



softMC Compact 302

Controller Software User Manual

Revision History

Document Revision	Date	Remarks
1.0	January 2024	Initial release.

Copyright Notice

© 2020 Servotronix Motion Control Ltd. All rights reserved.

No part of this work may be reproduced or transmitted in any form or by any means without prior written permission of Servotronix Motion Control Ltd.

Disclaimer

The information in this manual was accurate and reliable at the time of its release. Servotronix Motion Control Ltd. reserves the right to change the specifications of the product described in this manual without notice at any time.

Contact Information

Servotronix Motion Control Ltd.

Address: Room 605, Building B2, Kexing Science Park, Nanshan District, Shenzhen, China

Tel: 400-111-8669

Website: www.servotronix.cn

Email: servotronix@midea.com

Technical Support

If you need assistance with the installation and configuration of the SC302 motion controller, contact Servotronix technical support. Tel: 400-111-8669

Customer Service

Servotronix is committed to delivering quality customer service and support for all our products. Our goal is to provide our customers with the information and resources so

that they are available, without delay, if and when they are needed. In order to serve in the most effective way, we recommend that you contact your local sales representative for order status and delivery information, product information and literature, and application and field technical assistance. If you are unable to contact your local sales representative for any reason, please use the most relevant of the contact details below:

For all other inquiries regarding SC302 motion controller or other Servotronics products, contact: m.servotronics.com.cn

Unpacking

Upon arrival, please open the package and remove all packaging materials. Check the product to ensure that there is no visible damage. If damage is found, please inform the carrier immediately.

Contents

1. Safety Precautions.....	1
1.1 Safety statement	1
1.2 Definition of security level	1
1.3 Safety precautions.....	1
2. Introduction.....	8
2.1 SC302 Overview.....	8
2.1.1 Product overview	8
2.1.2 Controller features	9
2.1.3 Software functional characteristics.....	9
2.2 CODESYS Overview	10
2.2.1 Introduction of CODESYS.....	10
2.2.2 Connecting CODESYS to hardware	11
2.2.3 Software Acquisition and Installation	13
2.2.4 Installation steps	13
3. Introduction to Hardware Parameters.....	15
3.1 Hardware Parameter Specification	15
3.1.1 Hardware interface overview.....	15
3.1.2 Application environment specifications.....	16
3.1.3 Software performance index specifications.....	17
3.1.4 External dimensions	18
3.2 Hardware interface details.....	18
3.2.1 Digital input	18
3.2.2 Digital output.....	19
3.2.3 CAN	20
3.2.4 RS485	21
3.2.5 RS232	24
3.2.6 24V DC Power Supply	25
3.2.7 USB	25
3.2.8 EtherCAT	26
3.2.9 SD.....	28
3.2.10 RESET	28
3.2.11 RUN/STOP.....	29
3.2.12 LED status indicator light.....	29
3.2.13 PORT1 EtherNET	31
3.2.14 PORT2 EtherNET	31

3.2.15 RTC Clock	32
3.2.16 Power down holding function	32
4. Configuring EtherCAT Servo	33
4.1 Description of Installing SC302 Equipment	33
4.2 Adding EtherCAT Master softmotion	34
4.3. Configuring EtherCAT Master.....	36
4.4 Scanning devices	37
4.5 Adding 402 Axis	39
4.6 Configuring axis parameters	39
5. IO Module.....	41
5.1 Input/output Introduction	41
5.2 Installation Equipment Description and library File.....	45
5.3 Adding LocalHSIO Device.....	47
5.4 Configuring LocalHSIO Device	49
5.5. Introduction to high speed output function block.....	54
5.5.1.HSMC_Power.....	54
5.5.2.HSMC_Reset.....	55
5.5.3.HSMC_MoveAbsoulte	55
5.5.4.HSMC_MoveRelative.....	57
5.5.5.HSMC_MoveVelocity.....	58
5.5.6.HSMC_MoveJog	59
5.5.7.HSMC_Home	60
5.5.8.HSMC_SetPosition.....	63
5.5.9.HSMC_FollowPostion	64
5.5.10.HSMC_FollowVelocity	65
5.5.11.HSMC_Stop.....	66
5.5.12.HSMC_PWM.....	67
5.5.13.HSMC_WriteParameter	68
5.5.14.HSMC_ReadParameter.....	70
5.6. High-speed output application examples	72
5.6.1. Absolute positioning motion.....	72
5.6.2. Jog motion	75
5.6.3. Follow velocity motion	77
5.6.4. PWM output	78
5.7. Introduction to high-speed input function block.....	80
5.7.1. HS_Counter	80
5.7.2. HS_PresetValue	84
5.7.3. HS_SetCompare.....	85

5.7.4. HS_CompareInterrupt	87
5.7.5. HS_MeasurePulseWight.....	88
5.7.6. HS_Latch.....	89
5.7.7. HS_WriteParameter	91
5.8. High speed input application.....	93
5.8.1. Counter phase AB counting application.....	93
5.8.2. Counter comparison output application.....	94
5.9.HSIO_ERROR_CODE error code.....	97
6. Use USB disk to update applications.....	98
6.1 Operating procedure	98

1. Safety Precautions

1.1 Safety statement

1) This chapter explains the safety matters that need to be paid attention to when using this product. Before using this product, please read the instruction manual and correctly understand the relevant information of safety precautions. Failure to comply with the agreed items in safety matters may lead to death, serious injury or equipment damage.

2) The words “Danger”, “Warning” and “Caution” in the manual do not represent all safety matters that should be observed, but only serve as supplements to all safety precautions.

3) This manual should be used in an environment that meets the requirements of design specifications, otherwise it may cause failure, and abnormal function or partial damage caused by non-compliance with relevant regulations is not within the product quality assurance scope.

4) Our company will not bear any legal responsibility for personal safety accidents and property losses caused by non-compliance with the contents of this manual and illegal operation of products.

1.2 Definition of security level



means that if you don't follow the fixed principles, it will lead to death or serious physical injury.



means that if you don't follow the fixed principles, it may lead to death or serious physical injury.



means that if you don't follow the fixed principles, it may lead to minor physical injury or equipment damage.

1.3 Safety precautions

1) In the illustrations in this manual, sometimes the product is in the state of removing the cover or safety cover in order to show the details of the product. When using, please be sure to install the cover or protective cover according to the requirements, and operate according to the fixed principles.

2) The pictures shown in this manual are only sample pictures, which may be slightly different from the products you ordered. The products actually ordered shall prevail.

Table 1-1

Unpacking acceptance	
 Warning	● When unpacking, if the product and accessories are found to be damaged, rusted or have traces of use, please do not install it! ● When unpacking, if water is found inside the product, parts are missing or damaged, please do not install it! ● Please check the packing list carefully. If the packing list does not match the product name, please do not install it!
 Caution	● Before unpacking, please check whether the outer packaging of the equipment is intact, and whether there is damage, soaking, deformation. ● Please open the package according to the hierarchical order, and it is forbidden to knock violently! ● Before unpacking, please check whether the surface of the equipment and accessories are damaged. ● After unpacking, please check the packing list and check whether the quantity and information of equipment and accessories are complete.
Storage and transportation	
 Warning	● Please use professional lifting equipment, and carry large or heavy products by qualified professionals. Otherwise, there is a risk of injury or product damage! ● Before lifting the product vertically, please ensure that the product components such as the front cover and terminal block have been firmly fixed with screws, otherwise the parts will fall off, which may lead to personal injury or product damage! ● When the product is lifted by lifting equipment, no one is allowed to stand or stay under the product!

- When lifting the product with wire rope, please lift it smoothly and evenly, so as to avoid the product from vibration and shock. Don't flip the product, and don't let it hang for a long time, otherwise there will be a risk of personal injury or product damage!

 Caution

- Please handle the product with care, and always pay attention to the objects under your feet to prevent tripping or falling, otherwise there is a risk of injury or product damage!
- When carrying the product by hand, please hold the product shell firmly to avoid the product parts falling off, otherwise there is a risk of injury!
- Please store and transport in strict accordance with the storage and transportation conditions required by the product, otherwise there is a risk of product damage!
- Avoid storing and transporting products in places such as splashing rain, direct sunlight, strong electric field, strong magnetic field and strong vibration.
- Avoid storing the product for more than 3 months. If the storage time is too long, please carry out strict protection and necessary inspection.
- Please pack the products strictly before vehicle transportation, and a closed box must be used for long-distance transportation!
- It is strictly forbidden to transport this product together with equipment or articles that may affect or damage this product!

Installation

 Danger

- Only professionals who have been trained in electrical equipment and have electrical knowledge can operate this equipment. Operation by non-professionals is strictly prohibited!

 Warning

- Please read the instruction manual and safety precautions carefully before installation!
- Please do not install this product in places with strong electric field or strong electromagnetic interference!
- Before installation, please ensure that the mechanical strength of the installation location is enough to support the weight of the equipment, otherwise it will lead to mechanical danger!
- Please don't wear loose clothes or accessories during installation, otherwise there may be the risk of electric shock!
- When installing the product in a closed environment (such as a cabinet), please use a cooling device (such as a cooling fan or a cooling air conditioner) to fully cool it to meet the requirements of the installation environment, otherwise it may lead to overheating or fire!
- No modification of this product is allowed!

Installation	
- Do not twist the fixing bolts of product parts and components and the bolts marked in red! - When this product is installed in a cabinet or terminal equipment, the cabinet or terminal equipment should be equipped with corresponding protective devices such as fire-proof enclosure, electrical protective enclosure and mechanical protective enclosure, and the protection class should meet the requirements of relevant IEC standards and local laws and regulations. - When it is necessary to install transformers and other equipment with strong electromagnetic interference, please install shielding devices to avoid misoperation of this product! - Please install the product on metal and other flame-retardant objects, and do not let flammable materials touch or attach to the product, otherwise there will be a risk of fire!	
 Caution	- When installing, please cover the top of the product with cloth or paper to prevent foreign objects such as metal chips, oil and water from entering the product during drilling, thus causing product failure. After the operation, please remove the cover to prevent the cover from blocking the ventilation hole, affecting the heat dissipation and causing the product to generate abnormal heat. - When a machine running at a constant speed runs at a variable speed, resonance may occur. At this time, installing anti-vibration rubber under the motor base or using vibration suppression function can effectively weaken resonance.
Wiring	
 Danger	- Non-professionals are forbidden to install, wire, maintain, check the equipment, or replace parts of equipment! - Please cut off the power supply of all equipment before wiring. After the power supply is cut off, there is residual voltage in the internal capacitor of the equipment. Please wait at least the time specified in the product warning label before wiring and other operations. Measure the DC voltage of the main circuit, and ensure it is below the safe voltage, otherwise there will be danger of electric shock! - Please connect the wires, remove the product cover or touch the circuit board when the power supply is cut off, otherwise there will be danger of electric shock! - Please ensure that the equipment and products are well grounded, otherwise there will be danger of electric shock!
 Warning	It is forbidden to connect the input power supply to the output end of

equipment or products, otherwise it will cause equipment damage and even fire. ● When the driving equipment is connected to the motor, please ensure that the sequence of the product and the motor terminal is accurate and consistent, so as to avoid the reverse rotation of the motor. ● The cable used in wiring must meet the requirements of corresponding wire diameter and shielding, and the shielding layer of the cable needs to be reliably grounded at one end. ● Please tighten the terminal screws according to the tightening torque specified in the manual. Inadequate or excessive tightening torque may cause overheating, damage and fire. ● After wiring, please ensure that all cables are wired correctly, and there are no dropped screws, washers or exposed cables inside the product, otherwise there may be electric shock danger or damage to the product.
⚠ Caution ● Please follow the steps specified in ESD measures, and wear an electrostatic bracelet for wiring and other operations to avoid damaging the internal circuits of equipment or products. ● When wiring the control loop, please connect the shielding layer to the grounding terminal of the product by using the shielded twisted pair cable, otherwise the product will act abnormally.
Powering on
● Before powering on, please ensure that the product is installed, the wiring is firm, and the motor is allowed to restart! ● Before powering on, please ensure that the power supply meets the product requirements to avoid product damage or fire! ● It is forbidden to open the product cabinet door or product protective cover, touch any terminal of the product, or remove any device or parts of the product when it is energized, otherwise there is a danger of electric shock!
⚠ Warning ● After the wiring operation and parameter setting are completed, please carry out a trial run of the machine to confirm that the machine can run safely, otherwise personal injury or equipment damage may be caused. ● Please ensure that the rated voltage of the product is consistent with the power supply voltage before powering on. If the power supply voltage is not used properly,, there will be a fire hazard. ● Please ensure that there are no people around the products, motors and machinery before powering on, otherwise personal injury or death may be caused.
Running

 Danger ● Non-professionals are strictly forbidden from running the product, otherwise it will lead to personal injury or death! ● It is forbidden to touch any terminal of the equipment and remove any device or parts of the equipment and products during operation, otherwise there is a danger of electric shock!
 Warning ● Do not touch the device housing, fan or resistor to test the temperature, otherwise it may cause burns! ● During operation, avoid other articles or metal objects falling into the equipment, otherwise it may lead to fire or product damage!
Maintenance
 Danger ● Non-professionals are forbidden to install, wire, maintain, check the equipment, or replace parts of equipment! ● It is forbidden to carry out equipment maintenance under the power-on state, otherwise there is a danger of electric shock! ● After cutting off the power supply of all equipment, please wait at least the time specified in the warning label on the product before checking, maintaining and other operations on the equipment.
 Warning ● Please maintain the equipment according to the product warranty agreement. ● In case of fuse blowing, circuit breaker tripping or earth leakage circuit breaker (ELCB) tripping, please wait at least the time specified in the warning label on the product before turning on the power supply or operating the machine, otherwise it may cause casualties and equipment damage. ● When the equipment breaks down or is damaged, professionals are required to troubleshoot and repair the equipment and products according to the maintenance instructions, and keep maintenance records. ● Please follow the instructions for replacing the easily damaged parts of the product. ● Please do not continue to use the damaged machine, otherwise it may cause casualties or damage to the product to a greater extent. ● After replacing the device, be sure to check the device wiring and set parameters again.
Scrapping
 Warning ● Please scrap the equipment and products according to the relevant national regulations and standards, so as to avoid property losses or casualties! ● Please recycle the equipment and products according to the industrial waste

treatment standards to avoid environmental pollution.

2. Introduction

2.1 SC302 Overview

2.1.1 Product overview

The SC302 motion controller from SERVOTRONIX provides users with intelligent automation solutions. SC302 adopts IEC61131-3 programming language system and supports PLCopen programming language. SC302 uses expandable structure and supports 16 local ET series IO modules through the internal bus protocol; besides, other devices can be connected through various industrial field buses such as EtherCAT and CANopen. This product can realize high-performance motion control function through EtherCAT bus and provides single-axial acceleration and deceleration control function, electronic gear function and electronic cam function. It also supports CAN, RS485, RS232, Ethernet, USB and other communication functions. SC302 supports up to 32 axes of motion through bus expansion. It is a flexible product which can meet the diverse application needs of users. SC302 supports at most 12 digital inputs (8 high-speed DI inputs and 4 low-speed DI inputs) and 8 high-speed DO outputs, and supports the maximum frequency of 200KHz.



Figure 2-1 Side View of SC302 Motion Controller + ET Series IO Module



Figure 2-2 Front View of SC302 Motion Controller + ET Series IO Module

2.1.2 Controller features

- 1) A variety of motion control, such as point-to-point motion, gear, linear interpolation, electronic cam;
- 2) Support up to 10000+ I/Os;
- 3) Adequate program capacity and data storage area;
- 4) Fast instruction execution;
- 5) Support more high-end field buses (EtherCAT, CANopen, EtherNet, OPC-UA);
- 6) Reliable and easy-to-use software, meeting the needs of different users;
- 7) Support online debugging mode;
- 8) Support online editing mode.

2.1.3 Software functional characteristics

Table 2-1 SC302 Software Functional Characteristics

Product model	DI	DO	Size of power failure data storage	Number of motion control axis	Function registration level
SC-E08-302-0103	12 inputs	8 outputs	1MB	2ms16	Softmotion+ MulCore
SC-E08-302-0104	12 inputs	8 outputs	1MB	2ms16	Softmotion+WebVisu+ MulCore
SC-E08-302-0105	12 inputs	8 outputs	1MB	2ms16	CNC+Robotics +MulCore

Note: All models of bus EtherCAT, CANOpen, and OPCUA are supported.

2.2 CODESYS Overview

2.2.1 Introduction of CODESYS

CODESYS stands for controller development system. It is a development environment for programming controller applications developed by 3S Company. The version currently used by our company is CODESYS V3.5 SP17 Patch4 platform, which provides a complete set of configuration, programming, debugging and monitoring environment for programming controllers. It also can deal with the powerful IEC language.

CODESYS provides the following configuration functions:

- 1) I/O module configuration;
- 2) EtherCAT bus configuration;
- 3) CANOpen bus configuration;
- 4) OPC-UA configuration.

It supports functions such as program writing, downloading, and debugging, providing the following conveniences for programmers.

- Standardized programming (in line with IEC 61131-3 standard)

Supporting a variety of programming languages: Structured Text (ST), Ladder Diagram (LD), Sequential Function Chart (SFC), and the IEC 61131-3 programming language, Continuous Function Chart (CFC).

- Flexible function block library

Comprehensive function block library, supporting user-defined libraries.

- Offline simulation function

No need to connect PLC hardware to complete program debugging and simulation.

- Intelligent debugging and error checking

Pre-compile and compile to check for errors; quickly locate programming errors; diagnose and record.

- Sampling tracking

Establishment of timing diagram of process variables.

2.2.2 Connecting CODESYS to hardware

Programming devices can use CODESYS software to write user programs via Ethernet (which can go through hubs, switches, etc.). Download the program to PLC, then monitor the program and control PLC.



Figure 2-3 Controller and PC Connected via Network Cable

When downloading CODESYS application using PORT2 EtherNet port, please modify the IP of the computer network port, which must be located in the same network segment as the controller IP. For example, the default IP address of controller PORT2 is 192.168.0.1; the subnet mask is 255.255.255.0, and the IP of PC needs to be changed to 192.168.0.XX (XX cannot be 1, i.e. it cannot be the IP address of the controller).

Figure 2-4 Network Setting for PC Connected to Controller

Alternatively, you can also connect through other network ports, as long as it is the same network segment as the IP address of the connected network port. Open CODESYS IDE and install the device description XML file of the controller (get it from SERVOTRONIX sales staff), then you can scan the controller and download the application program.

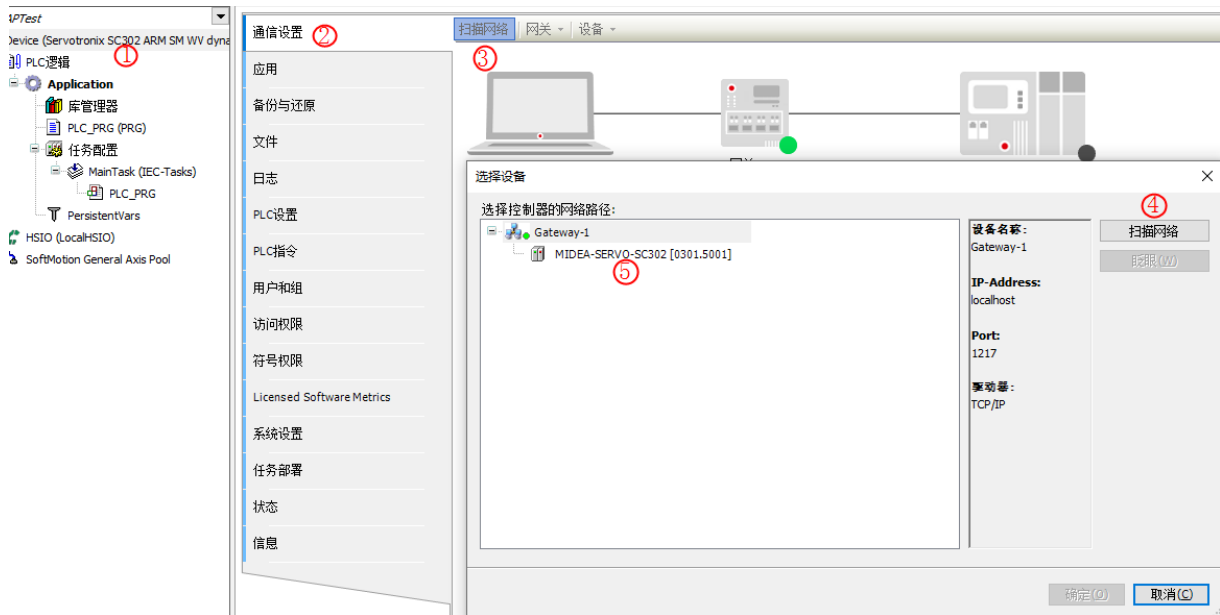


Figure 2-5 CODESYS Online Scanning Controller

2.2.3 Software Acquisition and Installation

CODESYS software is a free software, which can be installed by downloading CODESYS V3.5SP17 Patch 4 from CODESYS official website. CODESYS official website: <https://store.codesys.com/codesys.html#product.attributes.wrapper>; or you can contact the sales staff of Midea SERVOTRONIX to obtain it.

The desktop PC or laptop PC on which the software is installed requires the following:

- ✧ Window10 operating system, 64bit operating system is recommended;
- ✧ Memory: at least 8GB;
- ✧ Space: at least 20G of free hard disk space;
- ✧ It is suggested that the CPU frequency of computer should be above 2 GHz, otherwise it will affect the running speed;
- ✧ It is suggested that the firewall function of Windows be turned off.

2.2.4 Installation steps

Preparation before installation:

When installing CODESYS for the first time, please check the free space on the computer hard disk, and ensure that the target disk to be installed has more than 20 GB of free space, and then you can install directly. If you are upgrading and installing CODESYS, please back up your existing working files, and then uninstall the old version of CODESYS, and restart your computer before starting to install the new version of the software.

Installation:

During installation, it is recommended to connect the computer to the Internet. Click the installation package with administrator privileges, select the installation directory, and then click Next until the end.

3. Introduction to Hardware Parameters

3.1 Hardware Parameter Specification

3.1.1 Hardware interface overview

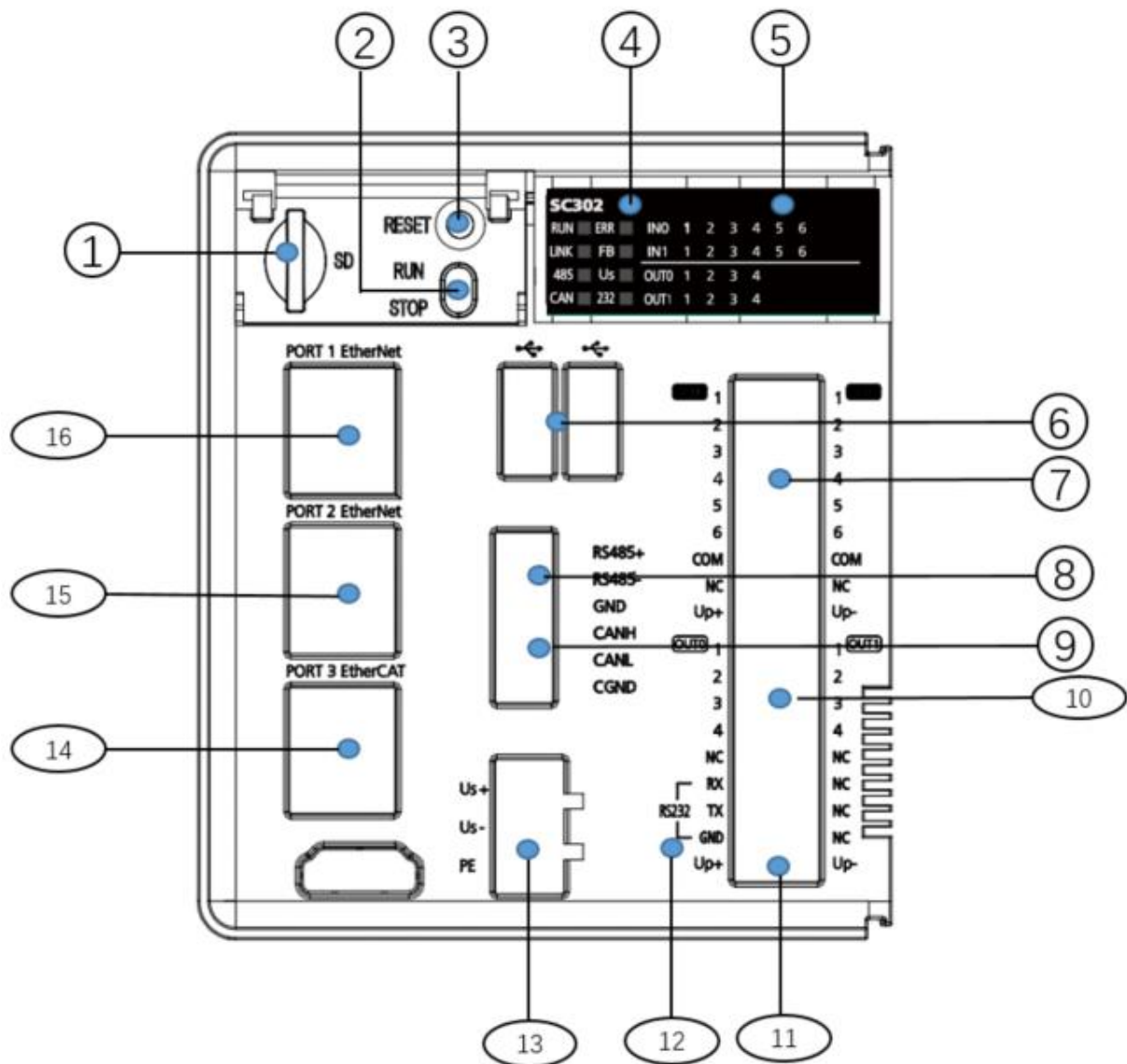


Figure 3-1 SC302 Controller Hardware Interface

Table 3-1 SC302 Hardware Interface Descriptions

No.	Name	Function
-----	------	----------

No.	Name	Function
1	SD card interface	1. Users can insert the SD card by themselves and store the CODESYS application when the SD is inserted. (Note: The factory default does not come with an SD card)
2	Dip switch	Control the start and stop of CODESYS application
3	RESET button	Reset the default IP/delete CODESYS application
4	System indicator light	Running indicator light RUN
		Error indicator light ERR
		Extension IO module indicator light LINK
		EtherCAT bus indicator light FB
		RS485 indicator light
		Power indicator Us
		CANopen indicator light CAN
		RS232 indicator light
5	IO indicator light	IO indicator light
6	USB port	Plugable U disk, USB to serial port module
7	DI	DI interface
8	RS485 interface	RS485 interface, supporting ModbusRTU protocol and free protocol
9	CAN interface	CANopen protocol
10	DO	DO interface
11	IO power supply	IO 24V power supply
12	RS232 interface	RS232 interface, supporting ModbusRTU protocol and free protocol
13	CPU module 24V DC power interface	24V DC voltage input
14	EtherCAT	EtherCAT protocol
15	PORT2 EtherNET	Standard Ethernet function/Modbus-TCP protocol /OPC-UA/User program download and debugging (IPv4 only).
16	PORT1 EtherNET	Modbus-TCP protocol/User program download and debugging (IPv4 only).

3.1.2 Application environment specifications

Table 3-2 Application Environment Specifications

	Parameter type	Operating environment	Transportation environment	Storage environment
	Environmental Class	IE33	IE33	IE33
	Temperature	-20-55°C	-40-85°C	-40-85°C
	Humidity	10-95%, no condensation		

Environmental parameters (IEC60721-3)	Vibration	Frequency	5-150Hz	2M2	1M2
		Displacement	3.5mm (direct installation) (< 9Hz)		
		Acceleration	1g (direct installation) (> 9Hz)		
		Direction	3 axial directions		
	Impact (collision)		15 g, 11 ms, half sine wave, 3 axial directions		
	Altitude/Pressure		0-2000m	0-3000m (>70kPa)	

3.1.3 Software performance index specifications

Table 3-3 SC302 Basic Specifications

Item	Rule description					
Programming method	IEC 61131-3 programming language (LD, FBD, IL, ST, SFC, CFC)					
Program execution mode	Compilation					
Power down holding data capacity	1M Bytes					
SD card storage capacity	8G for general SD card (default factory without SD)					
Components and features		Compon ents	Name	Number	Storage characteristics	
					Default	Attribute s can be changed
	I	Input relay	64K Words	Not save	No	X: 1 bit B: 8-bit W: 16-bit D: 32-bit L: 64-bit
	Q	Output relay	64K Words	Not save	No	
	M	Auxiliary relay	240K Words	Save	Yes	
	SM	Special marking	10000 bits	Save	Special use	Special marking

	<table><tr><td>SD</td><td>Special register</td><td>10000 Words</td><td>Save</td><td>Special use</td><td>Special register</td></tr><tr><td colspan="6"></td></tr></table>	SD	Special register	10000 Words	Save	Special use	Special register						
SD	Special register	10000 Words	Save	Special use	Special register								
Program power down mode	EMMC holding (if power down occurs when the power-on time is less than 30s, the power down holding will not be performed.)												

3.1.4 External dimensions

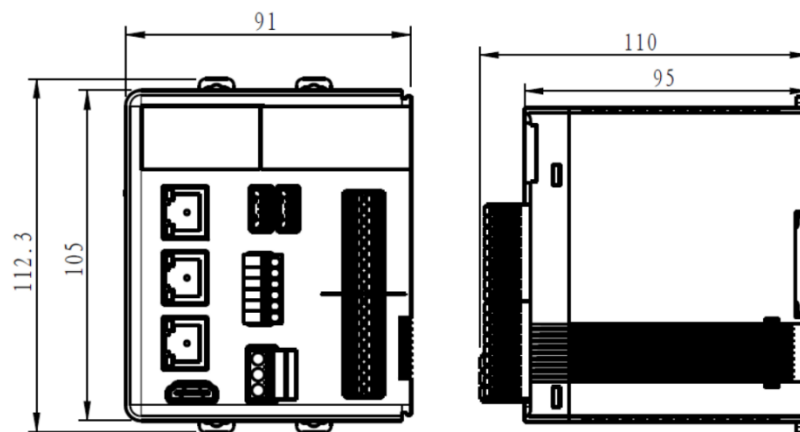


Figure 3-2 SC302 External Dimensions

3.2 Hardware interface details

3.2.1 Digital input

Table 3-4 Basic Specifications

Item	Specifications	
Input connection method	12-point terminal	
Input type	Digital input	
Input mode	NPN or PNP, choose the configuration through COM	
Input channel	8 high speed	4 low speed
Maximum input frequency	200kHz	—
Input voltage level	24V DC (up to 28.8V)	
Input current (typical)	4.22mA	5.3mA

ON voltage	>15V DC	>15V DC
OFF voltage	< 5V DC	< 5V DC
Port filter time	Software configurable	10ms
Input impedance	—	4.3K ohms
Input signal form	Support NPN and PNP input	Support NPN and PNP input
Isolation mode	Optocoupler isolation	Optocoupler isolation
Input display	When the input is in the driving state, the input indicator light is on	When the input is in the driving state, the input indicator light is on

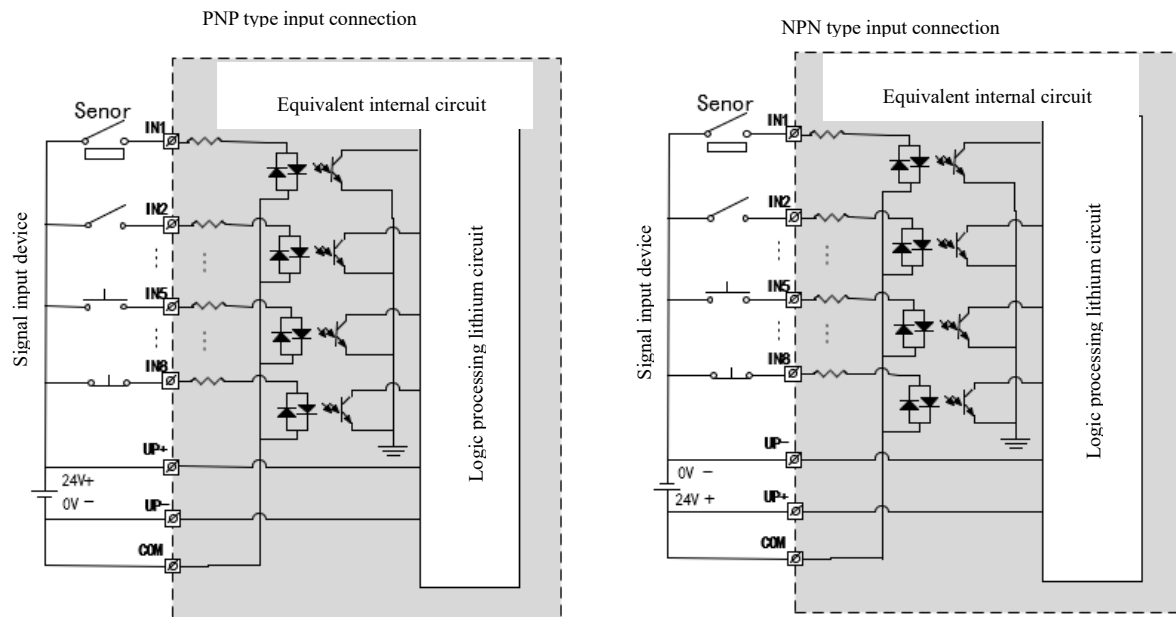


Figure 3-3 Wiring

3.2.2 Digital output

Table 3-5 Basic Specifications

Item	Specifications
Output channel	8
Output connection method	8-point terminal
Output type	Transistor NPN output
Output mode	NPN type
Maximum output frequency	200kHz
Voltage	24V DC (-15% ~ +20%)
Output voltage level	24V DC (-15% ~ +20%)
Maximum output current	500mA
Maximum leakage current at OFF	Under 100uA

Item		Specifications
ON→OFF response time		1 us (hardware response time)
OFF→ON response time		2 us (hardware response time)
Maximum load	Resistance load	0.5A/ point
	Inductive load	7.2W/point
	Lamp load	5W/point
Isolation mode		Digital isolation
Output display		When outputting, the input indicator light is on
Preventing short-circuit output		Restore short circuit protection
Compatibility		Compatible with low speed DO

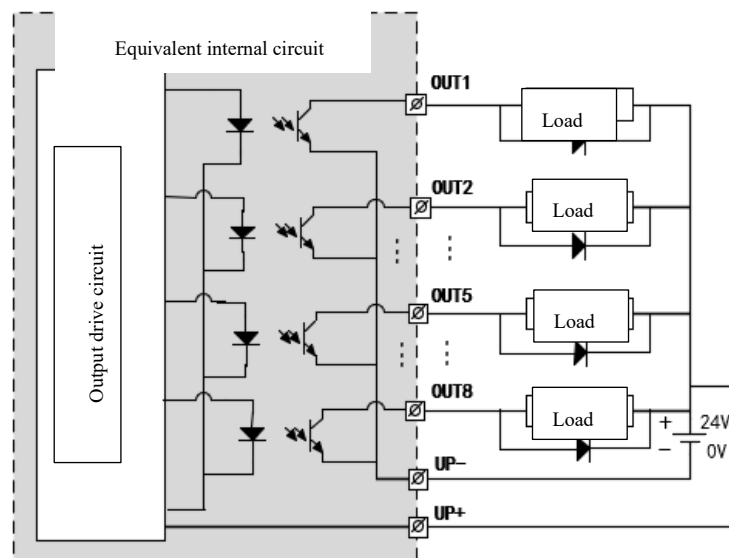


Figure 3-4 Wiring

3.2.3 CAN

Table 3-6 CAN Support Protocol

	NetWork	0
	Baud rate	100/125/250/500/800/1000Kbps
	Supported protocols	CANopen Master or free protocol

Table 3-7 CAN Interface Definition

CAN interface pin definition			
	Pin	Signal name	Function
	CANH	CANH	CAN_H

	CANL	CANL	CAN_L
	CGND	CGND	Grounded
Internal circuit: 120Ω terminal resistance			

CAN bus connection topology is as follows. For CAN bus, it is recommended to use shielded twisted pair cable, and two 120Ω terminal matching resistors are connected at both ends of the bus to prevent signal reflection. The shielding layer generally uses single-point grounding. When fixing cables, don't tie them together with AC power lines and high-voltage cables, so as to avoid interference of communication signals.

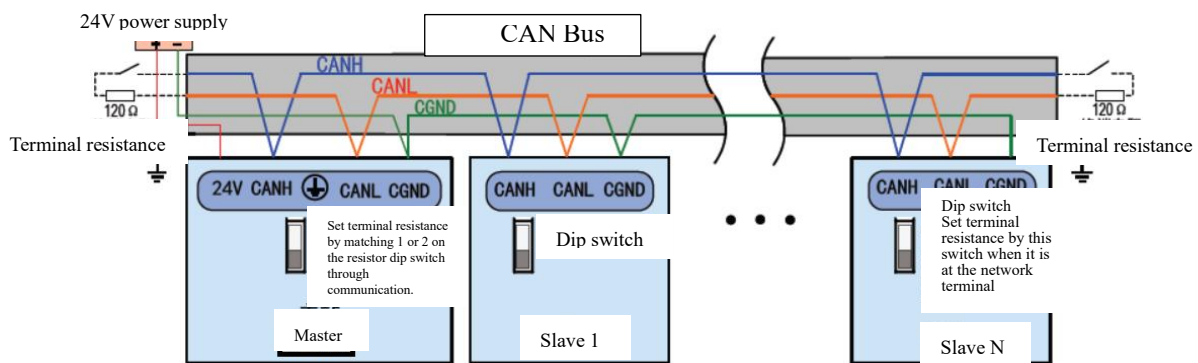


Figure 3-5 CAN Wiring

The relationship between CANopen transmission rate and transmission distance is shown in the following table:

Table 3-8 Relationship between CAN Baud Rate and Distance

Baud rate (bit/s)	Maximum length of bus (m)
1M	30
500k	80
250k	150
125k	300
50k	1000

3.2.4 RS485

Table 3-9 RS485 Support Protocol

	Program port	COM4
	Baud rate	9600-115200
	Data bit	8
	Check bit	None
	Stop bit	1
	Supported protocols	Free protocol, Modbus RTU Slave, Modbus RTU Master

Table 3-10 RS485 Interface Definitions

RS485 interface pin definition			
	Pin	Signal name	Function
	RS485+	A	RS485 differential to positive signal
	RS485-	B	RS485 differential to negative signal
	GND	GND	Power ground
Internal circuit: 120Ω terminal resistance			

RS485 bus connection topology is as follows. For RS485 bus, it is recommended to use shielded twisted pair cable, and two 120Ω terminal matching resistors are connected at both ends of the bus to prevent signal reflection. The shielding layer generally uses single-point grounding.

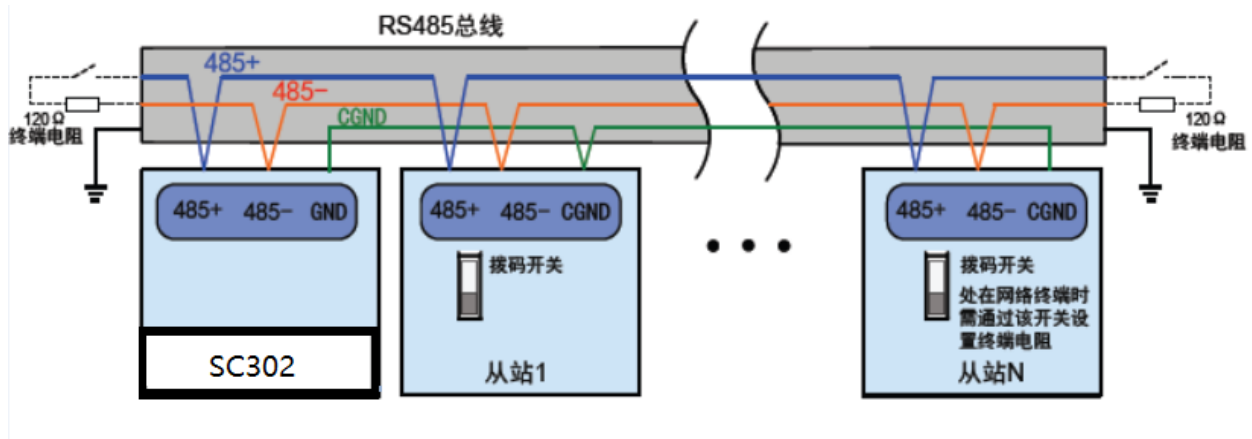


Figure 3-6 RS485 Communication Connection Diagram

- Multi-node connection
- When there are many nodes, the 485 bus must be daisy-chained connection. When branch line connection is needed, the branch length from bus to node should be as

short as possible, and it is recommended not to exceed 3m. Put an end to star connection. The common bus structure is shown below:

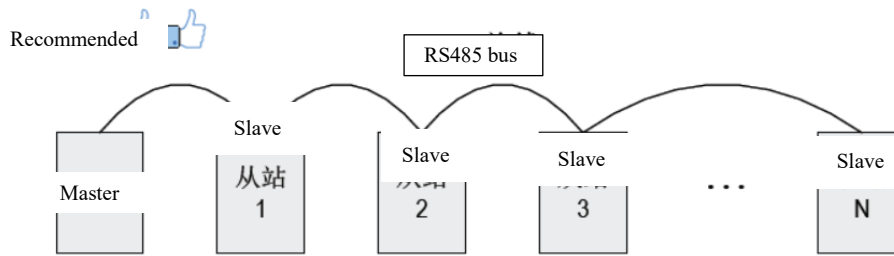


Figure 3-7 Daisy Chain Connection Structure

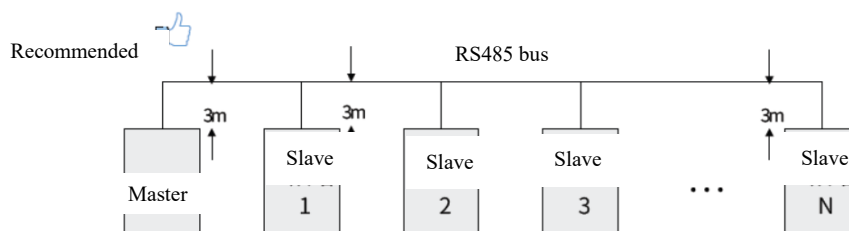


Figure 3-8 When branch line connection is used, the branch line is recommended not to exceed

3m

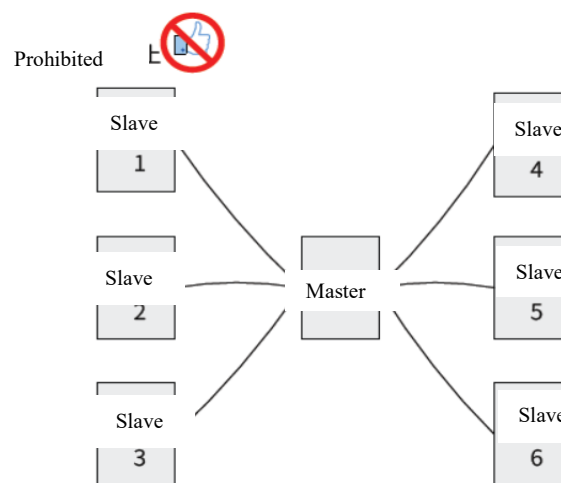


Figure 3-9 Incorrect Star Connection

● Terminal wiring

(1) Nodes with CGND wiring point at the port

Check whether the field 485 bus contains three cables connected with 485+, 485- and CGND terminals, and ensure that the terminals are not connected backwards or wrongly.

When a shielded cable is used, it is especially important to note that the shielding layer

must also be connected to the CGND terminal. In any node or midway position, the shielding layer is forbidden to be connected to any other place (including the field casing and equipment grounding terminal, etc.) except the CGND of the node. Due to the attenuation of cables, it is recommended to use AGW26 or thicker cables for cables with connection length greater than 3m, and it is recommended to use twisted pair cables for 485+ and 485- connection cables at any time.

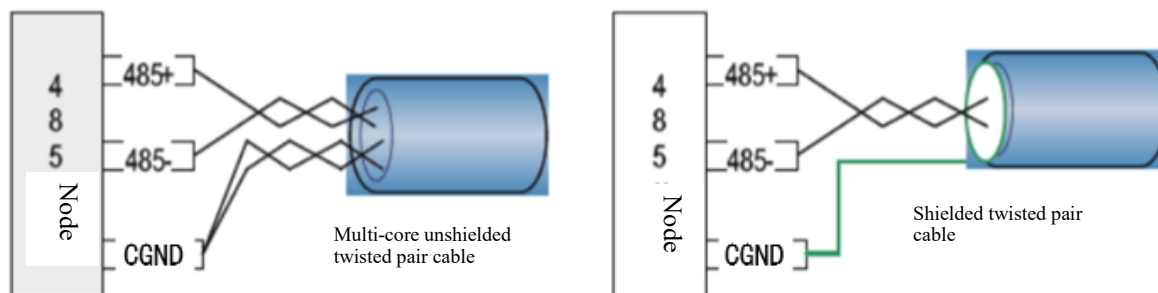


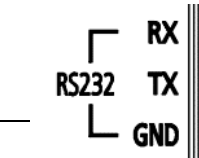
Figure 3-10 RS485 Communication Cable Terminal Wiring Diagram

Recommended wiring cable 1: Multi-core cable with twisted pair cables, in which one pair of twisted pair cables is used as the connecting wire of 485+ and 485-, and the other redundant cables are twisted together as the connecting wire of CGND.

Recommended wiring cable 2: Twisted pair cables with shielding layer. The twisted pair cables are used as the connecting wire of 485+ and 485-, and the shielding layer is used as the connecting wire of CGND. In the case of using shielded cables as connecting cables, it is especially important to note that the shielding layer can only be connected to CGND, but not to the site earth.

3.3.5 RS232

Table 3-23 RS232 Support Protocol

	Port	COM5
	Baud rate	9600-115200
	Data bit	8
	Check bit	None
	Stop bit	1
	Supported protocols	Free protocol, Modbus RTU Slave, Modbus

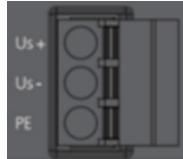
		RTU Master
--	--	------------

Table 3-24 RS232 Interface Definitions

RS232 interface pin definition			
	Pin	Signal name	Function
	RX	RX	Receiving data
	TX	TX	Sending data
	GND	GND	Power ground

3.2.6 24V DC Power Supply

Table 3-11 24V DC Interface Definition

DC24V interface pin definition			
	Pin	Signal name	Function
	Us+	24 DC positive	Power supply positive
	Us-	24 DC negative	Power supply negative
	PE	GND	Grounded

3.2.7 USB

Table 3-12 USB Interface Definition

USB2.0 interface pin definition			
	Pin	Signal name	Function
	1	VCC	Power supply
	2	DATA-	USB 2.0 differential data signal
	3	DATA+	
	4	GND	Power ground

SC302 provides 2 USB ports, both of which support plug-and-play and hot swapping functions; both of them conform to USB 2.0 standard and USB EHCI, Rev.2 standard.

The USB interface supports the following functions:

- (1) Insert the USB disk (only supports NTFS/FAT32 format)

When the U disk is inserted, the system will automatically mount the U disk to the

/udisk directory.

(2) Insert USB to serial port module

When the USB is converted to a serial port module, it will be automatically mapped to the COM6 and COM7 ports. The first inserted module port is COM6 and the second inserted module port is COM7.

3.2.8 EtherCAT

Table 3-13 EtherCAT Network Port Definition

Network card name	Default IP	Supported protocols	LED
eth2	192.168.39.220	TCP, UDP, OPC UA Server, WebVisu Server, Modbus TCP Slave, EtherCAT	Connecting: yellow light is on. Data communication: The yellow light flashes. MByte: Green light is on Gigabit: Red light is on

Table 3-14 Specific EtherCAT Performance Index

Item	Description
Communication protocol	EtherCAT protocol
Supporting service	CoE (PDO/SDO)
Minimum synchronization period	2000us (16 axes)
Synchronization mode	Servo adopts distributed clock (DC); input and output synchronization for IO
Physical layer	1000BASE-TX
Baud rate	1000Mbit/s
Duplex mode	Full duplex
Topological mode	Ring topology
Transmission medium	Network cable
Transmission distance	Less than 100M between bright nodes
Number of slave stations	65535
EtherCAT frame length	44 bytes - 1498 bytes
Process data	A single Ethernet frame has a maximum of 1486 bytes.
Synchronization jitter of two slave stations	<1us
Refresh time	About 30us for 1000 switches input and output (100us for about 32 servos)

- Wirings

How to make network cable

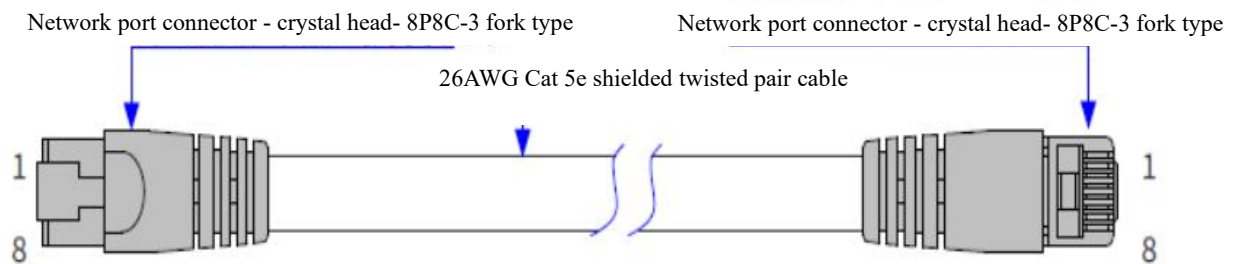


Figure 3-11 Requirements for Making EtherCat Network Cable

Table 3-15 Signal Lead Distribution

Pin	Signal	Signal direction	Signal description
1	TD+	Output	Data transmitting+
2	TD-	Output	Data transmitting-
3	RD+	Input	Data receiving+
4	--	--	No use
5	--	--	No use
6	RD-	Input	Data receiving-
7	--	--	No use
8	--	--	No use

- Length requirements

According to FastEthernet technology, when using EtherCAT bus, the length of cables between devices should not exceed 100 meters, which will attenuate signals and affect normal communication.

- Technical requirements

100% conductivity test; there is no short circuit, open circuit, dislocation and poor contact. The cable length is within the allowable tolerance range.

EtherCAT bus uses cables with shielding layer for network data transmission, and the specific specifications are shown in the following table:

Table 3-16 Specifications

Item	Specification
------	---------------

Cable type	Elastic crossover cable, S-FTP, Cat 5e
Standards	EIA/TIA568A, EN50173, ISO/IEC11801 EIA/TI Abulletin TSB, EIA/TIA SB40-A&TSB36
Cable section	AWG26
Cable type	Twisted pair cables
Pairs	4

3.2.9 SD

Table 3-17 SD Interface

Use	Use Description	Operation description	Remarks
1	SD storage	1. Users can insert the SD card by themselves and store the CODESYS application when the SD is inserted.	The factory default does not come with an SD card;

3.2.10 RESET

Table 3-18 RESET Interface

Use	Use Description	Operation description	Remarks
1	Restore default IP	When the linux system starts normally (the system will start completely for about 20s), short press the RESET key for 5s to restore the IP address. Procedures: Under normal operation, short press the RESET key for 5s, and the red light 1Hz of RUN flashes 5 times; then release the RESET key and the default IP address of the controller will be restored. It will take effect after power-on again. Note: Pay attention to avoid the RESET key jitter or unexpected power-off operation during operation.	Check the factory default IP
2	Normal deletion program	When the CODESYS starts normally, long press RESET key for about 10s to delete the CODESYS application. Procedures: Under normal operation, long press the RESET key for about 10s and the red light 0.5Hz of RUN flashes 5 times; then release the RESET key and the CODESYS application in the controller will be deleted. Note: Pay attention to avoid the RESET key jitter or	If the application causes the runtime to crash, this function may fail.

Use	Use Description	Operation description	Remarks
		unexpected power-off operation during operation.	
3	Force deletion program	Before CODESYS starts normally, long press the RESET key until the green light of RUN flashes, and the CODESYS application will be deleted. Procedures: After the controller is powered on for about 5s, long press the RESET key until the red light 0.5Hz of RUN flashes 5 times; then release the RESET key and the CODESYS application in the controller will be deleted. Note: Pay attention to avoid the RESET key jitter or unexpected power-off operation during operation.	If you encounter a serious CODESYS program error and the runtime crashes, try this method.

Note: Press and hold the RESET key for a long time until the timing begins. If you release, repeat the operation.

3.2.11 RUN/STOP

Table 3-19 RUN/STOP Interface

Direction	Use	Remarks
STOP (down)	CODESYS stopping	Equivalent to the STOP function in CODESYS IDE; if the hardware has been in Stop, you can also click Start to start running under IDE online mode.
RUN (up)	CODESYS running	Equivalent to the Run function in CODESYS IDE; if the hardware has been in RUN, you can also click Stop to stop running under IDE online mode, or debug in separate step

Note: If you turn the RUN/STOP switch for one or more times and then log in to the CODESYS to change the application download, the CODESYS IDE Start/Stop key shall prevail.

Note: If you log in to the CODESYS online and click Start in the IDE, it is still in the STOP state, please check whether the hardware dip switch is in the STOP state.

3.2.12 LED status indicator light

Table 3-20 LED Status Indicator Light

Indicator light name	Colour	Status	Function	Remarks
RUN (2-color light)	Red	Normally on	(1) When the PLC stops, the red light of RUN is normally on (2) Normally on when the data is saved during power down (3) During system startup (hardware default control)	
		Flash 5 times at 1Hz	(1) When the system is started normally, short press Reset for about 5s and the light will flash for 5 times at 1Hz; then release the key and the default IP will be restored	
		Flash 5 times at 0.5Hz	(1) When the system is started normally, long press Reset for about 10s and the light will flash for 5 times at 0.5Hz; then release the key and the CODESYS project will be deleted	
		OFF	PLC is running normally	
	Green	Normally on	Normal operation of PLC (normal operation of CODESYS application); green light is normally on During system startup (hardware default control)	
		1Hz Flashing	There is no CODESYS application; PLC is not running, and the RUN green light flashes at 1Hz	
		OFF	Linux system is not running or CODESYS starts abnormally; light off	
LINK	Green	Normally on	Local extension IO module works normally, and the green light is normally on.	
		OFF	Local extension IO module is abnormal; light off	
485	Green	Normally on	When the 485Tx/RX sends and receives data for normal communication, the green light is always on.	
		1Hz Flashing	485 Tx or RX is abnormal	
		OFF	The 485Tx and RX have no data communication, and the lights are off.	
CAN	Green	Normally on	CAN is running normally, and the green light is normally on.	
		OFF	CAN runs abnormally; light off	
ERR	Red	Normally on	PLC CODESYS fault; PLC stops running; normally on	
		OFF	PLC is running normally, no fault, light off	
FB	Green	Normally on	EtherCAT bus indicator light: Normal - normally on	
		1Hz	Ethercat bus indicator light: abnormal - 1Hz flashing	

Indicator light name	Colour	Status	Function	Remarks
		Flashing	(such as unplug the network cable)	
Us	Green		CPU module power indicator light	
232	Green	Normally on	When the 232Tx/RX sends and receives data for normal communication, the green light is always on.	
		1Hz Flashing	232 Tx or RX is abnormal	
		OFF	The 232Tx and RX have no data communication, and the lights are off.	

3.2.13 PORT1 EtherNET

Table 3-21 Ethernet Port

Network card name	Default IP	Supported protocols	LED
eth1	90.0.0.1 Subnet mask: 255.255.255.0	TCP, UDP, Modbus TCP Slave	Connecting: yellow light is on. Data communication: Green light flashes

Note: When downloading an application project as CODESYS, the downloading speed may be slow due to the large CODESYS application project. It is recommended to use PORT 2 EtherNET for CODESYS program download.

3.2.14 PORT2 EtherNET

Table 3-22 Ethernet Port

Network card name	Default IP	Supported protocols	LED
eth0	192.168.0.1	TCP, UDP, OPC UA Server, WebVisu Server, Modbus TCP Slave	Connecting: yellow light is on. Data communication: The yellow light flashes. MByte: Green light is on Gigabit: Red light is on

Note: This network port is recommended to be used as the download and debugging port for CODESYS application.

3.2.15 RTC Clock

Hardware supports RTC power down saving, and the service life of button cell is 3 years. If it needs to be replaced, please change the battery by yourself. Pull out the left shell of the controller and pull out the battery for replacement, as shown below.



Figure 3-11 Installation Position of Button Cell

3.2.16 Power down holding function

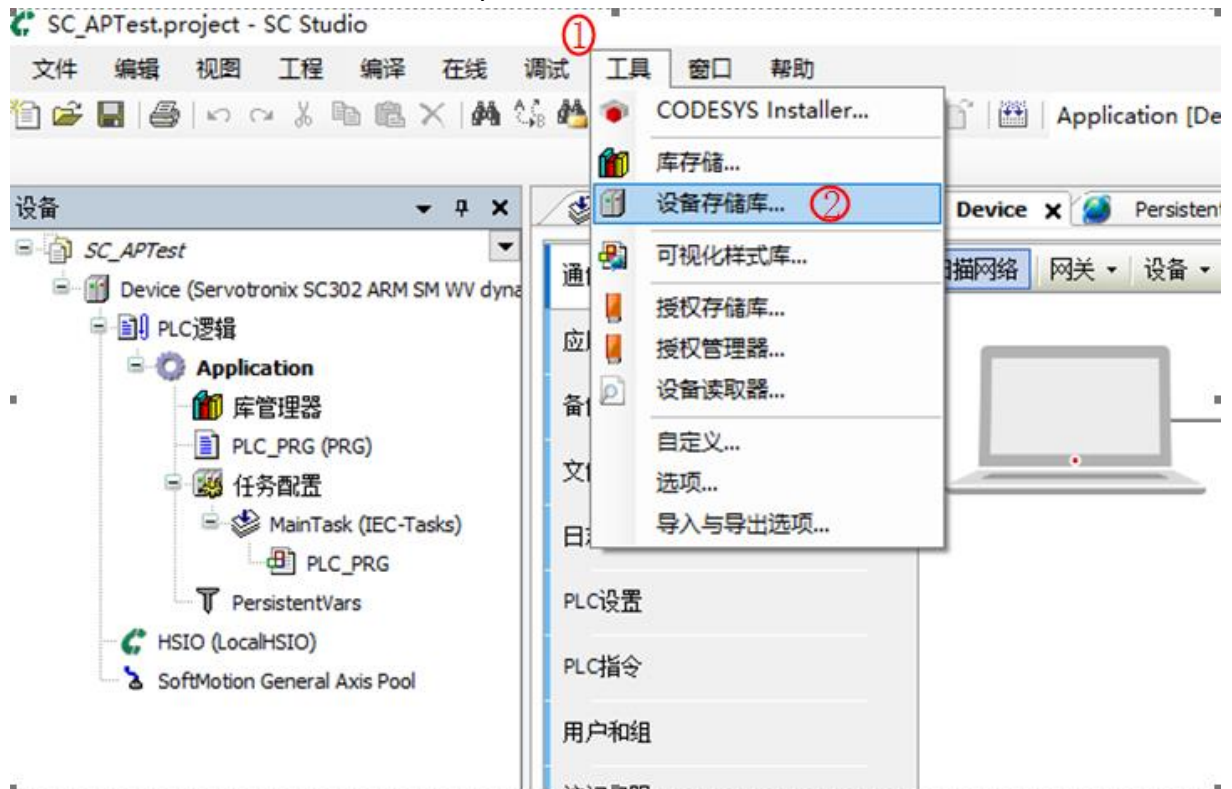
Hardware supports accidental power down data holding, with a maximum data volume of 1 MBytes, without external UPS power supply. Power-on is less than 30S. Due to insufficient charging time, saving data after power-off may fail.

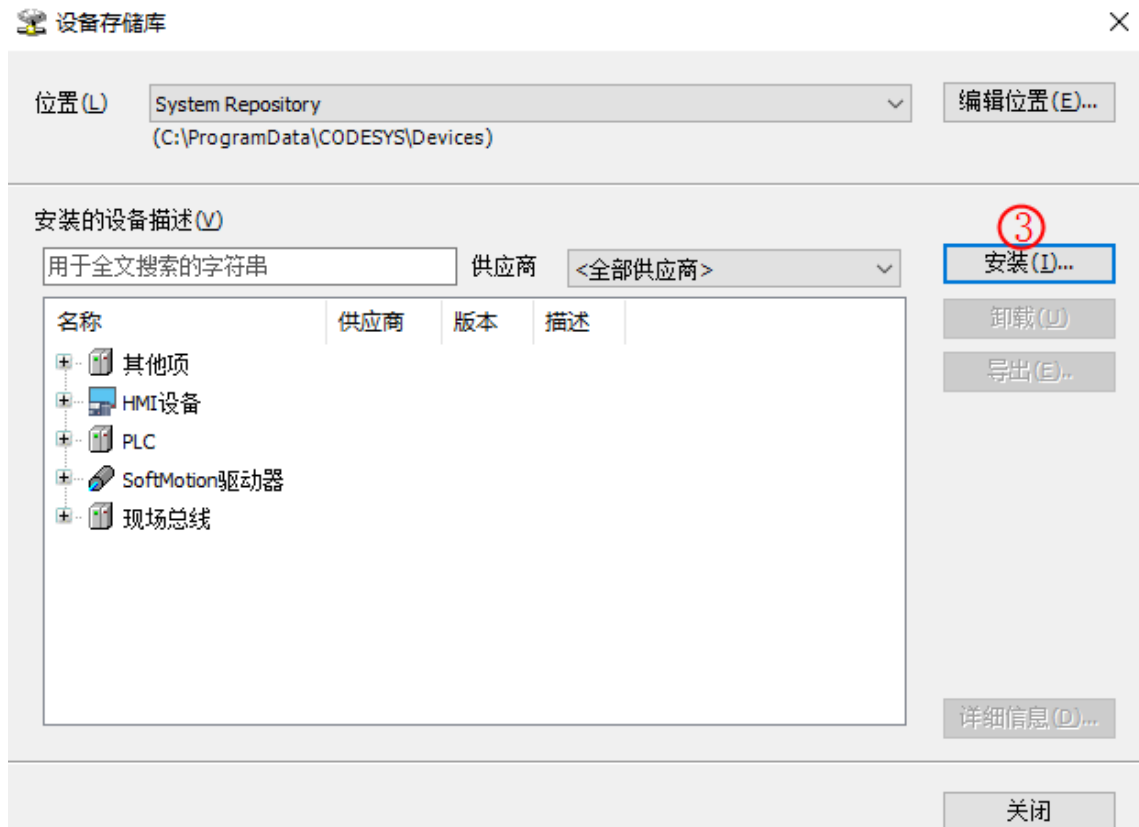
4. Configuring EtherCAT Servo

4.1 Description of Installing SC302 Equipment

SC302 controller description file:

Servotronics SC302 ARM SM WV dynamic V1.0.1.devdesc.xml





4.2 Adding EtherCAT Master softmotion

Right-click and select Device in the device tree; select Add Device, select EtherCAT Master softmotion, and select Add Device.

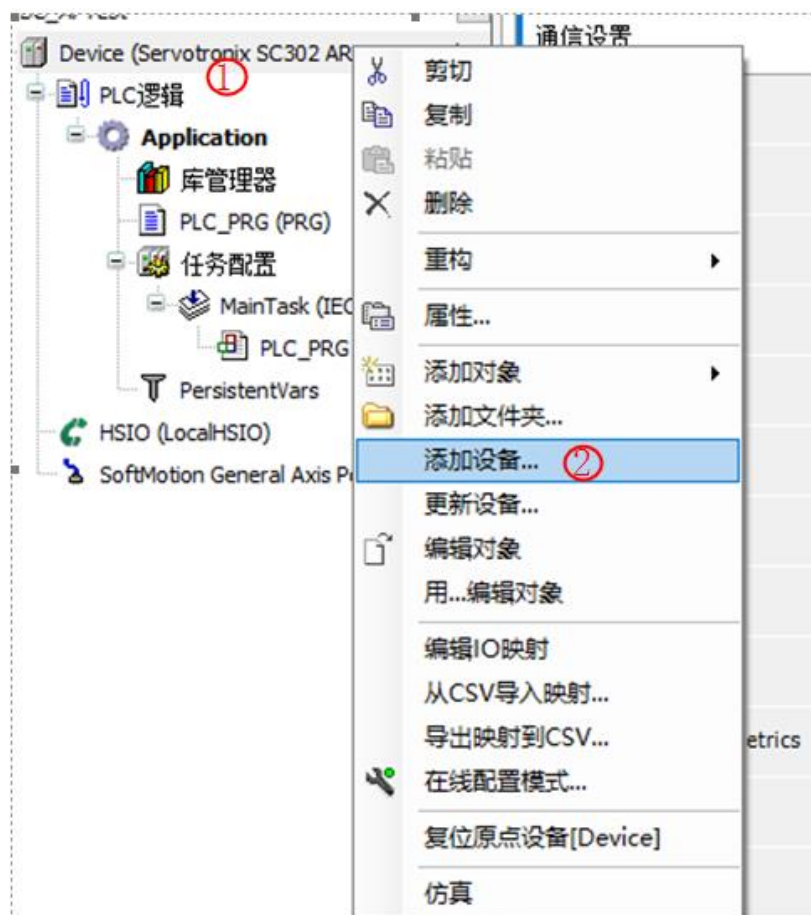


Figure 4-1 Adding EtherCAT Master Softmotion (1)

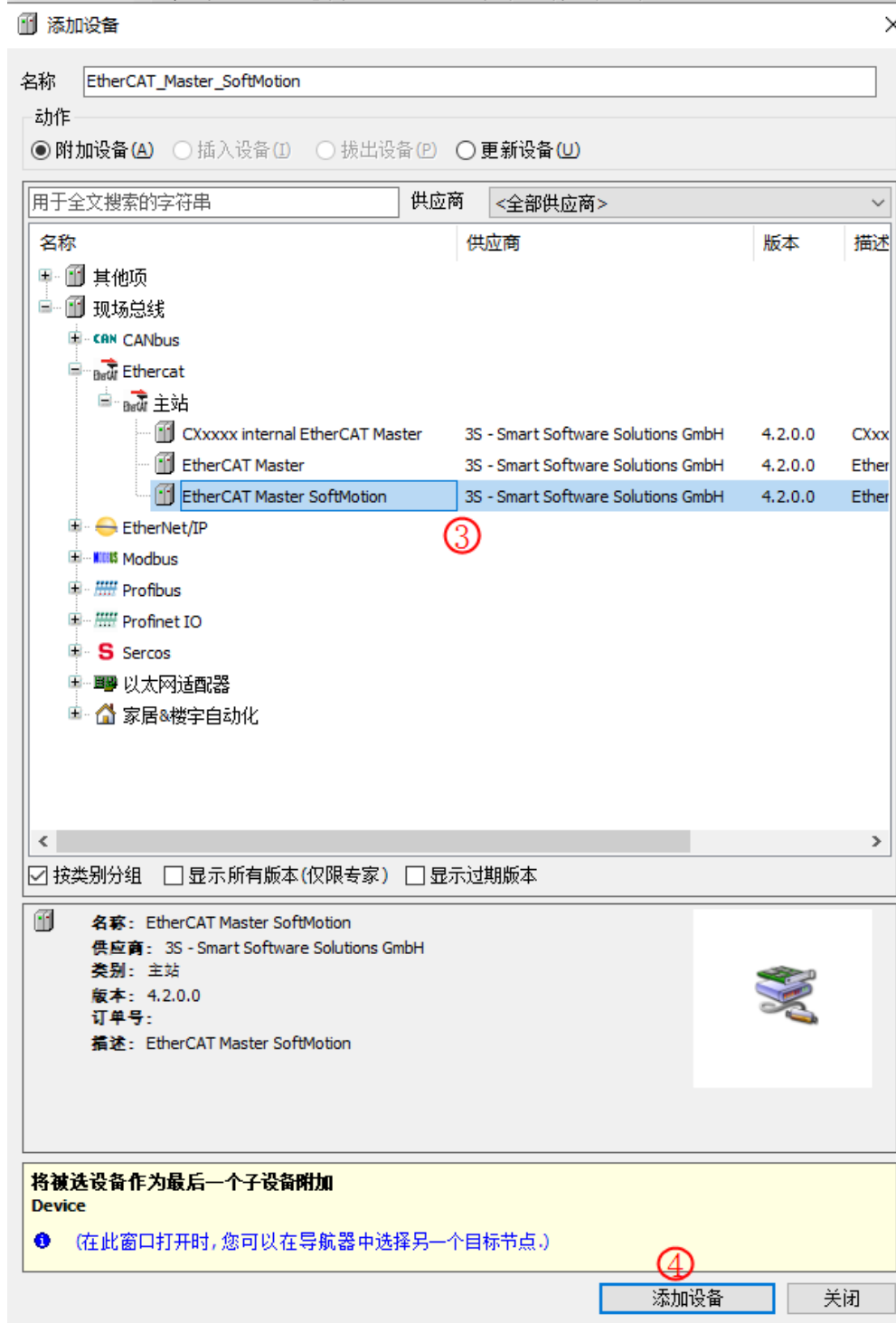


Figure 4-2 Adding EtherCAT Master Softmotion (2)

4.3. Configuring EtherCAT Master

Double-click EtherCAT_Master_Softmotion in the device tree; in the general options, select Select Network by Name, and fill in eth2 for the network name. eth2 is the EtherCAT

network port for connecting EtherCAT devices. Set the EtherCAT bus cycle. Bus cycle: SC302 controller supports various scanning cycles. Users can choose the appropriate bus cycle according to the number of connected slaves.



Figure 4-3 Configuring EtherCAT Network Port

4.4 Scanning devices

Before scanning devices, you need to log in the software first, then right click EtherCAT Master Softmotion in the device tree, and select Scan:

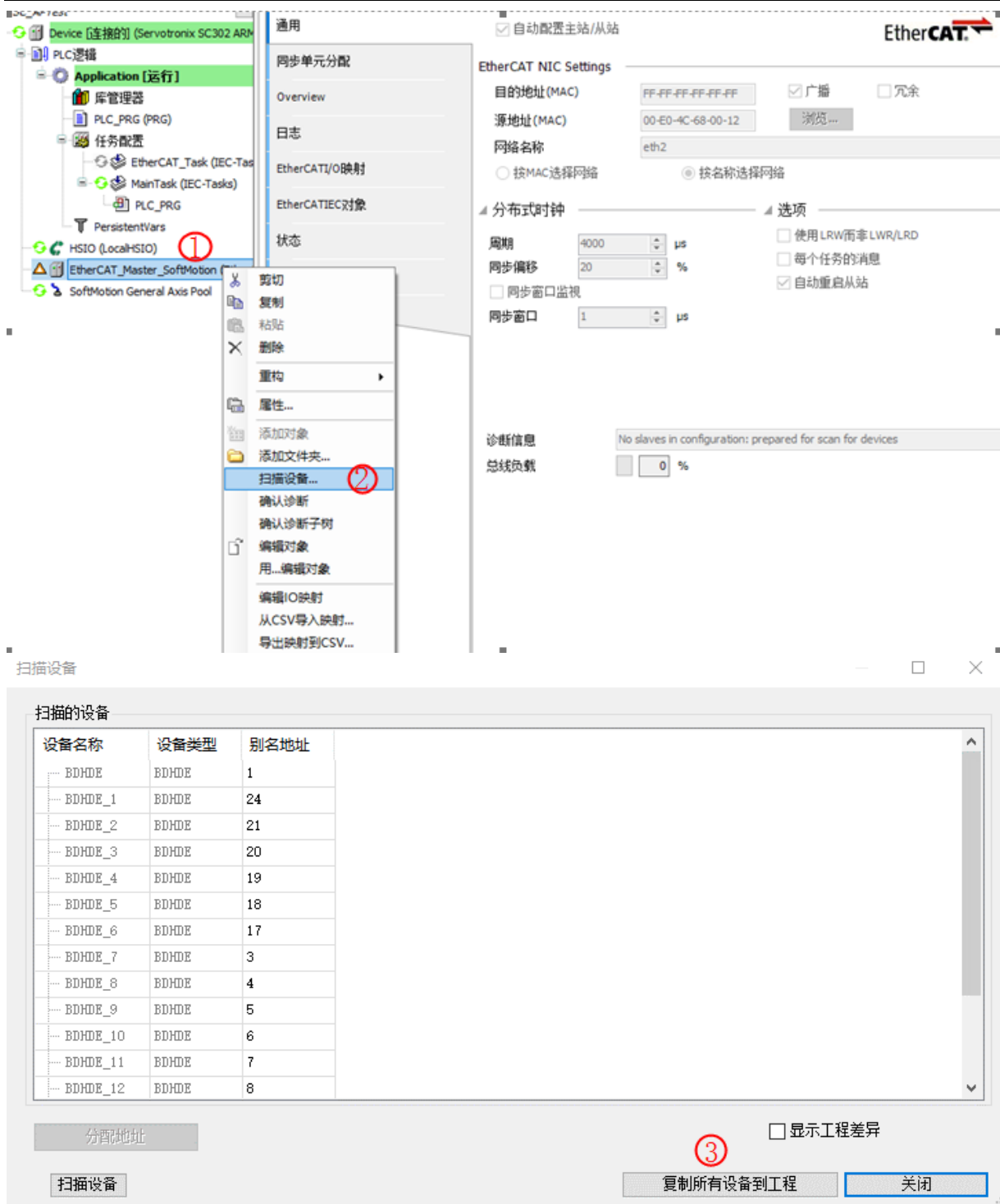


Figure 5-4 Automatic Scanning

In the pop-up interface, select all modules. Click Copy All Devices to Project, and the corresponding modules will appear under therCAT MasterSoftmotion after adding successfully.

4.5 Adding 402 Axis

Right click the servo, then Add SoftMotion CiA402 Axis

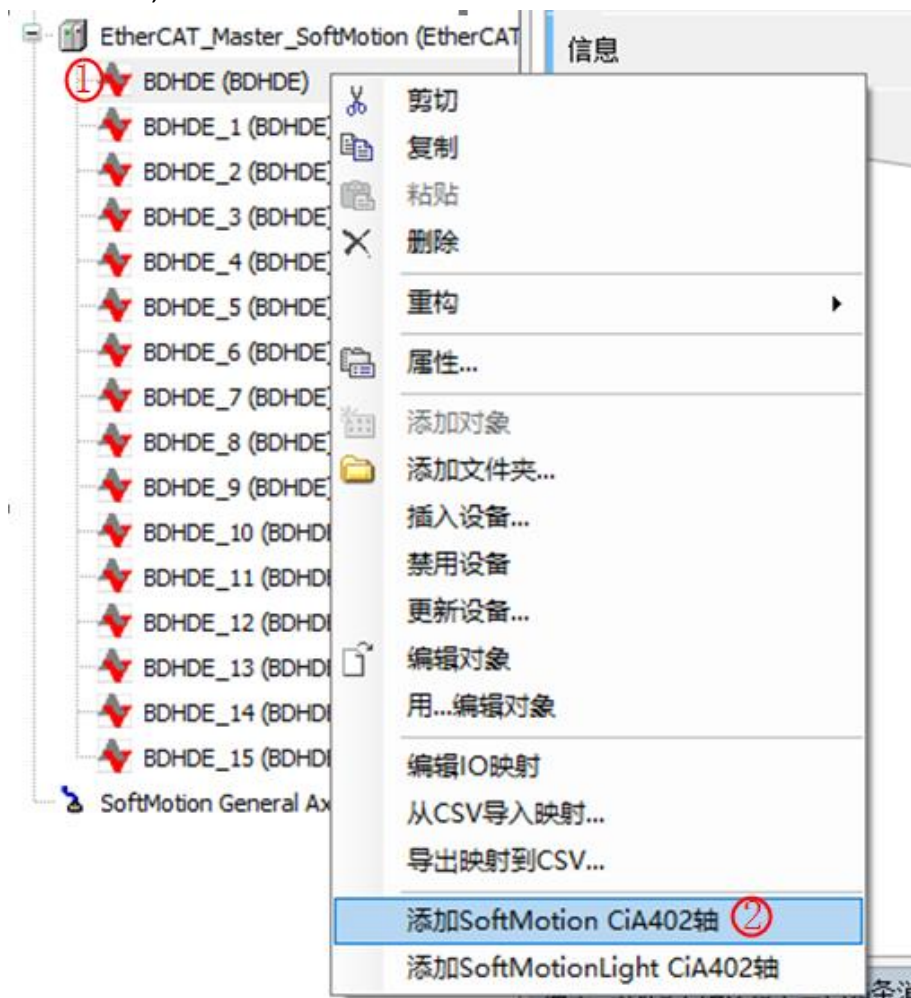


Figure 4-5 Adding 402 Axis

4.6 Configuring axis parameters

Configure the motor resolution according to the actual motor parameters, and configure the reduction ratio according to the actual needs.

SC_API test

Device (Servotronics SC302 ARM SM WV dyn)

PLC逻辑

Application

库管理器

PLC_PRG (PRG)

任务配置

EtherCAT_Task (IEC-Tasks)

MainTask (IEC-Tasks)

PLC_PRG

PersistentVars

HSIO (LocalHSIO)

EtherCAT_Master_SoftMotion (EtherCAT)

BDHDE (BDHDE)

SM_Drive_GenericDSP402 (SM)

BDHDE_1 (BDHDE)

BDHDE_2 (BDHDE)

BDHDE_3 (BDHDE)

BDHDE_4 (BDHDE)

BDHDE_5 (BDHDE)

BDHDE_6 (BDHDE)

BDHDE_7 (BDHDE)

SoftMotion驱动:通用

SoftMotion驱动:缩放/映射

调试

SM_Drive_ETC_GenericDSP402:I/O

映射

SM_Drive_ETC_GenericDSP402:IEC

对象

状态

信息

电机类型

☒ 旋转

☐ 线性

比例缩放

☐ Invert direction

16#10000

increments <=> motor turns

1

1

motor turns <=> gear output turns

1

1

gear output turns <=> units in application

1

映射

☒ 自动映射

输入:

周期性对象	对象数	地址	类型
status word (in.wStatusWord)	16#6041:16#00	'%IW2'	'UINT'
actual position (diActPosition)	16#6064:16#00	'%ID2'	'DINT'
actual velocity (diActVelocity)	16#606C:16#00	'%ID3'	'DINT'
actual torque (wActTorque)	16#6077:16#00	'%IW8'	'INT'
Modes of operation display (OP)	16#6061:16#00	'%IB18'	'SINT'
digital inputs (in.dwDigitalInputs)	16#60FD:16#00	"	"
Touch Probe Status	16#60B9:16#00	'%IW12'	'UINT'
Touch Probe 1 rising edge	16#60BA:16#00	'%ID7'	'DINT'
Touch Probe 1 falling edge	16#60BB:16#00	"	"
Touch Probe 2 rising edge	16#60BC:16#00	"	"
Touch Probe 2 falling edge	16#60BD:16#00	"	"
Following error (A632)	16#60F4:16#00	'%ID5'	'DINT'

输出:

Figure 5-6 Configuring 402 Axis

5. IO Module

5.1 Input/output Introduction

The IO hardware of the controller body comes with 8 high-speed inputs, 4 low-speed inputs, and 8 high-speed outputs. All IOs on the controller can be used as ordinary IO. In high-speed input and output pulse control mode, it supports a maximum pulse frequency of 200K.

In the figure, the green area is the high-speed input and output terminal, and the blue area is the low-speed input terminal. IN1-1 and IN1-2 are phase AB input terminals; OUT1-1 and OUT1-2 are phase AB output terminals.

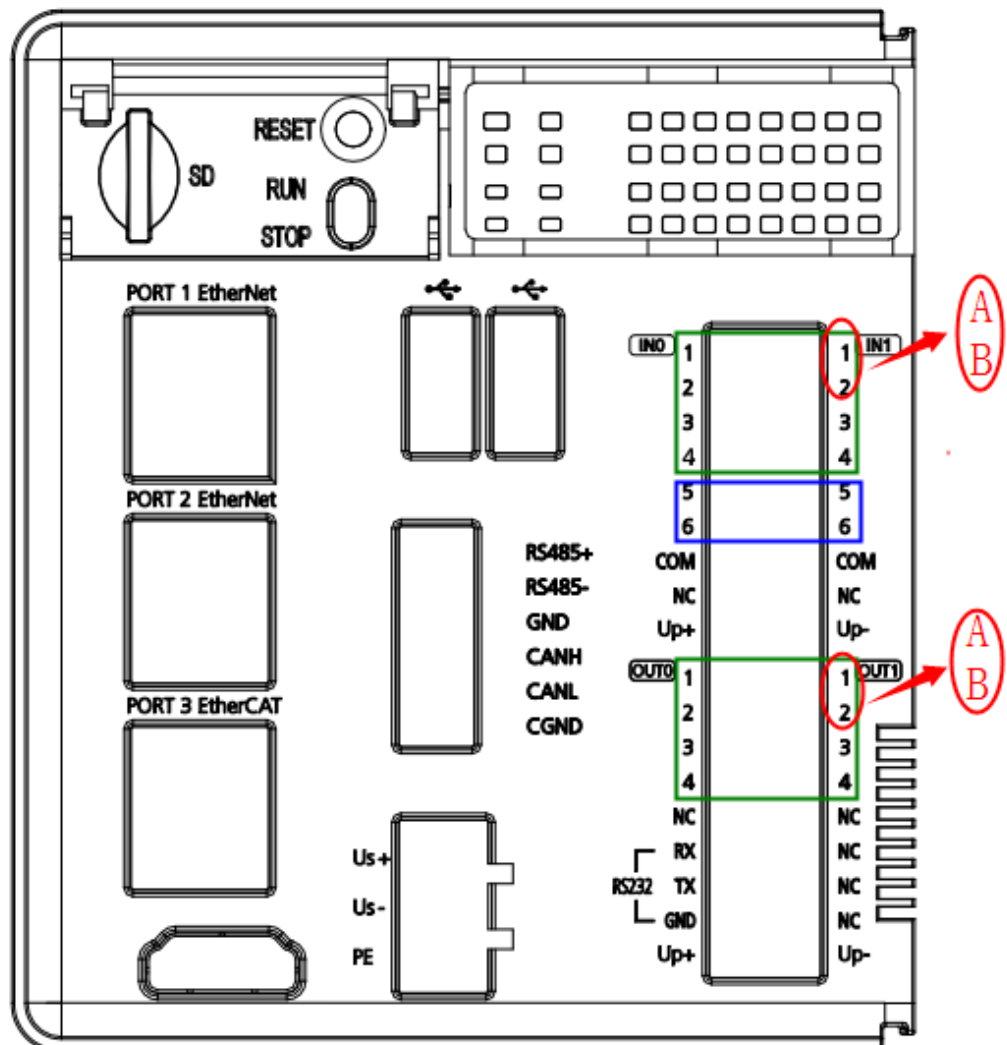


Figure 5-1-1 Appearance Diagram of Main Body IO

- Main Body IO channel definition

Table 5-1-2 Definition of Input Terminals on the Main Body

Definition	IN0 terminal		Definition	IN1 terminal
High speed input	IN0-1		High speed input	IN1.1
High speed input	IN0-2		High speed input	IN1.2
High speed input	IN0.3		High speed input	IN1.3
High speed input	IN0.4		High speed input	IN1.4
Low speed input	IN0.5		Low speed input	IN1.5
Low speed input	IN0.6		Low speed input	IN1.6
Common terminal	COM		Common terminal	COM
Reserved	NC		Reserved	NC
24V	UP+		0V	UP-

Table 5-1-3 Definition of Output Terminals on the Main Body

Definition	OUT0 terminal		Definition	OUT1 terminal
High speed output	OUT0.1		High speed output	OUT1.1
High speed output	OUT0.1		High speed output	OUT1.2
High speed output	OUT0.3		High speed output	OUT1.3
High speed output	OUT0.4		High speed output	OUT1.4
Reserved	NC		Reserved	NC
Reserved	NC		Reserved	NC
Reserved	NC		Reserved	NC
Reserved	NC		Reserved	NC
24V	UP+		0V	UP-

- High-speed input channel definition

Table 5-1-4 High Speed Input Channel Definition

Model	Counter 0	Counter 1	Counter 2	Counter 3	Counter 4	Counter 5	Counter 6	Counter 7
Single phase counting	IN0-1 pulse	IN0-2 pulse	IN0-3 pulse	IN0-4 pulse	IN1-1 pulse	IN1-2 pulse	IN1-3 pulse	IN1-4 pulse
Pulse+dire ction	IN0-1 pulse IN0-2 direction	--	IN0-3 pulse IN0-4 direction	--	IN1-1 pulse IN1-2 direction	--	IN1-3 pulse IN1-4 direction	-- --
Phase AB counting	IN0-1 phase A IN0-2 phase B	--	IN0-3 phase A IN0-4 phase B	--	IN1-1 phase A IN1-2 phase B	--	IN1-3 phase A IN1-4 phase B	--
Phase AB double frequency	IN0-1 phase A IN0-2 phase B	--	IN0-3 phase A IN0-4 phase B	--	IN1-1 phase A IN1-2 phase B	--	IN1-3 phase A IN1-4 phase B	--
Phase AB quadruple frequency	IN0.1 phase A IN0-2 phase B	--	IN0-3 phase A IN0-4 phase B	--	IN1-1 phase A IN1-2 phase B	--	IN1-3 phase A IN1-4 phase B	--
CW/CCW	IN0-1 CW IN0-2 CCW	--	IN0-3 CW IN0-4 CCW	--	IN1-1 CW IN1-2 CCW	--	IN1-3 CW IN1-4 CCW	--

Special note: Counter 1, Counter 3, Counter 5, and Counter 7 only support single-phase counting mode;

[Example]

If counter 0 is set to pulse + direction, the IN0.1 terminal of the controller should be connected to the external pulse signal, and the IN0.2 terminal should be connected to the external direction signal, and counter 1 will not be enabled at this time; if counter 4 is set to AB , then the IN1.1 terminal of the controller should be connected to the external phase A signal, and the IN1.2 terminal should be connected to the external phase B signal, and counter 5 will not be enabled at this time.

- High-speed output channel definition

Table 5-1-5 High Speed Output Channel Definition

Model	Axis0	Axis1	Axis2	Axis3	Axis4	Axis5	Axis6	Axis7
Single phase output	OUT0-1 Pulse	OUT0-2 Pulse	OUT0-3 Pulse	OUT0-4 Pulse	OUT1-1 Pulse	OUT1-2 Pulse	OUT1-3 Pulse	OUT1-4 Pulse
Pulse+direction	OUT0-1 pulse OUT0-2 direction	--	OUT0-3 pulse OUT0-4 direction	--	OUT1-1 pulse OUT1-2 direction	--	OUT1-3 pulse OUT1-4 direction	-- --
Phase AB output	OUT0-1 phase A OUT0-2 phase B	--	OUT0-3 phase A OUT0-4 phase B	--	OUT1-1 phase A OUT1-2 phase B	--	OUT1-3 phase A OUT1-4 phase B	--
CW/CCW	OUT0-1 CW OUT0-2 CCW	--	OUT0-3 CW OUT0-4 CCW	--	OUT1-1 CW OUT1-2 CCW	--	OUT1-3 CW OUT1-4 CCW	--
PWM	OUT0-1	OUT0-2	OUT0-3	OUT0-4	OUT1-1	OUT1-2	OUT1-3	OUT1-4

Special note: Axis1, Axis3, Axis5, and Axis7 only support single-phase pulse mode.

[Example]

If Axis0 is set to pulse + direction, the OUT0-1 terminal of the controller should be connected to the external pulse signal, and the OUT0-2 terminal should be connected to the external direction signal, and Axis1 will not be enabled at this time; if Axis4 is set to AB, the control The OUT1-1 terminal of the device should be connected to the external phase A signal, and the OUT1-2 terminal should be connected to the external phase B signal, and Axis5 will not be enabled at this time.

- Output port mode settings

PCI-BusIEC对象	参数	类型	值	默认值	单元	描述
HSIO参数	PortInFilter					输入端口滤波时间配置
HSIOI/O映射	InterruptCMD					输入中断类型配置
状态	PortFunc:PortOutMode					输出端口模式配置
信息	OUT0-1-Mode	Enumeration of BYTE	GeneralOut	GeneralOut		脉冲输出索引: 0
	OUT0-2-Mode	Enumeration of BYTE	GeneralOut	GeneralOut		脉冲输出索引: 1
	OUT0-3-Mode	Enumeration of BYTE	PWM	GeneralOut		脉冲输出索引: 2
	OUT0-4-Mode	Enumeration of BYTE	CW/CCW输出	GeneralOut		脉冲输出索引: 3
	OUT1-1-Mode	Enumeration of BYTE	AB相1倍频输出	GeneralOut		脉冲输出索引: 4
	OUT1-2-Mode	Enumeration of BYTE	脉冲+方向输出	GeneralOut		脉冲输出索引: 5
	OUT1-3-Mode	Enumeration of BYTE	单相输出	GeneralOut		脉冲输出索引: 6
	OUT1-4-Mode	Enumeration of BYTE	Cnt0CmpOut	GeneralOut		脉冲输出索引: 7
	Counter		Cnt1CmpOut			计数器配置
	Axis		Cnt2CmpOut			轴参数配置
			Cnt3CmpOut			
			Cnt4CmpOut			
			Cnt5CmpOut			
			Cnt6CmpOut			
			Cnt7CmpOut			

Figure 5-1-6 Output Port Mode Configuration

[Example]

If Axis0 is set to pulse + direction, the controller OUT0-1-Mode should be set to pulse + direction. The OUT0-2-Mode configuration is invalid (occupied by the direction of OUT0-1-Mode) and does not need to be configured. The external physical wiring mode: OUT0-1 terminal should be connected to the external pulse signal; OUT0-2 terminal should be connected to the external direction signal, and Axis1 will not be enabled at this time;

If Axis4 is set to AB, the controller OUT1-1-Mode should be set to phase AB 1 frequency output. The OUT1-2-Mode configuration is invalid (occupied by the phase B of OUT1-1-Mode) and does not need to be configured. At this time, the external physical wiring method is as follows: The OUT1-1 terminal should be connected to the external phase A signal, and the OUT1-2 terminal should be connected to the external phase B signal, and Axis5 will not be enabled at this time.

5.2 Installation Equipment Description and library File

For the description file of the installed IoDrvHSIO V1.0.0.4.devdesc.xml and above, please refer to the device description file of the 4.1 controller according to the method.

Install the high-speed IO library file

CmpHSIO.compiled-library-V1.0.0.8.compiled-library and above. The installation method of the library is as follows:

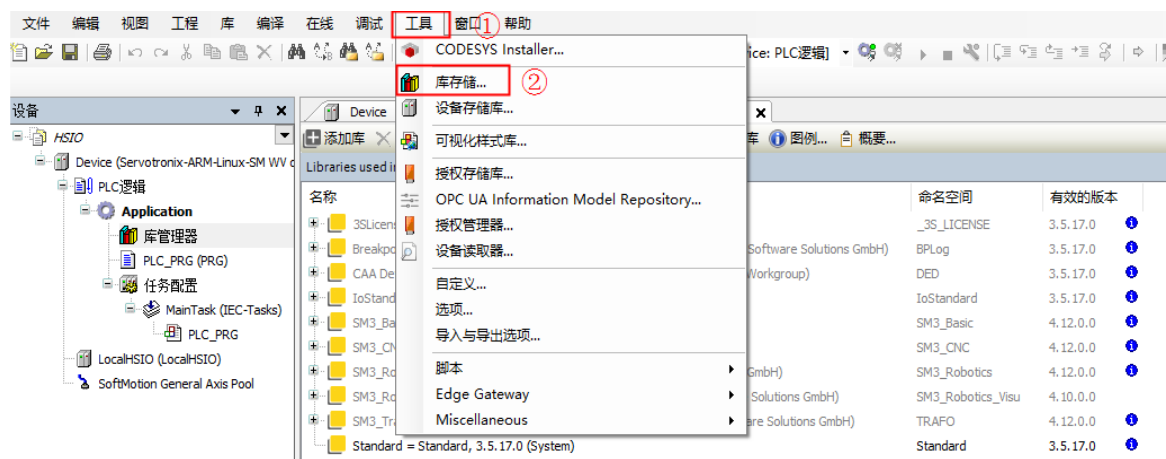


Figure 5-2-1 Adding High Speed IO External Library

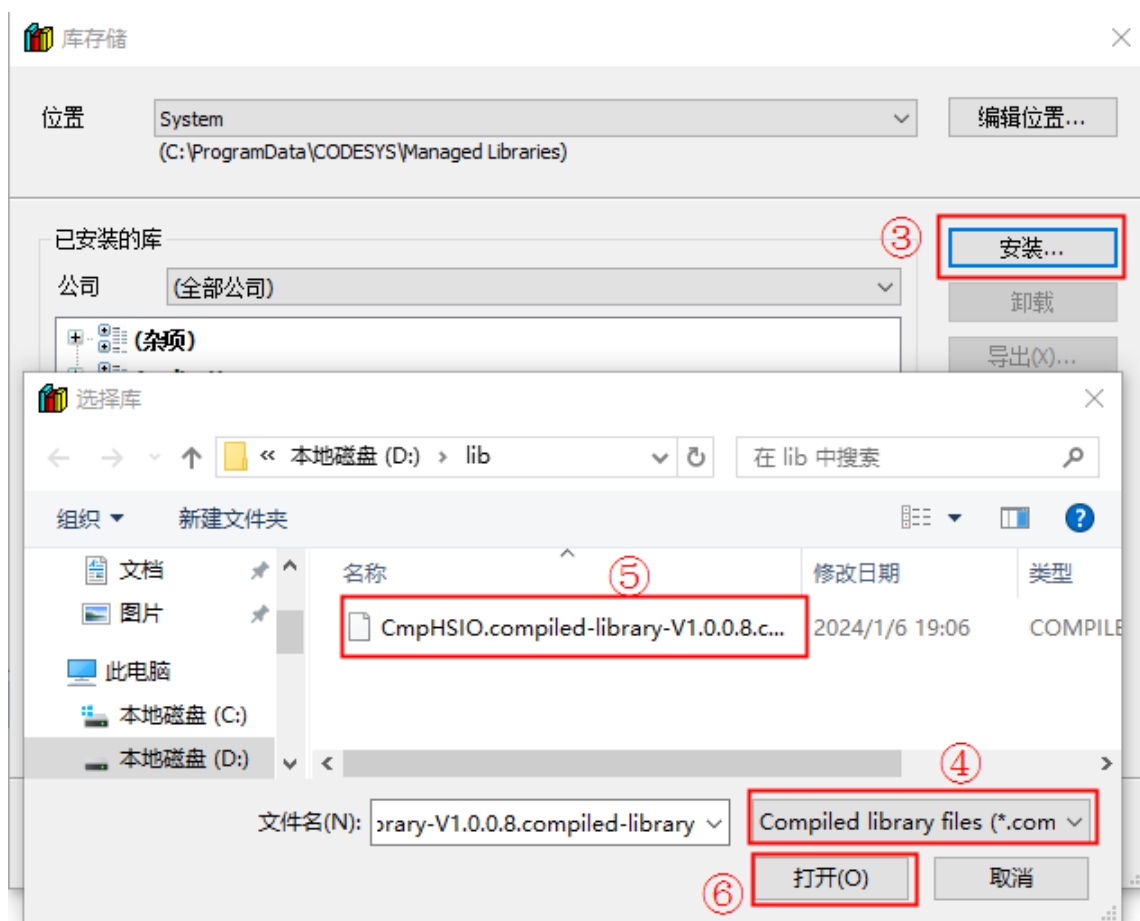


Figure 5-2-2 Adding High Speed IO External Library

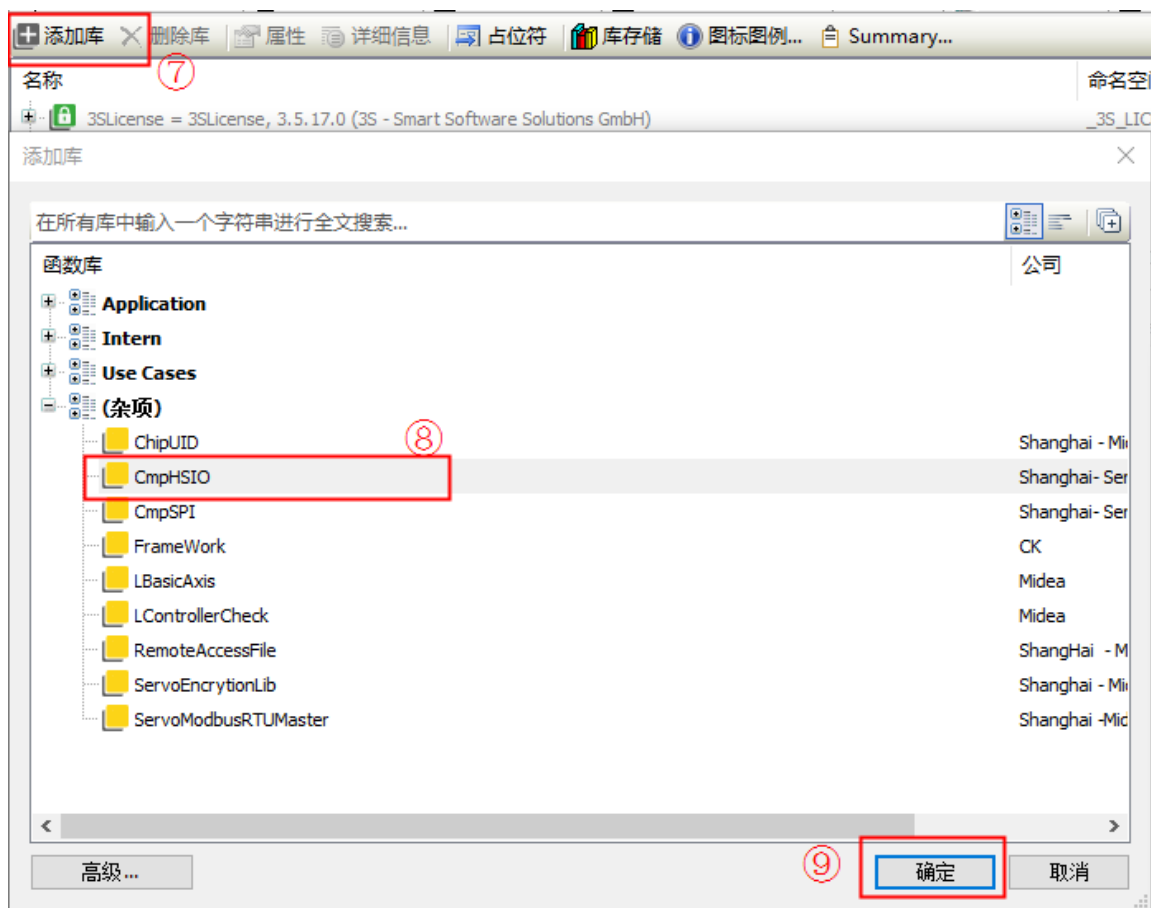


Figure 5-2-3 Adding High Speed IO External Library

5.3 Adding LocalHSIO Device

Right click and select Device (Servotronix-ARM-Linux SM EtherCAT) in the device tree; select Add Device, select LocalHSIO, and select Add Device.

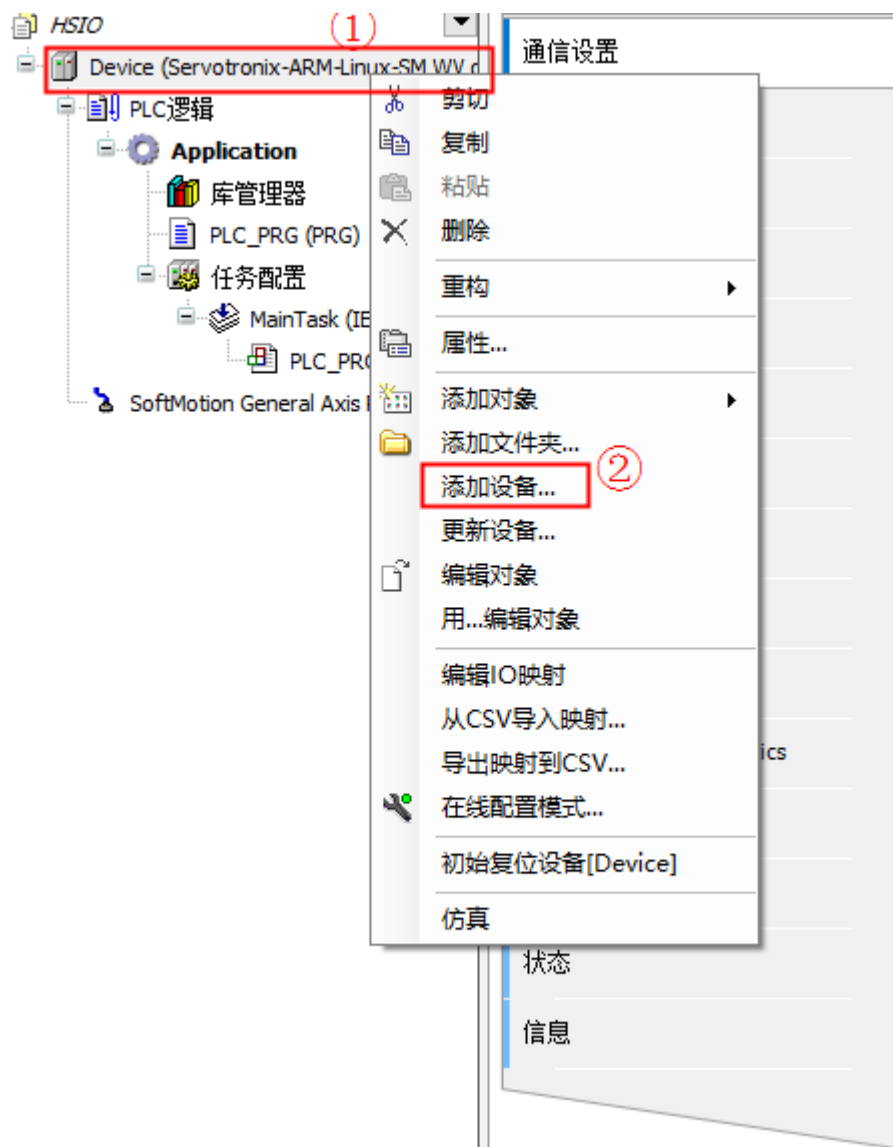


Figure 5-3-1 Adding LocalHSIO Device (1)

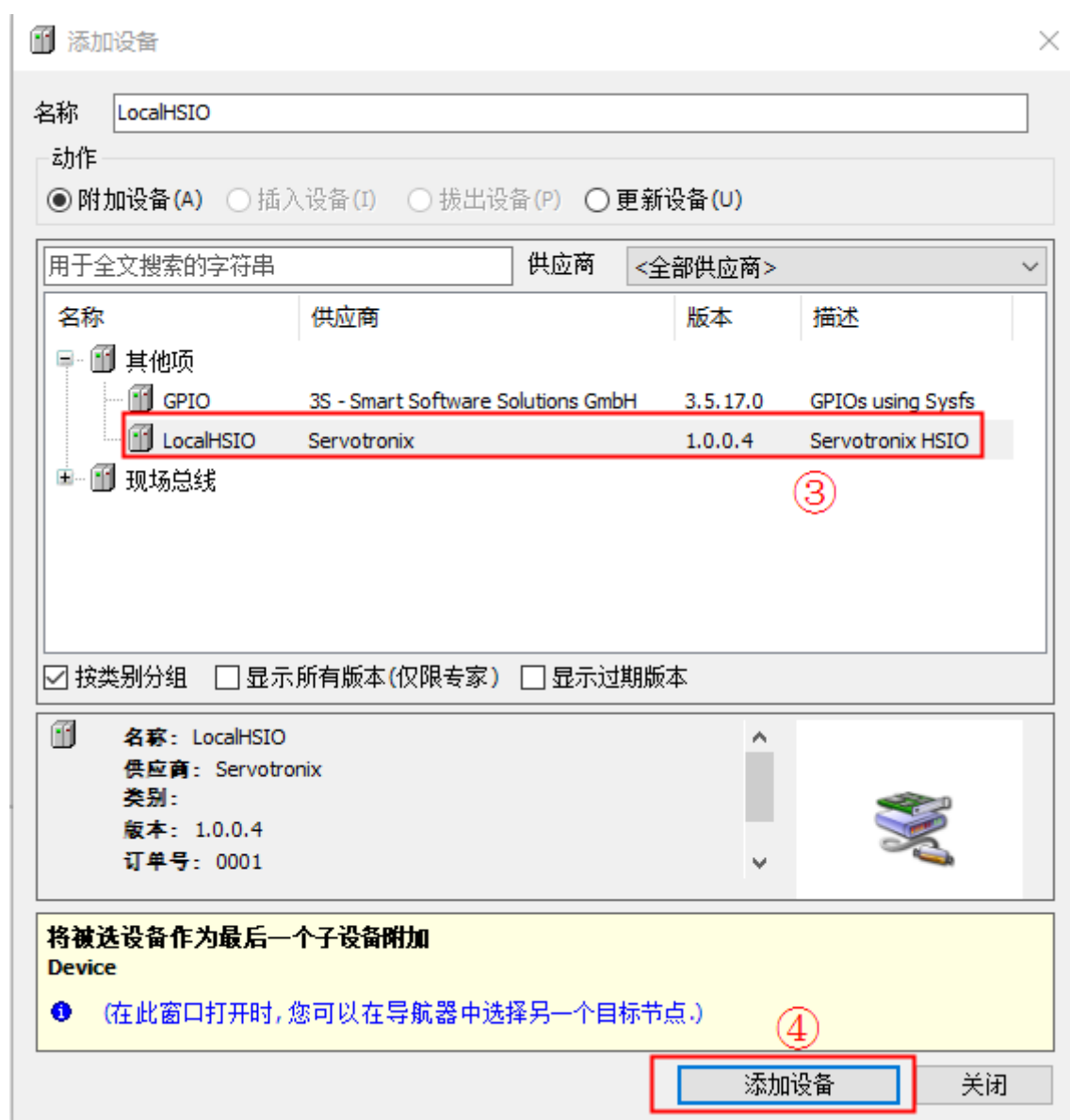


Figure 5-3-2 Adding LocalHSIO Device (2)

5.4 Configuring LocalHSIO Device

Double-click LocalHSIO in the device tree to configure HSIO Parameters. The parameters include input port filter time, input interrupt type configuration, output port configuration, counter configuration, and axis parameter configuration.

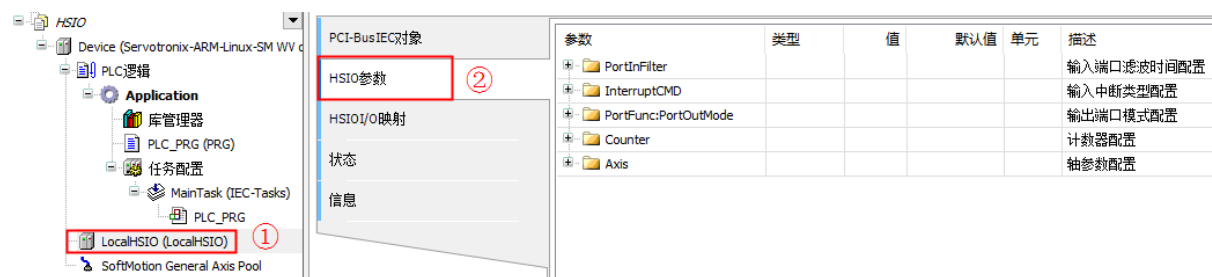


Figure 5-4-1 Configuring LocalHSIO device

● PortInFilter input port filtering time configuration

参数	类型	值	默认值	单元	描述
PortInFilter					输入端口滤波时间配置
IN0-1-filter	DWORD	2	3 us		
IN0-2-filter	DWORD	2	3 us		
IN0-3-filter	DWORD	2	3 us		
IN0-4-filter	DWORD	2	3 us		
IN0-5-filter	DWORD	500	100 us		
IN0-6-filter	DWORD	500	100 us		
IN1-1-filter	DWORD	2	3 us		
IN1-2-filter	DWORD	2	3 us		
IN1-3-filter	DWORD	2	3 us		
IN1-4-filter	DWORD	2	3 us		
IN1-5-filter	DWORD	500	100 us		
IN1-6-filter	DWORD	500	100 us		

Figure 5-4-2 Input Port Filtering Time Configuration

The filter time configuration of the 12 input ports IN0~IN1 of the main body can be set by entering the value in the 'Value' item on the page. The filter time supports a maximum of 64000us.

● InterruptCMD input interrupt trigger type configuration

参数	类型	值	默认值	单元	描述
PortInFilter					输入端口滤波时间配置
InterruptCMD					输入中断类型配置
IN0-1-Interrupt	Enumeration of BYTE	Disable	FallingEdge		
IN0-2-Interrupt	Enumeration of BYTE	RisingEdge	FallingEdge		
IN0-3-Interrupt	Enumeration of BYTE	FallingEdge	FallingEdge		
IN0-4-Interrupt	Enumeration of BYTE	DoubleEdge	FallingEdge		
IN1-1-Interrupt	Enumeration of BYTE	Disable	FallingEdge		
IN1-2-Interrupt	Enumeration of BYTE	FallingEdge	FallingEdge		
IN1-3-Interrupt	Enumeration of BYTE	FallingEdge	FallingEdge		
IN1-4-Interrupt	Enumeration of BYTE	FallingEdge	FallingEdge		
PortFunc:PortOutMode					输出端口模式配置
Counter					计数器配置
Axis					轴参数配置

Figure 5-4-3 Input Interrupt Trigger Type Configuration

The interrupt trigger type configuration page for high speed input ports does not support the interrupt type configuration for low speed input ports.

When the input port is occupied by the counter function, the interrupt type configuration is invalid. The interrupt type configuration of the corresponding port will only take effect when the corresponding counter is inactive or set to level sampling.

There are four ways to set the input interrupt signal trigger mode:

- FallingEdge falling edge trigger;
- RisingEdge rising edge trigger;
- DoubleEdge double edge trigger;
- Disable off interrupts trigger function.

● PortOutMode output port mode configuration

参数	类型	值	默认值	单元	描述
PortInFilter					输入端口滤波时间配置
InterruptCMD					输入中断类型配置
PortFunc:PortOutMode					输出端口模式配置
OUT0-1-Mode	Enumeration of BYTE	GeneralOut	GeneralOut		脉冲输出索引: 0
OUT0-2-Mode	Enumeration of BYTE	GeneralOut	GeneralOut		脉冲输出索引: 1
OUT0-3-Mode	Enumeration of BYTE	PWM	GeneralOut		脉冲输出索引: 2
OUT0-4-Mode	Enumeration of BYTE	CW/CCW输出	GeneralOut		脉冲输出索引: 3
OUT1-1-Mode	Enumeration of BYTE	AB相1倍频输出	GeneralOut		脉冲输出索引: 4
OUT1-2-Mode	Enumeration of BYTE	脉冲+方向输出	GeneralOut		脉冲输出索引: 5
OUT1-3-Mode	Enumeration of BYTE	单相输出	GeneralOut		脉冲输出索引: 6
OUT1-4-Mode	Enumeration of BYTE	Cnt0CmpOut	GeneralOut		脉冲输出索引: 7
Counter		Cnt1CmpOut			计数器配置
Axis		Cnt2CmpOut			轴参数配置
		Cnt3CmpOut			
		Cnt4CmpOut			
		Cnt5CmpOut			
		Cnt6CmpOut			
		Cnt7CmpOut			

Figure 5-4-4 Output Port Mode Configuration

Table 5-4-5 Port Mode Introduction

Port mode type	Description
GeneralOut	Ordinary output, level output mode
PWM	Pulse frequency output type
CW/CCW output	CW/CCW output type
Phase AB single frequency output	Phase AB output mode
Pulse+direction	Pulse signal and direction signal output mode
Single phase output	Single-phase pulse signal output mode

Cnt0CmpOut	Comparison output port of counter 0
Cnt1CmpOut	Comparison output port of counter 1
Cnt2CmpOut	Comparison output port of counter 2
Cnt3CmpOut	Comparison output port of counter 3
Cnt4CmpOut	Comparison output port of counter 4
Cnt5CmpOut	Comparison output port of counter 5
Cnt6CmpOut	Comparison output port of counter 6
Cnt7CmpOut	Comparison output port of counter 7

● Counter counter configuration

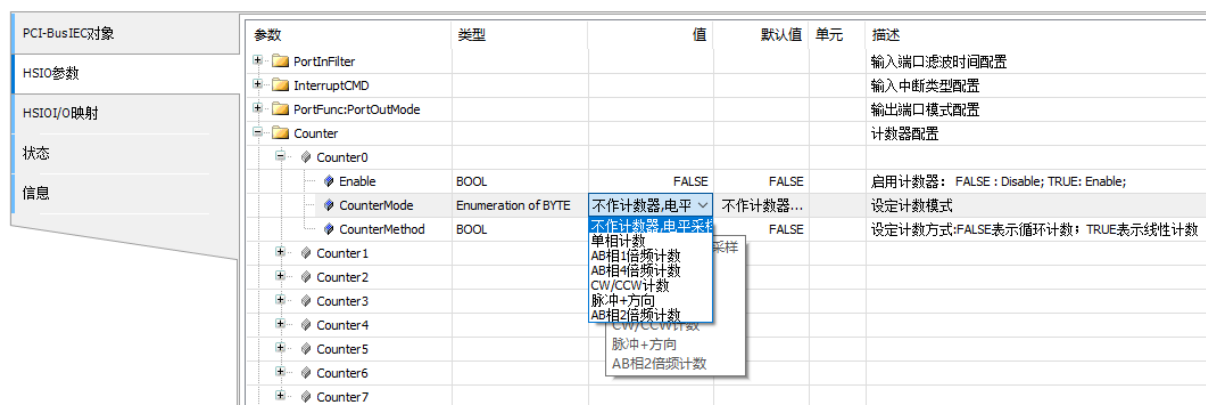


Figure 5-4-6 Counter Configuration Page

The Counter counter supports a maximum of 8 counters. After Enable in the Counter configuration item is set to TRUE, the counter-related function blocks in the CODESYS function block are valid. Otherwise, corresponding errors and fault codes will be generated after the function blocks are used.

Counter counting mode supports the following types:

Table 5-4-7 Counter Counting Type Description

Counter type	Description
Not used as a counter or level	Ordinary input level sampling method
Single phase input count	Single phase pulse counting type
CW/CCW counting	CW/CCW counting type
Phase AB single frequency	Two-phase AB single frequency frequency

counting	counting method
Phase AB double frequency counting	Two-phase AB double frequency counting method
Phase AB quadruple frequency counting	Two-phase AB quadruple frequency counting method
Pulse+direction count	Pulse signal and direction signal counting method

The counting method supports two methods: circular counting and linear counting.

- a) Loop counting method: The counting range set by the function block is [-2147483648, 2147483647], which can include -2147483648 and 2147483647. When the count exceeds the maximum value, the counter starts counting in reverse.
- b) Linear counting mode: The counting range set by the function block (-2147483648, 2147483647) does not include -2147483648 and 2147483647. When the counter value reaches the counting limit, a corresponding overflow signal will be generated.

● Axis configuration

PCI-BusIEC对象	参数	类型	值	默认值	单元	描述
HSIO参数	PortInFilter					输入端口滤波时间配置
HSIOI/O映射	InterruptCMD					输入中断类型配置
状态	PortFunc:PortOutMode					输出端口模式配置
信息	Counter					计数器配置
	Axis					轴参数配置
	Axis0					
	Direction	BOOL	FALSE	FALSE		设定旋转方向: FALSE表示轴正向运行; TRUE表示轴负...
	HomeMode	Enumeration of BYTE	回零模式1	回零模式1		设定回零模式
	ZeroPort	Enumeration of BYTE	IN0-1	IN0-1		选择原点信号输入端子
	Axis1					
	Axis2					
	Axis3					
	Axis4					
	Axis5					
	Axis6					
	Axis7					

Table 5-4-8 High-speed Pulse Axis Configuration Page

The high-speed pulse axis supports up to 8 axes (all axis output port modes are configured as single-phase output).

- a) Direction is the rotation direction of the pulse axis. When the value is FALSE, it means that when moving in the positive direction, the controller outputs positive pulses to the pulse axis; when the value is TRUE, it means when moving in the positive direction, the controller outputs negative pulses to the pulse axis. The specific pulse axis

rotation direction also needs to refer to the internal motor parameter settings of the servo drive.

b) HomeMode: Home mode, currently supports three home modes: Home Mode 1, Home Mode 2, and Home Mode 3. Please refer to Chapter 5.5.7 HSMC_Home for details.

c) ZeroPort: Terminal 3 of the origin input signal, currently supports 12 input port configurations IN0~IN1.

[Note]

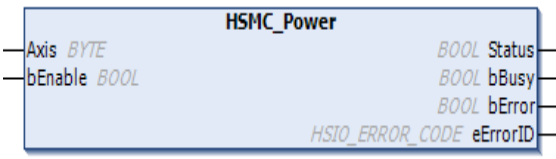
When the counter corresponding to the input port is valid (non-level mode), the input port can no longer be configured as the origin input signal, otherwise the HSMC_Home function block will generate errors and fault codes.

5.5. Introduction to high speed output function block

5.5.1.HSMC_Power

Pulse axis enable function block.

● Command format

Commands	Name	LD performance
HSMC_Power	Enable axis control	

● Input variables

Name	Description	Type	Range	Initial value
Axis	Pulse axis index	BYTE	0 - 7	-
bEnable	Enable	BOOL	TRUE, FALSE	FALSE

● Output variable

Name	Description	Type	Range	Initial value
Status	Axis enable flag	BOOL	TRUE, FALSE	-
bBusy	Execution flag	BOOL	TRUE,FALSE	

bError	Fault flag	BOOL	TRUE,FALSE	
eErrorID	Error code	HSMC_ERROR_CODE	-	-

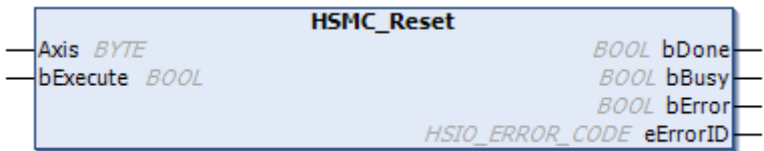
- Function block description

It sets the enabled state of the pulse axis, valid at high level. When bEnable is set to TRUE, the axis enters the enabled state, the Status signal of the command is valid, and then the axis operation related function blocks can be executed.

5.5.2.HSMC_Reset

Pulse axis reset function block.

- Command format

Commands	Name	LD performance
HSMC_Reset	Axis reset	

- Input variables

Name	Description	Type	Range	Initial value
Axis	Pulse axis index	BYTE	0 - 7	-
bExecute	Execution status	BOOL	TRUE, FALSE	FALSE

- Output variable

Name	Description	Type	Range	Initial value
bDone	Done flag	BOOL	TRUE,FALSE	-
bBusy	Execution flag	BOOL	TRUE,FALSE	
bError	Fault flag	BOOL	TRUE,FALSE	
eErrorID	Error code	HSMC_ERROR_CODE	-	-

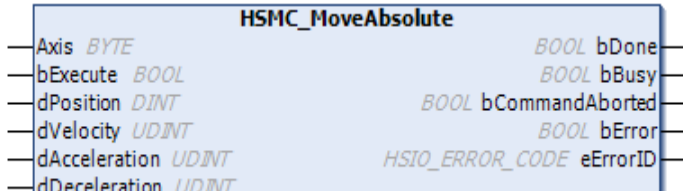
- Function block description

It performs the internal reset function of the pulse axis.

5.5.3.HSMC_MoveAbsoulte

Axis absolute position motion function block.

- Command format

Commands	Name	LD performance
		

HSMC_MoveAbsoulte	Axis absolute position motion	
-------------------	--	--

● Input variables

Name	Description	Type	Range	Initial value
Axis	Pulse axis index	BYTE	0 - 7	
bExecute	Execution switch, edge triggered	BOOL	TRUE, FALSE	FALSE
dPosition	Axis target position Unit: Pulse	DINT	Follow data type	0
dVelocity	Axis velocity, maximum 200K Unit: Pulse/s	UDINT	Follow data type	0
dAcceleration	Axial acceleration Unit: Pulse/s ²	UDINT	Follow data type	0
dDeceleration	Axial deceleration Unit: Pulse/s ²	UDINT	Follow data type	0

● Output variable

Name	Description	Type	Range	Initial value
bDone	Done flag	BOOL	TRUE,FALSE	-
bBusy	Execution flag	BOOL	TRUE,FALSE	-
bCommandAborted	Abort flag	BOOL	TRUE,FALSE	-
bError	Fault flag	BOOL	TRUE,FALSE	-
eErrorID	Error code	HSMC_ERROR_CODE	-	-

● Function block description

This function block supports the absolute position movement of the high-speed pulse axis. bExecute is the rising edge trigger function block and dPosition is the absolute position of the pulse axis.

1. The absolute motion trajectory is T-shaped routing, and the maximum position of dPosition is 2147483647. When the dPosition setting exceeds 2147483647, the axis will move in the reverse direction;

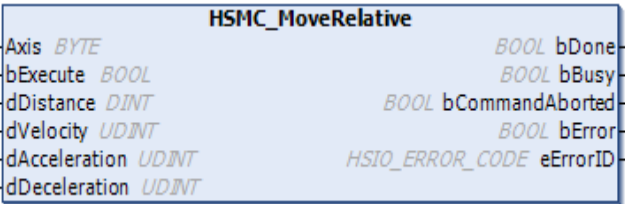
2. dVelocity is the speed command of axis motion, which supports a maximum of 200K;

3. dAcceleration is the acceleration of the axis motion, and dDeceleration is the deceleration of the axis motion. It is recommended that dAcceleration and dDeceleration be set consistent.

5.5.4.HSMC_MoveRelative

Pulse axis relative position motion function block.

● Command format

Commands	Name	LD performance
HSMC_MoveRelative	Axis relative position motion	

● Input variables

Name	Description	Type	Range	Initial value
Axis	Pulse axis index	BYTE	0 - 7	
bExecute	Execution switch, edge triggered	BOOL	TRUE, FALSE	FALSE
dDistance	Axis relative motion distance Unit: Pulse	DINT	Follow data type	0
dVelocity	Axis velocity, maximum 200K Unit: Pulse/s	UDINT	Follow data type	0
dAcceleration	Axial acceleration Unit: Pulse/s ²	UDINT	Follow data type	0
dDeceleration	Axial deceleration Unit: Pulse/s ²	UDINT	Follow data type	0

● Output variable

Name	Description	Type	Range	Initial value
bDone	Done flag	BOOL	TRUE,FALSE	-
bBusy	Execution flag	BOOL	TRUE,FALSE	-
bCommandAborted	Abort flag	BOOL	TRUE,FALSE	-
bError	Fault flag	BOOL	TRUE,FALSE	-
eErrorID	Error code	HSMC_ERROR_CODE	-	-

● Function block description

This function block supports the relative position motion of the high-speed pulse axis. bExecute is the rising edge trigger function block and dDistance is the relative motion distance of the pulse axis.

1. The relative motion trajectory is T-shaped routing, and the maximum position of dDistance is 2147483647. When dDistance is set to exceed 2147483647, the axis will move in the reverse direction;

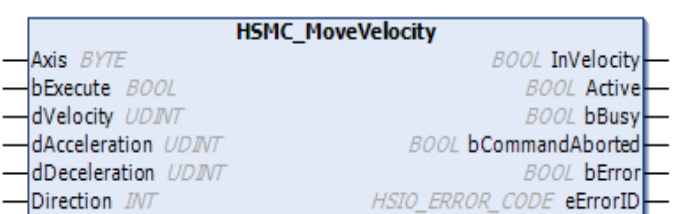
2. dVelocity is the speed command of axis motion, which supports a maximum of 200K;

3. dAcceleration is the acceleration of the axis motion, and dDeceleration is the deceleration of the axis motion. It is recommended that dAcceleration and dDeceleration be set consistent.

5.5.5.HSMC_MoveVelocity

Pulse axis velocity mode motion function block.

- Command format

Commands	Name	LD performance
HSMC_MoveVelocity	Axis velocity motion	

- Input variables

Name	Description	Type	Range	Initial value
Axis	Pulse axis index	BYTE	0 - 7	
bExecute	Execution switch, edge triggered	BOOL	TRUE, FALSE	FALSE
dVelocity	Axis velocity, maximum 200K Unit: Pulse/s	UDINT	Follow data type	0
dAcceleration	Axial acceleration Unit: Pulse/s^2	UDINT	Follow data type	0
dDeceleration	Axial deceleration Unit: Pulse/s^2	UDINT	Follow data type	0
bDirection	Axis movement	BOOL	TRUE, FALSE	FALSE

	direction			
--	-----------	--	--	--

● Output variable

Name	Description	Type	Range	Initial value
bDone	Done flag	BOOL	TRUE,FALSE	-
bBusy	Execution flag	BOOL	TRUE,FALSE	-
bCommandAborted	Abort flag	BOOL	TRUE,FALSE	-
bError	Fault flag	BOOL	TRUE,FALSE	-
eErrorID	Error code	HSMC_ERROR_CODE	-	-

● Function block description

This function block supports the relative position motion of the high-speed pulse axis. bExecute is the rising edge trigger function block and dVelocity is the running speed of the pulse axis.

1. dVelocity is the speed command of axis motion, which supports a maximum of 200K;

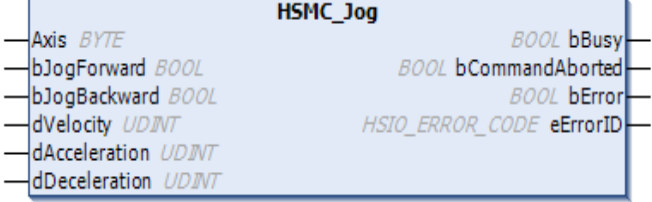
2. dAcceleration is the acceleration of the axis motion, and dDeceleration is the deceleration of the axis motion. It is recommended that dAcceleration and dDeceleration be set consistent;

3. bDirection is the axis velocity running direction; FALSE means running at positive speed, and TRUE means running at negative speed.

5.5.6.HSMC_MoveJog

Pulse axis jogging function block.

● Command format

Commands	Name	LD performance
HSMC_MoveJog	Axis jogging motion	

● Input variables

Name	Description	Type	Range	Initial value
Axis	Pulse axis index	BYTE	0 - 7	

bJogForward	Jog runs forward, level triggered	BOOL	TRUE, FALSE	FALSE
bJogBackward	Jog runs backward, level triggered	BOOL	TRUE, FALSE	FALSE
dVelocity	Axis velocity, maximum 200K Unit: Pulse/s	UDINT	Follow data type	0
dAcceleration	Axial acceleration Unit: Pulse/s^2	UDINT	Follow data type	0
dDeceleration	Axial deceleration Unit: Pulse/s^2	UDINT	Follow data type	0

● Output variable

Name	Description	Type	Range	Initial value
bDone	Done flag	BOOL	TRUE,FALSE	-
bBusy	Execution flag	BOOL	TRUE,FALSE	-
bCommandAborted	Abort flag	BOOL	TRUE,FALSE	-
bError	Fault flag	BOOL	TRUE,FALSE	-
eErrorID	Error code	HSMC_ERROR_CODE	-	-

● Function block description

This function block supports the jogging motion of the high-speed pulse axis, and executes the jogging motion according to the set dVelocity running speed.

1. bJogForward is jog forward motion, level triggered; jog motion is executed on the rising edge, and jog stop is executed on the falling edge;

2. bJogBackward is a negative jog motion, level-triggered. The jog motion is executed on the rising edge, and the jog stop is executed on the falling edge. When bJogForward and bJogBackward are high level at the same time, the function block generates an error;

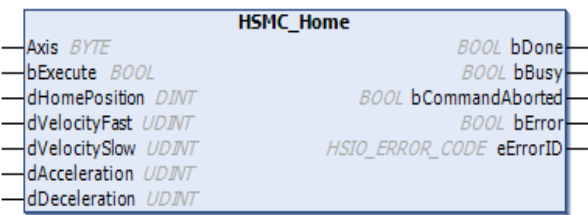
3. dVelocity is the speed command of axis motion, which supports a maximum of 200K;

4. dAcceleration is the acceleration of the axis motion, and dDeceleration is the deceleration of the axis motion. It is recommended that dAcceleration and dDeceleration be set consistent.

5.5.7.HSMC_Home

Pulse axis homing motion function block.

● Command format

Commands	Name	LD performance
HSMC_MoveHome	Axis homing motion	

● Input variables

Name	Description	Type	Range	Initial value
Axis	Pulse axis index	BYTE	0 - 7	
bExecute	Execution switch, edge triggered	BOOL	TRUE, FALSE	FALSE
dHomePosition	Origin position Unit: Pulse	DINT	Follow data type	0
dVelocityFast	Homing fast speed , maximum 200K Unit: Pulse/s	UDINT	Follow data type	0
dVelocitySlow	Homing slow speed , maximum 200K Unit: Pulse/s	UDINT	Follow data type	0
dAcceleration	Axial acceleration Unit: Pulse/s ²	UDINT	Follow data type	0
dDeceleration	Axial deceleration Unit: Pulse/s ²	UDINT	Follow data type	0

● Output variable

Name	Description	Type	Range	Initial value
bDone	Done flag	BOOL	TRUE,FALSE	-
bBusy	Execution flag	BOOL	TRUE,FALSE	-
bCommandAborted	Abort flag	BOOL	TRUE,FALSE	-
bError	Fault flag	BOOL	TRUE,FALSE	-
eErrorID	Error code	HSMC_ERROR_CODE	-	-

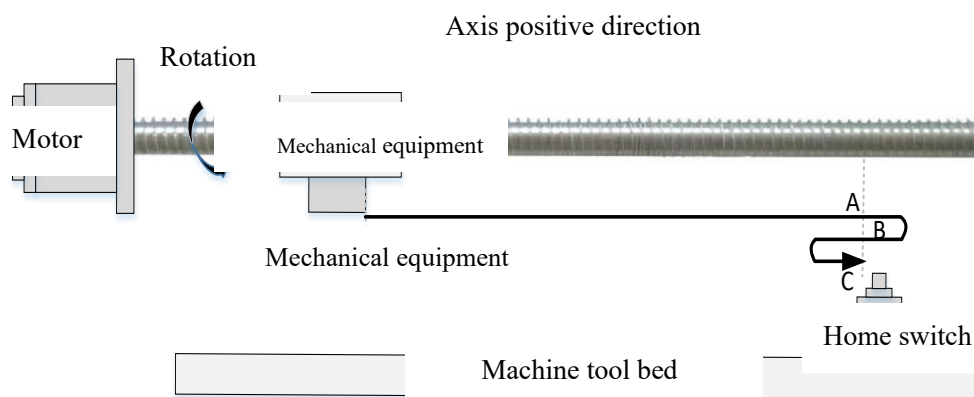
● Function block description

This function block mainly implements the homing (also called return to origin) motion of the high-speed pulse axis. dHomePosition is the origin position after returning to the origin. Currently, it supports three homing modes: homing mode 0, homing mode 1, and homing mode 2.

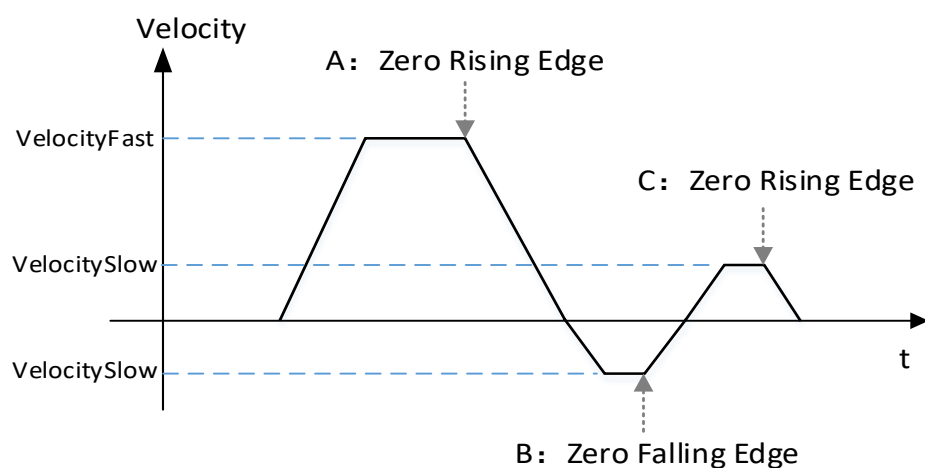
- Homing mode 0: Mode0 homing immediately

After the function block performs the homing operation, the current position is used as the zero position. The function has the same effect as HSMC_SetPosition.

- Homing mode 1: Mode1 homing in the positive direction, and the origin is the home switch. When the starting position is on the right side of the origin, the homing process cannot be performed.

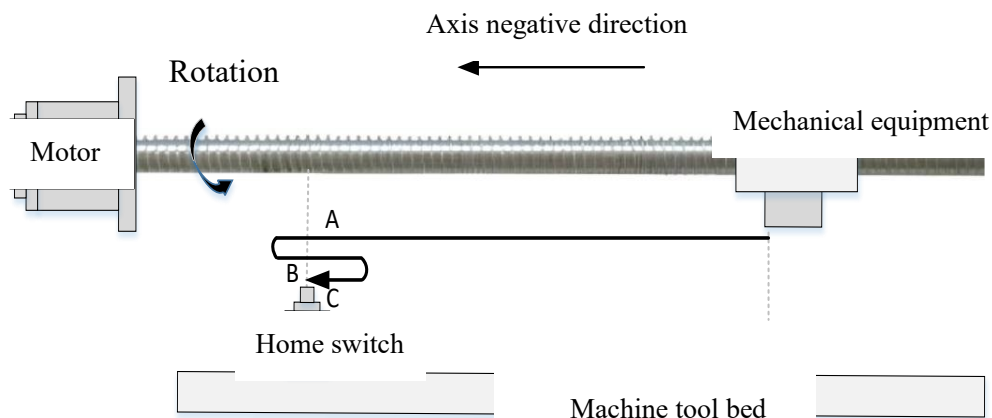


Mode1 workbench homing effect diagram

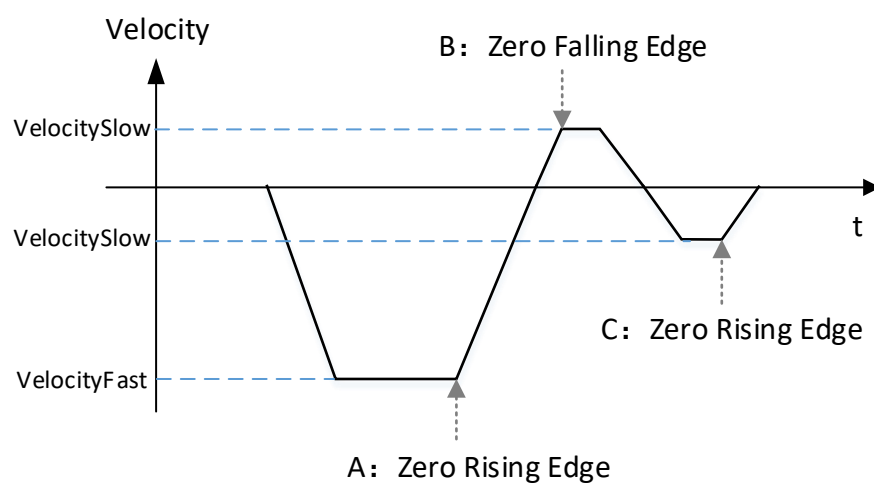


Mode1 homing velocity curve

Homing mode 2: Mode2 homing in the negative direction, and the origin is the home switch. When the starting position is to the left of the origin, the homing process cannot be performed.



Mode2 workbench homing effect diagram

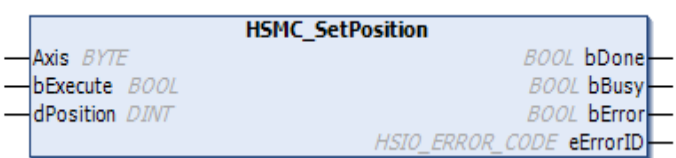


Mode2 homing velocity curve

5.5.8.HSMC_SetPosition

Pulse axis setting coordinate position function block.

● Command format

Commands	Name	LD performance
HSMC_SetPosition	Set axis coordinate position	

- Input variables

Name	Description	Type	Range	Initial value
Axis	Pulse axis index	BYTE	0 - 7	
bExecute	Execution switch, edge triggered	BOOL	TRUE, FALSE	FALSE
dPosition	Axis target position Unit: Pulse	DINT	Follow data type	0

- Output variable

Name	Description	Type	Range	Initial value
bDone	Done flag	BOOL	TRUE,FALSE	-
bBusy	Execution flag	BOOL	TRUE,FALSE	-
bError	Fault flag	BOOL	TRUE,FALSE	-
eErrorID	Error code	HSMC_ERROR_CODE	-	-

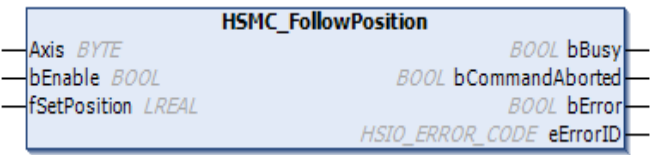
- Function block description

This function block sets the coordinate position of the pulse axis. When setting the coordinate position of the pulse axis, the axis will not move.

5.5.9.HSMC_FollowPostion

Pulse axis follows position motion function block.

- Command format

Commands	Name	LD performance
HSMC_FollowPosition	Axis follows position motion	

- Input variables

Name	Description	Type	Range	Initial value
Axis	Pulse axis index	BYTE	0 - 7	
bEnable	Level control	BOOL	TRUE, FALSE	FALSE
fSetPosition	Axis target position Unit: Pulse	LREAL	Follow data type	0

- Output variable

Name	Description	Type	Range	Initial value
------	-------------	------	-------	---------------

bBusy	Execution flag	BOOL	TRUE,FALSE	-
bCommandAborted	Abort state	BOOL	TRUE,FALSE	-
bError	Fault flag	BOOL	TRUE,FALSE	-
eErrorID	Error code	HSMC_ERROR_CODE	-	-

- Function block description

This function block is used to periodically set the coordinate position fSetPosition of the pulse axis.

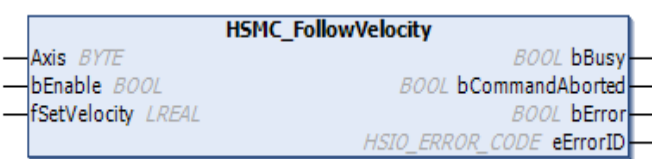
1. bEnable in the function block is level enable. When bEnable is set to TRUE, the axis target position coordinate value fSetPosition will be given in each cycle, and the unit is Pulse;

2. The pulse axis velocity is calculated as the position increment between two task cycles. Pulse axis velocity $V = \Delta S / \Delta T$ (ΔT is the time interval between the two tasks before and after; ΔS is the difference between the given position of the current task cycle minus the given position of the previous task cycle); when the pulse axis velocity v is greater than 200K, an error occurs inside the function block.

5.5.10.HSMC_FollowVelocity

Pulse axis setting speed function block.

- Command format

Commands	Name	LD performance
HSMC_FollowVelocity	Axis follows speed motion	

- Input variables

Name	Description	Type	Range	Initial value
Axis	Pulse axis index	BYTE	0 - 7	
bEnable	Level control	BOOL	TRUE, FALSE	FALSE
fSetVelocity	Axis following speed, maximum 200K Unit: Pulse/s	LREAL	Follow data type	-

- Output variable

Name	Description	Type	Range	Initial value
bBusy	Execution flag	BOOL	TRUE,FALSE	-
bCommandAborted	Abort state	BOOL	TRUE,FALSE	-
bError	Fault flag	BOOL	TRUE,FALSE	-
eErrorID	Error code	HSMC_ERROR_CODE	-	-

- Function block description

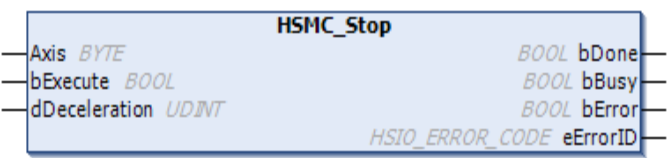
This function block is used to periodically set the running velocity of the pulse axis fSetVelocity.

1. bEnable in the function block is level enable. When bEnable is set to TRUE, the target axis velocity fSetVelocity will be given in each cycle, and the unit is Pulse/s;
2. The maximum pulse axis velocity is 200K. When the axis velocity exceeds 200K, an error occurs inside the function block.

5.5.11.HSMC_Stop

Pulse axis stop motion function block.

- Command format
- Input variables

Commands	Name	LD performance
HSMC_Stop	Axis stop motion	

Name	Description	Type	Range	Initial value
Axis	Pulse axis index	BYTE	0 - 7	
bExecute	Execution switch, edge triggered	BOOL	TRUE, FALSE	FALSE
dDeceleration	Axis stop deceleration Pulse/s ²	UDINT	Follow data type	-

- Output variable

Name	Description	Type	Range	Initial value
bDone	Done flag	BOOL	TRUE,FALSE	
bBusy	Execution flag	BOOL	TRUE,FALSE	-

bError	Fault flag	BOOL	TRUE,FALSE	-
eErrorID	Error code	HSMC_ERROR_CODE	-	-

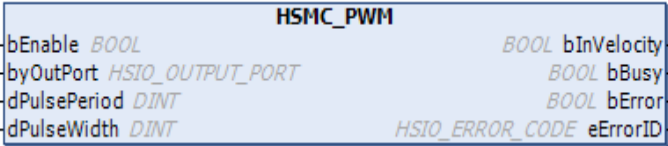
- Function block description

This function block performs the deceleration and stop process of the pulse axis under normal operation conditions. In the function block, bExecute is the edge triggered. When bExecute is set to TRUE, the pulse axis decelerates according to the set deceleration dDeceleration. During the execution of axial deceleration, the output status of bBusy is TRUE; after the axial deceleration stops, the output status of bDone changes to TRUE, and at the same time, the output status of bBusy changes to FALSE.

5.5.12.HSMC_PWM

PWM output control function block.

- Command format

Commands	Name	LD performance
HSMC_PWM	PWM output control	

- Input variables

Name	Description	Type	Range	Initial value
Axis	Pulse axis index	BYTE	0 - 7	
bEnable	Level control	BOOL	TRUE, FALSE	FALSE
byOutPort	The hardware output port number supports up to 8 PWM output ports. The hardware output port number can be selected according to the HSIO_OUTPUT_PORT type definition.	HSIO_OUTPUT_PORT	-	0
dPulsePeriod	Full square wave pulse period Unit: us	DINT	Follow data type	0
dPulseWidth	Full square wave high level pulse width Unit: us	DINT	Follow data type	0

- Output variable

Name	Description	Type	Range	Initial value
bInVelocity	Frequency reaches setting mark	BOOL	TRUE,FALSE	-
bBusy	Execution flag	BOOL	TRUE,FALSE	-
bError	Fault flag	BOOL	TRUE,FALSE	-
eErrorID	Error code	HSMC_ERROR_CODE	-	-

- Function block description

This function block is used for PWM pulse output control.

1. In the function block, bEnable is level control. When the bEnable value is TRUE, PWM pulse output starts to be generated. When the pulse output frequency reaches the set value, the bInVelocity output state changes to TRUE; when the bEnable value is FALSE, the PWM pulse output control is turned off.

2. PWM pulse output supports up to 8 ports for simultaneous output. byOutPort is the hardware output port number. The corresponding hardware output port can be specified according to the HSIO_OUTPUT_PORT type definition.

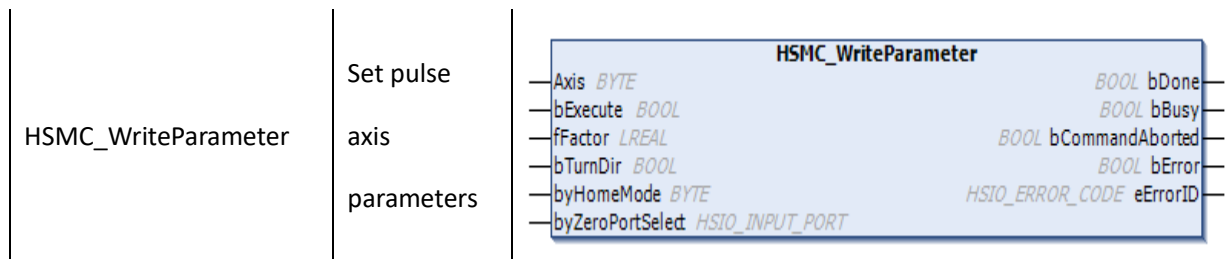
3. The full square wave period of the PWM pulse is set to dPulsePeriod (in us). The maximum frequency output supports 200K, i.e. the minimum period of dPulsePeriod is 5us. When the function block sets the square wave period to be less than 5us, the function block generates an error. The high-level width of PWM square wave is set to dPulseWidth (in us), and the set value dPulseWidth should be less than dPulsePeriod. When dPulseWidth is set to be greater than or equal to dPulsePeriod, the function block generates an error, and the bError output state becomes TRUE, and the corresponding fault code is generated.

5.5.13.HSMC_WriteParameter

Axis parameter setting function block.

- Command format

Commands	Name	LD performance
----------	------	----------------



● Input variables

Name	Description	Type	Range	Initial value
Axis	Pulse axis index	BYTE	0 - 7	
bExecute	Execution switch, edge triggered	BOOL	TRUE, FALSE	FALSE
fFactor	Output command scaling	LREAL	-	0
bTurnDir	Motor rotational direction FALSE: The motor rotates in the positive direction; TRUE: The motor rotates in the negative direction.	BOOL	Follow data type	FALSE
byHomeMode	Home mode currently supports Mode0, Mode1, and Mode2. Please refer to explanation of the HSMC_Home function block for details.	BYTE	Follow data type	0
byZeroPortSelect	Zero signal input port number Select the corresponding port number according to the HSIO_INPUT_PORT port definition	HSIO_INPUT_PORT	-	0

● Output variable

Name	Description	Type	Range	Initial value
bDone	Parameters written successfully	BOOL	TRUE, FALSE	
bCommandAborted	Abort flag	BOOL	TRUE, FALSE	
bBusy	Execution flag	BOOL	TRUE, FALSE	-
bError	Fault flag	BOOL	TRUE, FALSE	-
eErrorID	Error code	HSMC_ERROR_CODE	-	-

● Function block description

This function block is used for setting pulse axis parameters.

1. The function block bExecute is edge triggered. When bExecute is a rising edge, the function performs the pulse axis parameter writing operation. After the function block parameter writing is completed, the output status of bDone changes to TRUE.

2. The function block command scaling ratio fFactor is used to set the position command ratio and velocity ratio received by the pulse axis. For example, the position command output by the function block is POS, and the velocity command output by the function block is V. After the pulse axis is adjusted by the fFactor ratio, the actual received position command $ActPos = POS * fFactor$, and the received velocity value $ActV = V * fFactor$.

3. The function block sets the motor rotation direction bTurnDir. When the value is FALSE, it means that the motor rotates in the forward direction (actually, the servo parameter setting needs to be combined to determine the axis rotation direction); when the value is TRUE, it means that the motor rotates in the reverse direction.

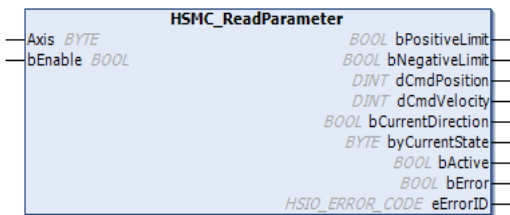
4. The motor rotation direction, homing mode, and Zero signal input port setting methods support the LocalHSIO device HSIO parameter setting method and the function block HSMC_WriteParameter setting method. When the HSMC_WriteParameter function block of the pulse axis is not executed, the corresponding axis configuration in the HSIO parameter is used by default; When the function block of pulse axis HSMC_WriteParameter takes effect, the corresponding axis configuration in HSIO parameter is no longer used.

HSIO 参数	参数	类型	值	默认值	单元	描述
HSIO1/O映射	PortInFilter					输入端口滤波时间配置
状态	InterruptCMD					输入中断类型配置
信息	PortFunc:PortOutMode					输出端口模式配置
	Counter					计数器配置
	Axis					轴参数配置
	Axis0					
	Direction	BOOL	FALSE	FALSE		设定旋转方向: FALSE表示轴正向运行; TRUE表示轴负向运行;
	HomeMode	Enumeration of BYTE	回零模式1	回零模式1		设定回零模式
	ZeroPort	Enumeration of BYTE	IN0-1	IN0-1		选择原点信号输入端子
	Axis1					
	Axis2					
	Axis3					
	Axis4					
	Axis5					
	Axis6					
	Axis7					

5.5.14.HSMC_ReadParameter

Pulse axis reading parameter function block.

● Command format

Commands	Name	LD performance
HSMC_ReadParameter	Reading pulse axis parameters	

● Input variables

Name	Description	Type	Range	Initial value
Axis	Pulse axis index	BYTE	0 - 7	
bEnable	Level control	BOOL	TRUE, FALSE	FALSE

● Output variable

Name	Description	Type	Range	Initial value
bPositiveLimit	Positive limit trigger flag	BOOL	TRUE, FALSE	
bNegativeLimit	Negative limit trigger flag	BOOL	TRUE, FALSE	
dCmdPosition	Command output position Unit: Pulse	DINT	Follow data type	-
dCmdVelocity	Command output velocity Unit: Pulse/s	DINT	Follow data type	-
bCurrentDirection	Current axis motion direction FALSE: run in positive direction TRUE: run in negative direction	BOOL	TRUE, FALSE	
byCurrentState	PLCopen State	BYTE	-	-

bActive	Function block valid flag	BOOL	TRUE, FALSE	
bError	Fault flag	BOOL	TRUE, FALSE	
eErrorID	Error code	HSMC_ERROR_CODE	-	

- Function block description

This function block is used to read pulse axis parameter values periodically.

When the bEnable value in the function block is TRUE, it means starting to read the pulse axis parameter value periodically, and the bActive output state will also change to TRUE; byCurrentState is a PLCOpen state machine, used to record the internal execution status of the high-speed pulse axis.

5.6. High-speed output application examples

5.6.1. Absolute positioning motion

[Example]

The pulse axis Axis0 performs absolute positioning motion, and the output control is pulse + direction mode. For example, the following steps need to be performed:

1. Pulse output type and Axis configuration

Double-click to select the LocalHSIO device, enter the HSIO parameter page, and find the pulse + direction of OUT0-1 Mode and OUT0-2-Mode; at this time, OUT0-1 is the pulse signal output by default, and OUT0-2 is the direction signal output. Finally, select the appropriate pulse axis rotation direction according to your needs.

HSIO 参数	参数	类型	值	默认值	单元	描述
HSIOI/O映射	PortInFilter					输入端口滤波时间配置
状态	InterruptCMD					输入中断类型配置
信息	PortFunc:PortOutMode					输出端口模式配置
	OUT0-1-Mode	Enumeration of BYTE	脉冲+方向输出	GeneralOut		脉冲输出索引: 0
	OUT0-2-Mode	Enumeration of BYTE	脉冲+方向输出	GeneralOut		脉冲输出索引: 1
	OUT0-3-Mode	Enumeration of BYTE	GeneralOut	GeneralOut		脉冲输出索引: 2
	OUT0-4-Mode	Enumeration of BYTE	GeneralOut	GeneralOut		脉冲输出索引: 3
	OUT1-1-Mode	Enumeration of BYTE	GeneralOut	GeneralOut		脉冲输出索引: 4
	OUT1-2-Mode	Enumeration of BYTE	GeneralOut	GeneralOut		脉冲输出索引: 5
	OUT1-3-Mode	Enumeration of BYTE	GeneralOut	GeneralOut		脉冲输出索引: 6
	OUT1-4-Mode	Enumeration of BYTE	Cnt0CmpOut	GeneralOut		脉冲输出索引: 7
	Counter					计数器配置
	Axis					轴参数配置
	Axis0					
	Direction	BOOL	FALSE	FALSE		设定旋转方向: FALSE表示...
	HomeMode	Enumeration of BYTE	回零模式1	回零模式1		设定回零模式
	ZeroPort	Enumeration of BYTE	IN0-1	IN0-1		选择原点信号输入端子
	Axis1					
	Direction	BOOL	FALSE	FALSE		设定旋转方向: FALSE表示...
	HomeMode	Enumeration of BYTE	回零模式1	回零模式1		设定回零模式
	ZeroPort	Enumeration of BYTE	IN0-1	IN0-1		选择原点信号输入端子
	Axis2					
	Axis3					
	Axis4					
	Axis5					
	Axis6					
	Axis7					

Figure 5-6-1-1 Pulse Output Configuration and Axis Configuration Diagram

2. Hardware wiring

If the controller output needs to be connected to a real pulse axis device, the controller output port OUT0-1 is the pulse signal output and is connected to the pulse signal input signal bit of the pulse axis; the controller output port OUT0-2 is the direction signal output, connected to the direction input signal bit of the pulse axis.

3. Function block execution

The absolute position of the pulse axis Axis0 is $dPosition = 100000$; the velocity is 10000 Pulse/s, and the acceleration and deceleration is 100000 Pulse/s^2 .

When the HSMC_Power function execution power_on is TRUE, the positioning motion can be started by executing the rising edge of bExecute of the HSMC_MoveAbsolute absolute positioning function block. The HSMC_ReadParamter function block can periodically monitor the output position command, output velocity command and motion direction.

```

1  ● HSMC_Power (
2      Axis 0 := 0,
3      bEnable TRUE := power_on F TRUE,
4      Status TRUE => status TRUE,
5      bBusy=> ,
6      bError=> ,
7      eErrorID=> );
8
9  ● HSMC_MoveAbsolute (
10     Axis 0 := 0,
11     bExecute TRUE := abs_move_execute F TRUE,
12     dPosition 100000 := 100000,
13     dVelocity 10000 := 10000,
14     dAcceleration 100000 := 100000,
15     dDeceleration 100000 := 100000,
16     bDone=> ,
17     bBusy=> ,
18     bCommandAborted=> ,
19     bError=> ,
20     eErrorID=> );
21
22  ● HSMC_ReadParameter (
23     Axis 0 := 0,
24     bEnable TRUE := read_parameter_enable F TRUE,
25     bPositiveLimit=> ,
26     bNegativeLimit=> ,
27     dCmdPosition=> ,
28     dCmdVelocity=> ,
29     bCurrentDirection=> ,
30     byCurrentState=> ,
31     bActive=> ,
32     bError=> ,

```

Figure 5-6-1-2 Function Block Execution Display

表达式	类型	值	准备值	地址	注释
✚ HSMC_SetPosition	CmpHSIO.HSMC_SetPosition				
☒ HSMC_ReadParameter	CmpHSIO.HSMC_ReadParameter				
✚ Axis	BYTE	0			脉冲轴索引值
✚ bEnable	BOOL	TRUE			值为TRUE表示一直执行
✚ bPositiveLimit	BOOL	FALSE			正向限位触发
✚ bNegativeLimit	BOOL	FALSE			负向限位触发
✚ dCmdPosition	DINT	100000			发送位置，单位 Pulse
✚ dCmdVelocity	DINT	0			发送速度，单位 Pulse/S
✚ bCurrentDirection	BOOL	FALSE			当前运行方向，FALSE:正转 TRUE:反转
✚ byCurrentState	BYTE	3			PLCopen状态
✚ bActive	BOOL	TRUE			
✚ bError	BOOL	FALSE			
✚ eErrorID	HSIO_ERROR_CODE	NO_ERROR			

Figure 5-6-1-3 Periodic Reading of Axis Status

5.6.2. Jog motion

[Example]

The pulse axis Axis0 performs Jog motion. For example, if the output control is in pulse+direction mode, the following steps need to be performed:

1. Pulse output type and Axis configuration

Please refer to the configuration in 5.6.1 Absolute positioning motion example;

2. Hardware wiring

If the controller output needs to be connected to a real pulse axis device, you can refer to the hardware wiring in “5.6.1 Absolute positioning motion example”;

3. Function block execution

The pulse axis Axis0 executes Jog forward motion; the velocity is 10000 Pulse/s, and the acceleration and deceleration is 100000 Pulse/s².

When the HSMC_Power function execution power_on is TRUE, and the HSMC_Jog operation function block bJogForward is TRUE, the forward jog motion can be started. The HSMC_ReadParamter function block can periodically monitor the output position command, output velocity command and motion direction.

```

1  ● HSMC_Power(
2      Axis 0 := 0,
3      bEnable TRUE := power_on F TRUE,
4      Status TRUE => status TRUE,
5      bBusy=> ,
6      bError=> ,
7      eErrorID=> );
8
9  ● HSMC_Jog(
10     Axis 0 := 0,
11     bJogForward TRUE := jog_forward_enable F TRUE,
12     bJogBackward:= ,
13     dVelocity 10000 := 10000,
14     dAcceleration 100000 := 100000,
15     dDeceleration 100000 := 100000,
16     bBusy=> ,
17     bCommandAborted=> ,
18     bError=> ,
19     eErrorID=> );
20
21 ● HSMC_ReadParameter(
22     Axis 0 := 0,
23     bEnable TRUE := read_parameter_enable F TRUE,
24     bPositiveLimit=> ,
25     bNegativeLimit=> ,
26     dCmdPosition=> ,
27     dCmdVelocity=> ,
28     bCurrentDirection=> ,
29     byCurrentState=> ,
30     bActive=> ,
31     bError=> ,
32     eErrorID=> );
33 ● RETURN

```

Figure 5-6-2-1 Function Block Execution Display

表达式	类型	值	准备值	地...	注释
 HSMC_ReadParameter	CmpHSIO.HSMC_ReadParameter				
 Axis	BYTE	0			脉冲轴索引值
 bEnable	BOOL	TRUE			值为TRUE表示一直执行
 bPositiveLimit	BOOL	FALSE			正向限位触发
 bNegativeLimit	BOOL	FALSE			负向限位触发
 dCmdPosition	DINT	1245560			发送位置，单位 Pulse
 dCmdVelocity	DINT	10000			发送速度，单位 Pulse/S
 bCurrentDirection	BOOL	FALSE			当前运行方向，FALSE:正转 TRUE:反转
 byCurrentState	BYTE	5			PLCopen状态
 bActive	BOOL	TRUE			
 bError	BOOL	FALSE			

Figure 5-6-2-2 Periodic Reading of Axis Status

5.6.3. Follow velocity motion

[Example]

The pulse axis Axis2 follows the velocity of the pulse axis Axis0. Axis0 performs Jog forward motion. The output control of Axis0 and Axis2 is pulse + direction mode. For example, the following steps need to be performed:

1. Pulse output type and Axis configuration

Please refer to the configuration in “5.6.1 Absolute positioning motion example”.

2. Hardware wiring

If the controller output needs to be connected to a real pulse axis device, you can refer to the hardware wiring in “5.6.1 Absolute positioning motion example”.

3. Function block execution

The pulse axis Axis0 executes Jog forward motion with a velocity of 10000 Pulse/s and an acceleration and deceleration of 100000 Pulse/s². The pulse axis Axis2 follows the velocity of Axis0.

When the HSMC_Power function execution power_on is TRUE, and the HSMC_Jog operation function block bJogForward is TRUE, the forward jog motion can be started. The HSMC_ReadParamter function block of Axis0 can periodically assign dCmdVelocity to dVelocity, and transmit the dVelocity value into the HSMC_FollowVelocity function block. When the function block bEnable is set to TRUE, the real-time follow-velocity operation function of Axis2 can be completed.

```

1  ● HSMC_Power(
2      Axis[0] := 0,
3      bEnable[TRUE] := power_on[TRUE],
4      Status[TRUE] => status[TRUE],
5      bBusy=> ,
6      bError=> ,
7      eErrorID=> );
8
9  ● HSMC_Jog(
10     Axis[0] := 0,
11     bJogForward[TRUE] := jog_forward_enable[TRUE],
12     bJogBackward:= ,
13     dVelocity[10000] := 10000,
14     dAcceleration[100000] := 100000,
15     dDeceleration[100000] := 100000,
16     bBusy=> ,
17     bCommandAborted=> ,
18     bError=> ,
19     eErrorID=> );
20
21 ● HSMC_ReadParameter(
22     Axis[0] := 0,
23     bEnable[TRUE] := read_parameter_enable[TRUE],
24     bPositiveLimit=> ,
25     bNegativeLimit=> ,
26     dCmdPosition=> ,
27     dCmdVelocity[10000] => dVelocity[10000],
28     bCurrentDirection=> ,
29     byCurrentState=> ,
30     bActive=> ,
31     bError=> ,
32     eErrorID=> );
33
34 ● HSMC_FollowVelocity(
35     Axis[2] := 2,
36     bEnable[TRUE] := follow_enable[TRUE],
37     fSetVelocity[1E+04 ▶] := dVelocity[10000],
38     bBusy=> ,
39     bCommandAborted=> ,
40     bError=> ,
41     eErrorID=> );
42 ● RETURN

```

Figure 5-6-3 Function Block Execution Display

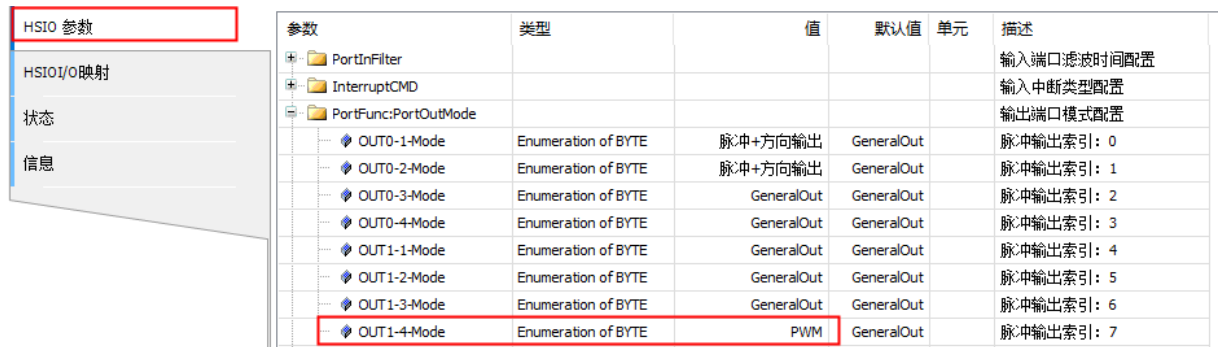
5.6.4. PWM output

[Example]

The pulse port OUT1-4 output ports perform PWM output. The PWM period is 1000us and the duty cycle is 50% (dPulseWidth high level width is 500us). The following steps need to be performed:

1. Pulse output type configuration

Double-click to select the LocalHSIO device, enter the HSIO parameter page, set OUT1-4-Mode in PortOutMode to PWM, then the output of the corresponding controller hardware output OUT1-4 ports is set to PWM output mode, as shown in the following figure:



参数	类型	值	默认值	单元	描述
PortInFilter					输入端口滤波时间配置
InterruptCMD					输入中断类型配置
PortFunc:PortOutMode					输出端口模式配置
OUT0-1-Mode	Enumeration of BYTE	脉冲+方向输出	GeneralOut		脉冲输出索引: 0
OUT0-2-Mode	Enumeration of BYTE	脉冲+方向输出	GeneralOut		脉冲输出索引: 1
OUT0-3-Mode	Enumeration of BYTE	GeneralOut	GeneralOut		脉冲输出索引: 2
OUT0-4-Mode	Enumeration of BYTE	GeneralOut	GeneralOut		脉冲输出索引: 3
OUT1-1-Mode	Enumeration of BYTE	GeneralOut	GeneralOut		脉冲输出索引: 4
OUT1-2-Mode	Enumeration of BYTE	GeneralOut	GeneralOut		脉冲输出索引: 5
OUT1-3-Mode	Enumeration of BYTE	GeneralOut	GeneralOut		脉冲输出索引: 6
OUT1-4-Mode	Enumeration of BYTE	PWM	GeneralOut		脉冲输出索引: 7

Figure 5-6-4-1 Output Port Mode Configuration

2. Hardware wiring

If the controller output needs to be connected to a real PWM hardware device, the corresponding OUT1-4 output port should be connected to an external device. If there is no need to connect to a real device, this step can be ignored.

3. Function block execution

The PWM output port of the HSMC_Pwm function block is OUT1_4. For the corresponding definition, see the HSIO_OUTPUT_PORT enumeration variable. The PWM period dPulsePeriod is set to 1000us, the PWM high-level pulse width dPulseWidth is set to 500us, and the bEnable enable bit is set to TRUE to start PWM output. If it is necessary to dynamically adjust the PWM output frequency and duty cycle, the new PWM configuration can take effect by directly setting different dPulsePeriod and dPulseWidth in the state of bEnable being TRUE; when bEnable is FALSE, the PWM output is turned off.

```

3  ● HSMC_PWM(
4      bEnable TRUE := pwm_enable TRUE,
5      byOutPort OUT1_4_POR := CmpHSIO.HSIO_OUTPUT_PORT.OUT1_4_PORT,
6      dPulsePeriod 1000 := 1000,
7      dPulseWidth 500 := 500,
8      bInVelocity TRUE => bInVelocity TRUE,
9      bBusy=> ,
10     bError=> ,
11     eErrorID=> );
12

```

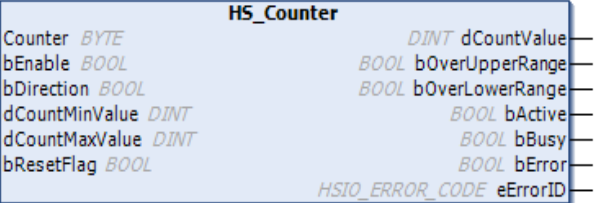
Figure 5-6-4-2 Function Block Execution Display

5.7. Introduction to high-speed input function block

5.7.1. HS_Counter

Counter counting start and end function block.

- Command format

Commands	Name	LD performance
HS_Counter	Counter counting start and end	

- Input variables

Name	Description	Type	Range	Initial value
Counter	Pulse axis index	BYTE	0 - 7	
bEnable	Level control	BOOL	TRUE, FALSE	FALSE
bDirection	Counting direction FALSE: When the pulse input is positive, the count value increases; TRUE: When the pulse input is in the positive direction, the count value decreases.	BOOL	TRUE, FALSE	FALSE
dCounterMinValue	Minimum count value	DINT	Follow data type	0
dCounterMaxValue	Maximum count value	DINT	Follow data type	0

bResetFlag	Counter overflow reset	BOOL	TRUE, FALSE	FALSE
------------	------------------------	------	-------------	-------

● Output variable

Name	Description	Type	Range	Initial value
bActive	Function block valid flag	BOOL	TRUE, FALSE	
bBusy	Execution flag	BOOL	TRUE, FALSE	
dCountValue	Count value	DINT	Follow data type	-
bOverUpperRange	Count exceeding the maximum value flag	BOOL	TRUE, FALSE	-
bOverLowerRange	Count exceeding the minimum value flag	BOOL	TRUE, FALSE	
bError	Fault flag	BOOL	TRUE, FALSE	
eErrorID	Error code	HSMC_ERROR_CODE	-	

● Function block description

This function block is used for the start and end of counter counting. When bEnable in the function block is set to TRUE, it means starting the counter counting. When bEnable is set to FALSE, it means ending the counter counting.

1. The function block Counter is the counter index number. The relationship between the counter mode, counter index number, and corresponding hardware input port is as shown in the following table:

Model	Counter 0	Counter 1	Counter 2	Counter 3	Counter 4	Counter 5	Counter 6	Counter 7
Single phase counting	IN0.1 pulse	IN0.2 pulse	IN0.3 pulse	IN0.4 pulse	IN1.1 pulse	IN1.2 pulse	IN1.3 pulse	IN1.4 pulse
Pulse+direction	IN0.1 pulse IN0.2 direction	--	IN0.3 pulse IN0.4 direction	--	IN1.1 pulse IN1.2 direction	--	IN1.3 pulse IN1.4 direction	-- --
Phase AB 1/2/4 frequency	IN0.1 phase A IN0.2	--	IN0.3 phase A IN0.4	--	IN1.1 phase A IN1.2	--	IN1.3 phase A IN1.4	--

	phase B		phase B		phase B		phase B	
CW/CCW	IN0.1	--	IN0.3	--	IN1.1	--	IN1.3	--
	CW		CW		CW		CW	
	IN0.2		IN0.4		IN1.2		IN1.4	
	CCW		CCW		CCW		CCW	

Counter 1, Counter 3, Counter 5, and Counter 7 only support single-phase counting mode.

[Example]

If Counter 2 is set to pulse + direction, the IN0.3 terminal of the controller should be connected to the external pulse signal, and the IN0.4 terminal should be connected to the external direction signal, and Counter 3 will not be enabled at this time;

If Counter 4 is set to single-phase counting, the IN1.1 terminal of the controller should be connected to an external single-phase signal. At this time, Counter 5 can only use single-phase counting or the counter is not enabled. If Counter 5 enables single-phase counting, IN1. Terminal 2 should be connected to an external single-phase signal.

2. The counter configuration items in the HSIO parameter page include:

a) Enable: Whether the counter is enabled. When the counter is enabled, the function block HS_Counter can be used. If Enable in the HSIO configuration page is FALSE and bEnable in the function block HS_Counter is set to TRUE, the HS_Counter function block reports an error, the bError output status changes to TRUE, and a fault code is generated.

b) CounterMode: Counting mode. Single-phase counting, pulse + direction, phase AB single, double, quadruple frequency counting, CW/CCW counting are supported.

c) CounterMethod: counting method. FALSE means loop counting; TRUE means linear counting.

d) Loop counting method: The counting range set by the function block is [-2147483648, 2147483647], which can include -2147483648 and 2147483647, depending on the minimum and maximum values set by the function block. When the count exceeds the maximum value, the counter starts counting in reverse. For example,

when the counting range is set to $[-9999, 9999]$, if the counter reaches 9999 in the positive direction and continues to receive positive pulses, it will become 9998, 9997, and 9996 in the next moment and decrease in reverse order; when the negative count of the counter reaches -9999, if the negative pulse continues to be received, it will be -9998, -9997 and -9996, which will decrease in reverse order.

e) Linear counting method: The counting range set by the function block $(-2147483648, 2147483647)$, excluding -2147483648 and 2147483647, based on the minimum and maximum values set by the function block. When the count exceeds the maximum value, the output status of bOverUpperRange changes to TRUE. When the count exceeds the minimum value, the output status of bOverLowerRange changes to TRUE; when the rising edge of bResetFlag is executed, the output status of bOverLowerRange and bOverUpperRange can be set to FALSE.

HSIO 参数	参数	类型	值	默认值	单元	描述
HSIOI/O映射	PortInFilter					输入端口滤波时间配置
状态	InterruptCMD					输入中断类型配置
信息	PortFunc:PortOutMode					输出端口模式配置
	Counter					计数器配置
	Counter0					
	Enable	BOOL	TRUE	FALSE		启用计数器: FALSE: Disable; TRUE: Enable;
	CounterMode	Enumeration of BYTE	脉冲+方向	不作计...		设定计数模式
	CounterMethod	BOOL	FALSE	FALSE		设定计数方式:FALSE表示循环计数; TRUE表示线性计数
	Counter1					
	Enable	BOOL	TRUE	FALSE		启用计数器: FALSE: Disable; TRUE: Enable;
	CounterMode	Enumeration of BYTE	脉冲+方向	不作计...		设定计数模式
	CounterMethod	BOOL	FALSE	FALSE		设定计数方式:FALSE表示循环计数; TRUE表示线性计数
	Counter2					
	Enable	BOOL	TRUE	FALSE		启用计数器: FALSE: Disable; TRUE: Enable;
	CounterMode	Enumeration of BYTE	AB相1倍频计数	不作计...		设定计数模式
	CounterMethod	BOOL	FALSE	FALSE		设定计数方式:FALSE表示循环计数; TRUE表示线性计数
	Counter3					
	Enable	BOOL	TRUE	FALSE		启用计数器: FALSE: Disable; TRUE: Enable;
	CounterMode	Enumeration of BYTE	AB相1倍频计数	不作计...		设定计数模式
	CounterMethod	BOOL	FALSE	FALSE		设定计数方式:FALSE表示循环计数; TRUE表示线性计数
	Counter4					
	Enable	BOOL	TRUE	FALSE		启用计数器: FALSE: Disable; TRUE: Enable;
	CounterMode	Enumeration of BYTE	单相计数	不作计...		设定计数模式
	CounterMethod	BOOL	FALSE	FALSE		设定计数方式:FALSE表示循环计数; TRUE表示线性计数
	Counter5					
	Enable	BOOL	TRUE	FALSE		启用计数器: FALSE: Disable; TRUE: Enable;
	CounterMode	Enumeration of BYTE	单相计数	不作计...		设定计数模式
	CounterMethod	BOOL	FALSE	FALSE		设定计数方式:FALSE表示循环计数; TRUE表示线性计数
	Counter6					

[Note]

When the function block HS_WriteParameter is not executed, the counter configuration shall be based on the counter configuration content of the HSIO parameter page; when the function block HS_WriteParameter is executed, the corresponding counter

configuration shall be based on the HS_WriteParameter function block configuration and no longer based on the HSIO parameter page.

5.7.2. HS_PresetValue

Counter count value preset function block.

- Command format

Commands	Name	LD performance
HS_PresetValue	Counter count preset	

- Input variables

Name	Description	Type	Range	Initial value
Counter	Pulse axis index	BYTE	0 - 7	
bExecute	Execution switch, edge triggered	BOOL	TRUE, FALSE	FALSE
dCountPresetValue	Counter value preset initial value	DINT	Follow data type	0

- Output variable

Name	Description	Type	Range	Initial value
bBusy	Execution flag	BOOL	TRUE, FALSE	
bDone	Done flag	BOOL	TRUE, FALSE	-
bError	Fault flag	BOOL	TRUE, FALSE	
eErrorID	Error code	HSMC_ERROR_CODE	-	

- Function block description

This function block is used to set the initial value of the counter. When the function block bExecute is set to rising edge, the counter value will be set to dCountPresetValue. The counter preset value operation can be completed in the current task cycle. The output status of bBusy is TRUE in the preset process, and after the preset operation is completed, the output status of bDone becomes TRUE and the output status of bBusy becomes FALSE.

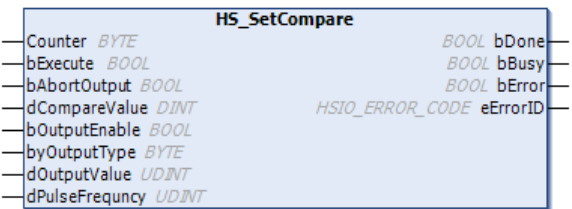
[Note]

The counter preset value should be a value between the minimum and maximum count value set by the counter HS_Counter function block, otherwise the counter counting will fail.

5.7.3. HS_SetCompare

Counter comparison output function block.

- Command format

Commands	Name	LD performance
HS_SetCompare	Counter comparison output	

- Input variables

Name	Description	Type	Range	Initial value
Counter	Pulse axis index	BYTE	0 - 7	
bExecute	Execution switch, edge triggered	BOOL	TRUE, FALSE	FALSE
bAbortOutput	Close the current count comparison output configuration	BOOL	TRUE, FALSE	FALSE
dCompareValue	Counter comparison value	DINT	Follow data type	0
bOutputEnable	Whether the hardware comparison output port is enabled FALSE: Do not perform hardware port output TRUE: Perform hardware port output	BOOL	TRUE, FALSE	FALSE
byOutputType	Compare output types 0 means setting time mode, 1 means output pulse mode	BYTE	0, 1	0
dOutputValue	Compare output sustained value It indicates the value from when the output is on to off after comparison; the unit of time is us, and the unit of pulse is unit.	DINT	Follow data type	0

dPulseFrequency	Pulse output frequency, effective only when byOutputType is 1.	UDINT	Follow data type	0
-----------------	--	-------	------------------	---

● Output variable

Name	Description	Type	Range	Initial value
bBusy	Execution flag	BOOL	TRUE, FALSE	
bDone	Done flag	BOOL	TRUE, FALSE	-
bError	Fault flag	BOOL	TRUE, FALSE	
eErrorID	Error code	HSMC_ERROR_CODE	-	

● Function block description

This function block is used for the counter comparison output function. The comparison output function takes effect when the function block bExecute is rising edge.

1. The function block bAbortOutput is used to close the comparison output configuration under the current comparison value dCompareValue. When bAbortOutput is TRUE, it closes the comparison output configuration with the previously configured comparison value dCompareValue;

2. The function block bOutputEnable is used to set whether the hardware output port is enabled. When bOutputEnable is FALSE, it means that the hardware output will not be enabled after the counter comparison is successful. When bOutputEnable is TRUE, it means that the hardware output will be enabled after the counter comparison is successful.

3. The function block byOutputType is used to set the execution mode of hardware output when the counter compares consistency. There are currently two execution methods, as follows:

a) The value is 0: Level output time mode, then the corresponding dOutputValue is the time from the level being set high to being pulled low (time unit: us);

b) The value is 1: Pulse output mode, then the corresponding dOutputValue is the number of pulse outputs; dPulseFrequency is the frequency of the output pulses, supporting a maximum of 200K, and the pulse duty cycle is 50%.

4. For the comparison output hardware port configuration, you can select the output mode of the corresponding port through the HSIO parameter configuration page PortOutMode. If the hardware output port mode is CNTXCPOUT (x: counter number), the configuration of the comparison output port of the corresponding counter can be completed.

[Example]

For example, if the hardware port of the comparison output of counter 0 is configured as OUT1-4 of the controller, you need to select the Cnt0CmpOut output item in OUT1-4-Mode in PortOutMode.

For example, if the hardware port of the comparison output of counter 1 is configured as OUT1-3 of the controller, you need to select the Cnt1CmpOut output item in OUT1-3-Mode in PortOutMode.

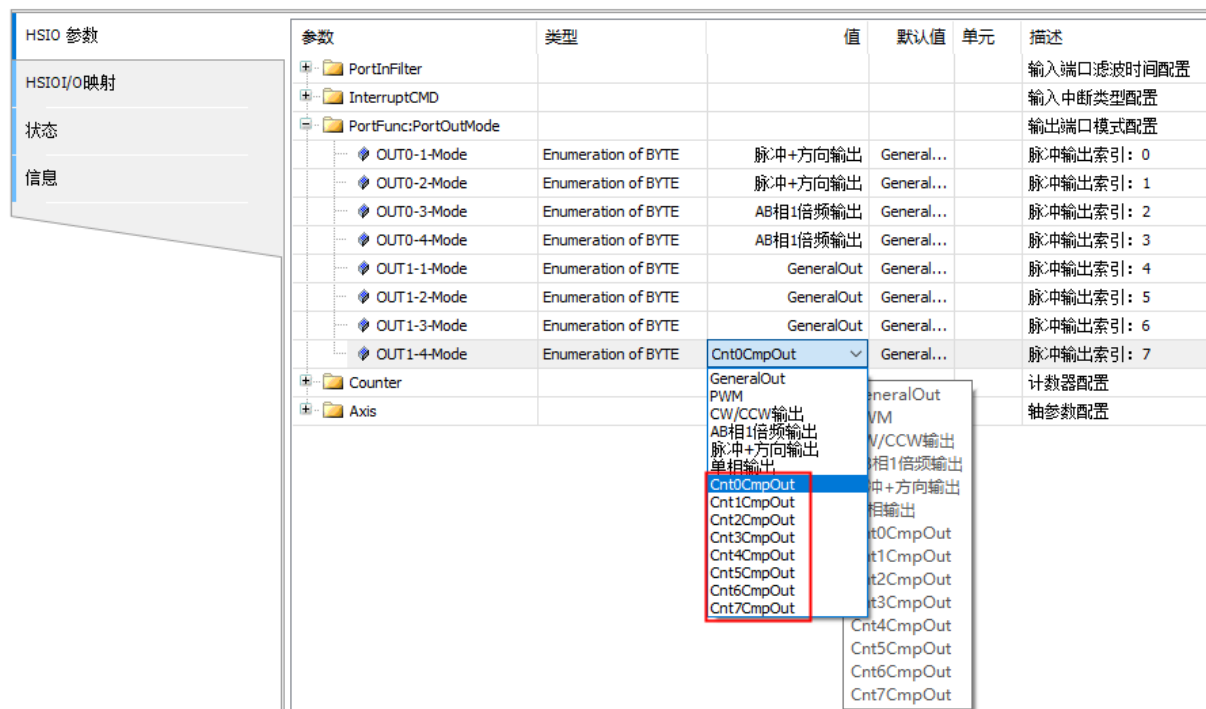
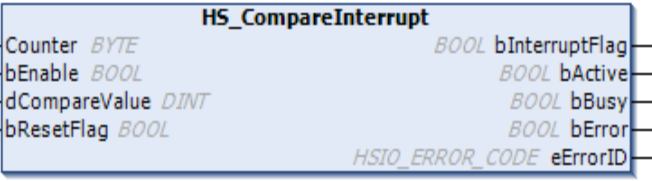


Figure 5-6-3 HSIO Parameter PortOutMode Configuration Page

5.7.4. HS_CompareInterrupt

Counter comparison output function block.

- Command format

Commands	Name	LD performance
HS_CompareInterrupt	Counter consistency comparison interrupt	

- Input variables

Name	Description	Type	Range	Initial value
Counter	Pulse axis index	BYTE	0 - 7	
bEnable	Level control	BOOL	TRUE, FALSE	FALSE
dCompareValue	Counter comparison value	DINT	Follow data type	0
bResetFlag	Reset flag for consistency comparison interrupt flag	BOOL	TRUE, FALSE	FALSE

- Output variable

Name	Description	Type	Range	Initial value
bInterruptFlag	Consistency comparison interrupt generation flag. When the value is TRUE, it indicates that the comparison output consistency interrupt is generated.	BOOL	TRUE, FALSE	
bActive	Function block valid flag	BOOL	TRUE, FALSE	
bBusy	Execution flag	BOOL	TRUE, FALSE	-
bError	Fault flag	BOOL	TRUE, FALSE	
eErrorID	Error code	HSMC_ERROR_CODE	-	

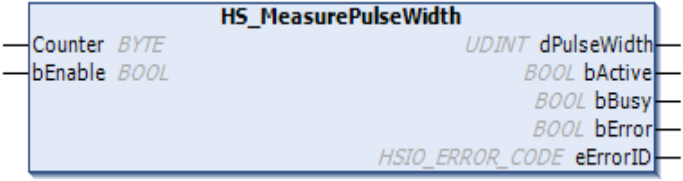
- Function block description

This function block is used to compare the consistency generation interrupt signal. The function block bEnable being level control and the value being TRUE indicates the consistency comparison output process. When the counter value is consistent with the set comparison value dCompareValue, the bInterruptFlag output state becomes TRUE. When the function block bEnable is TRUE, bInterruptFlag can be set to FALSE when the bResetFlag rising edge is executed; bActive and bBusy are always TRUE after the function block takes effect. After the function block is closed, bActive and bBusy are set to FALSE.

5.7.5. HS_MeasurePulseWight

Pulse width measurement function block.

- Command format

Commands	Name	LD performance
HS_MeasurePulseWight	Pulse width measurement	

- Input variables

Name	Description	Type	Range	Initial
------	-------------	------	-------	---------

				value
Counter	Pulse axis index	BYTE	0 - 7	
bEnable	Level control	BOOL	TRUE, FALSE	FALSE

● Output variable

Name	Description	Type	Range	Initial value
bPulseWidth	Full square wave period Unit: us	BOOL	TRUE, FALSE	
bActive	Function block valid flag	BOOL	TRUE, FALSE	
bBusy	Execution flag	BOOL	TRUE, FALSE	-
bError	Fault flag	BOOL	TRUE, FALSE	
eErrorID	Error code	HSMC_ERROR_CODE	-	

● Function block description

This function block is used to measure the pulse width of a full square wave of a high-speed input signal. When the function block bEnable is TRUE, it reads the pulse width value bPulseWidth (unit: us) of a full square wave in the current task cycle in real time; when the function block bEnable is FALSE, the pulse width measurement function is turned off.

1. Counter in the function block is the counter index number. See the correspondence between the counter index number and the high-speed input port in the details of the HS_Counter function block.

2. The corresponding relationship between counter configuration and pulse width measurement is as follows:

a) When the counter is configured for phase AB counting, the measured pulse width is the pulse width of a full square wave of the phase A signal;

b) When the counter is configured as pulse + direction counting, the measured pulse width is the pulse width of a full square wave of the pulse signal;

a) When the counter is configured as CW/CCW counting, the measured pulse width is the pulse width of a full square wave of the CW signal;

b) When the counter is configured as single-phase counting, the measured pulse width is the pulse width of a full square wave of the single-phase signal.

3. During the execution of the function block, bBusy and bActive are always TRUE. After the function block is closed, bBusy and bActive become FALSE.

5.7.6. HS_Latch

Counter latch function block.

● Command format

Commands	Name	LD performance
HS_Latch	Counter value latch	 The diagram shows the HS_Latch function block with the following inputs and outputs: - Inputs: Counter (BYTE), bEnable (BOOL), bResetFlag (BOOL) - Outputs: bLatchFlag (BOOL), dLatchValue (DINT), bActive (BOOL), bBusy (BOOL), bError (BOOL), HSIO_ERROR_CODE (eErrorID)

● Input variables

Name	Description	Type	Range	Initial value
Counter	Pulse axis index	BYTE	0 - 7	
bEnable	Level control	BOOL	TRUE, FALSE	FALSE
bResetFlag	After the latch is successful, reset bLatchFlag flag is FALSE	BOOL	TRUE, FALSE	FALSE

● Output variable

Name	Description	Type	Range	Initial value
bLatchFlag	Latch standard TRUE: It indicates completion of counter latch FALSE: It indicates that the latch is not completed	BOOL	TRUE, FALSE	
dLatchValue	Latched counter value	DINT	Follow data type	
bActive	Function block valid flag	BOOL	TRUE, FALSE	
bBusy	Execution flag	BOOL	TRUE, FALSE	-
bError	Fault flag	BOOL	TRUE, FALSE	
eErrorID	Error code	HSMC_ERROR_CODE	-	

● Function block description

This function block is used to latch the counter value after the signal is triggered. When the function block bEnable is TRUE, it means that the counter latch function is effective; when the function block bEnable is FALSE, the counter latch value function is turned off. According to the latch port and latch mode set by HS_WriteParameter, when the latch input port meets the set latch mode, dLatchValue is output as the counter value in the current state.

1. Counter in the function block is the counter index number. See the correspondence between the counter index number and the high-speed input port in the details of the HS_Counter function block.
2. After the counter latch is completed, the bLatchFlag output status changes to TRUE; when the latch port performs multiple latch operations, the latest latch value shall prevail;
3. bResetFlag in the function block is used for latch reset. When the bResetFlag input state is rising edge, the function block will actively reset the bLatchFlag latch state. The bLatchFlag output state changes to FALSE, and the latch operation can continue;
4. The counter latch port configuration and latch mode need to set the byLatchPortSel latch port and byLatchTrgMode latch mode through the HS_WriteParameter function block:
 - a) The byLatchPortSel latch input port configuration selects the corresponding input port number according to the HSIO_INPUT_PORT type definition;
 - b) byLatchTrgMode indicates that when the input port generates an edge change mode, it will start to record the current counter value accordingly.

Table 5-6-6 Latch Port Trigger Mode

byLatchTrgMode value	Input port trigger mode
0	Do not perform latch recording
1	Rising edge latch
2	Falling edge latch
3	Latch is performed on both rising and falling edges

5.7.7. HS_WriteParameter

Counter writing parameter function block.

● Command format

Commands	Name	LD performance
----------	------	----------------

HS_WriteParameter	Counter writing parameters	HS_WriteParameter Counter <i>BYTE</i> bExecute <i>BOOL</i> byCntMode <i>BYTE</i> bCntMethod <i>BOOL</i> byLatchPortSel <i>HSIO_INPUT_PORT</i> byLatchTrgMode <i>BYTE</i> <i>BOOL</i> bDone <i>BOOL</i> bBusy <i>BOOL</i> bError <i>HSIO_ERROR_CODE</i> eErrorID
-------------------	----------------------------	---

● Input variables

Name	Description	Type	Range	Initial value
Counter	Pulse axis index	BYTE	0 - 7	
bExecute	Execution switch, edge triggered	BOOL	TRUE, FALSE	FALSE
byCntMode	Counter counting mode 0: Not used as a counter, level sampling; 1: Single-phase counting; 2: Phase AB single frequency; 3: Phase AB quadruple frequency; 4: CW/CCW; 5: Pulse + direction; 6: Phase AB double frequency	BOOL	TRUE, FALSE	FALSE
bCntMethod	Counting method FALSE: Loop counting TRUE: Linear counting	BOOL	TRUE, FALSE	FALSE
byLatchPortSel	Counter latch port configuration Configure according to the detailed explanation of the HS_Latch function block	HSIO_INPUT_PORT	-	0
byLathcTrgMode	Counter latch mode Configure according to the detailed explanation of the HS_Latch function block	BYTE	0 - 3	0

● Output variable

Name	Description	Type	Range	Initial value
bDone	Done flag	BOOL	TRUE, FALSE	
bBusy	Execution flag	BOOL	TRUE, FALSE	-
bError	Fault flag	BOOL	TRUE, FALSE	
eErrorID	Error code	HSMC_ERROR_CODE	-	

● Function block description

This function block is used to configure high-speed counter parameters. The counter parameter setting is executed at the rising edge of the function block bExecute. When HS_WriteParameter is not enabled, the high-speed counter configuration is based on the HSIO parameter page value in the LocalHSIO device.

5.8. High speed input application

5.8.1. Counter phase AB counting application

[Example]

Counter 0 is set to the phase AB phase single frequency counting mode. The minimum value of the counter is -999999 and the maximum value is 999999. In the loop counting mode, the following steps need to be performed:

1. Counter configuration

The Enable of counter Counter0 is set to TRUE, which means starting Counter 0; CounterMode is set to phase AB single frequency counting; CounterMethod is set to FALSE loop counting mode.

HSIO 参数	参数	类型	值	默认值	单元	描述
HSIO1/O映射	PortInFilter					输入端口滤波时间配置
状态	InterruptCMD					输入中断类型配置
信息	PortFunc:PortOutMode					输出端口模式配置
	Counter					计数器配置
	Counter0					
	Enable	BOOL	TRUE	FALSE		启用计数器: FALSE : Disable; TRUE: Enable;
	CounterMode	Enumeration of BYTE	AB相1倍频计数	不作计数...		设定计数模式
	CounterMethod	BOOL	FALSE	FALSE		设定计数方式:FALSE表示循环计数; TRUE表示线性计数
	Counter1					
	Counter2					
	Counter3					
	Counter4					
	Counter5					
	Counter6					
	Counter7					

Figure 5-8-1-1 Counter Configuration Items

2. Hardware wiring

If the high-speed input control needs to be connected to an external AB input device, you need to connect IN0-1 to the phase A of the external device and IN0-1 to the phase B of the external device. In NPN mode, you also need to connect one common terminal COM on the input side to 24V.

3. Function block execution

Set the preset value of the HS_PresetValue function block of Counter 0 to 10000. The minimum value of dCountMinValue in the HS_Counter function block is set to -9999999, and the maximum value of dCountMaxValue is 9999999. The counting mode is FALSE, which means that when the positive pulse inputs, the counter value is positive accumulation. Firstly, the HS_PresetValue function block is executed to set the initial value of the counter, and then the HS_Counter function block bEnable is TRUE to start the counter counting function, and the function block dCounterValue feeds back the counter counting value in real time; when the HS_Counter function block bEnable is FALSE, the counter counting function is turned off.

```

1  ● HS_PresetValue (
2      Counter[0] := 0,
3      bExecute[TRUE] := preset_done[TRUE],
4      dCountPresetValue[10000] := 10000,
5      bDone[TRUE] => preset_done[TRUE],
6      bBusy=> ,
7      bError=> ,
8      eErrorID=> );
9
10 ● HS_Counter (
11     Counter[0] := 0,
12     bEnable[TRUE] := counter_enable[TRUE],
13     bDirection[FALSE] := FALSE,
14     dCountMinValue[-9999999] := -9999999,
15     dCountMaxValue[9999999] := 9999999,
16     bResetFlag:= ,
17     dCountValue[10000] => counter_value[10000],
18     bOverUpperRange=> ,
19     bOverLowerRange=> ,
20     bActive=> ,
21     bBusy=> ,
22     bError=> ,
23     eErrorID=> );

```

Figure 5-8-1-2 Function Block Execution Display

5.8.2. Counter comparison output application

[Example]

Set the comparison output port of Counter 6 to OUT1-4. After the comparison, the port output is pulse output. The number of output pulses is 100000, the pulse frequency is 10000, and the Counter 6 is in phase AB counting mode. The minimum value of the counter is -999999, and the maximum value is 999999. In loop counting mode, you need to perform the following steps:

1. Counter configuration

Please refer to the Counter configuration method in “5.8.1 Counter AB Counting Application”.

2. Output port configuration

Double-click to select the LocalHSIO device, enter the HSIO parameter page, and set the OUT1-4-Mode output type in PortOutMode to Cnt6CmpOut (the comparison output port of Counter 6).

HSIO 参数

HSIO I/O映射

状态

信息

参数	类型	值	默认值	单元	描述
PortInFilter					输入端口滤波时间配置
InterruptCMD					输入中断类型配置
PortFunc:PortOutMode					输出端口模式配置
OUT0-1-Mode	Enumeration of BYTE	AB相1倍频输出	GeneralOut		脉冲输出索引: 0
OUT0-2-Mode	Enumeration of BYTE	AB相1倍频输出	GeneralOut		脉冲输出索引: 1
OUT0-3-Mode	Enumeration of BYTE	GeneralOut	GeneralOut		脉冲输出索引: 2
OUT0-4-Mode	Enumeration of BYTE	GeneralOut	GeneralOut		脉冲输出索引: 3
OUT1-1-Mode	Enumeration of BYTE	GeneralOut	GeneralOut		脉冲输出索引: 4
OUT1-2-Mode	Enumeration of BYTE	GeneralOut	GeneralOut		脉冲输出索引: 5
OUT1-3-Mode	Enumeration of BYTE	GeneralOut	GeneralOut		脉冲输出索引: 6
OUT1-4-Mode	Enumeration of BYTE	Cnt6CmpOut	GeneralOut		脉冲输出索引: 7
Counter					计数器配置
Counter0					
Counter1					
Counter2					
Counter3					
Counter4					
Counter5					
Counter6					
Enable	BOOL	TRUE	FALSE		启用计数器: FALSE : Disabl...
CounterMode	Enumeration of BYTE	AB相1倍频计数	不作计数...		设定计数模式
CounterMethod	BOOL	FALSE	FALSE		设定计数方式:FALSE表示循...
Counter7					

3. Function block execution

Set the initial default value dCountPresetValue of counter Counter6 to 10000. Counter 6 is in phase AB counting mode. The minimum value of the counter is -999999 and the maximum value is 999999. The loop counting mode is used.

The comparison value dCompareValue of the HS_SetCompare function block is set to 20000. The enabled hardware output flag bOutputEnable is set to TRUE, the port

output mode is set to pulse output after the comparison. The corresponding byOutputType value is set to 1, the number of output pulses dOutputValue is set to 100000, and the pulse frequency dPulseFrequency is set to 10000 Pulse/s. The comparison output configuration is executed when bExecute performs rising edge.

Functional performance: When the count value of Counter6 is equal to the dCompareValue setting value, the Out1-4 hardware port generates pulse output, the pulse output frequency is the dPulseFrequency value, and the number of pulses generated is the dOutputValue value.

```

1  ● HS_PresetValue (
2      Counter6 := 6,
3      bExecute TRUE := preset_execute TRUE,
4      dCountPresetValue 10000 := 10000,
5      bDone TRUE => preset_done TRUE,
6      bBusy=> ,
7      bError=> ,
8      eErrorID=> );
9
10 ● HS_Counter (
11     Counter6 := 6,
12     bEnable TRUE := counter_enable TRUE,
13     bDirection FALSE := FALSE,
14     dCountMinValue -9999999 := -9999999,
15     dCountMaxValue 9999999 := 9999999,
16     bResetFlag:= ,
17     dCountValue 10000 => counter_value 10000,
18     bOverUpperRange=> ,
19     bOverLowerRange=> ,
20     bActive=> ,
21     bBusy=> ,
22     bError=> ,
23     eErrorID=> );
24
25 ● HS_SetCompare (
26     Counter6 := 6,
27     bExecute TRUE := compare_execute TRUE,
28     bAbortOutput FALSE := FALSE,
29     dCompareValue 20000 := 20000,
30     bOutputEnable TRUE := TRUE,
31     byOutputType 1 := 1,
32     dOutputValue 100000 := 100000,
33     dPulseFrequency 10000 := 10000,
34     bDone=> ,
35     bBusy=> ,
36     bError=> ,
37     eErrorID=> );
--

```

Figure 5-8-1-2 Function Block Execution Display

5.9.HSIO_ERROR_CODE error code

Currently, the error code types generated within the high-speed input and output function blocks are as follows. Users can solve the error message based on the error content and recommended processing methods.

输入/输出 文档

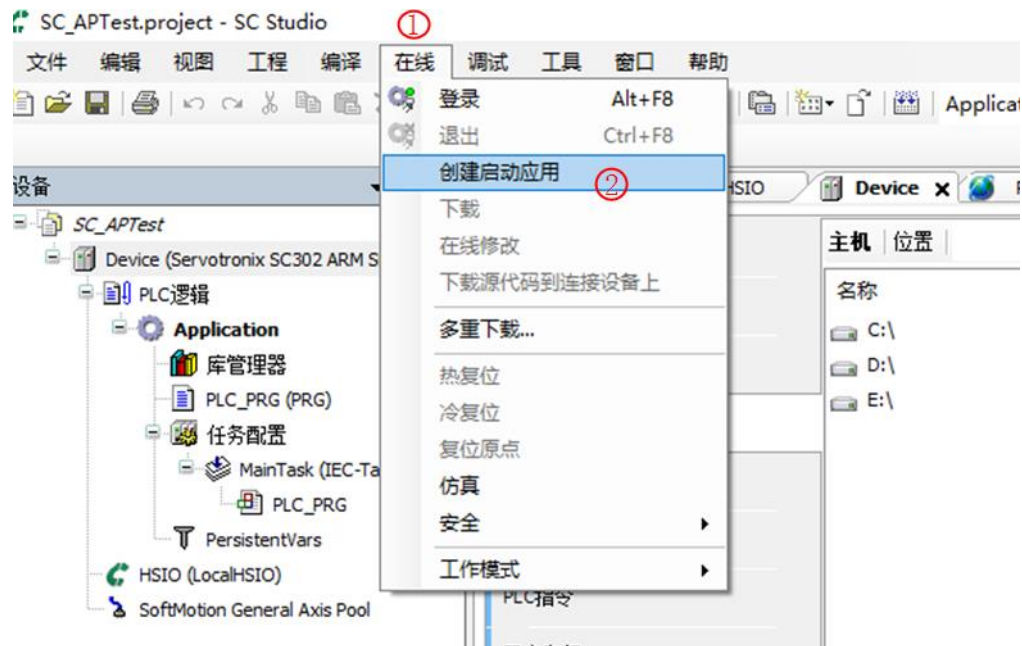
ENUM HSIO_ERROR_CODE					
名称	类型	继承自	地址	初始化	注释
NO_ERROR	INT			0	
PARAMETER_ERROR	INT			1	功能块参数设置错误
COMMAND_ERROR	INT			2	指令错误,如两个方向的JOG使能同时有效
HWLIMITS_ALARM	INT			3	限位触发
AXIS_BUSY	INT			4	轴繁忙, 处理其他指令中
HOME_VEL_NOT_MATCHING	INT			5	回原速度不匹配。如: 低速比高速高或者相等
ACC_DEC_SET_ABNOMAL	INT			6	功能块加速度与减速度需要相同, 否则可能出错
VELOCITY_EXCEEDED_200KHZ	INT			7	目标速度超过200KHZ
MOTION_PARAMETER_NO_CONFIG	INT			8	轴相关参数未配置, 请使用HMC_WriteParameter正确配置参数
MOTION_POWER_OFF	INT			9	轴未上电使能, 请执行HMC_POWER执行上电操作
HOME_PORT_SELECT_ERROR	INT			10	轴原点信号端口设置冲突, 高速计数器已占用, 请选择其他端口或关闭对应计数器使能
HOMING_ABNORMAL_ERROR	INT			11	轴回原点中止, 检测到异常原点信号有效, 请检查原点开关信号是否存在干扰
LOCALHSIO_COUNTER_CONFIG...	INT			12	当前计数器无效配置, 请检查LocalHSIO Counter配置是否正确, 或者参数配置功能块参数
LOCALHSIO_COUNTER_NO_ENABLE	INT			13	当前计数器未启用, 请检查LocalHSIO Counter Enable配置
LOCALHSIO_AXIS_OUTPUT_CONF...	INT			14	当前轴输出端口配置无效, 请检查LocalHSIO OutputMode配置
INVALID_TASK_CYCLE_TIME	INT			15	无效的插补任务周期,请检测LocalHSIO设备中的任务
SYSTEM_ERROR	INT			100	系统层逻辑故障, 请保持状态联系技术支持人员

Figure 5-9-1 HSIO_ERROR_CODE Error Code and Details

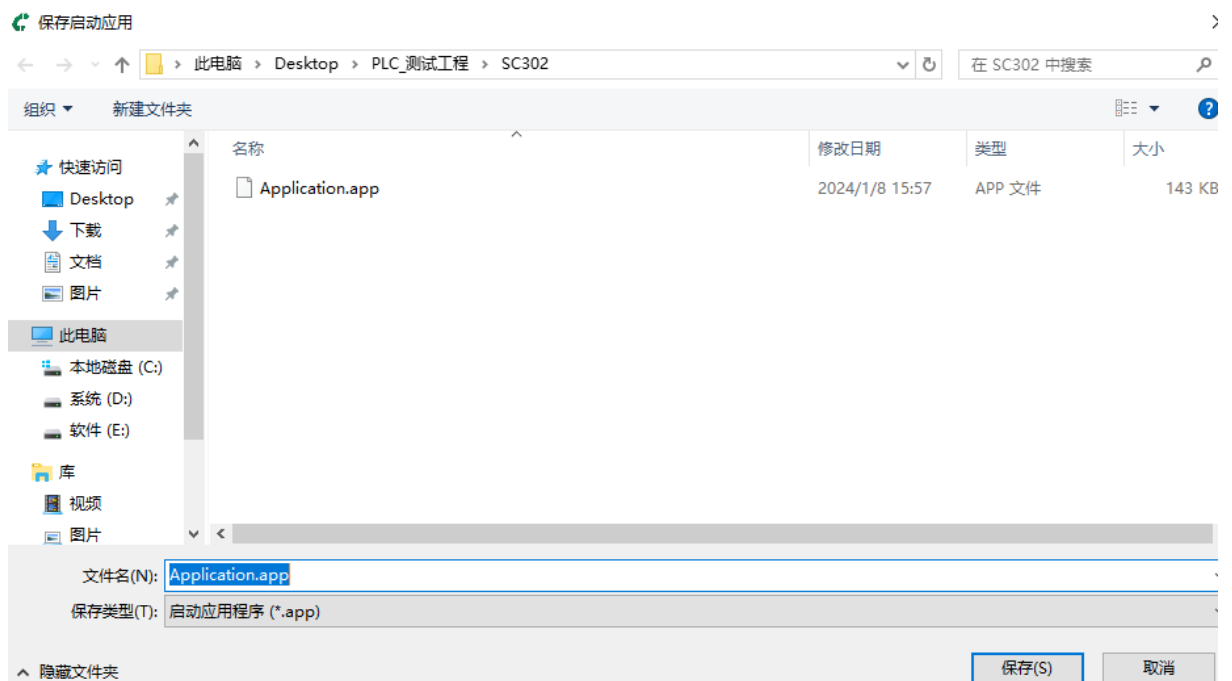
6. Use USB disk to update applications

6.1 Operating procedure

(1) Step 1: Compile the CODESYS project on the PC, create a startup application, and generate files



Then choose the save path:



PLC_测试工程 > SC302

名称	修改日期	类型	大小
Application.app	2024/1/8 15:57	APP 文件	143 KB
Application.crc	2024/1/8 15:57	CRC 文件	1 KB

在 SC302 中

The finally generated Application.app and Application.crc are the final files used.

(2) Step 2: Copy the update file to /updatefile/ in the specified root directory of the USB disk

➤ Create a new updatefile directory in the root directory of the U disk, and copy the application files Application.app and Application.crc of codesys to the /updatefile directory of the U disk;

Note: Only supporting NTFS/FAT32 format USB disks

(3) Step 3: Update on SC302 controller

➤ When the controller is not powered on, insert the USB disk and then power on to automatically complete the update of the CODESYS program. After the normal update, refer to the flashing condition of the RUN LED indicator, which is normally on.

致力于成为自动化设备领域的核心综合解决方案提供商

DEVOTING TO BEING A CORE COMPREHENSIVE SOLUTION PROVIDER IN THE FIELD OF INDUSTRIAL AUTOMATION EQUIPMENT