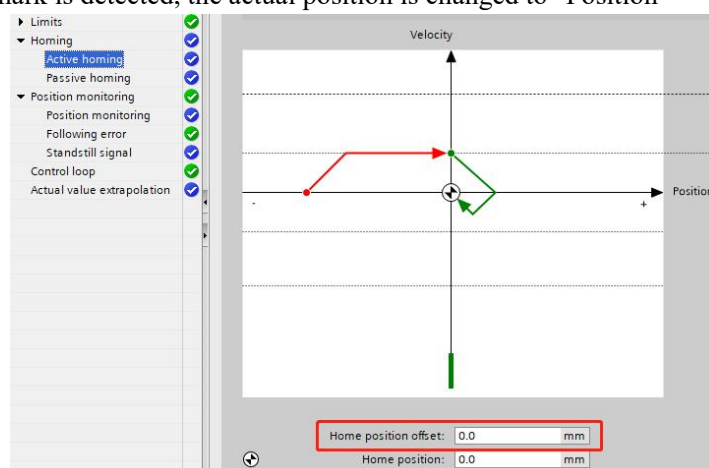
There are three modes for active homing:



- Use zero mark via PROFIdrive telegram (i.e. detecting zero mark)

Motion sequence:

- Start active homing via the "MC_Home" instruction.
- Look for the zero mark during motion.
- When zero mark is detected, the actual position is changed to "Position " - "Home position offset":



- The axis will then move a distance of the "Home position offset" and the actual position will be equal to "Position " after completion.
- Use reference cam and zero mark via PROFIdrive telegram (i.e. detecting zero mark after digital input is detected.)

Motion sequence:

- a. Start active homing via the "MC_Home" instruction.
- b. The axis detects digital input during motion.
- c. Decelerates when digital input is detected, and detect zero mark when the falling edge of digital input is detected.
- d. When zero mark is detected, the actual position is changed to "Position specified in MC_Home" - "Home position offset".
- e. The axis will then move a distance of the "Home position offset" and the actual position will be equal to "Position specified in MC_Home" after completion.

3. Use homing mark via digital input (i.e. detecting digital input)

Motion sequence:

- a. Start active homing via the "MC_Home" instruction.
- b. Detection of the rising edge at the digital input, while moving with homing velocity.
- c. Decelerates when the rising edge of digital input is detected; when the falling edge of digital input is detected, the actual position is changed to "Position" - "Home position offset".
- d. The axis will then move a distance of the "Home position offset" and the actual position will be equal to "Position" after completion.

13.4.6.3.5 Homing mode 6 (Absolute encoder adjustment- relative)

This mode is only for the connected encoder type is absolute value encoder, in this mode, after the MC_Home instruction is triggered, the axis will not run, and will not search for the home switch, it will set the current position value to "current position + Position specified in MC_Home". The absolute value is stored retentively in the CPU, and will not be lost after the CPU power off.

13.4.6.3.6 Homing mode 7 (Absolute encoder adjustment- absolute)

This mode is only for the connected encoder type is absolute value encoder, in this mode, after the MC_Home instruction is triggered, the axis will not run, and will not search for the home switch, it will set the current position value to "Position specified in MC_Home". The absolute value is stored retentively in the CPU, and will not be lost after the CPU power off.

13.4.6.4 MC_TorqueLimiting introduction

The CMD_q_Max [607300] and the Max_Torque% [607200] of the drive are associated. The maximum current (torque) can be limited using the MC_TorqueLimiting instruction.

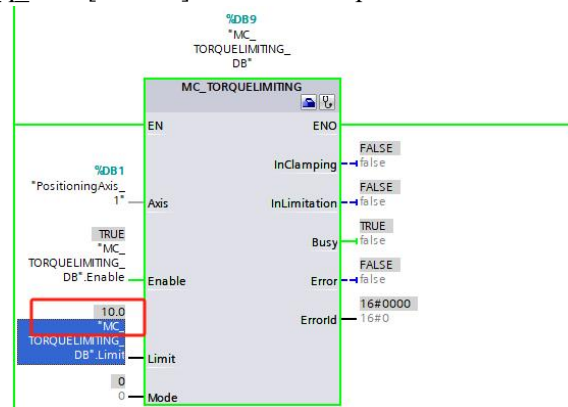
1. MC_TorqueLimiting pin definition are as follow:

Table 13-77 MC_TorqueLimiting pin definition

Pin name	Data type	Description
Axis	TO_SpeedAxis TO_PositioningAxis	Technology objects

Pin name	Data type	Description
	TO_SynchronousAxis	
Enable	Bool	= 1 Activate torque limiting function
Limit	LReal	Equal to CMD_q_Max [607300] value in Ap Range: 0 to drive / motor peak current
Mode	DInt	= 0 Torque limit mode = 1 —
InClamping	Bool	—
InLimitation	Bool	= 1 I_q [607800] reaches CMD_q_Max [607300]
Busy	Bool	= 1 Tasks in progress
Error	Bool	= 1 Error, see "ErrorID"
ErrorID	Word	Please consult the information system of TIA portal for help

2. Enable the torque limit function, "Enable" pin set to 1, in the "Limit" pin input target maximum current value 10 (unit Ap), then drive CMD_q_Max [607300] becomes 10 Ap:



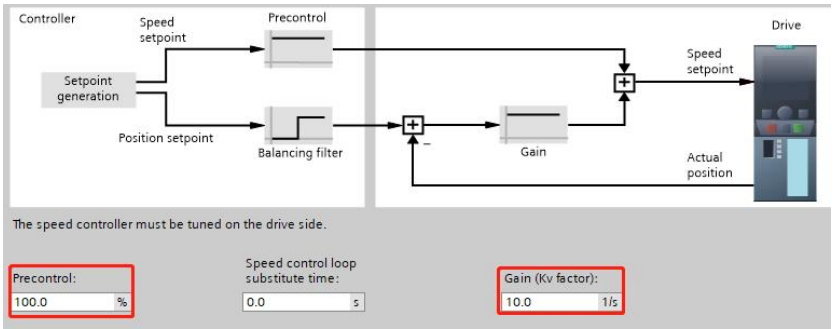
3. When I_q [607800] reaches CMD_q_Max [607300], the output pin "InLimitation" is set to 1, and the drive message status word MELDW's bit 1 is set to 0

Index	Description	Preset value	Current value
607300	CMD_q_Max		9.99

13.4.6.5 DSC (Dynamic servo control) introduction

When using telegram 3 and 102, the drive works in speed mode, the position loop calculation is completed by the PLC, and the calculation update cycle of the position loop is the same as the communication cycle of PROFINET. Telegram 5 and 105 support Dynamic Servo Control (DSC). When DSC is enabled, the drive works in interpolation mode, and the calculation and update cycle of the position loop is the local control cycle of the drive. The DSC function puts the PLC position loop calculation into the drive to complete, and the PLC only needs to perform the central interpolation calculation of the position, and then adjust the position loop calculation of the drive periodically through the PROFINET interface.

In the technology objects configuration interface, click "Extended parameters → Control loop" to set the parameters related to the DSC function, as shown in the following figure:



In the above figure, "Gain (Kv factor)" can not be 0, its specific value will not affect the servo's position loop proportional gain, keep it as the default value; "Pre-control" is the percentage of speed feedforward, and the role of K_Velocity_FF [60FB02] is the same:

Index	Description	Preset value	Current value
250809	Position_Windo...		10 DEC
606500	Max_Following_...		524288 DEC
606700	Target_Pos_Win...		327 DEC
60F400	Pos_Error	Write Prohibition	0 DEC
60FB01	Kpp[0]		10 DEC
60FB02	K_Velocity_FF		100 DEC
60FB03	K_Acc_FF		77.99 DEC
60FB05	Pos_Filter_N		1 DEC

Please note that when using the DSC function, if the type of encoder connected is a single-turn encoder (incremental encoder), it is necessary to set the Store_Position [60FB06] to 1, then write 0 to the Position_Shift [60FB07] and store the control parameters:

60FB06	Store_Position	0	D
60FB07	Pos_Shift	-15792	D



60FB06	Store_Position	1:store	
60FB07	Pos_Shift	0	

13.4.7 S7 200 SMART application

13.4.7.1 Application description

The PROFINET communication application in this chapter is based on the following conditions:

1. STEP 7-Micro/Win SMART V02.08.02.00_00.01
- PLC S7-200 SMART(CPU ST20 DC/DC/DC V02.04.01_00.00.03.00 firmware V2.6)

2. GSD file

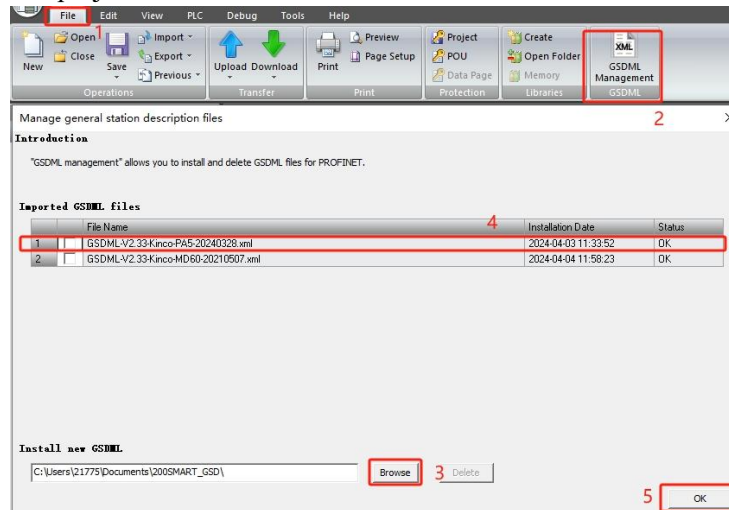
PN firmware version	Applicable GSD file
xxxx xxx5	GSDML-V2.33-Kinco-PA5-20240328

Table 13-70 describes the parameters related to the servo PROFINET communication.

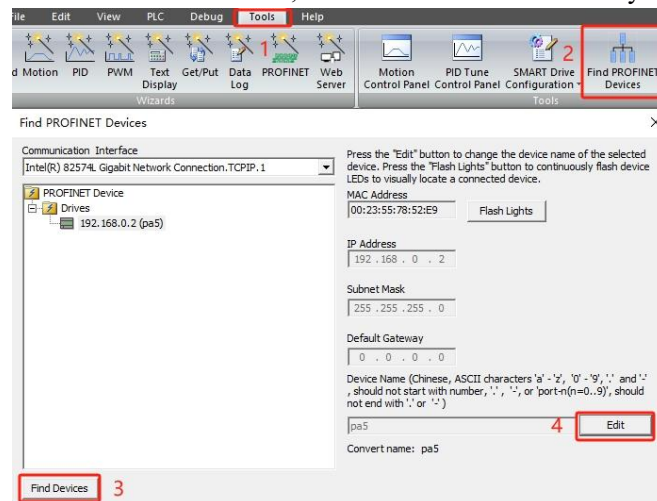
13.4.7.2 Acyclic communication

13.4.7.2.1 Project configuration

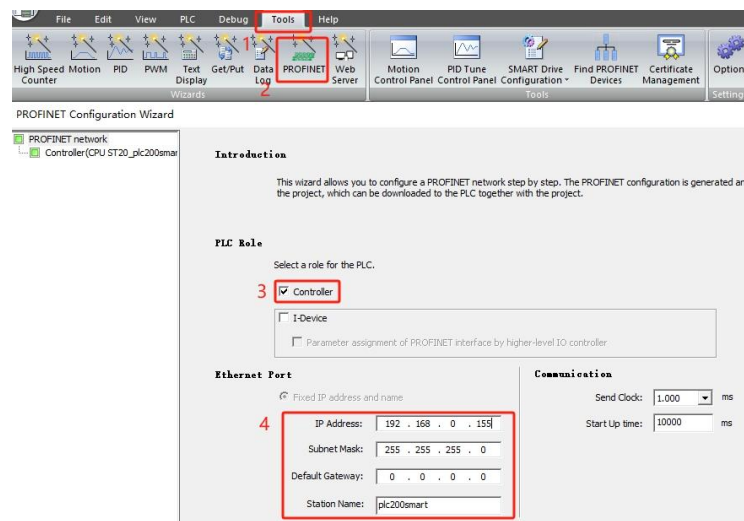
1. Click the menu "File → GSDML Management" to install the GSD file (GSDML-V2.33-Kinco-PA5-20240328). After successful installation, there is no need to repeat the installation for a new project:



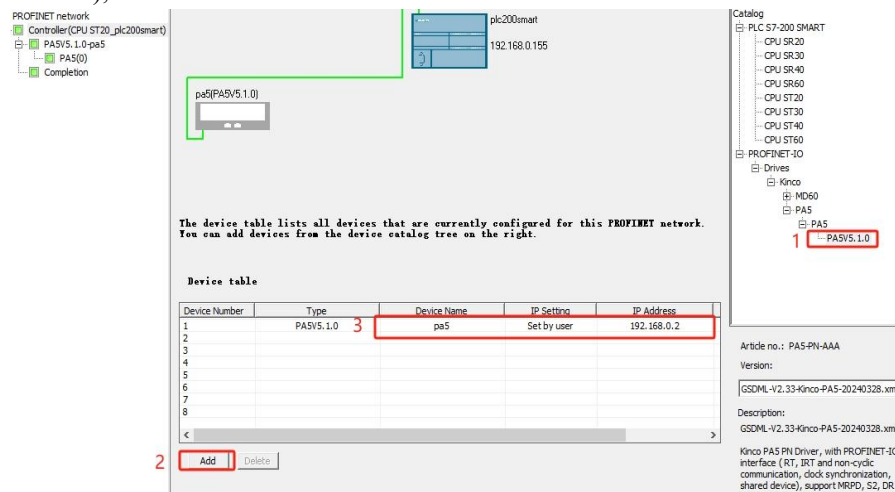
2. Click "Tools → Find PROFINET Devices" to ensure that the servo is connected. See Section 13.4.3.1 to modify the servo IP address and device name, or click "Edit" here to modify the device name:



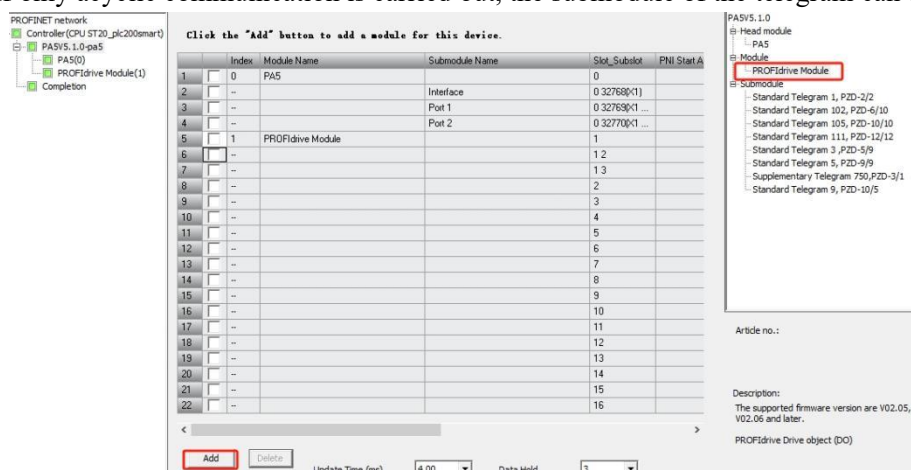
3. Click "Tools → PROFINET", select "Controller", set the PLC IP address and station name, and click "Next" :



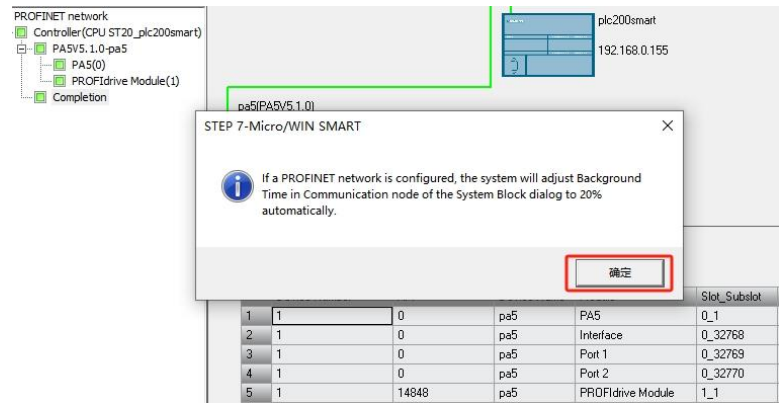
4. Add the servo to the device table, set the IP address and device name (need to be the same as the actual connected device), and then click "Next" :



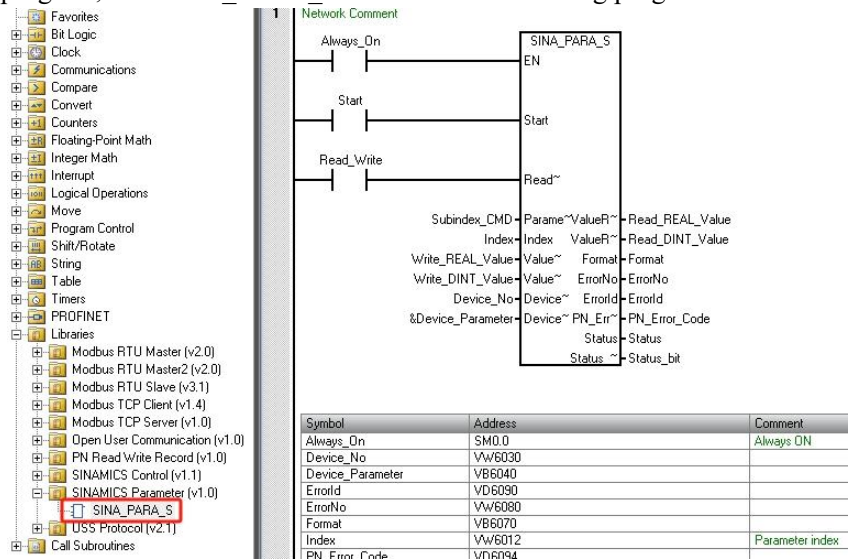
5. Add the PROFIdrive module for the servo, and the submodule of Standard Telegram 3 will be added by default. If only acyclic communication is carried out, the submodule of the telegram can be deleted:



6. Then keep clicking "Next" until it is successfully generated:



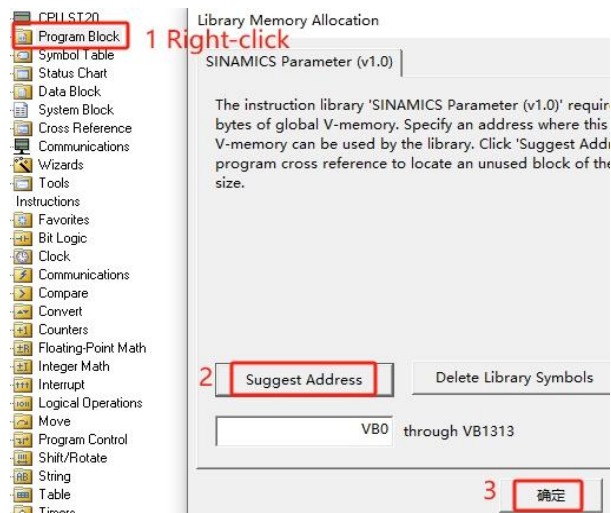
7. In the main program, call SINA_PARA_S and write the following program:



8. The symbol table address used in the program is defined as shown in the following figure:

Symbol Table			
	Symbol	Address	Comment
1	Start	V6000.0	Rising edge triggers read and write
2	Read_Write	V6000.1	0: read 1: write
3	Device_Parameter	VB6040	
4	Format	VB6070	
5	Status	VB6100	
6	Status_bit	VB6102	
7	Write_REAL_Value	VD6020	
8	Write_DINT_Value	VD6024	Data written to the servo
9	APINumber	VD6042	
10	Read_REAL_Value	VD6060	
11	Read_DINT_Value	VD6064	Data read from the servo
12	ErrorId	VD6090	
13	PN_Error_Code	VD6094	
14	Subindex_CMD	Vw6010	Parameter subindex + Command word
15	Index	Vw6012	Parameter index
16	Device_No	Vw6030	
17	SlotNumber	Vw6046	
18	SubSlotNumber	Vw6048	
19	ErrorNo	Vw6080	

9. Before downloading the program, assign the V-address area used by the library:



13.4.7.2.2 SINA_PARA_S introduction

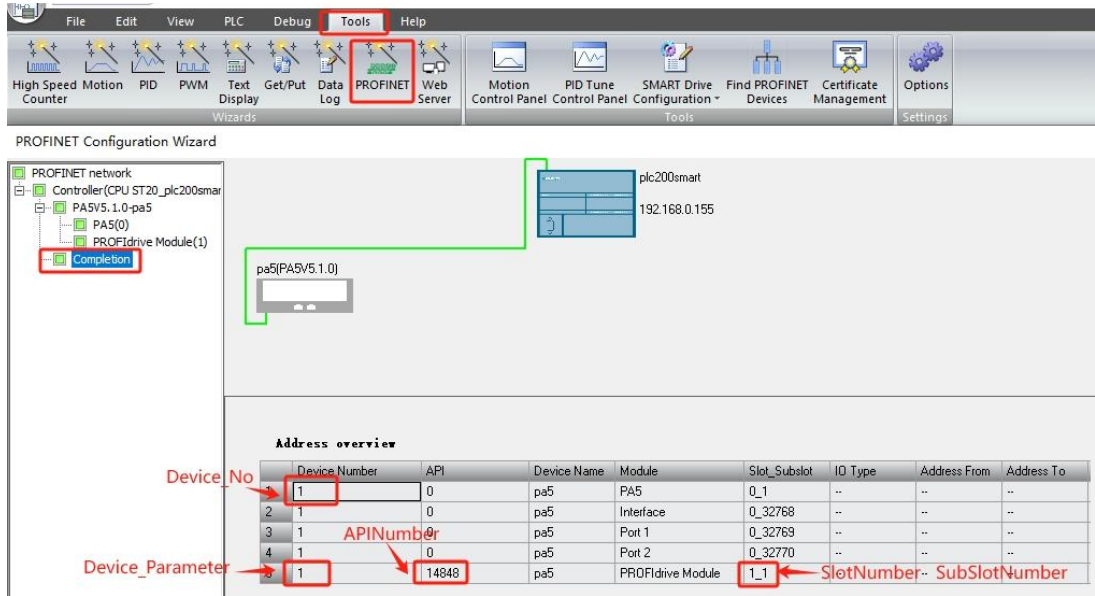
SINA_PARA_S pin definition are as follow:

Table 13-78 SINA_PARA_S pin definition

Pin name	Data type	Description
Start	Bool	The rising edge triggers data transmission
ReadWrite	Bool	0=Read; 1=Write
Parameter	Int	Bit8~15: Parameter subindex; Bit0~7: Command word
Index	Int	Parameter index
ValueWrite1	Real	Reserved
ValueWrite2	DInt	Data writing area, can write 1 to 4 bytes
DeviceNo	Word	Device number
Device_Parameter	DWord	PROFINET device parameters, byte offset as follows: 0: Axis number 1: Reserved 2~5: API number(Fixed to 14848) 6~7: Slot number 8~9: Subslot number
ValueRead1	Real	Reserved
ValueRead2	DInt	Data reading area
Format	Byte	Parameter format: 02H: Int 8 03H: Int 16 04H: Int 32 05H: Unsigned 8 06H: Unsigned 16 07H: Unsigned 32 08H: Float 10H: Octal string (16bit) 13H: Time diff (32bit) 41H: Byte 42H: Word 43H: Double word 44H: Error
ErrorNo	Word	Error number according to the PROFIdrive specification
ErrorID	DWord	First word: Binary encoding indicating a faulty parameter access Second word: Type of fault
PN_Error_Code	DInt	Error code according to the PROFINET protocol

Status	Byte	Bit0~4: System definition error codes for instructions RDREC and WRREC; Bit5: Error; Bit6: Request in progress
Status_bit	Byte	Indicates the status of the reading/writing parameter, which is 4 when the reading/writing is successful
Note: Command word—Read: 40H; Write 1 byte: 2FH; Write 2 bytes: 2BH; Write 4 bytes: 23H.		

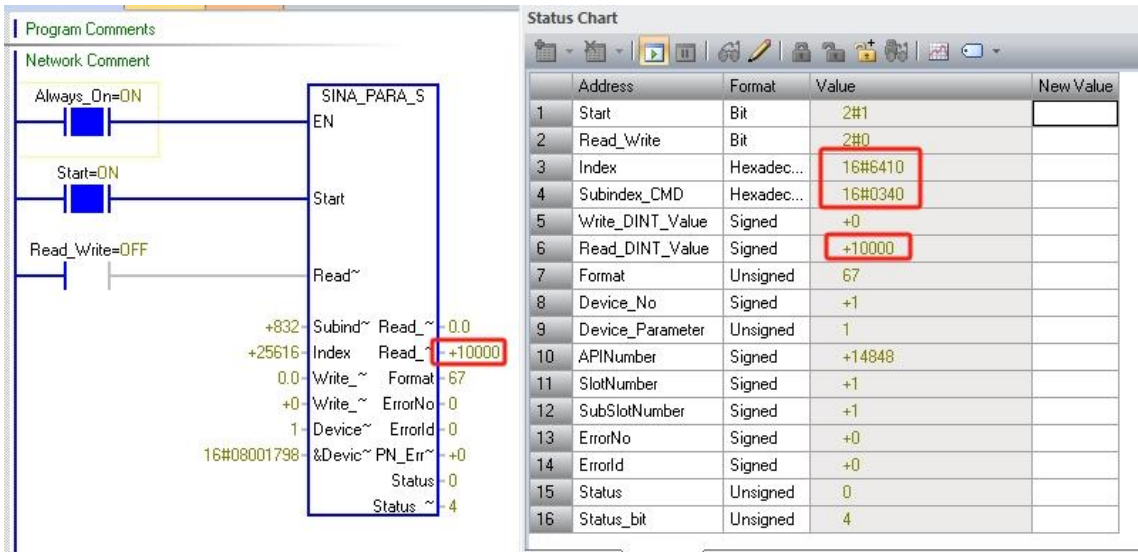
The Device_Parameter can be confirmed in the following figure:



13.4.7.2.3 Read -write parameter using SINA_PARA_S

- Read feedback_resolution: the parameter index is 0x6410, the parameter subindex is 0x03, and the read parameter command word is unified as 0x40. When the "Start" pin changes from 0 to 1, the "ValueRead2" pin in the data reading area automatically becomes 10000, and the "Status_bit" pin is 4, the data reading is successful.

Index	Description	Preset	Current value
641003	Feedback_Resolution		10000



- Write target_position: the parameter index is 0x607A, the parameter subindex is 0x00, and the data type is Int32, so the write parameter command word is 0x23, the "ReadWrite" pin is switched to 1. The data writing area is written to -10000. When the "Start" pin is changed from 0 to 1, the data is written. The "Status_bit" pin is 4, and the data is written successfully. Observing the KincoServo PRO software, the target position is changed to -10000.

The screenshot displays the KincoServo PRO software interface. On the left, a ladder logic program is shown with a network comment 'Always_On=ON' and a 'Read_Write=ON' condition connected to the 'Read' pin of the SINA_PARA_S block. The status chart on the right shows the following data:

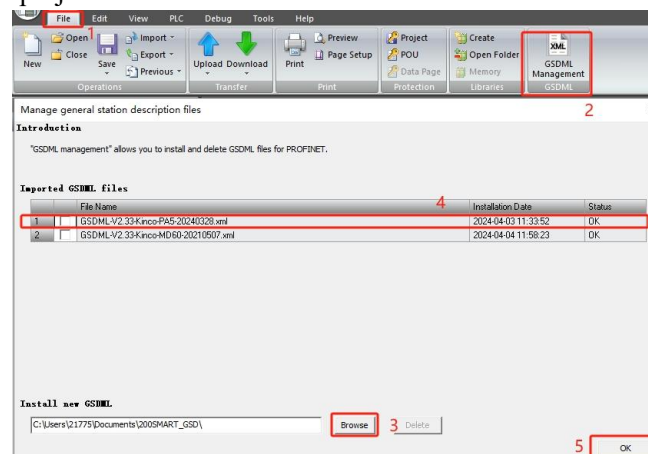
Address	Format	Value	New Value
1	Start	Bit	2#1
2	Read_Write	Bit	2#1
3	Index	Hexadec...	16#607A
4	Subindex_CMD	Hexadec...	16#0023
5	Write_DINT_Value	Signed	-10000
6	Read_DINT_Value	Signed	+0
7	Format	Unsigned	67
8	Device_No	Signed	+1
9	Device_Parameter	Unsigned	1
10	APINumber	Signed	+14848
11	SlotNumber	Signed	+1
12	SubSlotNumber	Signed	+1
13	ErrorNo	Signed	+0
14	ErrorId	Signed	+0
15	Status	Unsigned	0
16	Status_bit	Unsigned	4

Index	Description	Preset value ▼	Current value
607A00	Target_Position		-10000

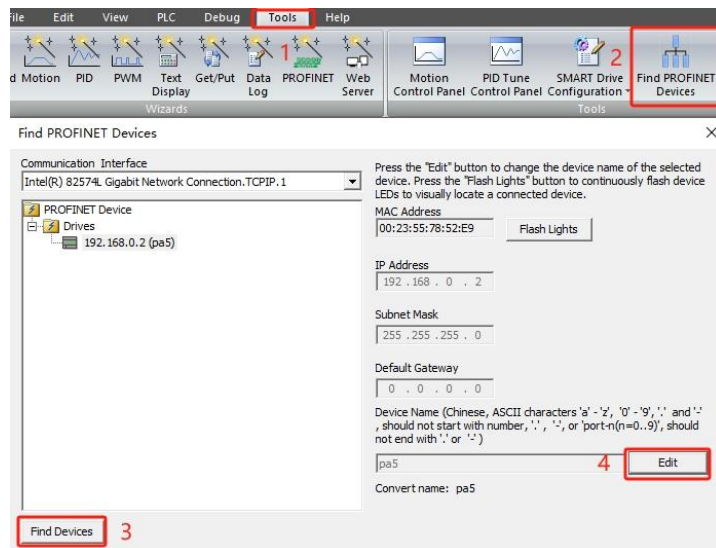
13.4.7.3 Telegram 1 application

13.4.7.3.1 Project configuration

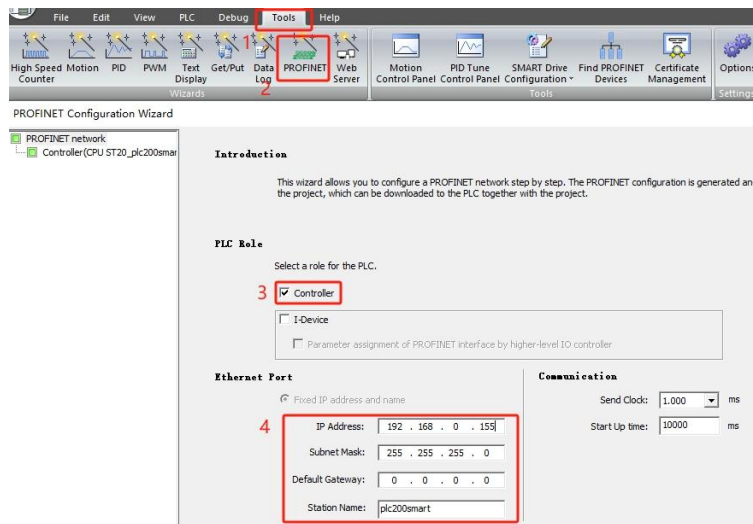
- Click the menu "File → GSDML Management" to install the GSD file (GSDML-V2.33-Kinco-PA5-20240328). After successful installation, there is no need to repeat the installation for a new project:



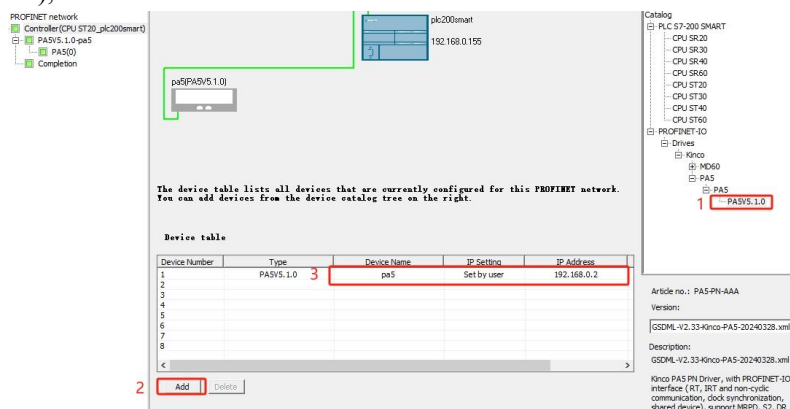
- Click "Tools → Find PROFINET Devices" to ensure that the servo is connected. See Section 13.4.3.1 to modify the servo IP address and device name, or click "Edit" here to modify the device name:



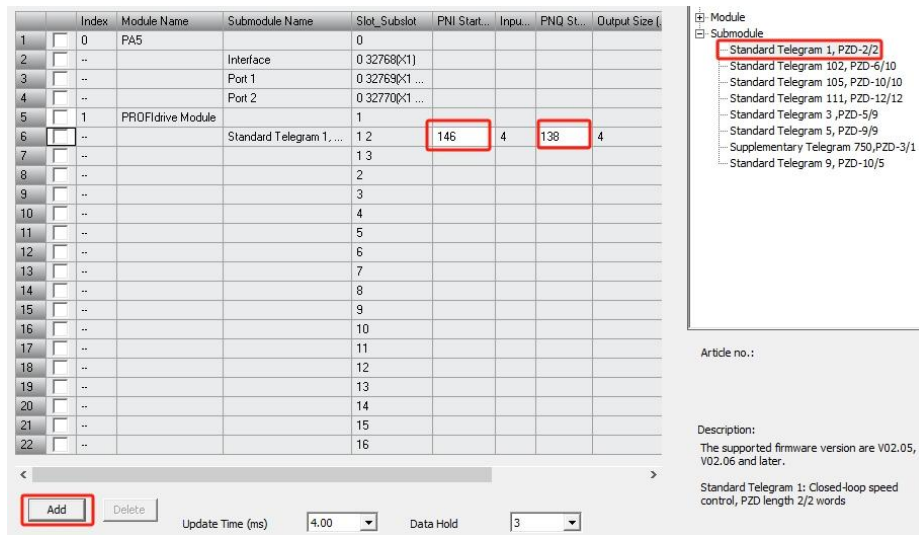
3. Click "Tools → PROFINET", select "Controller", set the PLC IP address and station name, and click "Next" :



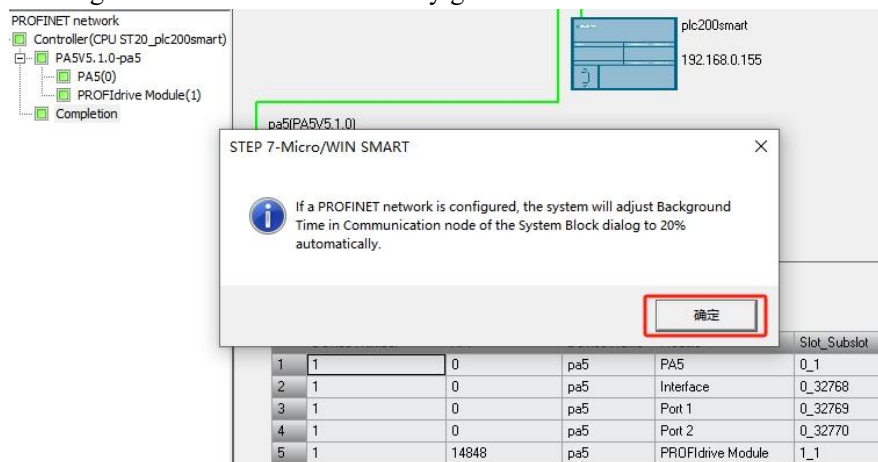
4. Add the servo to the device table, set the IP address and device name (need to be the same as the actual connected device), and then click "Next" :



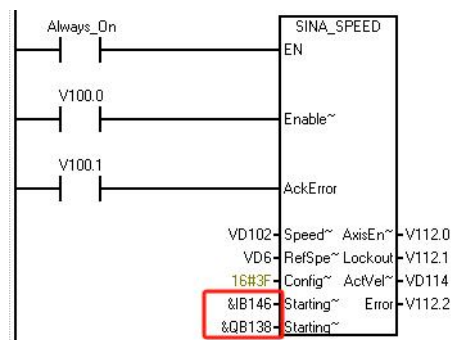
5. Add the PROFIdrive module to the servo, then select and delete the submodule of standard telegram 3 added by default, add the submodule of standard telegram 1, and note down the start address of the I/O data which is needed when calling SINA_SPEED:



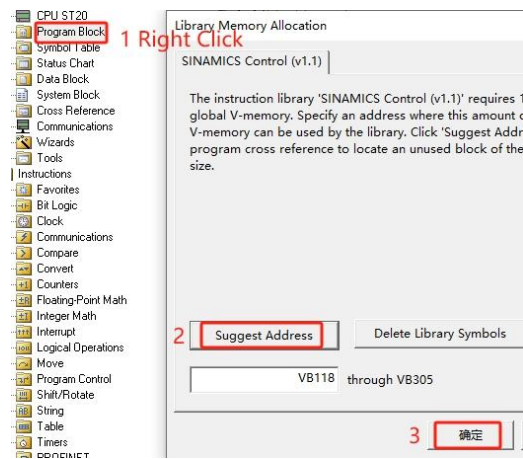
6. Then keep clicking "Next" until it is successfully generated:



7. In the main program, call SINA_SPEED and write the following program, where the inputs "Starting_I_add" and "Starting_Q_add" must correspond to the start address of the I/O data of telegram 1 (see step 5) :



8. Before downloading the program, assign the V-address area used by the library:



13.4.7.3.2 SINA_SPEED introduction

SINA_SPEED pin definition are as follow:

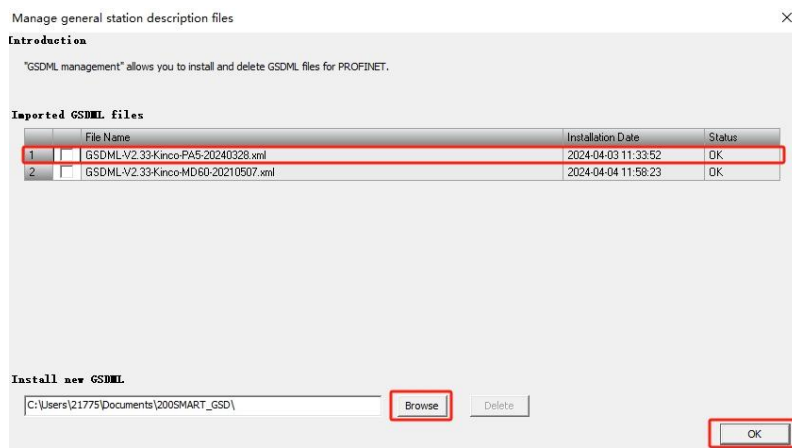
Table 13-79 SINA_SPEED pin definition

Pin name	Data type	Description																
EnableAxis	Bool	=1 Enable																
AckError	Bool	Reset fault, rising edge valid																
SpeedSp	Real	Speed setpoint in RPM																
RefSpeed	Real	Reference speed, must be 1/2 of the maximum speed [607F00] in RPM																
ConfigAxis	Word	The default is 16#3F, and the corresponding relationship with STW1 is as follows:<table><tr><th>Bit in ConfigAxis</th><th>Bit in STW1</th></tr><tr><td>Bit 0</td><td>Bit 1</td></tr><tr><td>Bit 1</td><td>Bit 2</td></tr><tr><td>Bit 2</td><td>Bit 3</td></tr><tr><td>Bit 3</td><td>Bit 4</td></tr><tr><td>Bit 4</td><td>Bit 5</td></tr><tr><td>Bit 5</td><td>Bit 6</td></tr><tr><td>Bit 6</td><td>Bit 11</td></tr></table>See Table 13-56 for the meaning of each bit in STW1	Bit in ConfigAxis	Bit in STW1	Bit 0	Bit 1	Bit 1	Bit 2	Bit 2	Bit 3	Bit 3	Bit 4	Bit 4	Bit 5	Bit 5	Bit 6	Bit 6	Bit 11
Bit in ConfigAxis	Bit in STW1																	
Bit 0	Bit 1																	
Bit 1	Bit 2																	
Bit 2	Bit 3																	
Bit 3	Bit 4																	
Bit 4	Bit 5																	
Bit 5	Bit 6																	
Bit 6	Bit 11																	
Starting_I_add	DWord	Start address of the PN I area of telegram 1																
Starting_Q_add	DWord	Start address of the PN Q area of telegram 1																
AxisEnabled	Bool	=1 Axis is enabled																
Lockout	Bool	=1 Switching on inhibited																
ActVelocity	Real	Speed actual value in RPM																
Error	Bool	=1 Fault present																

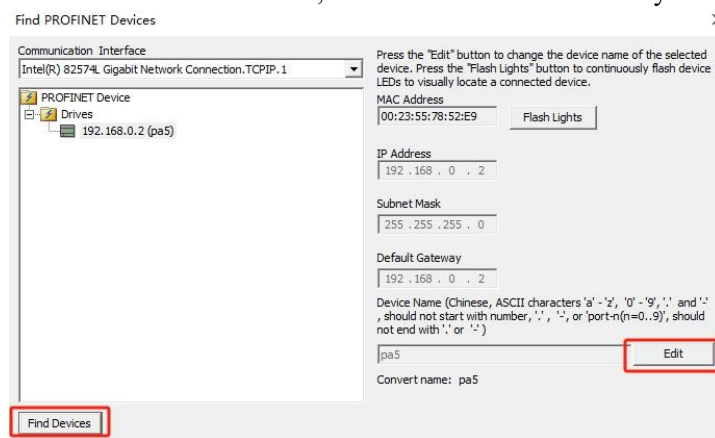
13.4.7.4 Telegram 111 application

13.4.7.4.1 Project configuration

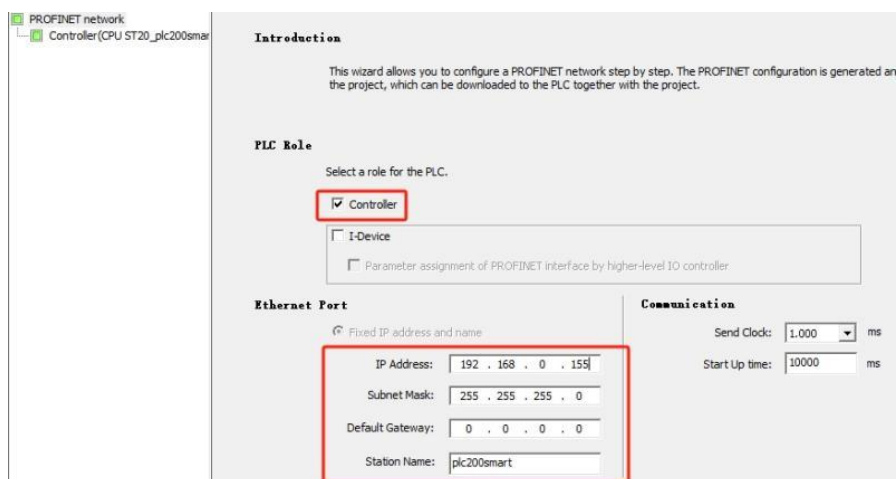
1. Click the menu "File → GSDML Management" to install the GSD file (GSDML-V2.33-Kinco-PA5-20240328). After successful installation, there is no need to repeat the installation for a new project:



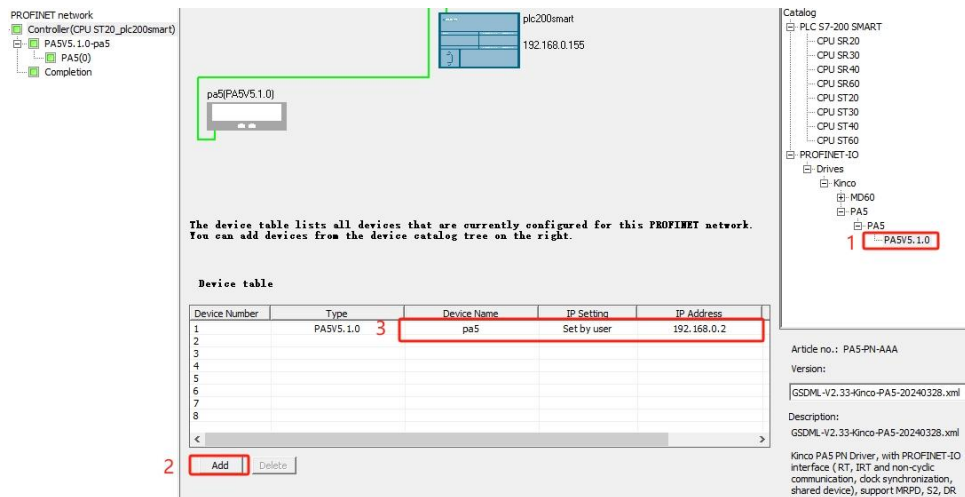
- Click "Tools → Find PROFINET Devices" to ensure that the servo is connected. See Section 13.4.3.1 to modify the servo IP address and device name, or click "Edit" here to modify the device name:



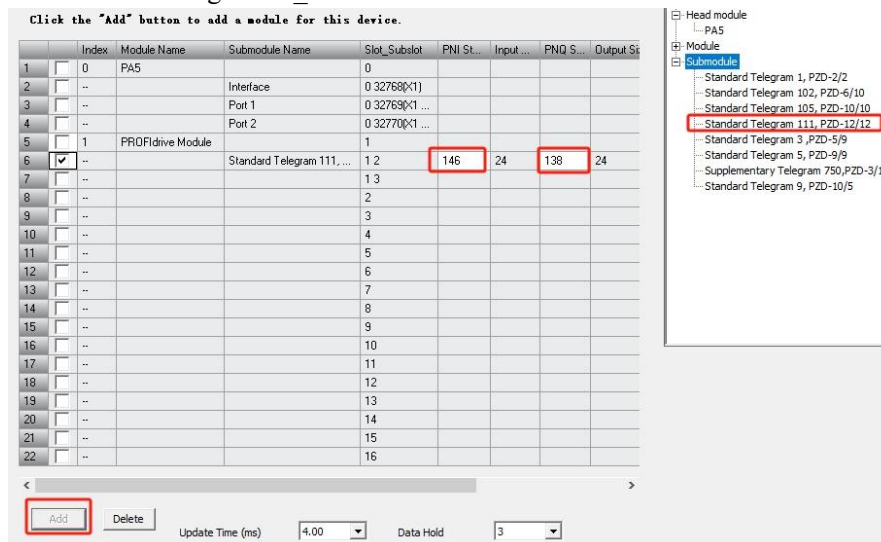
- Click "Tools → PROFINET", select "Controller", set the PLC IP address and station name, and click "Next" :



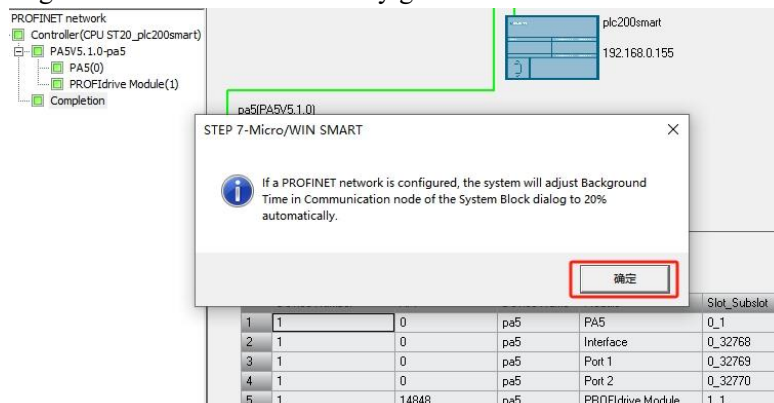
- Add the servo to the device table, set the IP address and device name (need to be the same as the actual connected device), and then click "Next" :



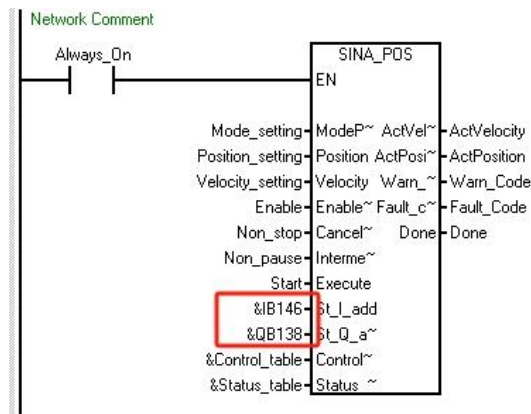
5. Add the PROFIdrive module to the servo, then select and delete the submodule of standard telegram 3 added by default, add the submodule of standard telegram 111, and note down the start address of the I/O data which is needed when calling SINA_POS:



6. Then keep clicking "Next" until it is successfully generated:



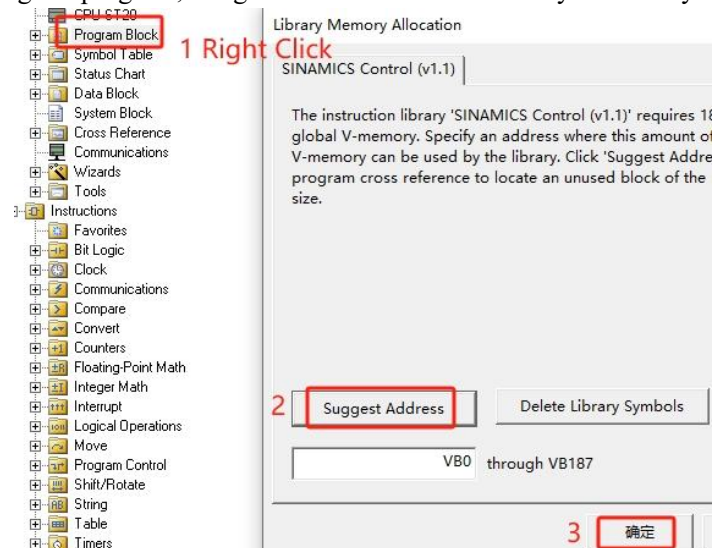
7. In the main program, call SINA_POS and write the following program, where the inputs " St_I_add " and "St_Q_add" must correspond to the start address of the I/O data of telegram 111 (see step 5) :



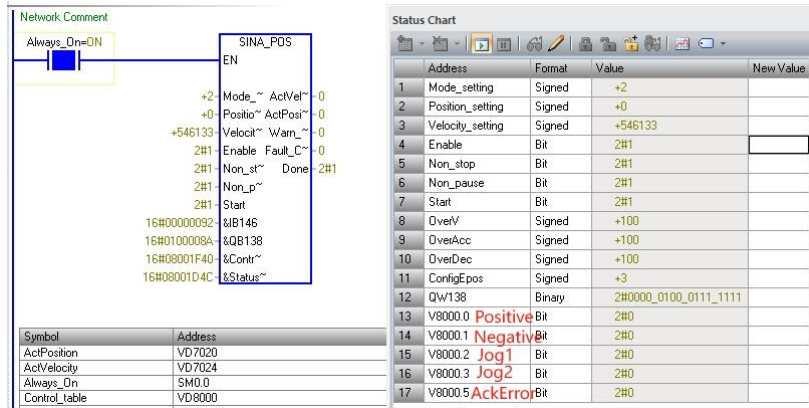
8. The symbol table address used in the program is defined as shown in the following table:

Mode_setting	VW7000
Position_setting	VD7002
Velocity_setting	VD7006
Enable	V7010.0
Non_stop	V7010.1
Non_pause	V7010.2
Start	V7010.3
Control_table	VD8000
Status_table	VD7500
ActPosition	VD7020
ActVelocity	VD7024
Warn_Code	VW7028
Fault_Code	VW7030
Done	V7032.0
OverV	VW8002
OverAcc	VW8004
OverDec	VW8006
ConfigEpos	VD8008

9. Before downloading the program, assign the V-address area used by the library:



10. After downloading the program, you can test it with a status chart:



13.4.7.4.2 SINA_POS introduction

SINA_POS pin definition are as follow:

Table 13-80 SINA_POS pin definition

Pin name	Data type	Description																																																																																				
ModePos	Int	Operating mode: =0 — =1 Relative positioning =2 Absolute positioning =3 — =4 Active homing =5 Direct homing =6 Position table (not supported) =7 Jog at the specified speed =8 —																																																																																				
Position	DInt	Position setpoint for operating mode 1, 2 Start/new task index of the position table when the operation mode is 6, ranging from 0 to 7																																																																																				
Velocity	DInt	Speed setpoint for operating mode 1, 2, 7																																																																																				
EnableAxis	Bool	=0 Disable, servo control word is 0xE =1 Enable, servo control word is 0xF																																																																																				
CancelTraversing	Bool	=1 Do not reject traversing task =0 Reject traversing task																																																																																				
IntermediateStop	Bool	=1 No intermediate stop =0 Intermediate stop																																																																																				
Execute	Bool	Activate operating mode, rising edge valid																																																																																				
St_I_add	DWord	Start address of the PN I area of telegram 111																																																																																				
St_Q_add	DWord	Start address of the PN Q area of telegram 111																																																																																				
Control_table	DWord	A pointer to the start address of Control table, for example &VD8000<table><tr><th>Byte</th><th>Bit7</th><th>Bit 6</th><th>Bit 5</th><th>Bit 4</th><th>Bit 3</th><th>Bit 2</th><th>Bit 1</th><th>Bit0</th></tr><tr><td>0</td><td>Reserved</td><td>Reserved</td><td>Ack-Error</td><td>Reserved</td><td>Jog2</td><td>Jog1</td><td>Negative</td><td>Positive</td></tr><tr><td>1</td><td colspan="8">Reserved</td></tr><tr><td>2</td><td colspan="8" rowspan="2">OverV: Percentage of Speed Setpoint at Operating Modes 1, 2, 7, unit %, range 0 to 199%</td></tr><tr><td>3</td></tr><tr><td>4</td><td colspan="8" rowspan="2">OverAcc: Acceleration percentage when operating mode is 1, 2 and 7, unit %, range 0~100%</td></tr><tr><td>5</td></tr><tr><td>6</td><td colspan="8" rowspan="2">OverDec: Deceleration percentage when operating mode is 1, 2 and 7, unit %, range 0~100%</td></tr><tr><td>7</td></tr><tr><td>8</td><td colspan="8">ConfigEpos —</td></tr><tr><td>9</td><td colspan="4">Bit0: Coast stop</td><td colspan="4">Bit1: Quick stop</td></tr><tr><td></td><td colspan="4">Bit2: Activate software limit</td><td colspan="4">Bit3: Activate hardware limit</td></tr></table>	Byte	Bit7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit0	0	Reserved	Reserved	Ack-Error	Reserved	Jog2	Jog1	Negative	Positive	1	Reserved								2	OverV: Percentage of Speed Setpoint at Operating Modes 1, 2, 7, unit %, range 0 to 199%								3	4	OverAcc: Acceleration percentage when operating mode is 1, 2 and 7, unit %, range 0~100%								5	6	OverDec: Deceleration percentage when operating mode is 1, 2 and 7, unit %, range 0~100%								7	8	ConfigEpos —								9	Bit0: Coast stop				Bit1: Quick stop					Bit2: Activate software limit				Bit3: Activate hardware limit			
Byte	Bit7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit0																																																																														
0	Reserved	Reserved	Ack-Error	Reserved	Jog2	Jog1	Negative	Positive																																																																														
1	Reserved																																																																																					
2	OverV: Percentage of Speed Setpoint at Operating Modes 1, 2, 7, unit %, range 0 to 199%																																																																																					
3																																																																																						
4	OverAcc: Acceleration percentage when operating mode is 1, 2 and 7, unit %, range 0~100%																																																																																					
5																																																																																						
6	OverDec: Deceleration percentage when operating mode is 1, 2 and 7, unit %, range 0~100%																																																																																					
7																																																																																						
8	ConfigEpos —																																																																																					
9	Bit0: Coast stop				Bit1: Quick stop																																																																																	
	Bit2: Activate software limit				Bit3: Activate hardware limit																																																																																	

Pin name	Data type	Description								
		10	Bit6: Reference point switch Bit8: The absolute positioning set point is changed immediately							
		11	Note that the initial value of the ConfigEpos is 3.							
Status_table	DWord	A pointer to the start address of Status_table, for example &VD7500								
		Byte	Bit7	Bit6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		0	Reserved	Over-range	Axis-Error	Axis-Warn	Lock-out	Axis-Ref	Axis-PosOk	Axis-Enabled
		1	Error ID—— 0: No error 1: Drive fault 2: Drive disabled 3: Unsupported mode 4: OverV, OverAcc and OverDec is overranged							
		2	ActMode: Current operating mode							
		3								
		4	POS_ZSW1 current value							
		5								
		6	POS_ZSW2 current value							
		7								
ActVelocity	DWord	Speed actual value								
ActPosition	DWord	Position actual value								
Warn_code	Word	Servo warn code								
Fault_code	Word	Servo error code [603F00] absolute positioning.								
Done	Bool	The target position is reached when operation mode is relative or absolute positioning.								

13.4.7.4.3 Operation condition and setting

- Enabled by entering pin "EnableAxis" = 1, "AxisEnabled" in "Status_table" is 1 if the servo is ready and fault-free ("AxisError" = 0).
- The input pin "CancelTraversing" and "IntermediateStop" are valid for all modes except jog mode, and should be set to 1 during operation, as follows:
 - When "CancelTraversing" = 0, it means that the current task is canceled, and the deceleration corresponds to the halt_mode [605D00]. The operating mode can be switched after the axis stops.
 - When "IntermediateStop" = 0, it means to pause the current task, and the deceleration corresponds to the halt_mode [605D00]. The axis will continue to run after resetting "IntermediateStop" = 1. The operating mode can be switched after the axis stops.
- The input pin "ModePos" is used to select the operating mode; the rising edge of "Execute" triggers the positioning movement.
- Activate software limit switch

After setting the reference point, if you need to use a software limit switch, you need to set bit 2 of the input pin "ConfigEPos" to 1 ("ConfigEPos" = 16#00000007) or set the En_SoftWare_Limit [30800C] to 1.

Activate the software limit function, set the soft positive limit [607D01] and the soft negative limit [607D02] for the servo.

30800C	En_SoftWare_Limit	1
607D01	Soft_Positive_Limit	10000000
607D02	Soft_Negative_Limit	-10000000

After you set soft positive limit more than soft negative limit and set the reference point, the software limit function is enabled.

5. Activate hardware limit switch

If you need to use a hardware limit switch, you need to set bit 3 of the input pin "ConfigEPos" to 1 ("ConfigEPos" = 16#0000000B):

Use KincoServo PRO software to define the positive limit and negative limit for the DIN of the servo.

The servo can only be operated when the hardware limit switch signal is high.

PIN number	Function	Actual input	Simulation input	Polarity	Valid input
DIN1	Homing Signal			High	
DIN2	Position limit +			High	
DIN3	Position limit -			High	

Please note that only positive and negative limits are configured in the KincoServo PRO software, but bit 3 of "ConfigEPos" is not set to 1, then the hardware limit function will not take effect.

13.4.7.4.4 Operation mode 1(Relative position)

Requirements:

1. Operating mode "ModePos" = 1
2. "ConfigEPos" = 16#00000003
3. "EnableAxis" = 1
4. "CancelTraversing" and "IntermediateStop" must be 1, and "Jog1" and "Jog2" must be 0.

Steps:

1. Specify the target position and target speed through the input pin "Position" and "Velocity". The unit of position and velocity is DEC. Refer to the servo drive manual for conversion, and the unit of velocity is transformed into: $DEC = RPM * 512 * feedback_resolution [641003] / 1875$.
2. Use "OverV", "OverAcc" and "OverDec" to scale the percentage of the target speed and profile acceleration and deceleration.
3. The direction of movement is determined by the positive or negative value set in "Position".
4. The movement is triggered by the rising edge of "Execute", and "AxisPosOk" is set to 1 after reaching the target position. If there is an error locating, "AxisError" is set to 1.
5. The currently running command can be replaced with a new command via the "Execute" rising edge. At any time, it can switch between operating mode 1 and 2. If you need to switch to another operating mode, the axis must be stationary.
6. Please note that "Execute" needs to be reset to 0 after the relative positioning motion is triggered by the rising edge of "Execute".

13.4.7.4.5 Operation mode 2(Absolute position)

Requirements:

1. Operating mode "ModePos" = 2
2. "ConfigEPos" = 16#00000003
3. "EnableAxis" = 1
4. "CancelTraversing" and "IntermediateStop" must be 1, and "Jog1" and "Jog2" must be 0.

Steps:

1. Specify the target position and target speed through the input pin "Position" and "Velocity". The unit of position and velocity is DEC. Refer to the servo drive manual for conversion, and the unit of velocity is transformed into: $DEC = RPM * 512 * feedback_resolution [641003] / 1875$.
2. Use "OverV", "OverAcc" and "OverDec" to scale the percentage of the target speed and profile acceleration and deceleration.
3. The servo will follow the shortest path to the target position when running. "Positive" and "Negative" must be 0.
4. The movement is triggered by the rising edge of "Execute", and "AxisPosOk" is set to 1 after reaching the target position. If there is an error locating, "AxisError" is set to 1.
5. The currently running command can be replaced with a new command via the "Execute" rising edge. At any time, it can switch between operating mode 1 and 2. If you need to switch to another operating mode, the servo must be stationary.
6. By setting bit 8 of "ConfigEPos" ("ConfigEPos" = 16#00000103), after updating "Position" on the PLC, there is no need to trigger "Execute", the new setting value will take effect immediately, and the servo will immediately execute the absolute positioning instruction according to the target position change.
7. Please note that "Execute" needs to be reset to 0 after the absolute positioning motion is triggered by the rising edge of "Execute".

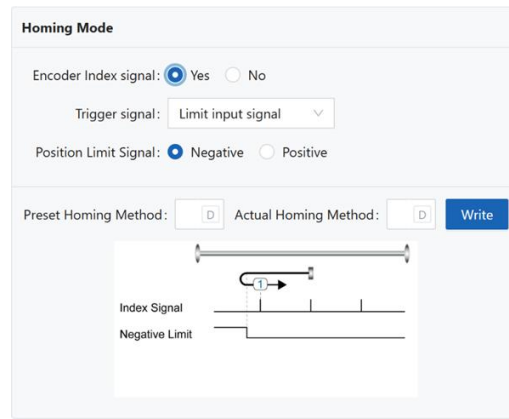
13.4.7.4.6 Operation mode 4(Active homing)

Requirements:

1. Operating mode "ModePos" = 4
2. "ConfigEPos" = 16#00000003. If hardware limits are used, "ConfigEPos" = 16#0000000B.
3. "EnableAxis" = 1
4. "CancelTraversing" and "IntermediateStop" must be 1, and "Jog1" and "Jog2" must be 0.
5. One of "Negative" and "Positive" must be set to 1.

Steps:

1. The way and the speed of returning to the reference point should be configured by KincoServo PRO software (PLC only sends the command, and the planning of returning to the reference point is set by the servo parameters).



2. A return to the reference point motion is triggered by the rising edge of "Execute", which should remain 1 during the motion. After the motion, "AxisRef" is set to 1, and if there is an error during the operation, "AxisError" is set to 1.

13.4.7.4.7 Operation mode 5(Direct homing)

Requirements:

1. Operating mode "ModePos" = 5
2. "ConfigEPos" = 16#00000003
3. "EnableAxis" = 1
4. The axis must be stationary.

Steps:

The reference point of the axis is set by the rising edge of the "Execute". When the homing operation has been done, the output pin "AxisRef" turns to be 1.

13.4.7.4.8 Operation mode 7 (Jog at the specified speed)

Requirements:

1. Operating mode "ModePos" = 7
2. "ConfigEPos" = 16#00000003
3. "EnableAxis" = 1
4. The axis must be stationary.

Steps:

1. Specify the jog speed through the input pin "Velocity", which must be a positive value in unit DEC.
2. Use "OverV", "OverAcc" and "OverDec" to scale the percentage of the jog speed and profile acceleration and deceleration.
3. "CancelTraversing" and "IntermediateStop" are not related to the jog mode. The running direction is independent of "Positive" and "Negative".
4. There is no need to trigger "Execute". Reverse jogging when "Jog1" = 1, forward jogging when "Jog2" = 1. "Jog1" and "Jog2" can only be triggered separately. When "Jog1" and "Jog2" are 0 or 1 at the same time, the axis stops.

13.4.8 Additional telegram 750 application

13.4.8.1 Application description

The PROFINET communication application in this chapter is based on the following conditions:

1. TIA V17 PLC S7-1500(CPU 1511T-1 PN 6ES7 511-1TK01-0AB0 firmware V2.8)
2. GSD file

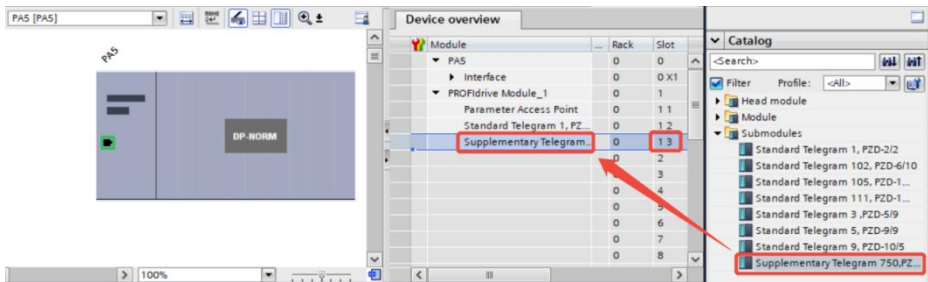
PN firmware version	Applicable GSD file
xxxx nxx5 ^a	GSDML-V2.43-Kinco-PA5-20240328 ^b
^a n is not zero	
^b If PLC does not support V2.43 GSD, use GSDML-V2.33-Kinco-PA5-20240328	

Table 13-70 describes the parameters related to the servo PROFINET communication.

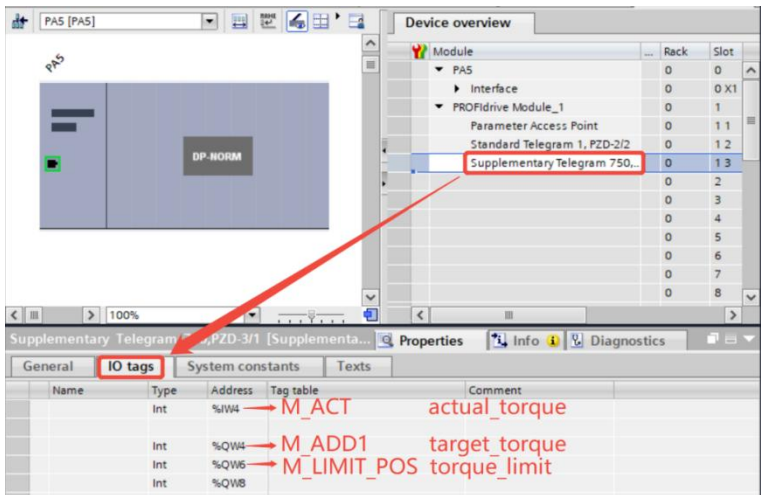
13.4.8.2 Telegram 1 + additional telegram 750

13.4.8.2.1 Project configuration

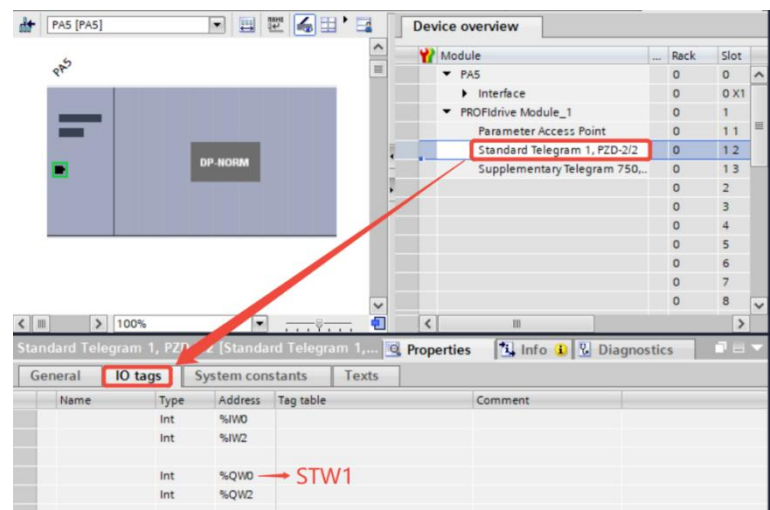
1. The configuration of Telegram 1 is shown in Section 13.4.3.2.
2. In the device view, select PA5, click "Hardware catalog → Module → PROFIdrive Module → Submodules", add telegram 750 to the corresponding position of slot 1 sub-slot 3:



3. Note down the data address of the additional telegram 750, select Additional telegram 750→ Right-click Properties →IO variables:

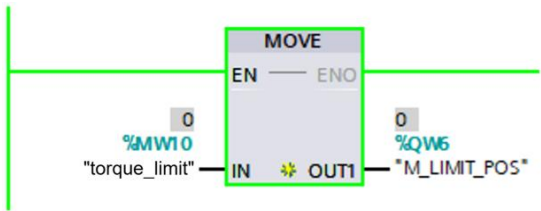


4. Note down the address of STW1, select telegram 1→ Right-click Properties →IO variables



13.4.8.2.2 Application example

1. After adding the additional telegram 750, the torque limit takes effect immediately. The default limit value is 0, so the Max_Torque %[607200] is 0, and at this time, the motor cannot rotate. Please note that iSMD does not support setting the forward limit value and the reverse limit value separately. By assigning a value to M_LIMIT_POS, the torque in both forward and reverse directions can be restricted simultaneously, while M_LIMIT_NEG is invalid.



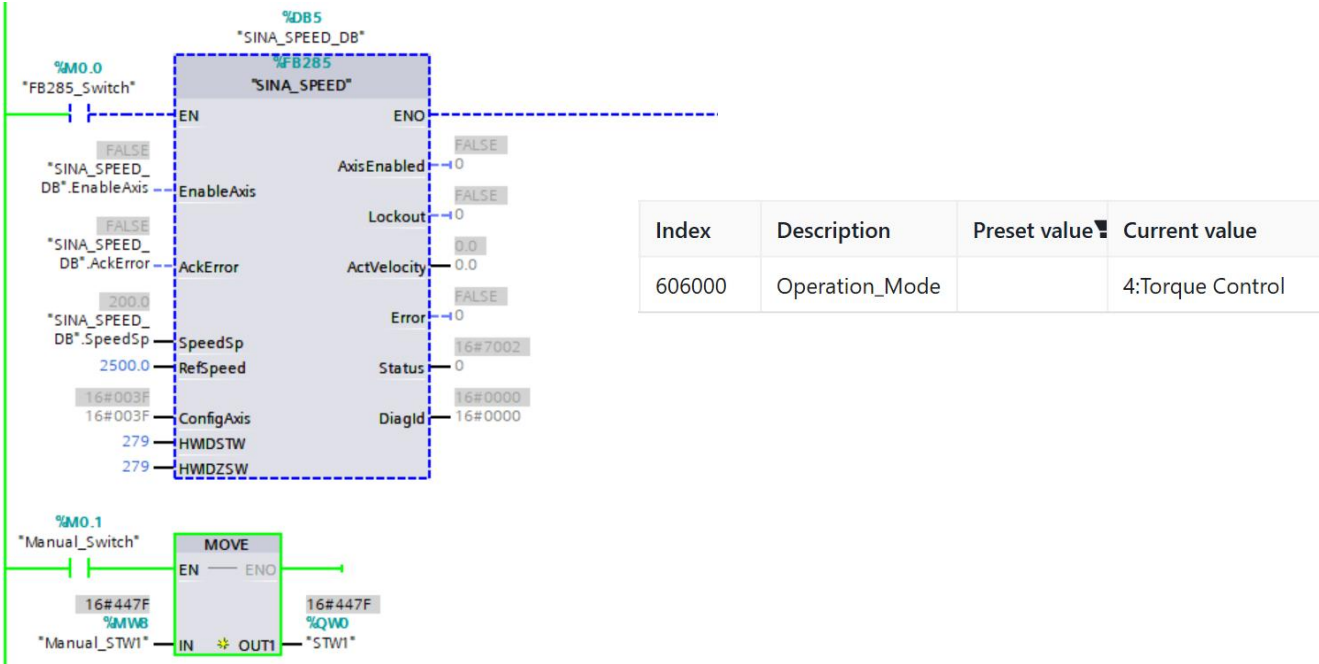
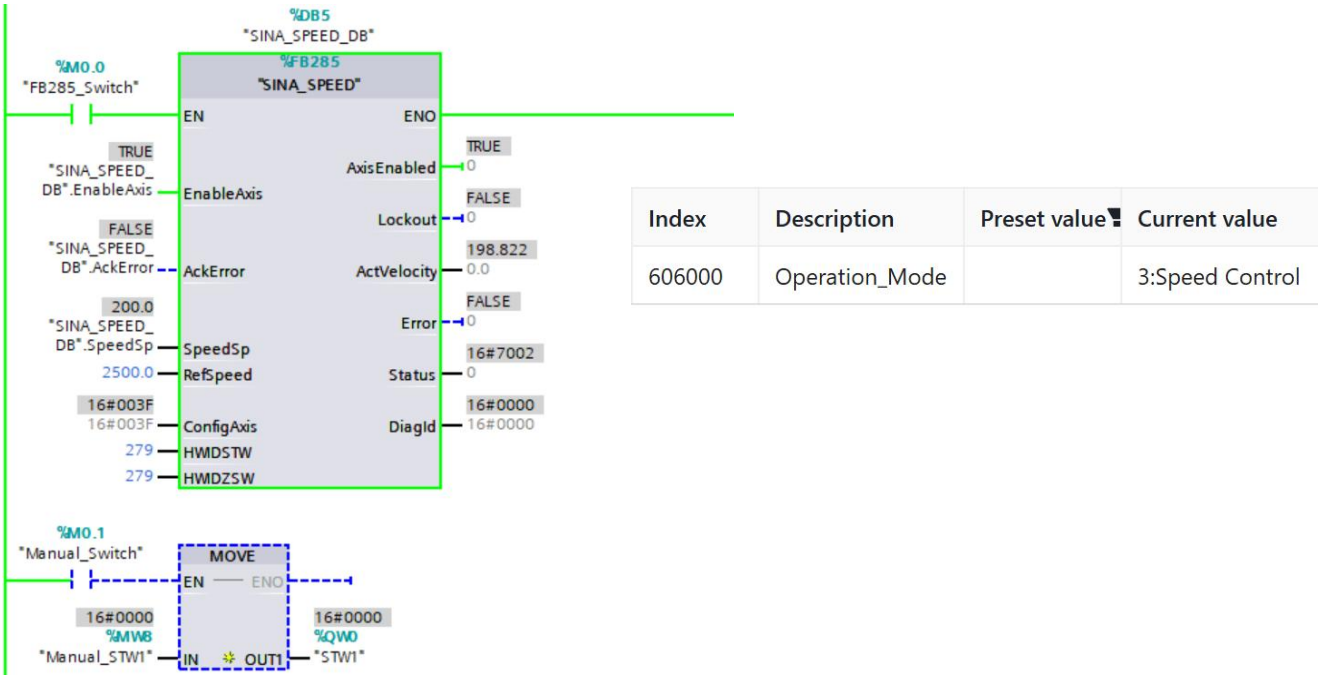
Index	Description	Preset value	Current value
607200	Max_Torque%		0

2. The torque limit value is an unsigned integer and is expressed as a percentage of the motor's rated torque. For example, if the torque limit M_LIMIT_POS is assigned a value of 500, then theMax_Torque% [607200] becomes 50%.

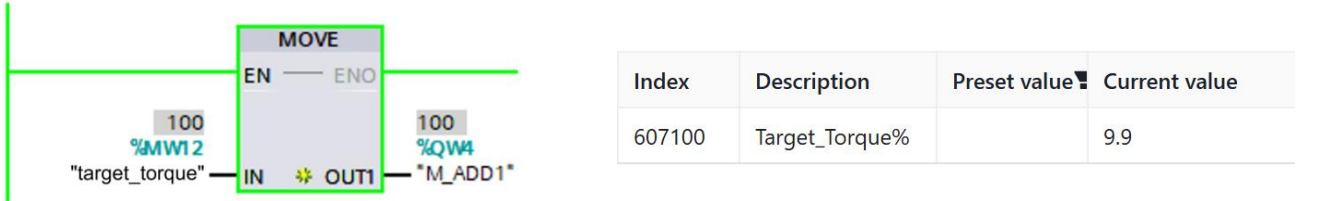


Index	Description	Preset value	Current value
607200	Max_Torque%		50

3. After adding the additional telegram 750, when Bit14 of STW1 is set to 0, the servo is in speed control, the target speed is valid, but the target torque M_ADD1 is invalid; when Bit14 of STW1 is set to 1, the servo is in torque control, the target speed is invalid, but the target torque M_ADD1 is valid, once the servo is enabled, it will rotate at the target torque. Please note the maximum speed limit for the servo before operation!



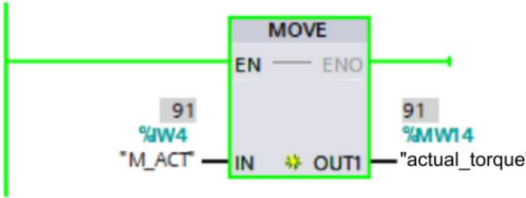
4. The target torque is a signed integer and is expressed as a percentage of the motor's rated torque. For example, if the target torque M_ADD1 is assigned a value of 100, then the Target_Torque %[607100] becomes 9.9%; if the target torque M_ADD1 is assigned a value of -100, then the Target_Torque %[607100] becomes -9.9%.





Index	Description	Preset value▼	Current value
607100	Target_Torque%		-9.9

5. The actual torque is a signed integer, with the unit being a thousandth of the motor's rated torque. For example, if the actual torque M_ACT is read as 91, it indicates that the Actual_Torque [607700] is 9.1%; The Actual_Torque M_ACT read is -91, indicating that the actual torque [607700] is -9.1%.



Index	Description	Preset value▼	Current value
607700	Actual_Torque	Write	9.1

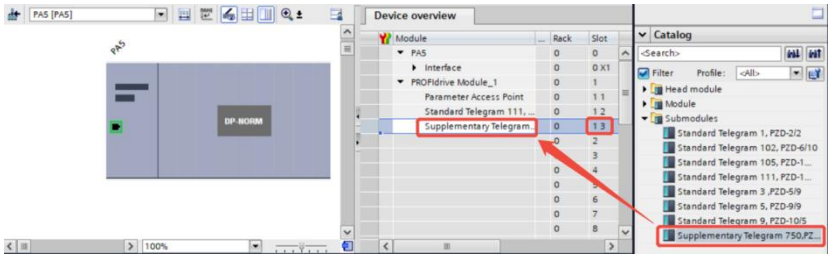


Index	Description	Preset value▼	Current value
607700	Actual_Torque	Write	-9.1

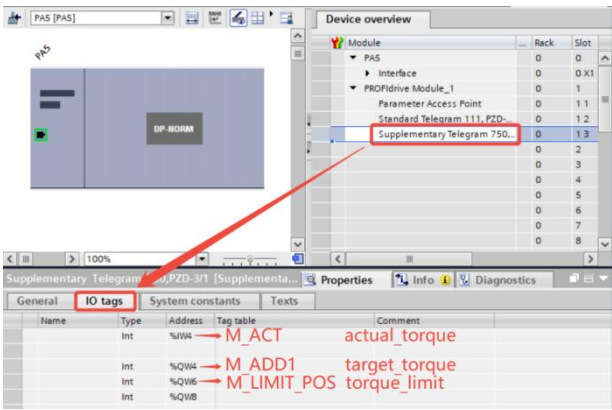
13.4.8.3 Telegram 111+ additional telegram 750

13.4.8.3.1 Project configuration

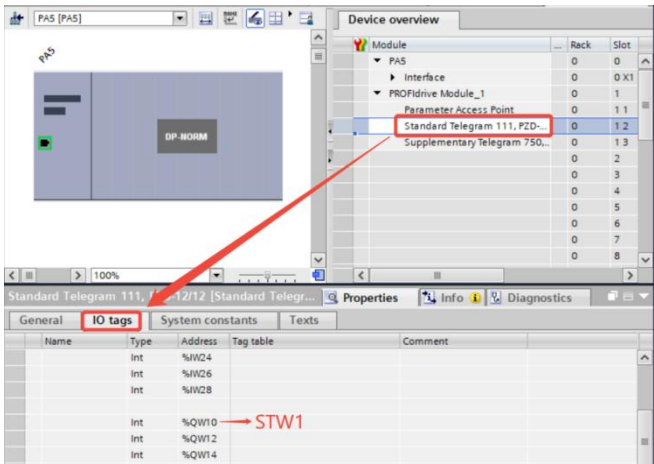
1. The configuration of Telegram 111 is shown in Section 13.4.3.2.
2. In the device view, select PA5, click "Hardware catalog → Module → PROFIdrive Module → Submodules", add telegram 750 to the corresponding position of slot 1 sub-slot 3:



3. Note down the data address of the additional telegram 750, select Additional telegram 750→ Right-click Properties →IO variables:

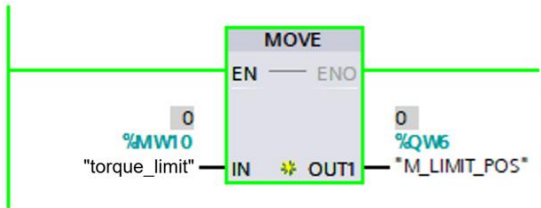


4. Note down the address of STW1, select telegram 111→ Right-click Properties →IO variables



13.4.8.3.2 Application example

- After adding the additional telegram 750, the torque limit takes effect immediately. The default limit value is 0, so the Max_Torque %[607200] is 0, and at this time, the motor cannot rotate. Please note that iSMD does not support setting the forward limit value and the reverse limit value separately. By assigning a value to M_LIMIT_POS, the torque in both forward and reverse directions can be restricted simultaneously, while M_LIMIT_NEG is invalid.



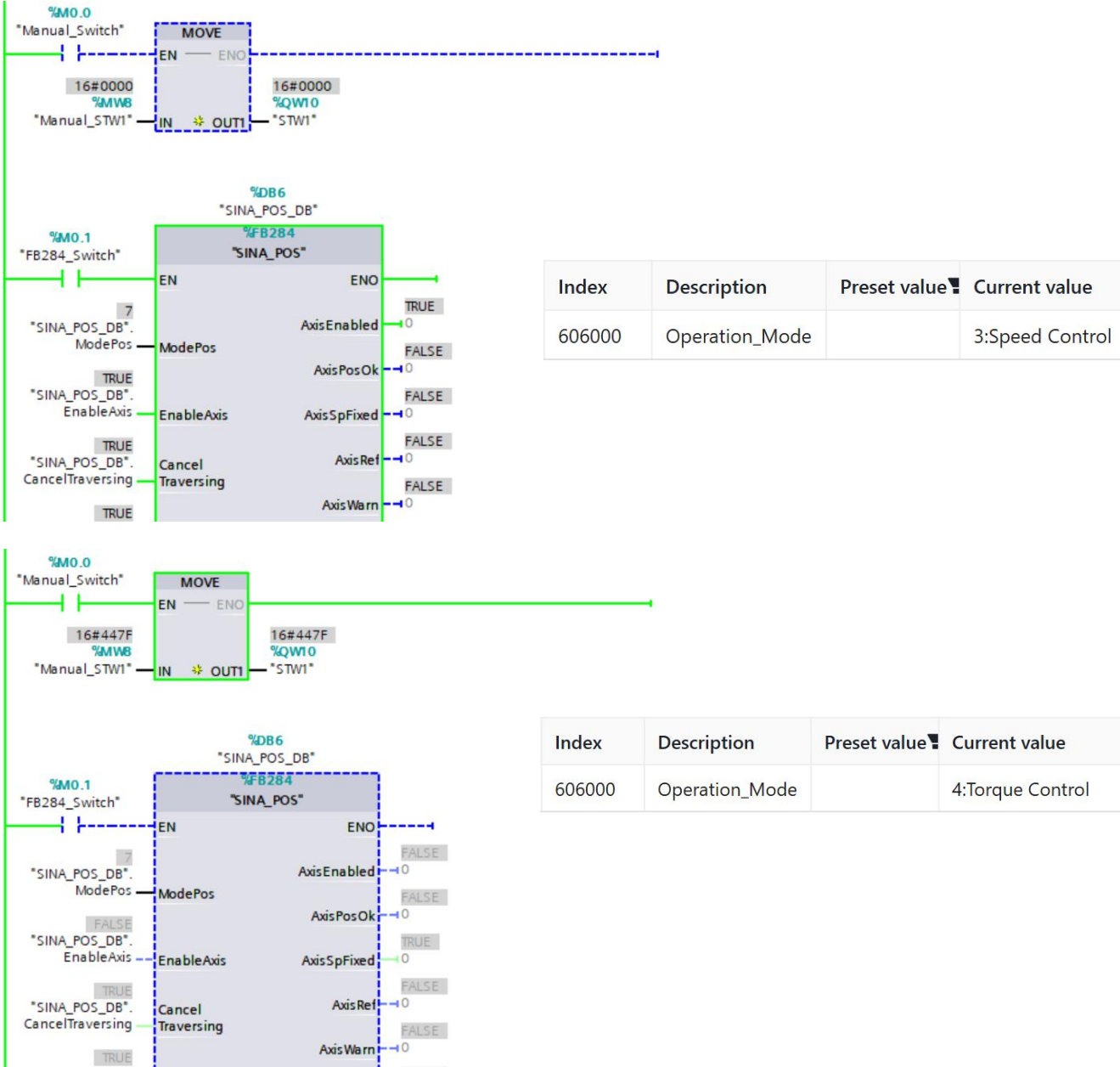
Index	Description	Preset value	Current value
607200	Max_Torque%		0

- The torque limit value is an unsigned integer and is expressed as a percentage of the motor's rated torque. For example, if the torque limit M_LIMIT_POS is assigned a value of 500, then theMax_Torque% [607200] becomes 50%.

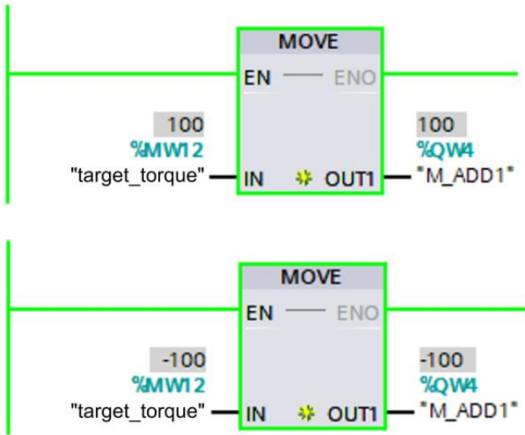


Index	Description	Preset value	Current value
607200	Max_Torque%		50

3. After adding the additional telegram 750, when Bit14 of STW1 is set to 0, the servo is in position/speed control, the position/speed is valid, but the target torque M_ADD1 is invalid; when Bit14 of STW1 is set to 1, the servo is in torque control, the position/speed is invalid, but the target torque M_ADD1 is valid, once the servo is enabled, it will rotate at the target torque. Please note the maximum speed limit for the servo before operation!



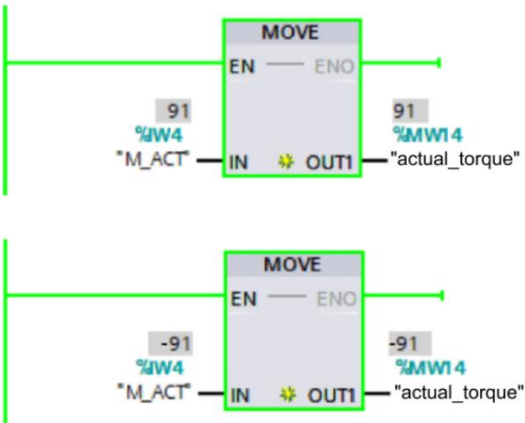
4. The target torque is a signed integer and is expressed as a percentage of the motor's rated torque. For example, if the target torque M_ADD1 is assigned a value of 100, then the Target_Torque %[607100] becomes 9.9%; if the target torque M_ADD1 is assigned a value of -100, then the Target_Torque %[607100] becomes -9.9%.



Index	Description	Preset value▼	Current value
607100	Target_Torque%		9.9

Index	Description	Preset value▼	Current value
607100	Target_Torque%		-9.9

5. The actual torque is a signed integer, with the unit being a thousandth of the motor's rated torque. For example, if the actual torque M_ACT is read as 91, it indicates that the Actual_Torque [607700] is 9.1%; The Actual_Torque M_ACT read is -91, indicating that the actual torque [607700] is -9.1%.



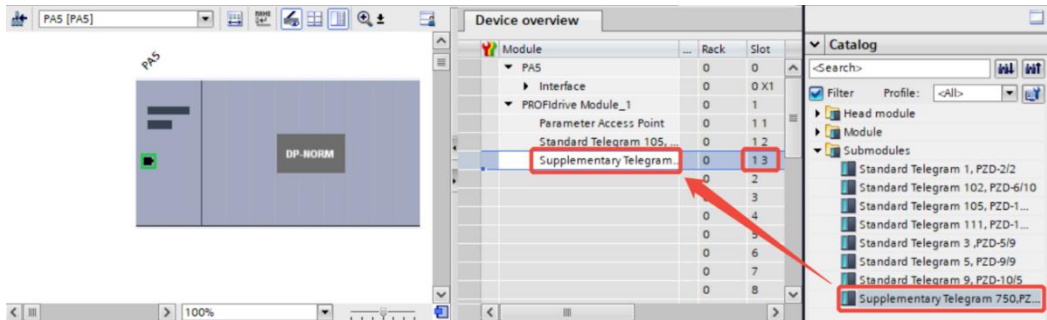
Index	Description	Preset value▼	Current value
607700	Actual_Torque	Write	9.1

Index	Description	Preset value▼	Current value
607700	Actual_Torque	Write	-9.1

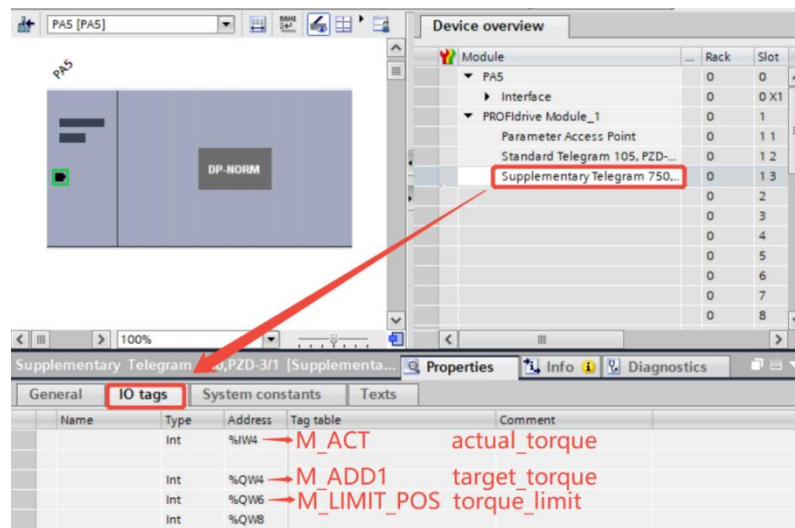
13.4.8.4 Telegram 105 + additional telegram 750

13.4.8.4.1 Project configuration

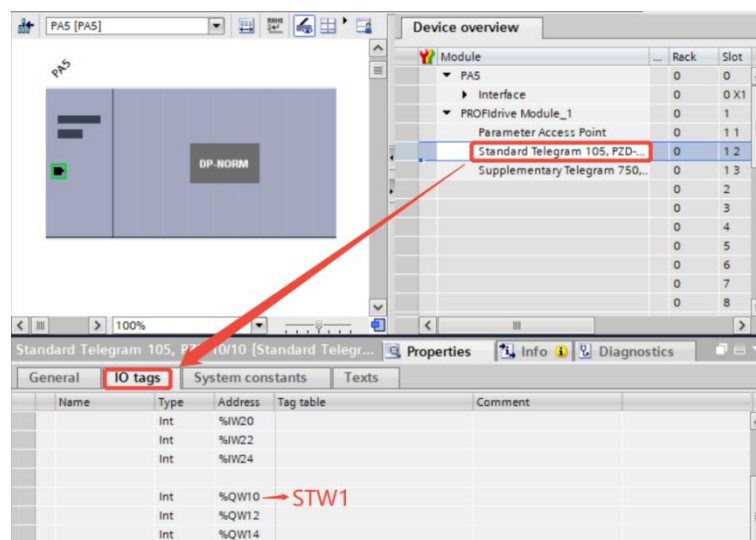
1. The configuration of Telegram 105 is shown in Section 13.4.3.2.
2. In the device view, select PA5, click "Hardware catalog → Module → PROFIdrive Module → Submodules", add telegram 750 to the corresponding position of slot 1 sub-slot 3:



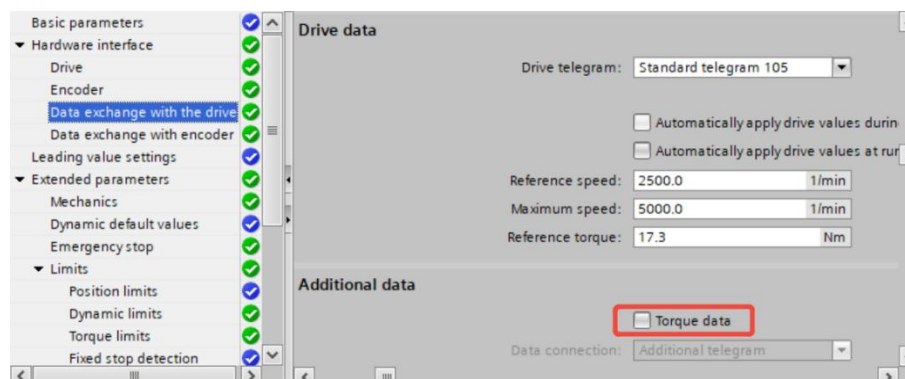
3. Note down the data address of the additional telegram 750, select Additional telegram 750→ Right-click Properties →IO variables:



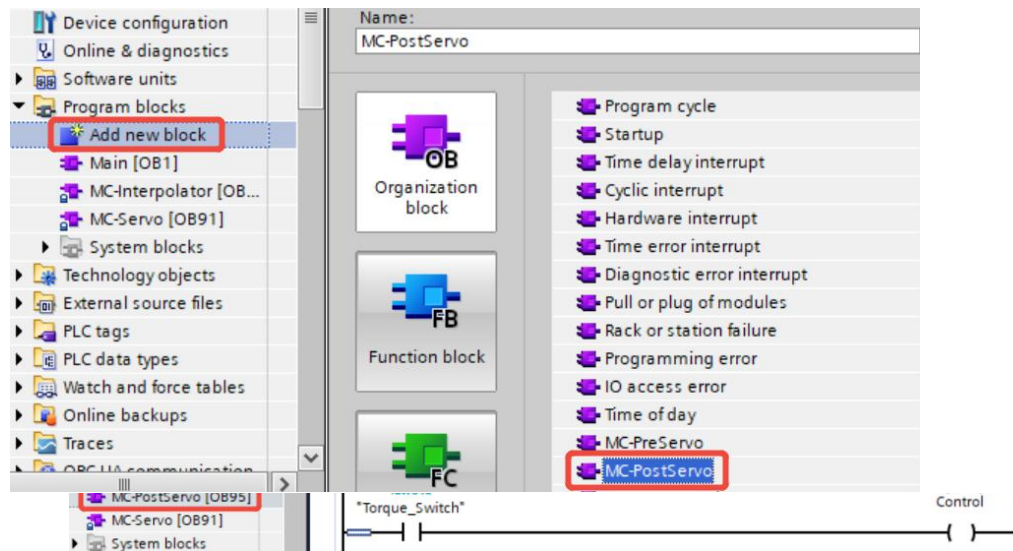
4. Note down the address of STW1 Bit14, select telegram 105 → Right-click Properties → IO variables, Then the Bit14 address is %Q10.6:



5. Please note that when configuring the process object, you need to uncheck "Torque Data" :

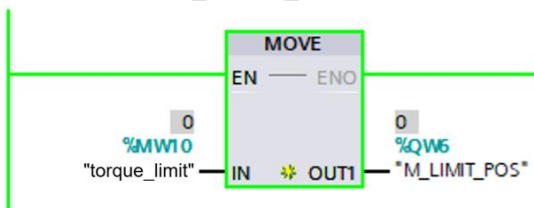


6. Add the MC-PostServo function block in the program and set the torque control mode in this function block:



13.4.8.4.2 Application example

1. Please note that iSMD does not support the MC_TorqueAdditive and MC_TorqueRange commands. Additionally, the priority of the torque limit setting in the additional message 750 is higher than that of MC_TorqueLimiting (refer to Section 13.4.6.4). That is, after adding the additional message 750, MC_TorqueLimiting becomes ineffective.
2. After adding the additional telegram 750, the torque limit takes effect immediately. The default limit value is 0, so the Max_Torque %[607200] is 0, and at this time, the motor cannot rotate. Please note that iSMD does not support setting the forward limit value and the reverse limit value separately. By assigning a value to M_LIMIT_POS, the torque in both forward and reverse directions can be restricted simultaneously, while M_LIMIT_NEG is invalid.



Index	Description	Preset value	Current value
607200	Max_Torque%		0

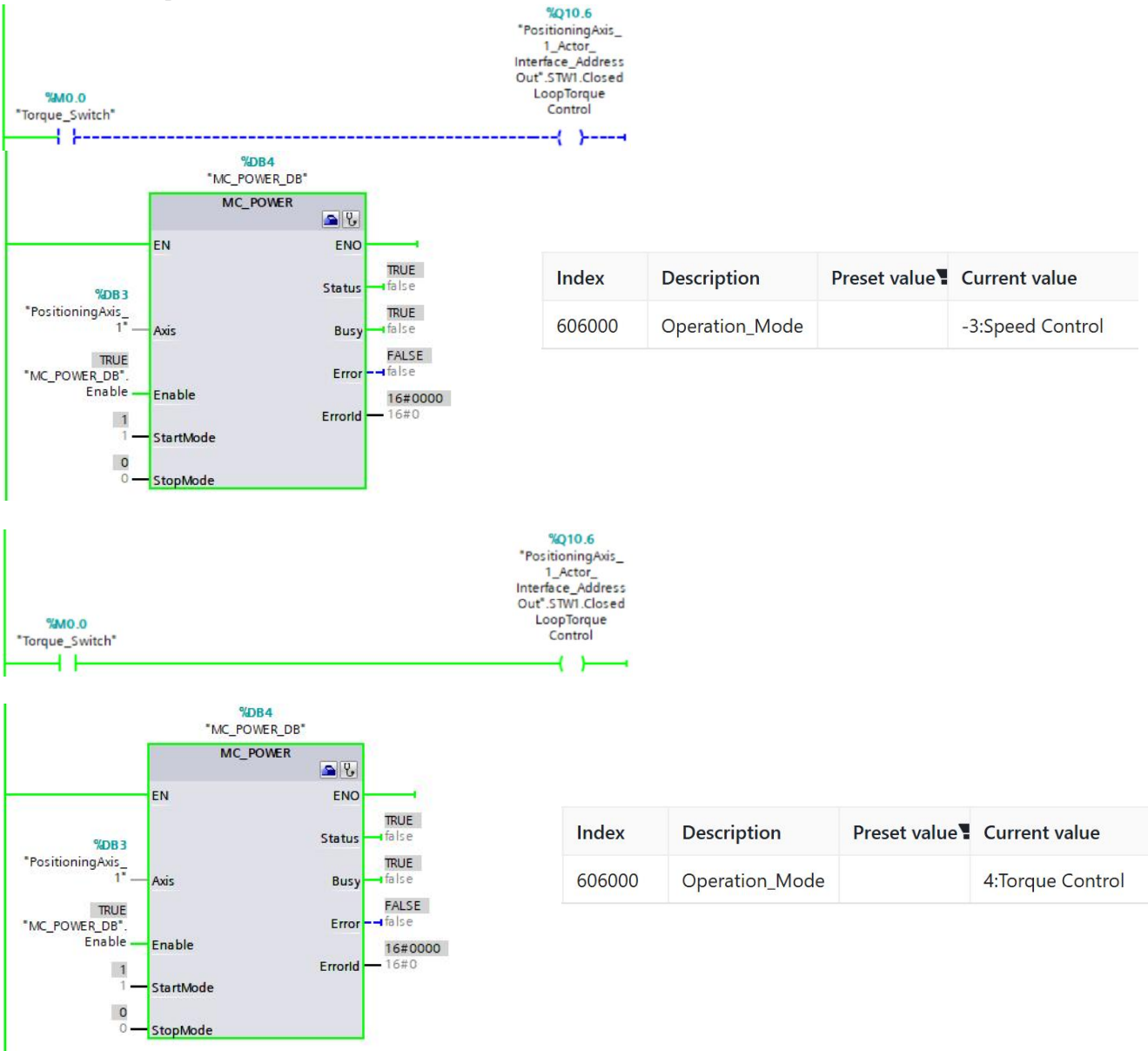
3. The torque limit value is an unsigned integer and is expressed as a percentage of the motor's rated torque. For example, if the torque limit M_LIMIT_POS is assigned a value of 500, then theMax_Torque% [607200] becomes 50%.



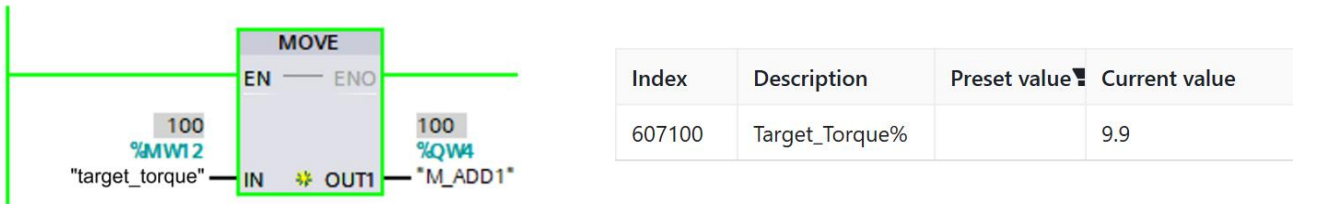
Index	Description	Preset value	Current value
607200	Max_Torque%		50

4. After adding the additional telegram 750, when Bit14 of STW1 is set to 0, the servo is in position/speed control, the position/speed is valid, but the target torque M_ADD1 is invalid; when Bit14 of STW1 is set to 1, the servo is in torque control, the position/speed is invalid, but the target torque M_ADD1 is valid, once

the servo is enabled, it will rotate at the target torque. Please note the maximum speed limit for the servo before operation!



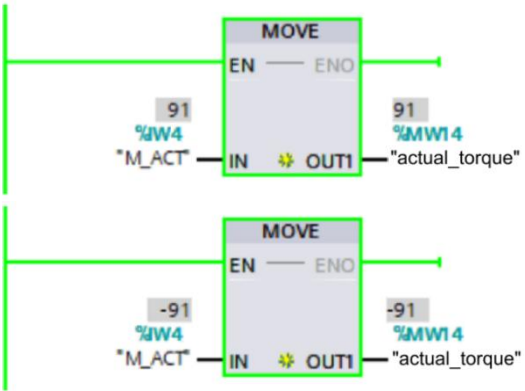
5. The target torque is a signed integer and is expressed as a percentage of the motor's rated torque. For example, if the target torque M_ADD1 is assigned a value of 100, then the Target_Torque %[607100] becomes 9.9%; if the target torque M_ADD1 is assigned a value of -100, then the Target_Torque %[607100] becomes -9.9%.





Index	Description	Preset value	Current value
607100	Target_Torque%		-9.9

6. The actual torque is a signed integer, with the unit being a thousandth of the motor's rated torque. For example, if the actual torque M_ACT is read as 91, it indicates that the Actual_Torque [607700] is 9.1%; The Actual_Torque M_ACT read is -91, indicating that the actual torque [607700] is -9.1%.



Index	Description	Preset value	Current value
607700	Actual_Torque	Write	9.1

Index	Description	Preset value	Current value
607700	Actual_Torque	Write	-9.1

Appendix I Commonly used formulas

The selection method of the trolley motor is suitable for the mechanism of the motor + reducer + wheel

Formula: $T \cdot n = \mu \cdot m \cdot g \cdot d / 2$	
Diameter of wheel d	m
The reduction ratio of the reducer n	1 : n
Torque of motor T	Nm, kgm^2/s^2
Full load capacity m	kg
friction coefficient μ	unitless
gravitational acceleration g	m/s^2

The relationship between the number of pulses and mechanical displacement in pulse mode

Formula: $N \cdot A / B = s \cdot n \cdot r / P$	
Gear ratio molecule A	unitless
Gear score B	unitless
Ball screw lead P	mm
The number of pulses per turn of the motor r	unitless
Reduction ratio $1 : n$	unitless
Mechanical displacement s	mm
Pulse count N	unitless

The relationship between speed and linear velocity

Formula: $n = v \div r \div \pi$	
Speed n	rpm
linear velocity v	mm/s
Radius r	mm

Parameter Name	Engineering unit	Internal unit	Conversion relation
Speed	rpm	DEC	$\text{DEC} = [(\text{RPM} \cdot 512 \cdot \text{encoder resolution}) / 1875]$
Acceleration		DEC	$\text{DEC} = [(\text{RPS}/\text{S} \cdot 65536 \cdot \text{encoder resolution}) / 4000000]$
Current	A	DEC	$1\text{Arms} = [2048 / (\text{drive peak current } I_{\text{peak}} / 1.414)]\text{dec}$

For example:

The speed engineering unit is rpm, the internal unit is dec, and the relationship between the two is that 1RPM is equal to 2730dec (encoder resolution 10000)! Assuming that the required speed is 10rpm, then the write speed is 27300dec.

The current engineering unit is Arms, the internal unit is dec, 1Arms is equal to about 60dec, if the current needs to be 10Arms, then the current needs to be written to 600dec when using communication control.