Diamond Plus Series Servo Drive

User Guide

Preface

Thanks for using Diamond Plus Series Servo Drive.

The Diamond Plus series servo drive is a compact AC servo drive with low power (0 – 750 W) and low voltage but high performance, developed by ISMC company. It supports EtherCAT, CANopen and Modbus communication and is equipped with intelligent debugging tools with powerful functions such as full Chinese interface, one-click import of parameters and graphic debugging. And it supports multiple motor types, including permanent magnet synchronous servo motor, DC brushless motor, linear motor, torque motor, voice coil motor and stepping motor, and rich feedback types, including incremental, absolute and Hall. This series of servo takes up little space but features high precision, high performance and high reliability, which is suitable for electronics manufacturing, robotics, semiconductor, warehousing, transportation, logistics, medical, traditional manufacturing, etc.

This user manual includes product introduction, installation, parameter settings, debugging, troubleshooting, notes related to daily maintenance, communication protocols and cases, etc. For users who do not use Diamond Plus series servo drive before, please carefully read this manual. For any questions about function and performance, please contact our technical team.

When opening the box, please carefully check the following:

- Please check if the model of the received product matches that of the ordered one according to the nameplate.
- Please check product accessories, and check the drive terminals and connectors (it is recommended to use the cables recommended by the manufacturer), and check the user manual.
- Please check if there is any damage to the product during shipping.

If there is a problem with any of the above, please contact your supplier or us.

Please note when using.

 Danger	● Before powering on, please check the wiring to make sure that the supply voltage is not greater than the voltage specified in the product specification and that the current is not greater than the maximum peak current. ● Please avoid short circuit and abnormal connection between U, V, W, PE, DC+ and DC-. Otherwise, the servo drive may be burnt or electric spark may be generated to cause personal injury or death. ● To avoid electric arc and hazards to personnel and electrical contacts, it is prohibited to plug and unplug all connector cables of the servo drive while the servo drive is powered on.

 Warning	● For heat dissipation, please make sure a certain distance between servo drives, and make sure that the working environment meets the environment requirements, and add the secondary heat sink according to the actual situation. ● Please insert or remove USB after powering off. Otherwise, your PC may be damaged due to the voltage difference between the servo drive and the PC. ● Please make sure that the elevation of the servo drive is not greater than 1000 m. ● To protect the servo drive from overvoltage damage due to energy feedback, please add a brake module.
 Anti-static	● Please use equipment that meets ESD protection and make sure that servo drives are free of electrostatic sources. ● When manually operating servo drives, it is recommended to wear electrostatic clothing to avoid contact with the components of the board.
 Grounding	Please make sure that the heat sink or housing is well connected to the shield ground. Otherwise, the servo drive may be damaged or other hazards may occur.

We reserves the right to make continuous improvements to the product without notice.

The Diamond Plus series servo drive is strictly tested in accordance with IEC60068 standards for environmental testing, EMC testing, reliability testing and life testing, so as to make sure that you can reliably and stably operate with low noise in complex industrial environments.

Environmental Standards		
Storage temperature	IEC60068-2-1	• -40°C - +85 °C • -70°C - +85°C
Working temperature	IEC60068-2-2	• 0°C - +50°C • -40°C - +50°C • -55℃ - +50℃ • -70℃ - +50℃
Humidity	IEC60068-2-78	95% (No condensation)
Vibration	IEC60068-2-6	• 2<f<9, 3.5 mm • 9<f<200, 5 g • 200<f<500, 5 g,10 min
Shock	IEC60068-2-27	15 g 11 ms half-sine pulse 3 times

EMC Standards		
Voltage fluctuation	IEC61000-4-29	±10%
Voltage dips / short interruption	IEC61000 -4 29	Drop 100%/0.1s
Static electricity	IEC61000-4-2	• 4 KV contact discharge • 8KV air discharge
Fast transient	IEC61000-4-4	• 2 kV / 5 kHz power terminal • 1 kV / 5 kHz signal terminal
Electromagnetic conduction	IEC61000-4-6	0.15 MHz - 80 MHz, 10 V, 80% AM(1 kHz)
Electromagnetic radiation	IEC61000-4-3	80 MHz – 1000 MHz, 10 V/m, 80% AM(1 kHz)
Electromagnetic leakage	CISPR-16-1	C3/C4

Version Records

Date	Version	Details
2022-03	V1.0	Release 1st edition

Contents

1 Basic Information

1.1 Overview

The basic features are as follows:

- Power of the Diamond Plus series servo drive: 0 - 750 W
- Communication mode: EtherCAT, CANopen, Modbus and pulse communication
- Digital inputs and outputs: 6 digital inputs and 4 digital outputs
- Motor type: permanent magnet synchronous motor, DC brushless motor, DC brush motor, coreless motor, linear motor, voice coil motor, stepping motor
- Encoder: absolute encoders (absolute encoder protocol like Hiperface, BISS-C, SSI, Panasonic, Endat2.1/2.2, Nikon, Sanyo, Tamagawa and other mainstream protocols), incremental encoders, analog sincos encoders and Hall sensors.

Figure 1-1 Servo drive appearance

1.2 Naming

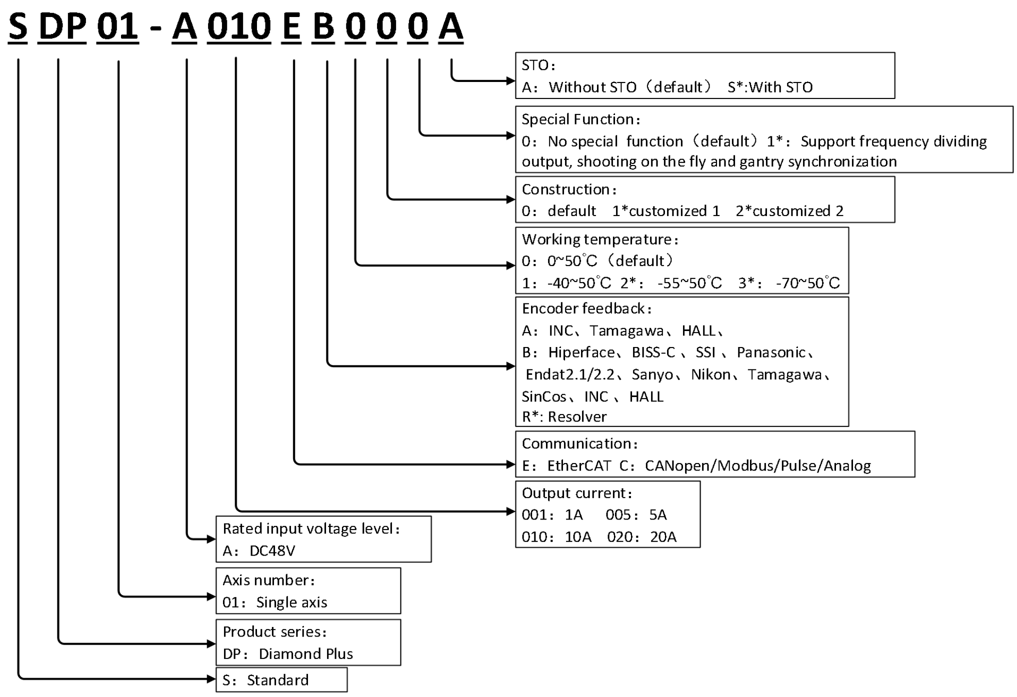


Figure 1-2 Definition of model name

Note:

1. *: it indicates non-standard.
2. R*: it refers to Resolver, only supported by CANopen communication. For EtherCAT communication, it needs to be customized.
3. Construction: 0: default (high power terminals and large heat sink); 1*: customized 1 (high power terminals and small heat sink); 2*: customized 2 (small power terminals and large heat sink).

1.3 Specification

Table 1-1 Specification

Product Model		SDP01-A001	SDP01-A005	SDP01-A010	SDP01-A020
Output power	Rated power (W)	0-50	0-200	0-400	0-750
	Rated current I_{rms} (A)	1	5	10	20
	Instantaneous peak current (A)	2.5	12.5	25	50
	Instantaneous peak current duration (s)	3	3	3	3
Input power	Rated input voltage (V)	12-60			
	Maximum input voltage (V)	72			
Communication mode	Remote	EtherCAT, CANopen, Modbus			
	Pulse	Pulse + direction control / PWM pulse width control			
	Local	USB			
Motor feedback	Absolute encoders (Hiperface, BISS-C, SSI, Panasonic, Endat2.1/2.2, Nikon, Sanyo, Tamagawa)				
	Incremental encoders (TTL, Sincos)				
	HALL				
Digital IO	Input	Isolated 6-way			
	Output	Isolated 4-way			
Simulation AI	Input	-10 - +10 V; 2-way; 12-bit resolution			
Status display	Dual color LED: red and green				
Motor control	Position, velocity and current control Current loop: 50 us; position / velocity loop: 200 us				
Protection	Short-circuit, over-current, over-temperature, over-voltage, under-voltage, over-speed, encoder communication error, etc.				
Size	62 mm * 56 mm * 34 mm				

1.4 Main Function

The product adopts advanced vector control algorithm, with high system response speed.

Table 1-2 Main function

Function	Details
Servo Control	1. The bandwidth of current loop is 4.5 kHz, and sampling frequency of current loop, velocity loop and position loop is 20 kHz, 5 kHz, 5 kHz respectively. 2. Support control of position loop, velocity loop and torque loop. 3. Support dynamic PID download. 4. Support debugging of filtering parameters. 5. Support real-time reading of servo system parameters. 6. Support dead-time compensation. 7. Support position angle compensation. 8. Position, velocity and torque commands all support 32-bit data: • Position: -2×10^9 - $+2\times10^9$ count • Speed: -2×10^9 - $+2\times10^9$ count/s • Acceleration: -2×10^9 - $+2\times10^9$ count/s^2
Motion Control	1. Based on EtherCAT (COE) protocol, fully support CIA402 position profile, velocity profile, torque profile, homing mode and cycle synchronization mode. 2. Support S-curves. 3. Support fine interpolation of position commands. 4. Motion command: please refer to "***User Manual of ISMC Servo Drive (EtherCAT Communication)***". 5. Upper computer software: support TwinCAT2.0 and TwinCAT3.0 software. 6. Debugging tool: refer to "***User Manual of ISMC Servo Drive Debugging Software***".
Feedback terminal	Supported feedback methods: 1. Absolute encoders Supported protocols: Hiperface, BISSC, SSI, Panasonic, Endat2.1/2.2, Nikon, Sanyo, Tamagawa. 2. Incremental encoders • Support digital quadrature encoding and maximum frequency 10 M without frequency multiplication and 40 M with frequency multiplication. • Support sincos analog encoders 3. Digital Hall sensor • Phase change frequency: 4 kHz • Logic level: 5 V 4. Incremental encoders + digital Hall sensors

Function	Details
Communication	Supported communication protocols: 1. EtherCAT slave • CoE (CANopen over EtherCAT) • Support distributed clocks • Support synchronization cycle mode with synchronization cycle of 1 - 4 ms 2. USB 2.0 3. Modbus 4. CAN Open
Digital IO	1. Configurable functions via DI: • Limit switch • Enable / disable movement • Motor over-temperature • Start • Stop / emergency stop • Forward / reverse • Zero calibration • Fault clearing • Reset • External fault input 2. Configurable functions via DO: • General DO • Fault output • Brake output • Target arrival

2 Installation

2.1 Dimension

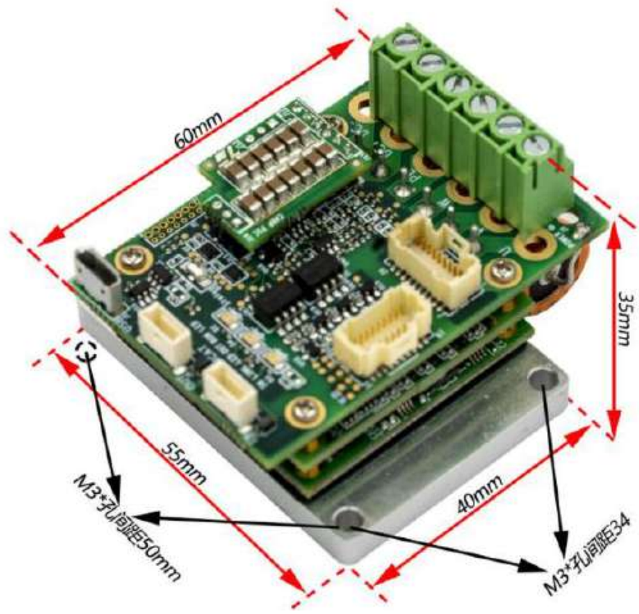


Fig. 2-1 Dimension (unit: mm)

2.2 Assembly and Disassembly

Before installation, please check the product package:

- Servo drive
- Connector corresponding to the servo drive
- User manual (scan the QR code card to access to the official website for download).

Please note the following during disassembly:

- Careful remove the box.
- Make sure the product is free of cosmetic damage.
  If there are such damages, please contact us.
- Check the model on the housing of the product and make sure the product is your desired one.
- Check if the rated voltage meets your demands.

2.3 Mechanical Installation

This servo is a pedestal type servo amplifier and failures may occur if installed in the wrong way.

2.3.1 Installation Site

- Please do not use this product in the vicinity of corrosive and flammable gas environments such as hydrogen sulfide, chlorine, ammonia, sulfur, chlorinated gases, acids, alkalis, salts, and combustible materials.
- Please do not install this product in the environments with high temperature, humid places and lots of dust and iron powder.
- Please do not use this product in a closed environment where will cause high temperature of the servo and shorten the service life.
- Please note the following:
  - ✓ For servo drives with 200W and below, there is no special requirements for installation.
  - ✓ For servo drives with 400W, please make sure that the temperature of the secondary cooling installation surface is below 55℃.
  - ✓ For servo drives with 750W, please make sure the temperature of the installation surface is below 55℃, and make sure at least 3 m/s wind convection with wind direction along the horizontal direction of PCB.

2.3.2 Environmental Condition

Table 2-1 Environmental conditions

Projects	Description
Ambient temperature	• 0: 0℃ - +50℃ • 1: -40℃ - +50℃ • 2: -55℃ - +50℃ • 3: -70℃ - +50℃
Environmental humidity	＜ 95% RH (no condensation)
Storage temperature	• -40℃ - +85℃ (no freezing) • 7-0℃ - +85℃ (no freezing)
Storage humidity	0% - 95% RH (no condensation)
Vibration	＜ 5 m/s^2

2.3.3 Installation Steps

Note: The servo drive should be vertically installed on the wall and M3 screws must be tightened. For other requirements, please refer to 2.3.1Installation Site.

1. On the back of the mounting plate, mark the position of screw hole.
   The hole spacing is shown as in Figure 2-1, and the specification of heat sink hole is M3,.
2. Tap threads according to the mark, and make sure threads have full contact.
   Note: The metal surface of the mounting plate should not be coated or painted, and if so, please scrape it off. Otherwise the electromagnetic compatibility will deteriorate.
3. Vertically mount the servo drive on the back of the mounting plate.
   Note: Please pay attention to the installation spacing, and make sure the mounting surface is in good contact.

3 System Wiring

3.1 Interface Definition

There are 9 interfaces on the servo drive. Among them, the following interfaces are external:

- J2: Pulse / frequency division interface
- J3-J4: CANpen / EtherCAT communication interface
- J5: USB communication interface
- J6: IO interface
- J7: Motor encoder feedback interface
- J9: Power input and output interface

The pin definition of each interface is shown in Figure 3-1:

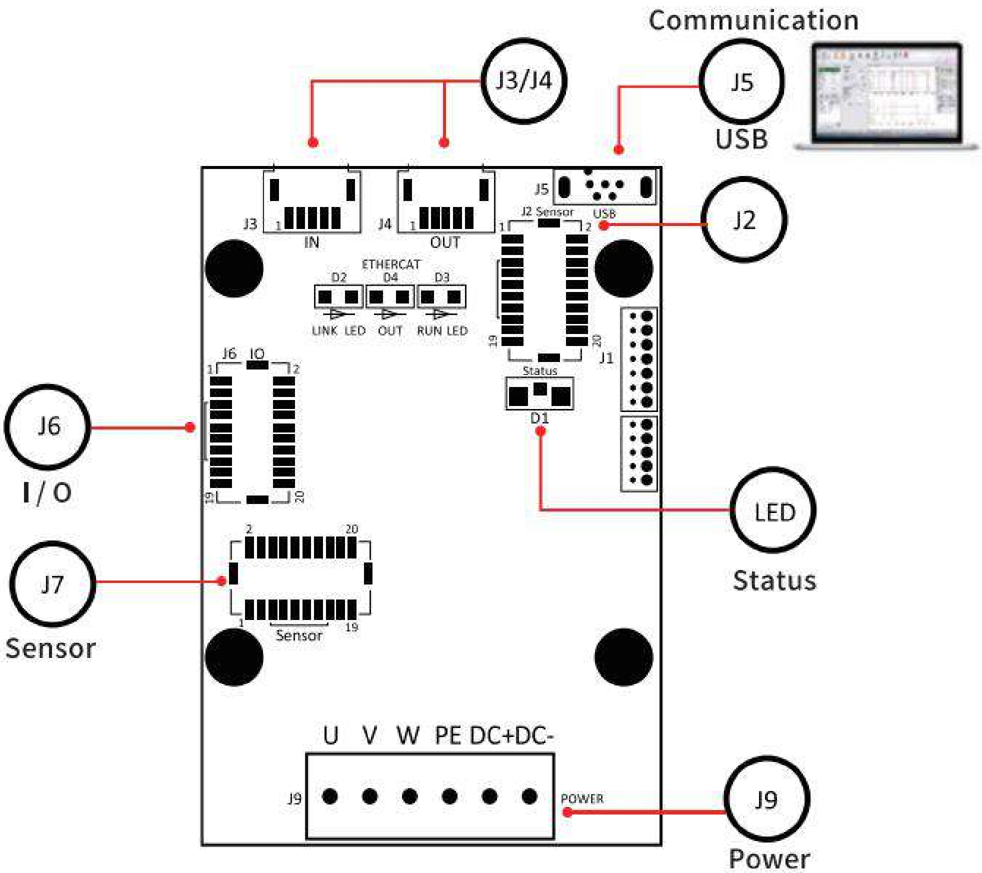


Figure3 -1 Interface definition

3.1.1 Pin Definition of J3 and J4

Table 3-1 Pin definition of J3 and J4

Interface	Pin	Pin Name
J3/J4-EtherCAT (J3-IN; J4-OUT)	1	TX+
	2	TX-
	3	RX+
	4	RX-
	5	PE
J3/J4-CANopen/Modbus	1	RS485_A
	2	RS485_B
	3	CAN_L
	4	CAN_H
	5	PE

3.1.2 Pin Definition of J5

You can communicate with the servo drive for parameter settings and debugging via the standard Micro-USB cable.

3.1.3 Pin Definition of J6

Table 3-2 Pin definition of J6

Interface	Pin	Pin Name
J6-IO interface	1	+24V_OUT
	2	DO_GND
	3	DO2_OUT
	4	DO3_OUT
	5	DO0_OUT
	6	DO1_OUT
	7	DI4_IN
	8	DI5_IN
	9	DI2_IN
	10	DI3_IN
	11	DI0_IN
	12	DI1_IN
	13	STO0
	14	DI_COM
	15	STO1
	16	STO_RET
	17	NC
	18	NC
	19	NC
	20	NC

3.1.4 Pin Definition of J7

Table 3-3 Pin definition of J7

Interface	Pin	Pin Name				
		Absolute	Incremental	Hall	Analog	Power supply
J7-motor encoder feedback interface	1					PE
	2					GND
	3		INC_A+/Sin+			
	4		INC_A-/Sin-			
	5		INC_B+/Cos+			
	6		INC_B-/Cos-			
	7		INC_Z+			
	8		INC_Z-			
	9			HALL_U		
	10					5V
	11			HALL_W		
	12			HALL_V		
	13	ABS_CLK+				
	14	ABS_CLK-				
	15				AI1+	
	16				AI1-	
	17				AI2+	
	18				AI2-	
	19	ABS_DATA+				
	20	ABS_DATA-				

3.1.5 Pin Definition of J2

Pin	Signal	Function
1	5V	5V
2	3.1.5.1 GND	GND
3	Encoder_2A+	PULSE+/PWM+
4	Encoder_2A-	PULSE-/PWM-
5	Encoder_2B+	SIGN+
6	Encoder_2B-	SIGN-
7	Encoder_2Z+	Encoder_2Z+
8	Encoder_2Z-	Encoder_2Z-
9	Encoder_3A+	PAO+
10	Encoder_3A-	PAO-
11	Encoder_3B+	PBO+
12	Encoder_3B-	PBO-
13	Encoder_3Z+	PZO+
14	Encoder_3Z-	PZO-
15	Encoder_4A+	Encoder_4A+
16	Encoder_4A-	Encoder_4A-
17	Encoder_4B+	Encoder_4B+
18	Encoder_4B-	Encoder_4B-
19	Encoder_4Z+	Encoder_4Z+
20	Encoder_4Z-	Encoder_4Z-

Note: J2 pulse / frequency division interface currently supports pulse input, PWM and frequency division output, the rest of interfaces are reserved for gantry function (coming soon).

3.1.6 Pin Definition of J9

Table 3-4 Pin definition of J9

Interface	Interface Pins	Pin Name
J9-power input and output interface	1	U
	2	V
	3	W
	4	PE
	5	DC+
	6	DC-

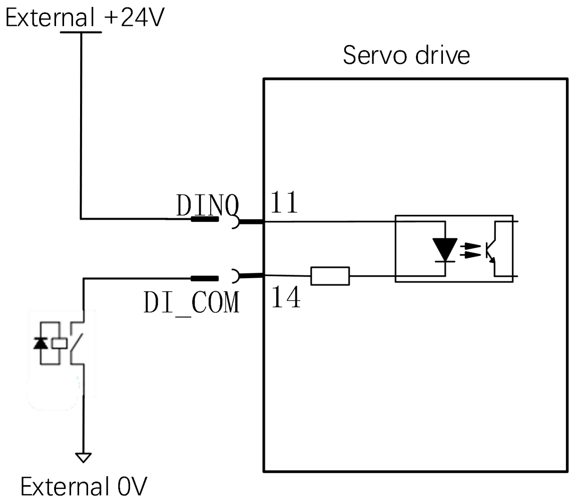
Caution:

1. For voice coil motor or DC brush motor, please connect the power cable to UV phase.
2. For two-phase four-wire stepping motor, please connect A+ and B+ of the power cable respectively to U phase and W phase, A- and B- to V phase.

3.2 IO wiring

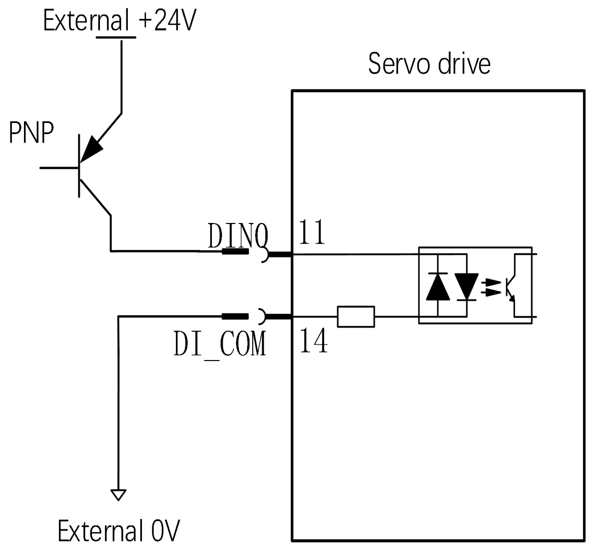
3.2.1 DI wiring

- When the upper unit uses relay to output (taking DI0 as an example)

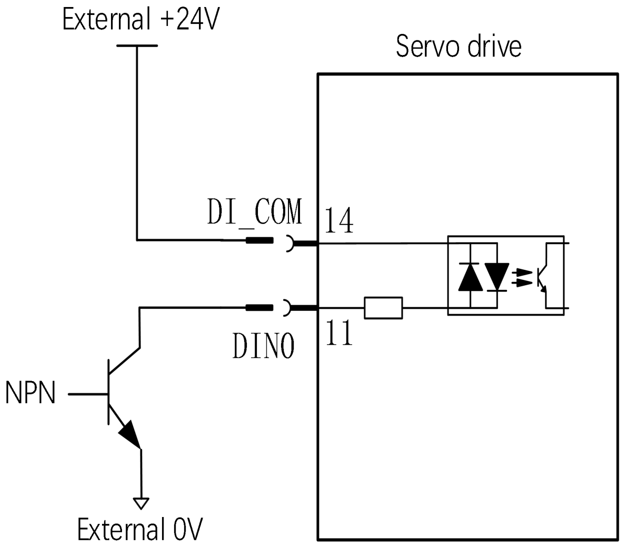


- When the upper unit uses open collector to output

  PNP connection



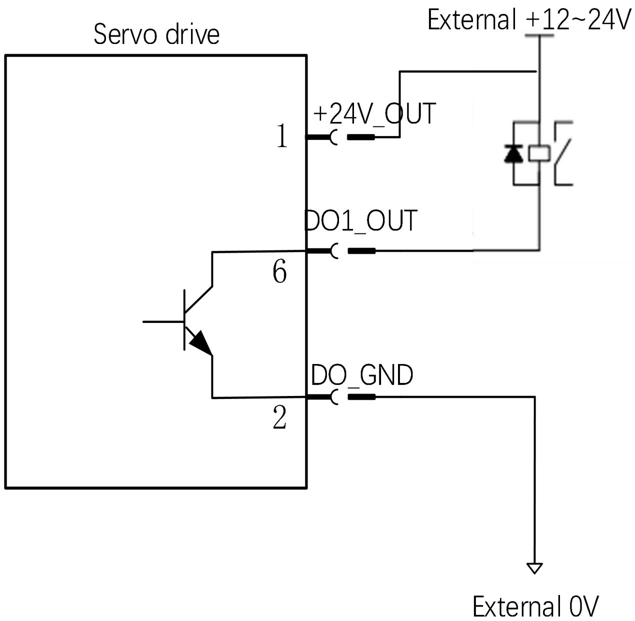
NPN connection



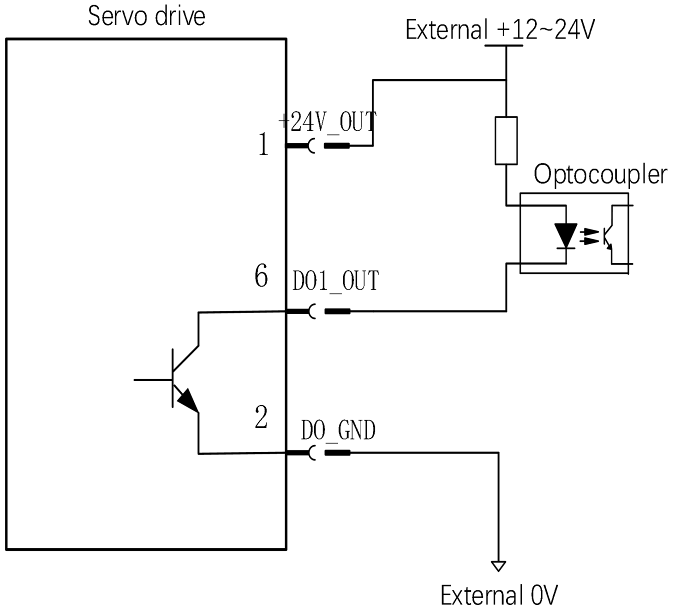
Note: The mixing case of PNP and NPN input is not supported.

3.2.2 DO Wiring

- When the upper unit uses relay to input (taking DO1 as an example).



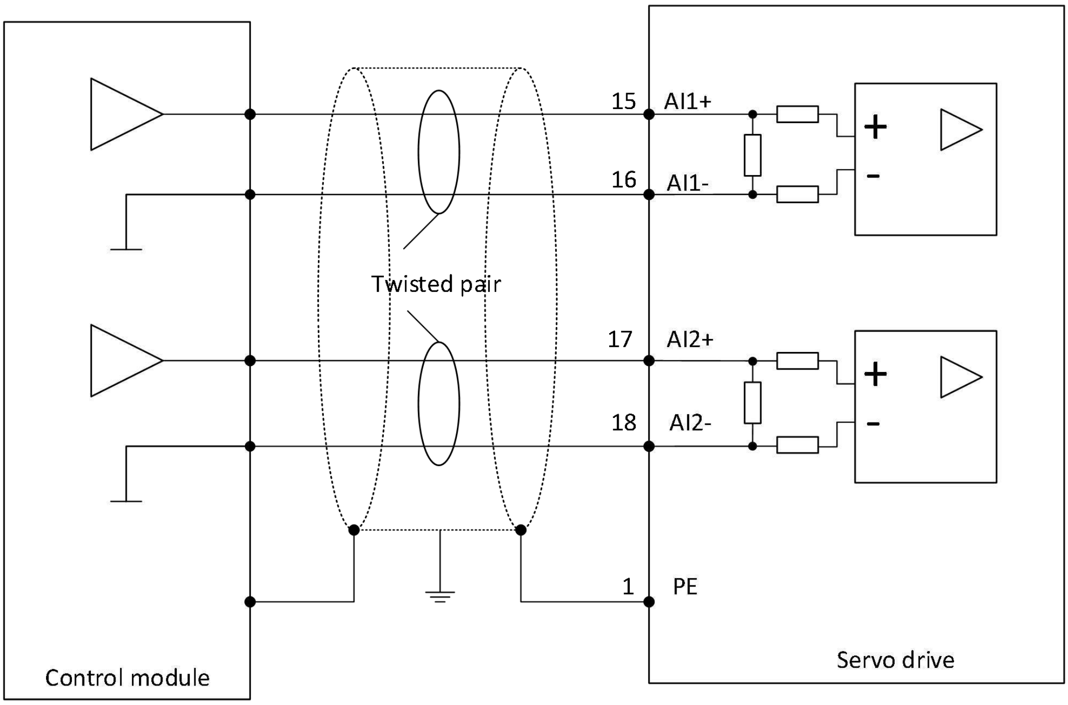
- When the upper unit uses optocoupler to input (taking DO1 as an example).



Note:

1. DO_OUT needs a pull-up resistor and current limiting resistor (it is optional. Please decide whether to use it according to the optocoupler specifications of the upper unit).
2. The maximum voltage and current of the internal optocoupler output is as follows.
   - Voltage: 30 V DC
   - Current: 400 mA DC

3.3 AI Wiring



Note:

1. The servo drive has 2 analog input circuits, i.e. AI1 and A12.
   - Input voltage: -10 - +10 V
   - AD accuracy: 12 bits
2. Please make sure the input voltage is within -12 - +12 V. Otherwise, damage to the circuit may occur.
3. Input impedance: 3.74 kΩ.
4. The upper unit can read values of 0x2413 (AI1) and 0x2414 (AI2) for external analog closed loop control.

3.4 Cables

Table 3-6 Cables

Name	Description	Length/m	Model
Power cables[*1]	Power cable between motor and servo drive with connector (for servo system ≤5 A)	1	SC-P0750100A00
		3	SC-P0750300A00
		5	SC-P0750500A00
		X (non-standard)	SC-P075xxx0A00
	Power cable between motor and servo drive with connector (for 10 A servo system)	1	SC-P1500100A00
		3	SC-P1500300A00
		5	SC-P1500500A00
		X (non-standard)	SC-P150xxx0A00
	Power cable between motor and servo drive with connector (for 20 A servo system)	1	SC-P2500100A00
		3	SC-P2500300A00
		5	SC-P2500500A00
		X (non-standard)	SC-P250xxx0A00
Encoder cables[*1]	Encoder signal cable between motor and servo drive with connector (For incremental encoders)	1	SC-E2010BA000
		3	SC-E2030BA000
		5	SC-E2050AA000
		X (non-standard)	SC-E2xxxBA000
	Encoder signal cable between motor and servo drive with connector and battery (3.6 V, it is recommended to replace it every 15 - 24 months) (For absolute encoders)	1	SC-E1010BAA00
		3	SC-E1030BAA00
		5	SC-E1050BAA00
		X (non-standard)	SC-E1xxxBAA00
Digital I/O cables[*1]	Digital I/O cable (6 inputs, 4 outputs), 2-way STO, with connector	1	SC-D010B000
		3	SC-D030B000
		5	SC-D050B000
		X (non-standard)	SC-DxxxB000

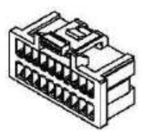
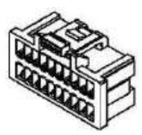
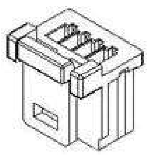
Name	Description	Length/m	Model
Communication cables*2	CANopen communication cable 1 between the servo drive and upper unit PLC	X	SC-C1XXXB000
	CANopen communication cable 2 between servo drives	X	SC-C1XXXBB00
Communication cables*2	EtherCAT communication cable 1 between the servo drive and upper unit PLC	X	SC-C2XXXBC00
	EtherCAT communication cable 2 between servo drives	X	SC-C2XXXBB00

Note:

1. It is recommended to purchase a set of cables for direct use. Otherwise, you need to purchase J2-J7 connectors separately.
2. It is recommended to purchase a set of bus communication cable, because the CAN / EtherCAT communication interface is not a standard RJ45 connector. And you can purchase USB cables by yourself.
3. Cable can be customized. For special needs, please contact us.

3.5 Wiring Accessories

Table 3-7 Wiring accessories

	Category	Connector Name	Splice pin
	Encoder connector	501189-2010 (Molex.)	5011937000
	I/O connector	501189-2010 (Molex.)	5011937000
	EtherCAT/ CAN connector	5019390500 (Molex)	5013340000

Note: The recommended cable is 28AWG.

3.6 Brake Module

3.6.1 Overview

The servo motor will generate regenerative energy in the following situations:

- The servo motor slows down and stops in acceleration and deceleration.
- The vertical axis continuously moves down
- The servo motor is generating power.

At this time, the brake module is required to consume the generated regenerative energy, so as to protect the equipment on the DC bus from overvoltage.

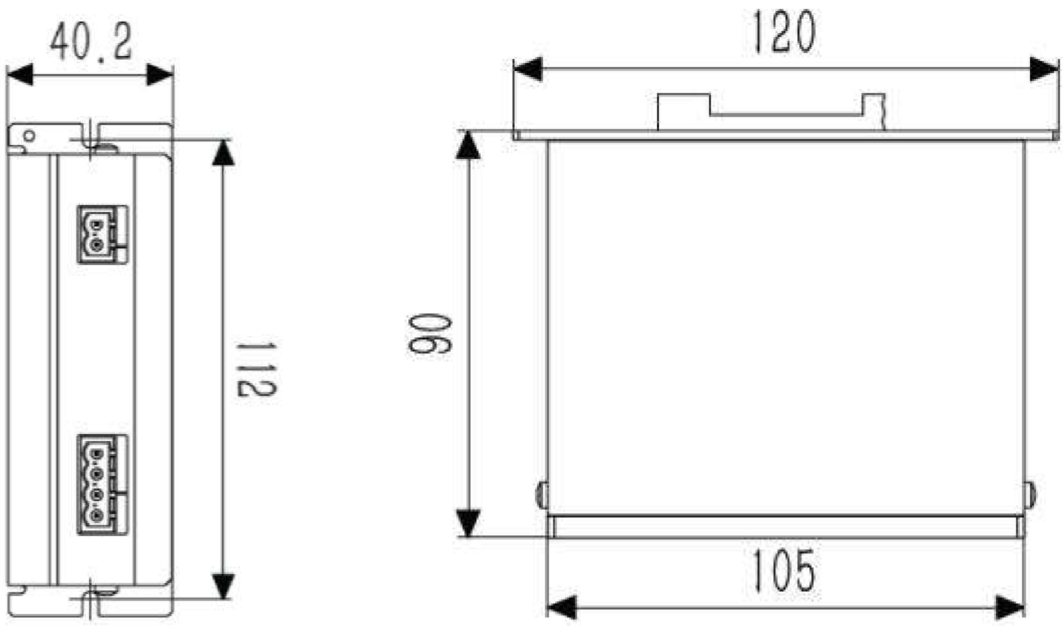


Figure3-3 Dimension of brake module

3.6.2 Principle

The brake module mainly consists of a power consumption unit, a comparator unit, and a switching tube unit. DC bus 48V enters from DC power supply by J1 and outputs to the servo drive by J2. The comparator unit monitors the bus voltage. When the bus voltage exceeds the maximum, the comparator unit outputs high level to make the switching tube unit conductive, at which time the voltage of bus pump is up and added to the unit and the power consumption unit consumes the excess energy on the bus. When the voltage returns to the minimum of normal voltage, the comparator unit outputs low level to turn off the switching tube unit. The energy returned from the servo drive will not transfer to the DC power supply due to a serial diode with high power and low conduction voltage drop. The diode can protect the DC power supply while also providing anti-reverse protection.

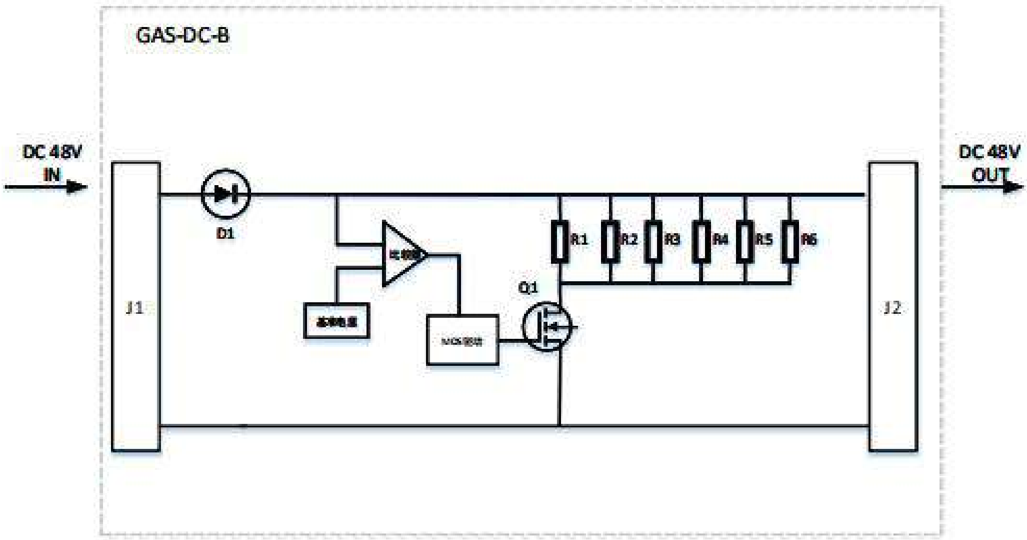


Figure3-4 Principle

A voltage comparator with hysteresis function is used in the comparator unit to avoid false operations due to signal interference. The range is 50 – 53 V.

3.6.3 Selection

In the position mode, the motor is controlled to move with a cycle of speed V. When braking, the kinetic energy is converted into electrical energy and fed back to the bus capacitance. When the independent brake voltage exceeds the releasing brake voltage, the brake module will consume the excess energy.

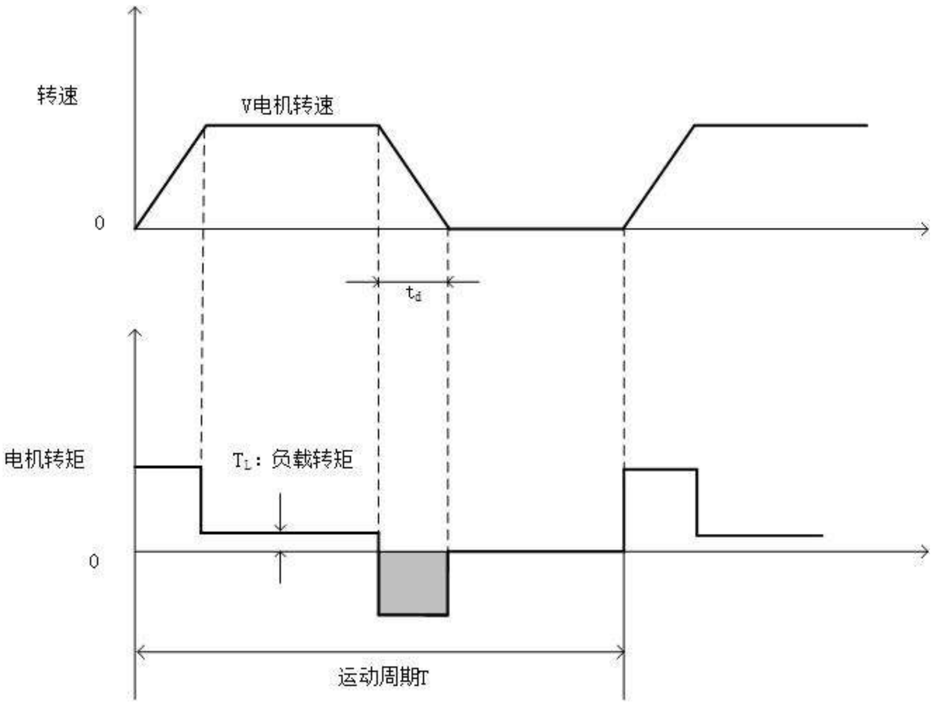


Figure 3-5 Reciprocating operation

转速：Speed
电机转矩：Motor torque
电机转速：Motor speed
负载转矩：Load torque
运动周期：Period of motion

The calculation steps are as follows.

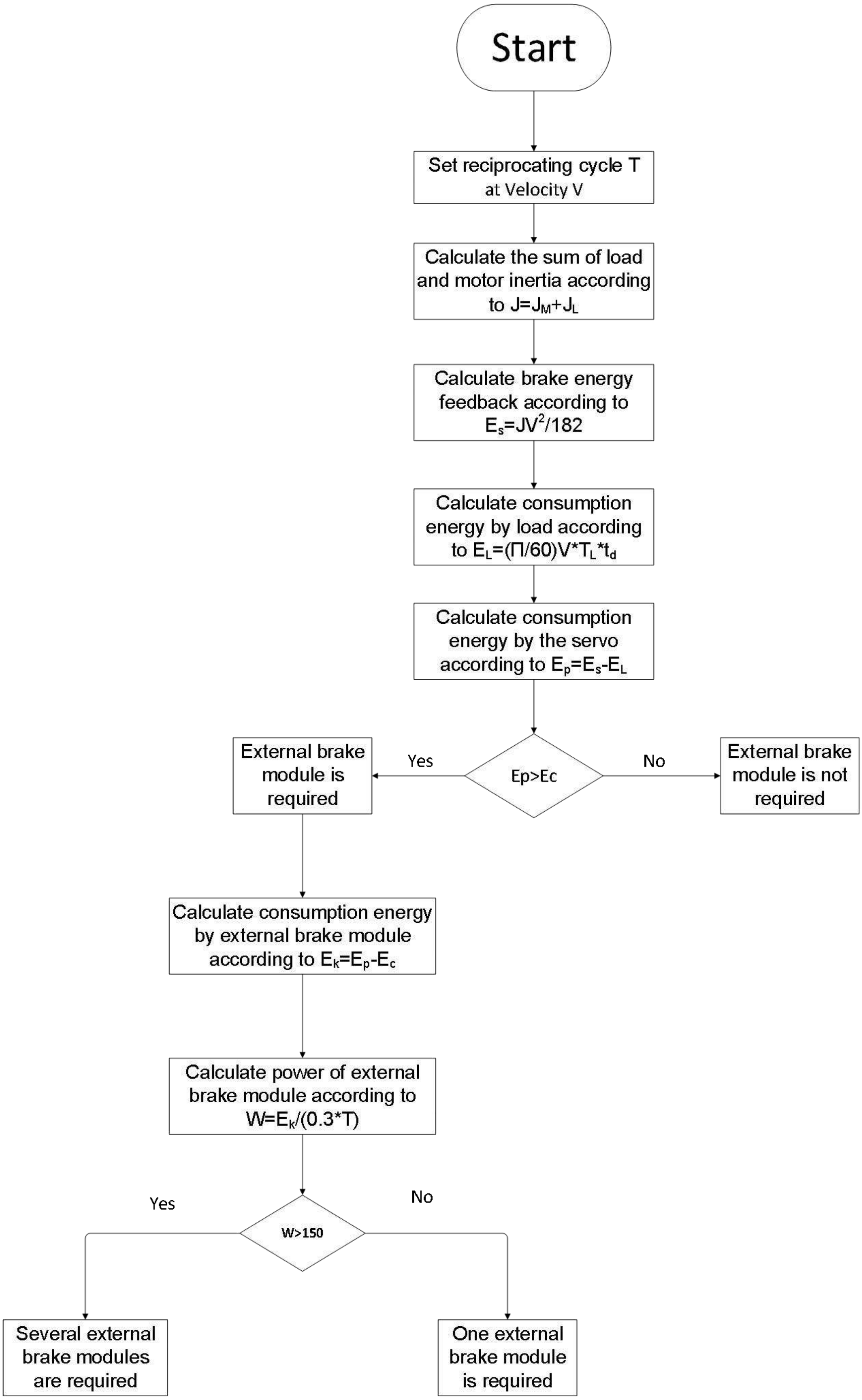


Figure 3-6 Calculation steps

Details:

Step	Calculated items	Symbols	Formula
1	Total inertia of the servo system	J	$J=J_M+J_L$ Note: Inertia is the sum of motor inertia and load inertia.
2	Total braking energy of the servo system	E_s	$E_s=JV^2/182$
3	Energy consumed by the load system during deceleration	E_L	$E_L=(\pi/60)V*T_L*td$ Note: When the loss energy of the load system is not clear, please assume E_L=0
4	Energy required to regenerate to the servo	E_p	$E_p=E_s-E_L$
5	Absorbable energy by servo drive	E_c	See the table below for details of the energy that can be absorbed by products with different specifications.
6	Energy consumed by regenerative resistor	E_k	$E_k=E_p-E_c$
7	Capacity of regenerative resistor	W	$W=E_k/(0.3*T)$ Note: 0.3 is used when the load rate of regenerative resistance is 30%.

The energy that can be absorbed by products with different specifications:

Product Series	Specification	Max Absorbable Energy (Ec/J)
Diamond/Diamond Plus	• SD-A010CA SDP-A010CA • SD-A010EA SDP-A010EA	0.084
	• SD-A020CA SDP-A020CA • SD-A020EA SDP-A020EA	0.2
Sapphire	• SS-A010CA • SS-A010EA	0.142

The brake module consists of six 25 W 120 Ω cement resistors in parallel, so a single brake module can consume 150 W energy feedback. When the brake energy of several servo drives is less than or equal to 150 W, a brake module is enough; When it is greater than 150 W, several brake modules are required and you should distribute the servo drives evenly to brake modules.

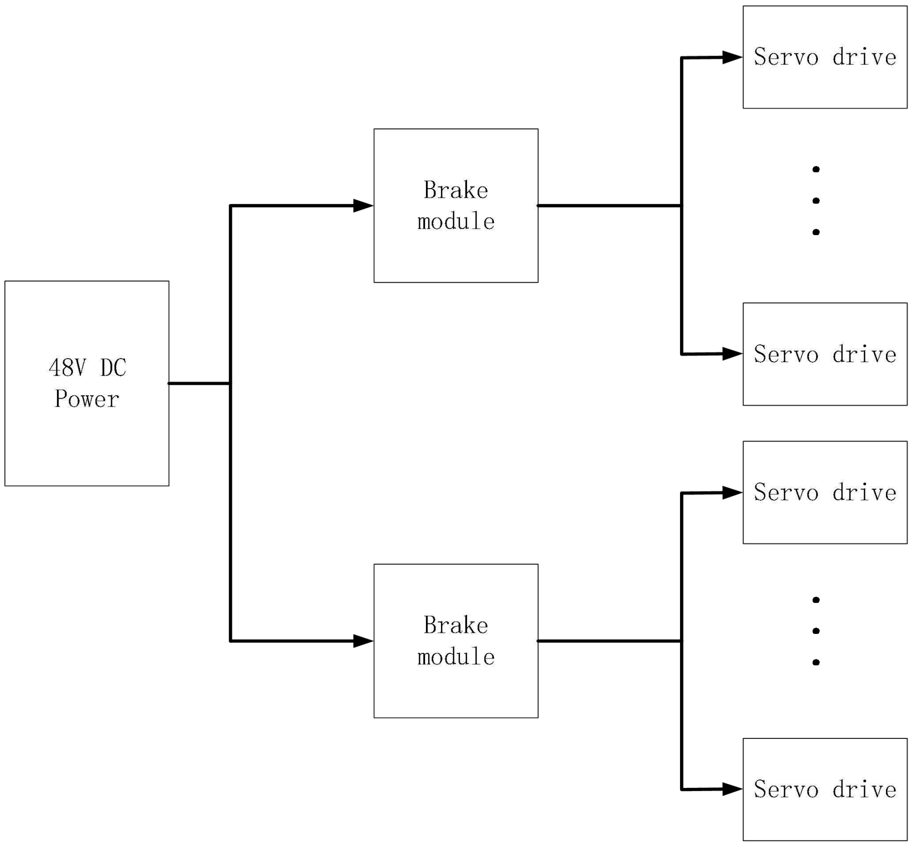
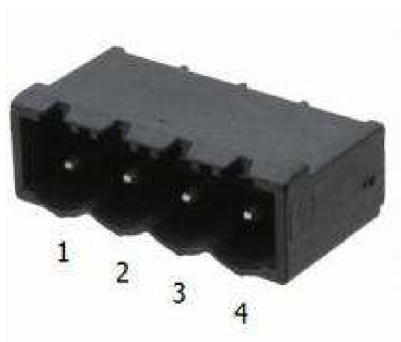


Figure 3-7 Wiring of brake module

3.6.4 Interface Definition

Figure 3-8 Interface definition

J1 power inlet	Pin	Definition
	1	48V IN+
	2	GND
• Connector (board): 1954919 (Phoenix Contact material number) • Connector (wire): 1754568 for 14 – 24 AWG or 0.2 - 1.5 mm² wire		

J2 power outlet	Pin	Definition
	1, 2	48V OUT+
	3, 4	GND
• Connector (board): 1954919 (Phoenix Contact material number) • Connector (wire): 1754584 for 14 - 24 AWG or 0.2 - 1.5 mm² wire		

3.6.5 Wiring

When the motor is working in a situation where a brake module is required, the brake module should be connected in series between the DC power supply and the power inlet wire of the servo drive.

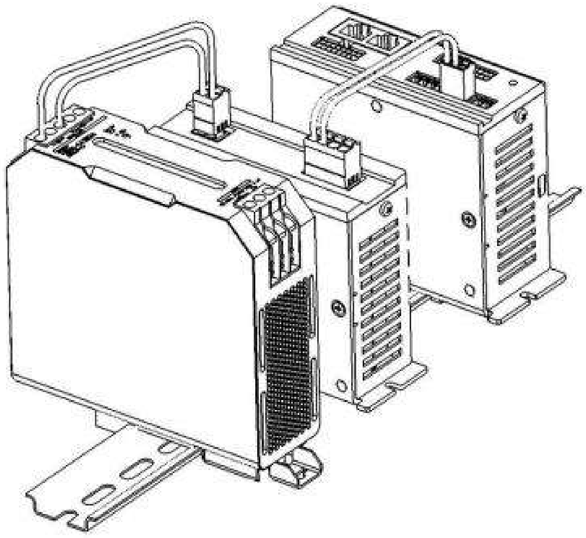


Figure 3-9 Wiring

4 Trial Run and Debugging

The iSMC software is used for trial run and debugging. For details, please refer to ***User Manual of Servo Debugging Software ISMC***.

Debugging steps:

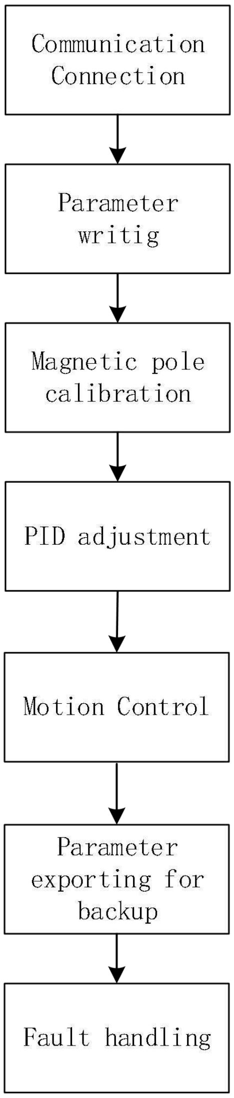


Figure 4-1 Parameter debugging steps

4.1 Communication Connection

1. Install the SMC software and USB driver.
2. Connect the upper computer and the servo drive via a USB cable (Mirco Type B).
3. Run **ISMC** software, enter the main interface, and select submenu **"Configuration"**:

Figure 4-2 Open the communication interface

4. Click **"Refresh"**, and select the port connected to the servo drive in drop-down box **"Port"**.
5. Click **"Connect"**. The result shows as in Figure 4-3.

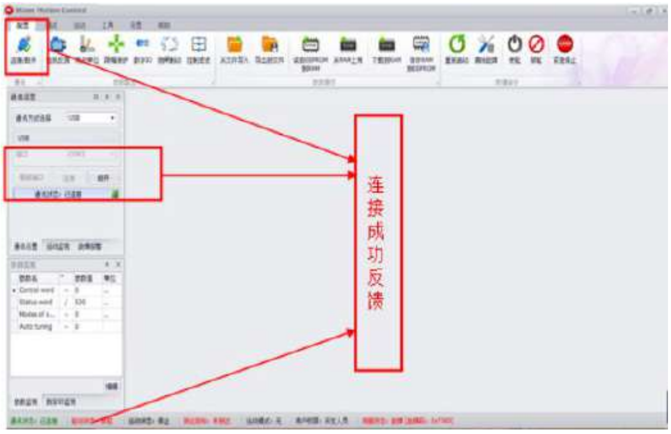


Figure 4-3 Communication connection succeeded

4.2 Parameter Writing

Parameter settings includes the settings of parameters about startup, motor feedback, limit protection and user unit.

4.2.1 Startup Parameters

Click **"Start Configuration"** in submenu **"Configuration"**, select the corresponding communication mode, and click **"OK"**.

Figure 4-4 Settings of startup parameters

4.2.2 Motor Feedback Parameters

Motor feedback parameters includes motor parameters and encoder parameters.

1. Select **"Motor Feedback"** in submenu **"Configuration".**
2. Set motor parameters and encoder parameters.
3. Click **"Download"** after settings are completed.

Figure 4-5 Settings of motor feedback parameters

Note: Please restart the servo drive after saving successfully, and reconnect the communication after restarting.

- **Motor parameters**

1) To conveniently configure motor parameters, SMC supports motor database in which you can directly call the motor parameters of known models and save the motor parameters of new models.

   Note: The database only supports saving and loading motor feedback parameters. "**Import**" command is used to import all servo parameters and "**Export**" command is used to export all servo parameters.

Figure 4-6 Motor parameters-1

2) Please input the values of target motor parameters into the software according to the motor nameplate and the motor parameter manual provided by the manufacturer, as shown in Figure 4-7:

Figure 4-7 Motor parameters-2

3) The motor types include rotary brushless motor, linear brushless motor, rotary DC brush motor and voice coil motor. The parameters and units may vary with the motor type.

Note: Please pay attention to parameter unit when writing.

- **Encoder parameters**

  According to the actual encoder type, in the encoder parameter interface, please select the encoder type and input the resolution.

Rotary encoder Linear encoder

Figure 4-8 Motor parameters-3

Encoder parameters are as shown in Table 4-1.

Table 4-1 Encoder parameters

Name	Unit	Definition
Absolute single-turn resolution	Bit	The pulse value output by one rotation of the encoder.
Absolute multi-turn resolution	Bit	The maximum number of turns recorded by the encoder.
Encoder multi-turn value reset		To clear the absolute encoder multi-turn value to zero.
Resolution	• Rotary: counts/revolution • Linear: counts/nm, um, mm	• The pulse value output by one rotation of the encoder. • The pulse value output by the grating ruler per unit distance.
Communication rate	M	The clock frequency at which data is sent to or received from the encoder.

4.2.3 Limit Protection

1. In **"Configuration"** submenu**,** select **"Limit Protection"**.
2. Set the peak current and the duration of the peak current.

   To protect motor i2t, please set the values according to the maximum current of the motor. Otherwise, damage to the motor may occur.

4.2.4 User Unit

1. In **"Configuration"** submenu, select **"User Unit"**.
2. Set parameter unit for motion control parameters, including position unit and velocity unit, and configure mechanical gear ratio, as shown in Figure 4-9:

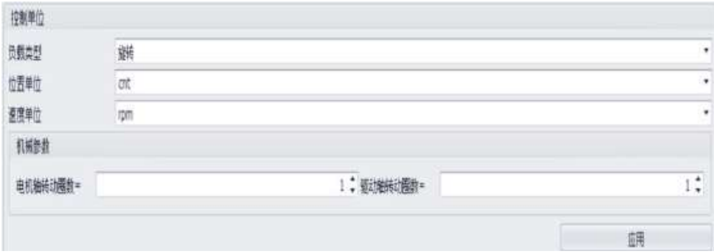


Figure 4-9 User units

3. After settings, click **"Apply".** The settings takes effect immediately.
   In the debugging mode and motion mode, the unit will be the same as the set unit.
   Please set user units for position and velocity modes according to the load type, as shown in Table 4-2.

Table 4-2 User units

Load Type / Motion Unit	Linear		Rotary	
Position unit	cnt	pulses	cnt	number of pulses
	um	microns	deg	angle
	mm	mm	rad	radian
	cm	centimeter	rev	Turn
	uu	customize	uu	customize
Velocity unit	cnt/s	pulses/sec	cnt/s	Pulses/sec
	um/s	μm/s	deg/s	angle/sec
	mm/s	mm/s	rad/s	radians/sec
	cm/s	cm/s	rpm	rpm
	uu	customize	rps	rev/sec
	-	-	uu	customize

4.3 Magnetic Pole Calibration

4.3.1 Phase Sequence Steering Detection

Phase sequence and motion direction of the incremental motors is required before motion control. With the phase sequence detection, the servo drive will automatically recognize UVW wire, and reverse the phase sequence and rotation direction according to the positive direction.

4.3.2 Hall Detection

When using the Hall sensor, the servo drive needs to automatically recognize the Hall angle. After that, the motor can be directly started with the Hall angle, which makes the motor start more smoothly, for it avoids the shock of magnetic pole identification when the incremental motor is powered on each time.

4.3.3 Commutation Offset Detection

Before motion control, detection of the magnetic pole zero is required. After calibration, the motion control can be performed normally. Otherwise, motor runway may occur.

Commutation offset detection is required after the phase sequence steering detection. Otherwise, calibration may fail. In this case, when the motion is enabled or started, the current feedback value observed in the motion monitoring is pretty large, the motor rotor is locked, or there is a risk of motor runaway. At this time, please set "**2002**" to "**1**" in "**Parameter Editor-PID**", to switch the phase sequence.

Parameter	Description
2002 Three-phase seq switch enable	The function of switching phase sequence. 0-no switching; 1-switching

The steps are as follows:

1. After setting parameters, in the parameter configuration interface, click "**Auto Commutation Offset**":

Figure 4-10 Automatic commutation detection

2. Wait for about 5 -10 seconds. The status turns green, as shown in figure 4-11, which indicates that the zero point calibration is completed.

Figure 4-11 Completion of automatic commutation offset detection

Description:

1. For an absolute encoder, an accurate calibration is required at the first time. After that, motion control can be performed directly after the servo drive is powered on.
2. For an incremental encoder (without Hall signal), the calibration is required every time the servo drive is powered on. Otherwise, motion control is not allowed.
   Calibration can be done by sending calibration commands and enable commands (see note 2 for logic) or manually clicking automatic commutation offset detection.
   During calibration, please do not perform other motion control related operations. Otherwise, the servo drive will report the corresponding error.
   The servo drive owns the function of automatic calibration after power-on. With it enabled, you can check 

   or set 0x2120 to 1. As a result, after saving, the servo drive will automatically start calibration every time it is powered on. After calibration, the servo is disabled, and the sign indicating the completion of calibration shows. At this time, please set 0x2121 to 1.
3. For an incremental encoder (with Hall signal), configuring HALL start (0x2103=1) and recognizing HALL angle are required at the first time. After that, motion control can be performed directly after the servo drive is powered on.
4. Setting of calibration current
   a. Gradually increase 0x2105 until the motor shaft can be fixed in a certain position quickly and stably.
   b. Rotate the shaft.
   c. Start calibration again several times until the position is basically the same (i.e. the value of 0x2102 is almost the same).

   Note:

   1. If the calibration current is not adjusted properly or the load of motor shaft is too large, the calibration will fail. For the error handling, please refer to ***Chapter 5***

Troubleshooting.

2. After 0x6060 (control mode) is set to 0, 0x2101 is written to 1, 0x6040 (control word) is executed according to the enable logic of 6 -> 7 -> 15, the servo enters calibration status. When 0x2101 turns to 0, it means the calibration process is completed.

Commutation offset related parameters

parameter	Description
2101 Calibrate commutation offset	The sign of manual zero calibration enable.
2102 Commutation offset	The value of zero calibration.
2103 HallModeSelect	Hall mode selection. 0: disable Hall;1: enable Hall.
2105 Commutation current ratio_1	D-axis calibration current amplitude = 2105 / 1000 * Rate current. Frequency: constant value
213E Hall_Angle	Hall calibration angle.
2120 AutoCalibrateAngle	Automatic calibration after power-on. 0-OFF; 1-ON.
2121 AutoCalibrateAngleFinish	The sign whether automatic calibration after power-on is completed. 0-Incomplete; 1-Complete.
2402 Commutation current ratio_2	Q-axis calibration current amplitude = 2402 / 1000 * 2105 Frequency: high frequency

4.4 PID Adjustment

If PID parameters are not set properly, the motor may vibrate or make abnormal noise. Thus, to achieve a better control effect, it is necessary to adjust PID parameters before controlling the motor.

The upper software SMC provides a function generator, which can output the given mode, wave form and step signal, and capture the given waveform and the feedback waveform for response analysis with an oscilloscope.

The whole debugging steps is as follows:

Adjustment for current loop → Adjustment for velocity loop → Adjustment for position loop

4.4.1 Current Loop

The first debugging is for the current loop.

1. Click **"Adjust"** in the main menu, select **"Current Loop"**. The debugging interface of the current loop shows as in Figure 4-12:

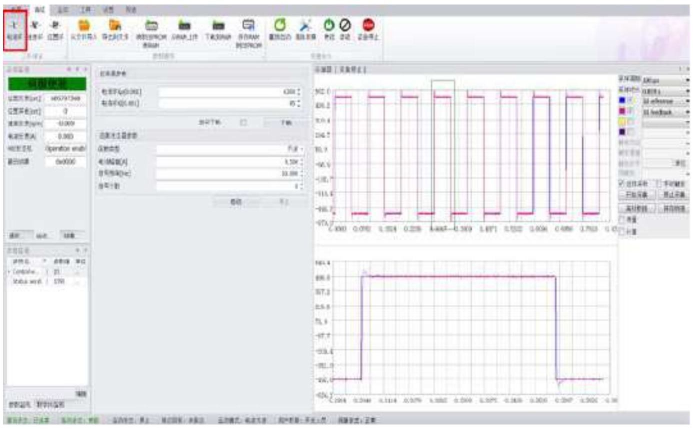


Figure 4-12 Current loop debugging interface

2. Adjust Kp.

   Main function: to increase the bandwidth with the increase of Kp. If it is too large, the motor makes noise, and if it is too small, the bandwidth is lowered.

   a. Set Ki to 0 and Kp to 100, and click "**Download**":

      Generally, you only need to slightly adjust the default values.

Figure 4-13 Control parameters of current loop

b. Set the function type to sine wave, current amplitude to 25% of the motor rated current (the following takes 1 A as an example) and frequency to 1500 Hz.

Figure 4-14 Function generator parameters of current loop

c. Turn on oscilloscope again, set the sampling channel to Id/Iq reference (current given value) and Id/Iq feedback (current feedback value), set the sampling period to 50 us, and check continuous sampling.

Figure 4-15 Oscilloscope sampling parameters of current loop

d. Enable the servo, start function generator, and click **"Start Acquisition"**.

e. Keep increasing Kp until the amplitude of Id/Iq feedback is between (0.707~1) of the amplitude of Id/Iq reference and the phase lag does not exceed 90°:

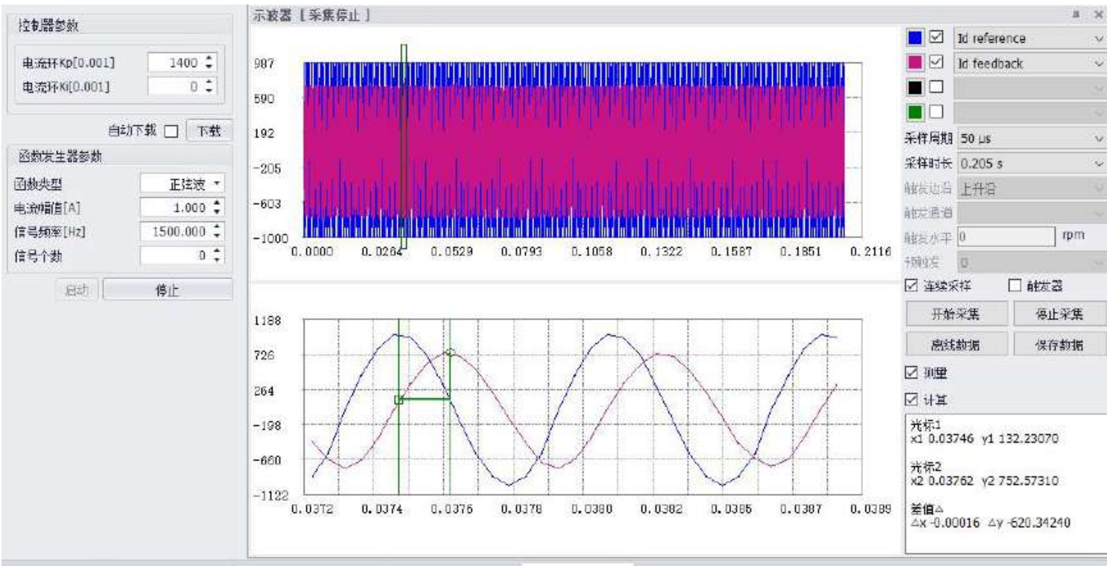


Figure 4-16 Current sampling waveform after adjusting Kp

3. Adjust Ki.

Main function: to eliminate the steady-state error. If it is too large, it will lead to overshoot and the motor will make noise.

a. Set the function type to square wave, current amplitude to 25% of the motor rated current (the following takes 1 A as an example) and frequency to 10 Hz.

Figure 4-1 7 Function generator parameters of current loop

b. Gradually increase Ki, (generally increase 100 each time), and repeat step **c** and **d** of adjusting Kp until the steady-state error is eliminated, the waveform of Id/Iq feedback almost coincides with that of Id/Iq reference waveforms, and the overshoot is within 5%:

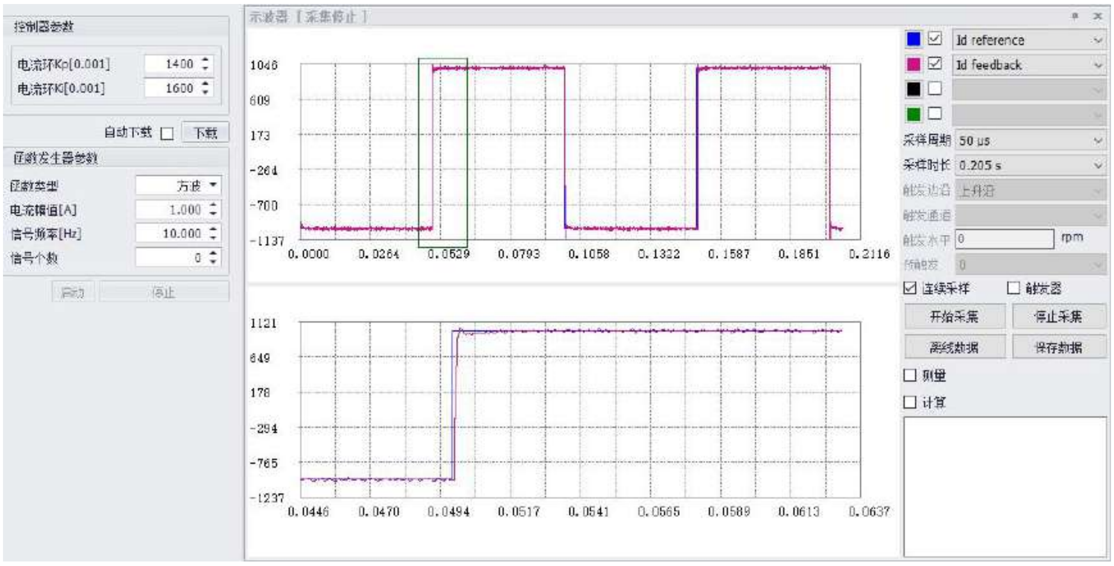


Figure 4-18 Current sampling waveform after adjusting Ki

Note: When adjusting the current loop, if the motor is a rotary brushless/linear motor, please select id for adjustment, and if the motor is DC brush/voice coil motor, please select iq for adjustment.

4.4.2 Velocity Loop

The second debugging is for the velocity loop.

1. Select **"Velocity Loop"**. The debugging interface of the velocity loop shows as in Figure 4-19:

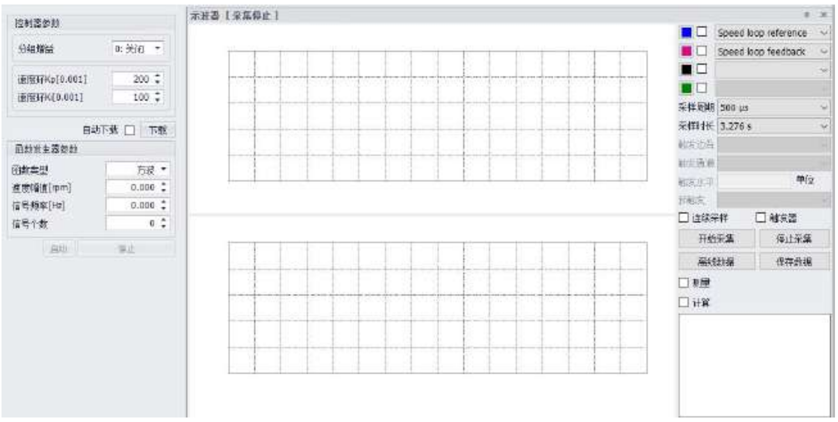


Figure 4-19 Velocity loop debugging interface

2. Do the following preparation works:
   a. Set the inertia ratio to 0X2422.
   b. Set the following parameters to 0:
      - ✓ 0x2020:01 Filter type of measured speed
      - ✓ 0x2021:01 Filter type of speed error filter
      - ✓ 2022:01 Filter type of speed error second filter
      - ✓ 2006 Feed forward method

3. Adjust Kp.
   a. Set Ki to 0 and Kp to 10, and click "**Download**":

Figure 4-20 Control parameters of velocity loop

   b. Set the function type to step signal and velocity amplitude to 300 rpm, and set the duration according to the limit of running distance, i.e 500 ms.

Figure 4-21 Function generator parameters of velocity loop

   c. Turn oscilloscope again, set the sampling channel to velocity loop reference (speed given value) and velocity loop feedback (velocity feedback value), set the sampling period to 200 us, check “**Trigger Acquisition**”, set the trigger edge to rising edge, set trigger channel to velocity loop reference, set trigger level to 10 rpm, and set pretrigger to 20%.

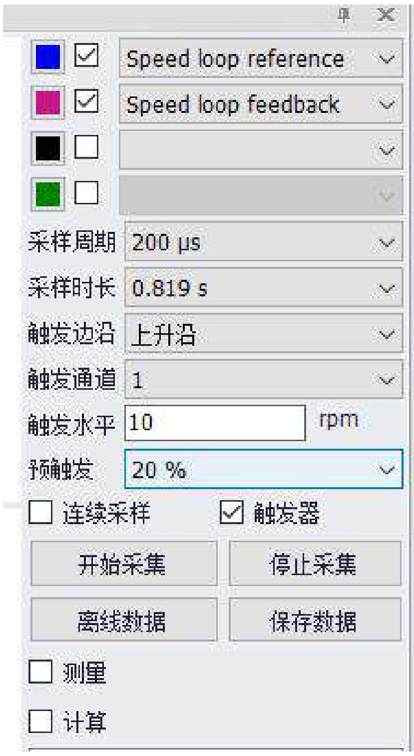


Figure 4-22 Oscilloscope sampling parameters of velocity loop

d. Enable the servo, start function generator, and click "Start Acquisition".
   When the function type is set to step signal, there will be a delay of 4 - 5 seconds, to make sure that there is enough time for the oscilloscope to start acquisition.

e. Keep increasing Kp (generally increase 10 digits each time) and observe the waveform of velocity loop reference (speed given value) and velocity loop feedback (speed feedback value) until the critical oscillation shows in the velocity waveform:

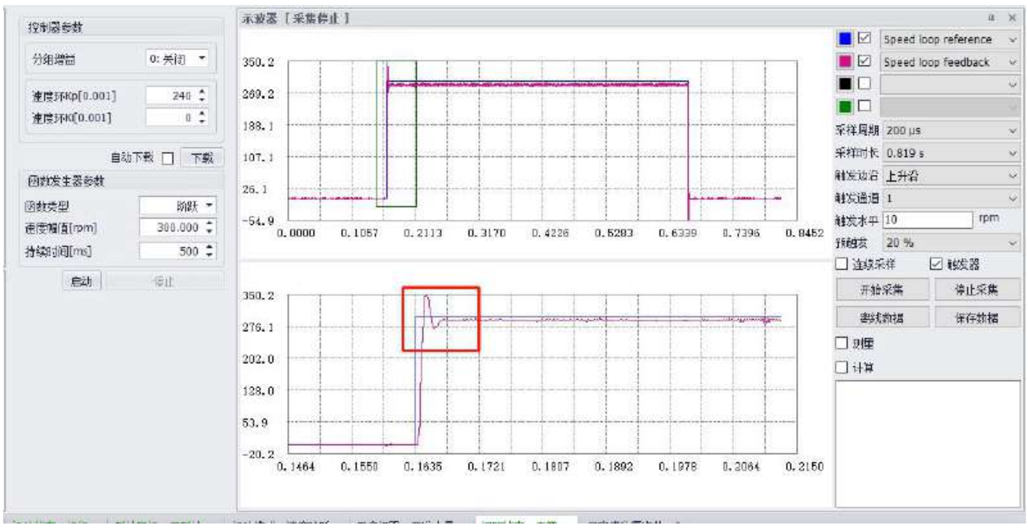


Figure 4-23 Velocity sampling waveform after adjusting Kp

f. Take 70% - 80% of the value of Kp, and stop the oscilloscope acquisition and function generator.

4. Adjust Ki.

Gradually increase Ki, and repeat step **c** and **d** of adjusting Kp until the steady-state error of velocity loop feedback is eliminated and the overshoot is within 30%:

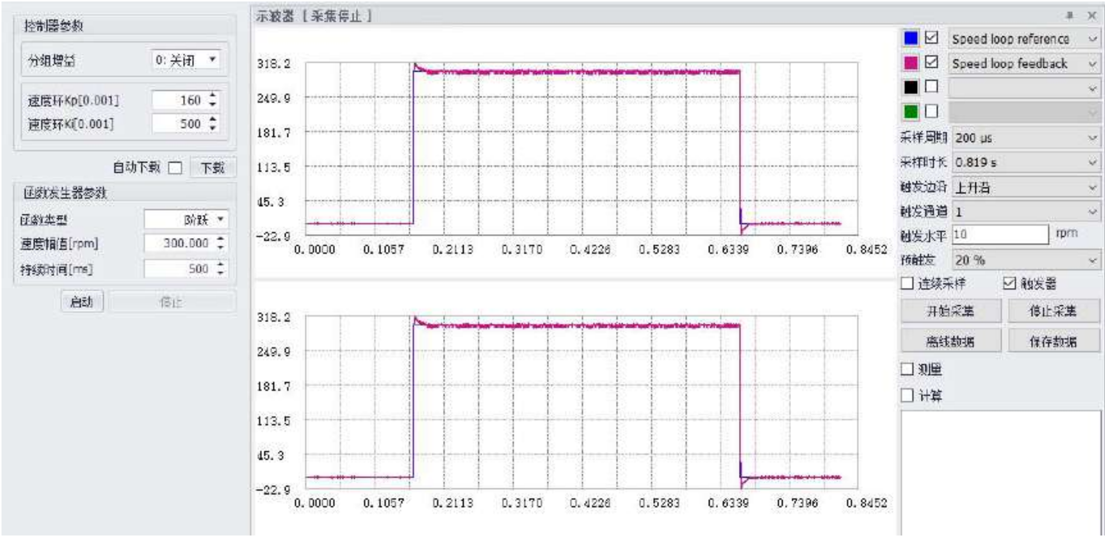


Figure 4-24 Velocity sampling waveform after adjusting Ki

To reduce the velocity deviation during acceleration, you can do debugging for torque feed forward as follows:

1. To enable the feed forward function, set 2006 to 2.
2. Set 2019 Torque feed forward time constant to a fixed value, and keep increasing 2016 Speed feed forward coefficient until a good result of velocity feed forward shows at a certain value.
3. Repeatedly adjust 2019 and 2016 to find a balance.

Note: Improper debugging will cause system oscillation. If oscillation or mechanical resonance occurs during debugging, you can set 0x2021 / 0x2022 Filter type of speed error filter to eliminate the oscillation frequency:

Parameter	Description
200C:01 Measured speed filter	The filter value of measured speed
200F:01 Speed error filter	The filter value of speed error
2010:01 Speed error second filter	The filter value 2 of speed error
2020:01 Filter type of measured speed filter	The filter type of measured speed
2020:02 Frequency of measured speed filter	The filter frequency of measured speed
2020:03 Quality factor of measured speed filter	The filter quality factor of measured speed
2021:01 Filter type of speed error filter	The filter type 1 of measured speed
2021:02 Frequency of speed error filter	The filter frequency 1 of measured speed
2021:03 Quality factor of speed error second filter	The filter quality factor 1 of measured speed
2022:01 Filter type of speed error filter	The filter type 2 of measured speed
2022:02 Frequency of speed error filter	The filter frequency 2 of measured speed
2022:03 Quality factor of speed error second filter	The filter quality factor 2 of measured speed
2421 Velocity Average Filtering	Velocity average filter (internal use)

4.4.3 Position Loop

The third debugging is for position loop.

1. Select **"Position Loop"**. The debugging interface of the position loop shows as in Figure 4-25.

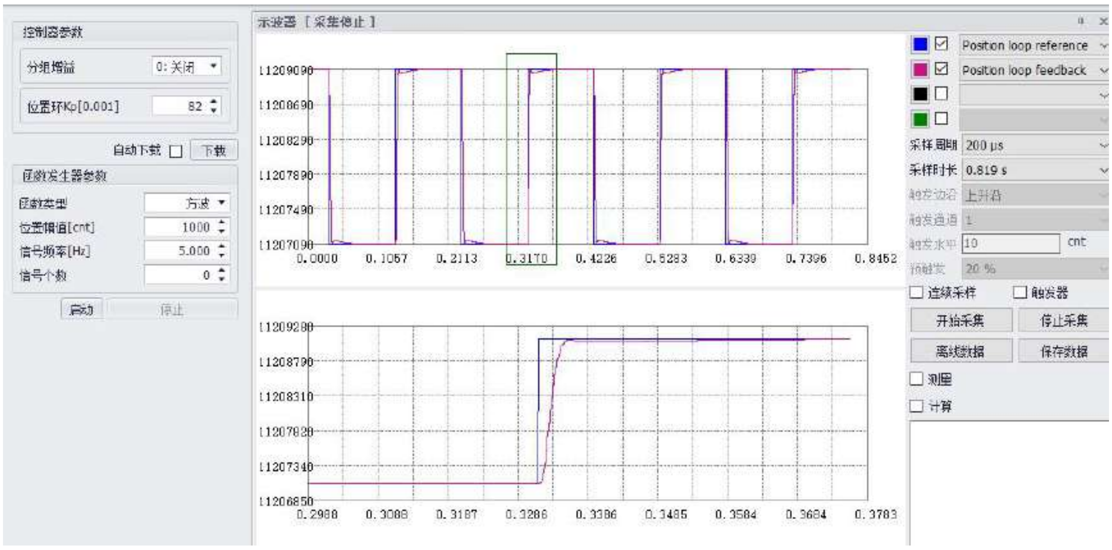


Figure 4-2 5 Debugging interface of position loop

2. Adjust Kp.
   a. Set Kp, and click "**Download**".
   It is recommended to use the default value 10 at first, and modify it after obtain the position curve.

Figure 4-26 Control parameters of position loop

   b. Set the function type to square wave signal, position amplitude to 1000 cnt (the current position is zero, the motion amplitude is 1000 cnt. Please pay attention to the mechanical stroke), and signal frequency to 5 Hz.

Figure 4-27 Function generator parameters of position loop

c. Turn on oscilloscope, set the sampling channel to position loop reference (position given value) and position loop feedback (position feedback value), set the sampling period and duration to proper values, and check continuous acquisition.

Figure 4-28 Oscilloscope sampling parameters of position loop

d. Observe the waveform of position loop reference (position given value) and position loop feedback (position feedback value), and adjust Kp as follows until the result of waveform is good with unsaturated current:

- ✓ Increase Kp when the position follow-up error is large or the response is slow.
- ✓ Reduce Kp when the position overshoot or jitter occurs.

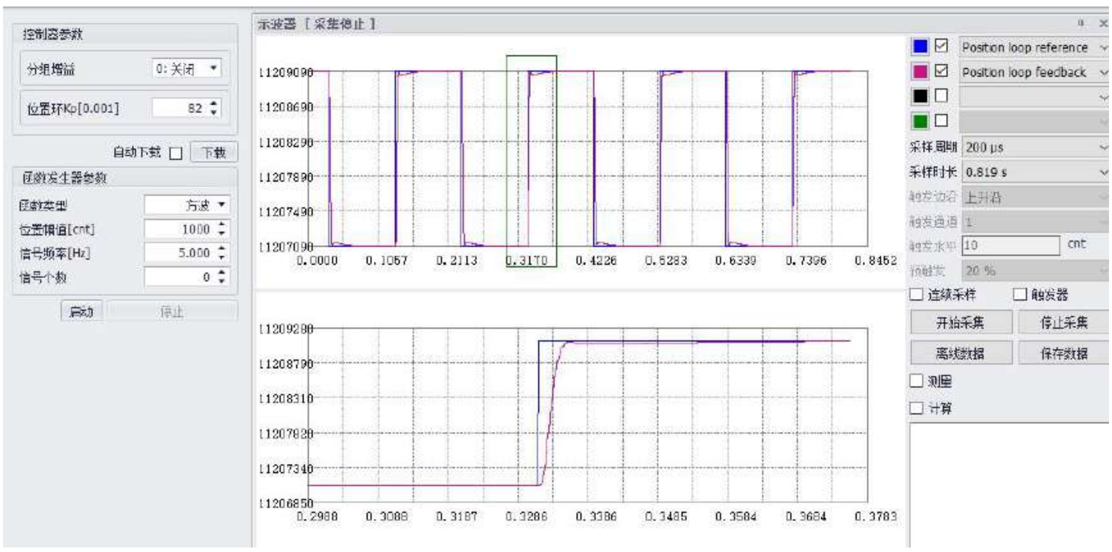


Figure 4-29 Position sampling waveform after adjusting Kp

In actual application, if not satisfied with the position follow-up error, you can carry out debugging for torque feed forward and speed feed forward as follows:

1. To enable the feed forward function, set 0x2006 to 2.
2. Set 0x2019 Torque feed forward time constant to a fixed value, and keep increasing 2016 velocity feed forward coefficient until a good result of velocity feed forward shows at a certain value.

3. Repeatedly adjust 0x2019 and 0x2016 to find a balance.

After adjusting the position loop gain, the motor makes low-frequency audible noise in the enabled but not running state, which will reduce the velocity loop Kp or the current loop Kp. If the value of position loop Kp is too small, the rigidity is weak.

4.4.4 Grouping Gain

Grouping gain can be set in the situation where the inertia load changes and a group of fixed gain parameters of velocity loop and position loop cannot satisfy high, medium and low speed. Its principle is as follows:

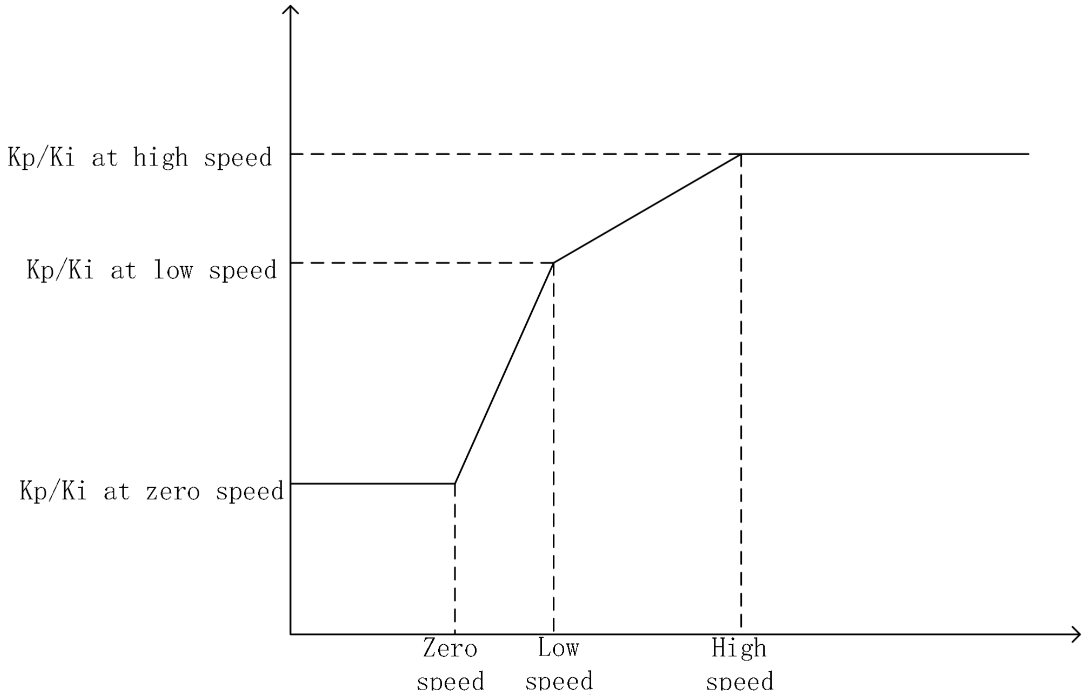


Figure 4-3 0 Principle of grouping gain

Take the velocity loop as an example: when setting the grouping gain, you can set the actual speed or the given speed:

Actual Speed / Given Speed:	Gain Parameter
0 - zero speed	Kp and Ki at zero speed
Zero speed - low speed	● Kp increases with the slope (Kp at low speed - Kp at zero speed) / (low speed - zero speed). ● Ki increases with the slope (Ki at low speed - Ki at zero speed) / (low speed - zero speed)
Low speed - high speed	● Kp increases with the slope (Kp at high speed - Kp at low speed) / (high speed - low speed). ● Ki increases with the slope (Ki at high speed - Ki at low speed) / (high speed - low speed)
＞ High speed	Kp and Ki at high speed

4.5 Motion Control

After setting motor parameters, encoder parameters and control parameters, the motor can be simply driven. The modes that the software controls the servo drive to drive the motor include the following:

- Position mode

- Speed mode
- Homing mode
- Torque mode

4.5.1 Position Control Mode

The process of motion control in position mode is as follows:

1. Click "**Motion**" in the main menu, and click "**Position Mode**"**.** The interface of motion control in position mode shows as in Figure 4-31.

Figure 4-31 Interface of motion control in position mode

2. Set the following parameters:
   - ✓ Motion mode: to set the position motion as unidirectional motion or reciprocating motion.
   - ✓ Target position: to control the distance of motor movement. When the motion mode is set to reciprocating motion, you need to set two target positions.
   - ✓ Speed: the movement speed of the motor.
   - ✓ Acceleration: the acceleration to start the motor.
   - ✓ Deceleration: the deceleration to stop the motor.
   - ✓ Deceleration for quick stop: the deceleration to stop the motor when the motor is directly disabled.
   - ✓ Command type: absolute, to start movement with zero point of the encoder as the start point; relative, to start movement with current position of the encoder as zero point.
   - ✓ Curve type: including linear ramp (straight line) and Jerk-limited ramp (S-shaped curve).
   - ✓ Waiting time: the waiting delay time for the arrival of the target position when the motion mode is set to reciprocating motion.
   - ✓ Cycle times: the number of reciprocating cycles when the motion mode is set to reciprocating motion. Infinite cycle means cycle will continue all the time.
3. To enable the servo drive, click "**Enable**".
4. To start motion control in position mode and start acquisition oscilloscope, click "**Start**" and **“Start Acquisition**".

4.5.2 Velocity Control Mode

The process of motion control in position mode is as follows:

1. Click **"Motion"** in the main menu, and click **"Velocity Mode".** The interface of motion control in velocity mode shows as in Figure 4-32.

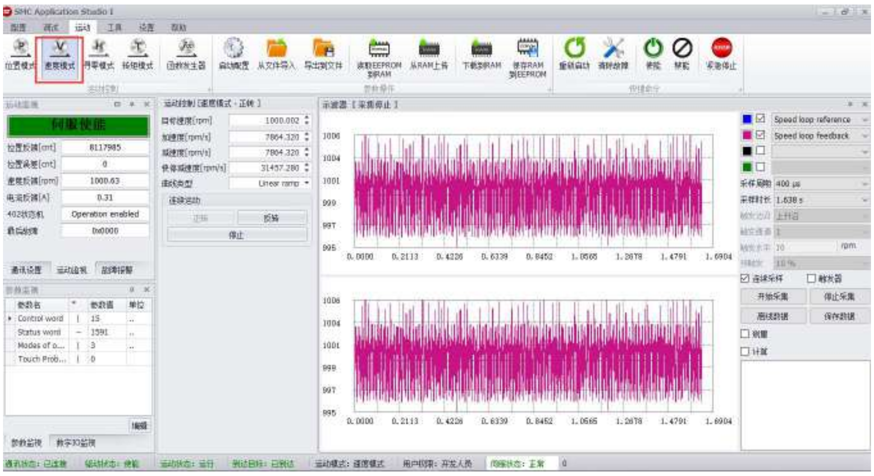


Figure 4-32 Interface of motion control in velocity mode

2. Set the following parameters:
   - ✓ Target speed: the movement speed of the motor.
   - ✓ Acceleration: the acceleration to start the motor.
   - ✓ Deceleration: the deceleration to stop the motor.
   - ✓ Deceleration for quick stop: the deceleration to stop the motor when the motor is directly disabled.
3. To enable the servo drive, click "**Enable**". **Servo Enable** shows in the interface.
4. To control the motor to move in the positive direction, click “**Forward**”, to control the motor to move in the opposite direction, click “**Reverse**”.

4.5.3 Homing Mode

The process of motion control in homing mode is as follows:

1. Click **"Motion"** in the main menu, and click **"Homing Mode".** The interface of motion control in homing mode shows as in Figure 4-33.

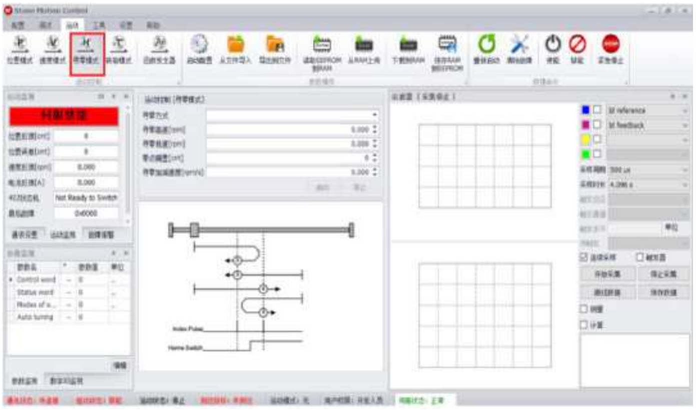


Figure 4-3 3 Interface of motion control in homing mode

2. Set the following parameters:
   - ✓ Homing method: there are 35 homing methods. When starting, the motor moves according to the selected homing method.
   - ✓ High speed for homing: when starting, the motor starts to find the zero point at high speed.
   - ✓ Low speed for homing: when starting, the motor moves to the zero point at low speed after it finds the zero point.
   - ✓ Zero offset: after setting the zero offset, the motor finally stops at the position behind the offset.
   - ✓ Acceleration and deceleration for homing: when starting, the acceleration and deceleration for homing.
3. To enable the servo drive, click "**Enable**". **Servo Enable** shows in the interface.
4. To make the motor move with the set homing method, click “**Start**”, to stop the motor, click “**Stop**”**.**

Homing

1. Homing method

- ✓ When using an incremental encoder, and when the servo does not know the position of the motor when it is powered on, homing is required every time it is powered on.
- ✓ When using an absolute encoder or incremental + Hall signal, homing is required only when the servo is powered on for the first time.

Note: The zero point calibration is the initial angle identification of the motor. If the initial angle identification is not performed, the motor may reverse or even run away. When using an incremental encoder, zero point calibration is required each time the power is on; when using an absolute encoder or incremental + Hall signal, zero point calibration is required only when the power is on for the first time.

2. Related concepts

Origin and zero point

- Home position: machine origin, which can represent origin switch or motor Z signal.
- Zero position: the position after homing finishes.

During homing, the motor stops at the home position. If the position deviation 607C is set, the motor stops at the zero position.

Zero position = Home position + 607C Home offset:

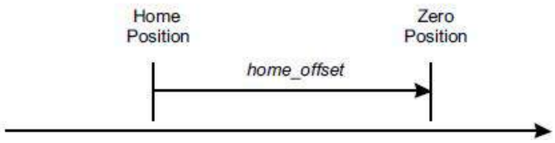


Figure 4-34 Relationship between origin and zero point

Speed

- High speed: the speed during finding the limit switch (different according to the origin mode). (6099-01h).
- Low speed: the speed during finding the origin after finding the limit switch. (6099-02h).
- Acceleration and deceleration: acceleration and deceleration during homing. (609A).

Direction

The direction in which the encoder value increases is the positive direction, and the direction in which the value decreases is the negative direction.

4.5.3.1 Homing Method

Note:

1. The numbers in the figure correspond to the corresponding homing methods. The same numbers indicates two ways of this homing method. For example, two ③ in the figure of method 3 indicates two different ways of the homing method 3.
2. The index pulse is the Z signal.
3. The bold color indicates homing at high speed.

4.5.3.2 Method 1: Homing on negative limit switch (falling edge) and index pulse

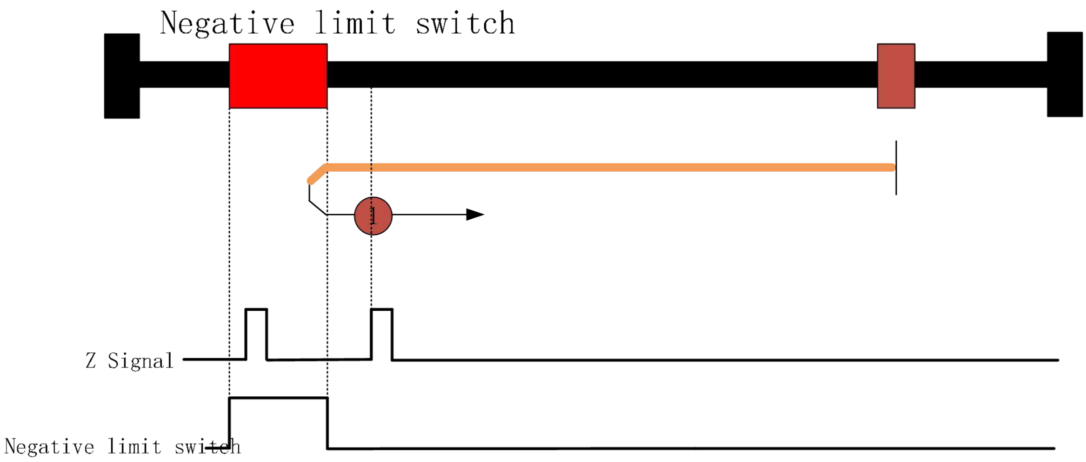


Figure 4-35 Method 1

When homing starts, the motor moves at a high speed (6099-01) in the negative direction. When the negative limit switch signal becomes high level, the motor decelerates to 0 with the homing deceleration (609A), moves in the positive direction with the homing acceleration (609A) to accelerate to a low speed (6099-02), and keeps moving in the positive direction at the low speed. After the negative limit switch signal becomes low level and the first Z signal shows, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.

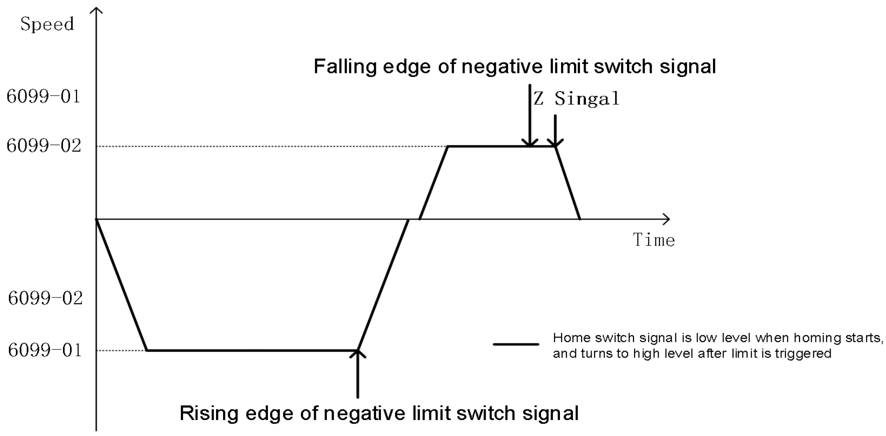


Figure 4-36 Speed-time curve of method 1

4.5.3.3 Method 2: Homing on positive limit switch (falling edge) and index pulse

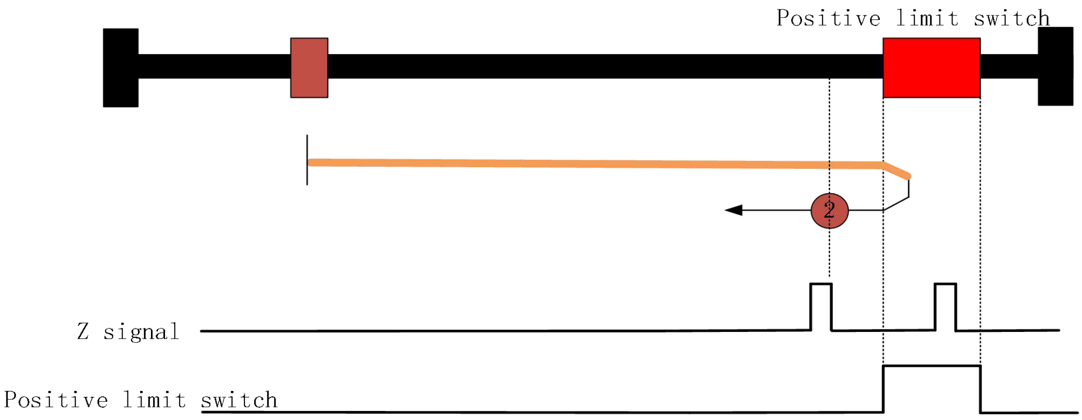


Figure 4-37 Method 2

When homing starts, the motor moves at a high speed (6099-01) in the positive direction. When the positive limit switch signal becomes high level, the motor decelerates to 0 with the homing deceleration (609A), moves in the negative direction with the homing acceleration (609A) to accelerate to a low speed (6099-02), and keeps moving in the negative direction at the low speed. After the positive limit switch signal becomes low level and the first Z signal shows, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.

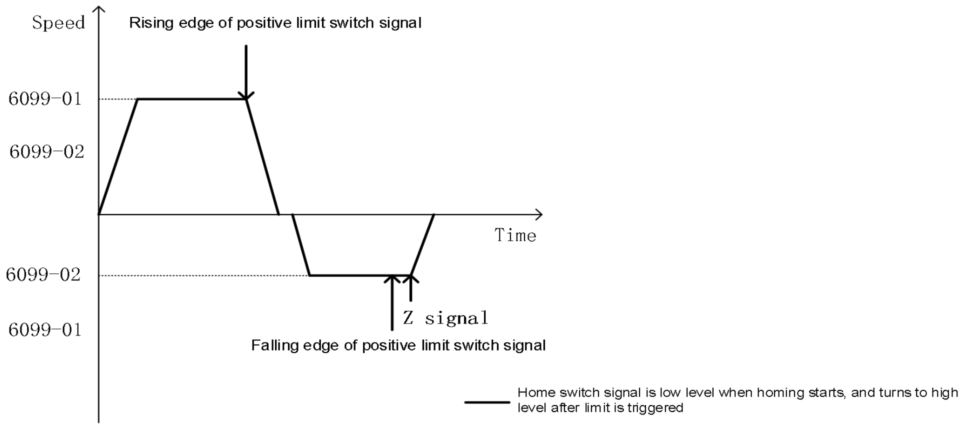


Figure 4-38 Speed-time curve of method 2

4.5.3.4 Method 3: Homing on positive home switch (falling edge) and index pulse

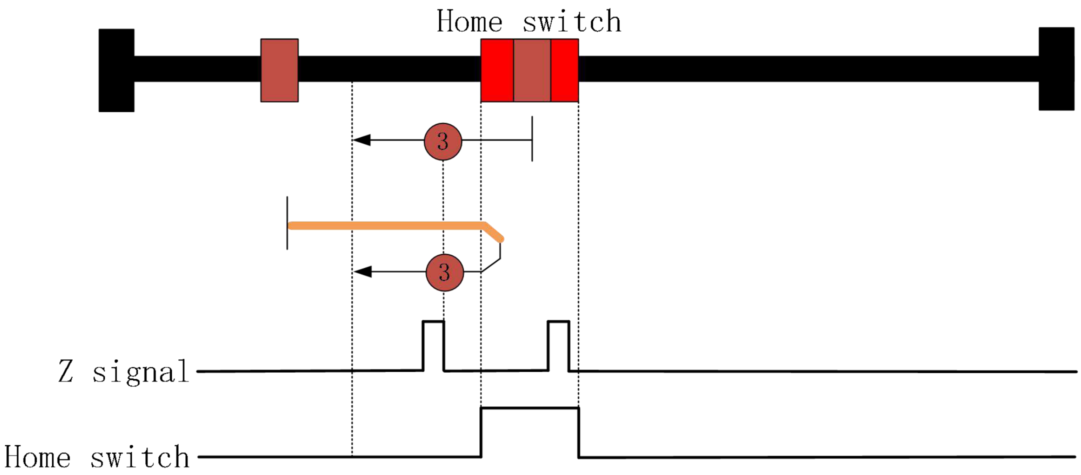


Figure 4-39 Method 3

- **When homing starts, if the home switch signal is low level,** the motor moves at a high speed (6099-01) in the positive direction. After the home switch signal becomes high level, the motor decelerates to 0 with the homing deceleration (609A), and moves in the negative direction with the homing acceleration (609A) to accelerate to a low speed (6099-02), and keeps moving in the negative direction at the low speed. After the positive home switch signal becomes low level and the first Z signal shows, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.
- **When homing starts, if the home switch signal is high level,** the motor moves at a low speed (6099-02) in the negative direction. After the home switch signal becomes low level and the first Z signal shows, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.

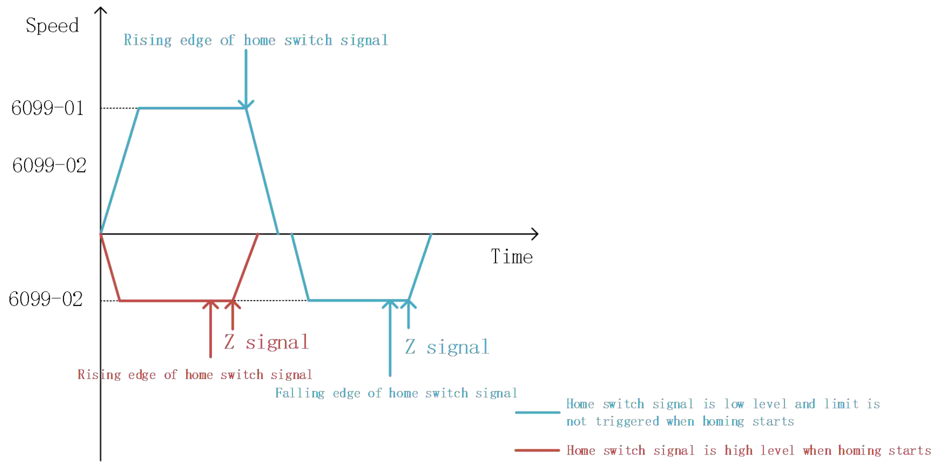


Figure 4-40 Speed-time curve of method 3

4.5.3.5 Method 4: Homing on positive home switch (rising edge) and index pulse

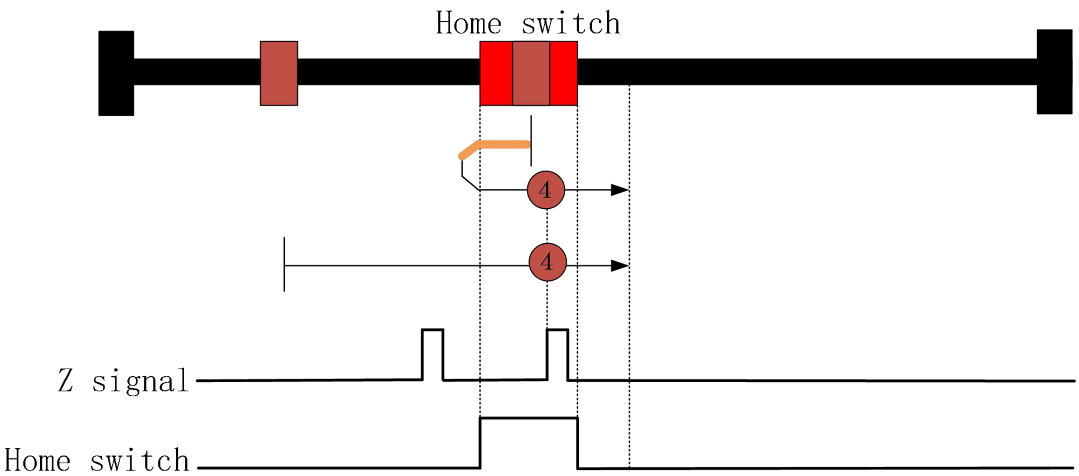


Figure 4-41 Method 4

原点开关：Home switch
Z 信号：Z signal

- **When homing starts, if the home switch signal is low level,** the motor moves at a low speed (6099-02) in the positive direction. After the home switch signal becomes high level and the first Z signal shows, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A) and finally return to the Z pulse latch position. The status word Target reached is set to 1 when the motor stops.
- **When homing starts, if the home switch signal is high level,** the motor moves at a high speed (6099-01) in the negative direction. After the home switch signal becomes low level, the motor decelerates to 0 with the homing deceleration (609A), and moves in the positive direction with the homing acceleration (609A) to accelerate to a low speed (6099-02), and keeps moving in the positive direction at the low speed. After the home switch signal becomes high level and the first Z signal shows, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A) and finally return to the Z pulse latch position. The status word Target reached is set to 1 when the motor stops.

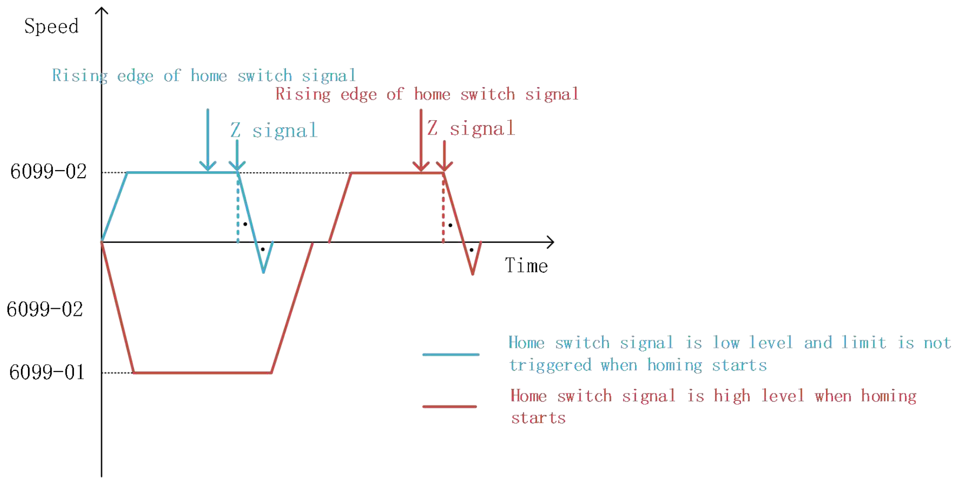


Figure 4-42 Speed-time curve of method 4

4.5.3.6 Method 5: Homing on negative home switch (falling edge) and index pulse

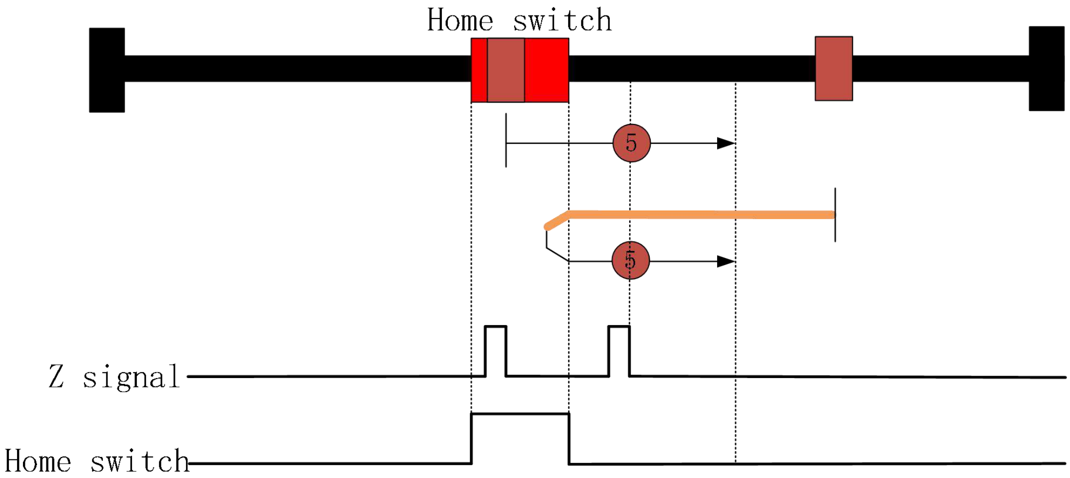


Figure 4-43 Method 5

- **When homing starts, if the home switch signal is low level,** the motor moves at a high speed (6099-01) in the negative direction. After the home switch signal becomes high level, the motor decelerates to 0 with the homing deceleration (609A), and moves in the positive direction with the homing acceleration (609A) to accelerate to a low speed (6099-02), and keeps moving in the positive direction at the low speed. After the home switch signal becomes low level and the first Z signal shows, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A) and finally returns to the Z pulse latch position. The status word Target reached is set to 1 when the motor stops.
- **When homing starts, if the home switch signal is high level,** the motor moves at a low speed (6099-02) in the positive direction. After the home switch signal becomes low level and the first Z signal shows, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A) and finally returns to the Z pulse latch position. The status word Target reached is set to 1 when the motor stops.

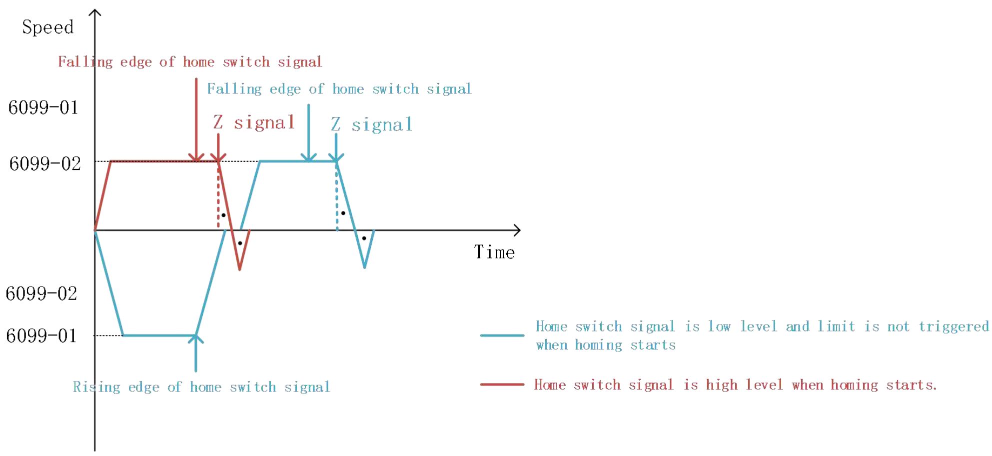


Figure 4-44 Speed-time curve of method 5

4.5.3.7 Method 6: Homing on negative home switch (rising edge) and index pulse

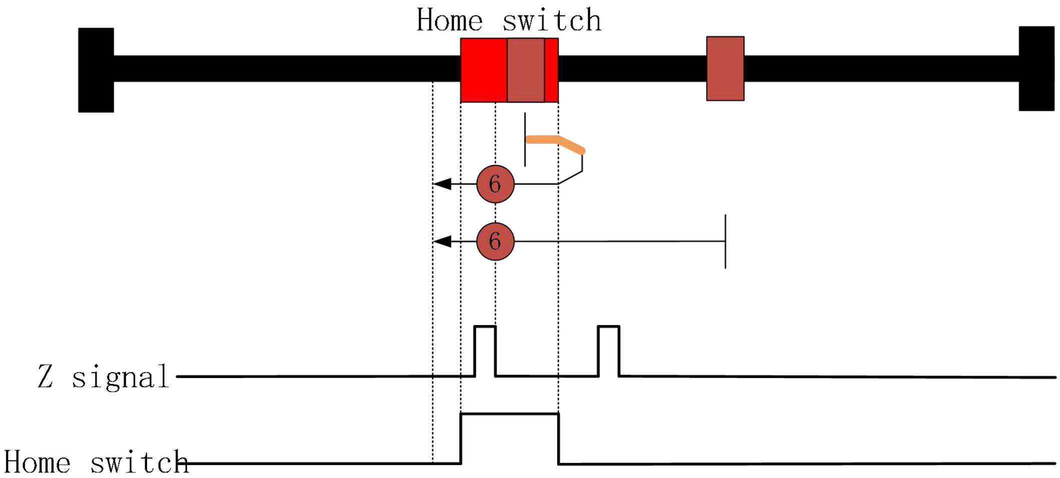


Figure 4-45 Method 6

- **When homing starts, if the home switch signal is low level,** the motor moves at a low speed (6099-02) in the negative direction. After the home switch signal becomes high level and the first Z signal shows, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.
- **When homing starts, if the home switch signal is high level,** the motor moves at a high speed (6099-01) in the positive direction. After the home switch signal becomes low level, the motor decelerates to 0 with the homing deceleration (609A), and moves in the negative direction with the homing acceleration (609A) to accelerate to a low speed (6099-02), and keeps moving in the negative direction at the low speed. After the home switch signal becomes high level and the first Z signal shows, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.

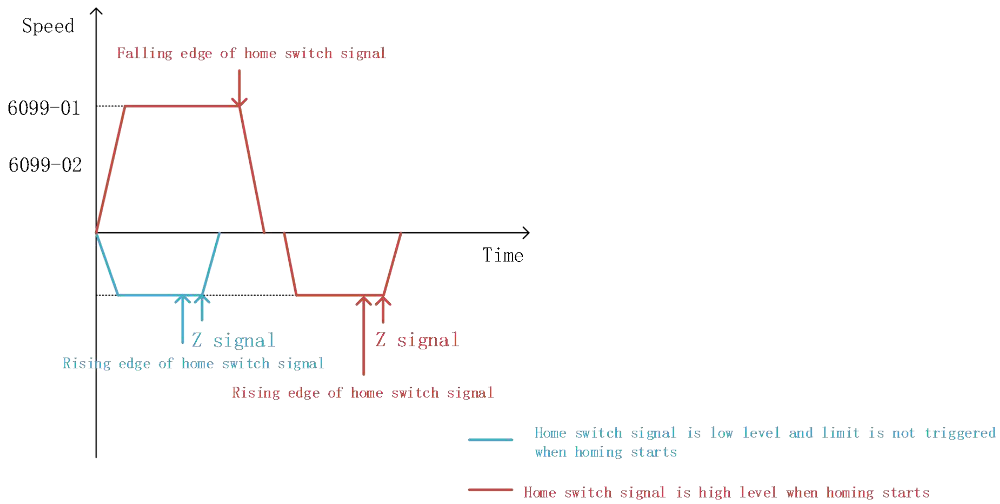


Figure 4-46 Speed-time curve of method 6

4.5.3.8 Method 7: Homing on negative home switch (falling edge) and index pulse-positive limit switch detection

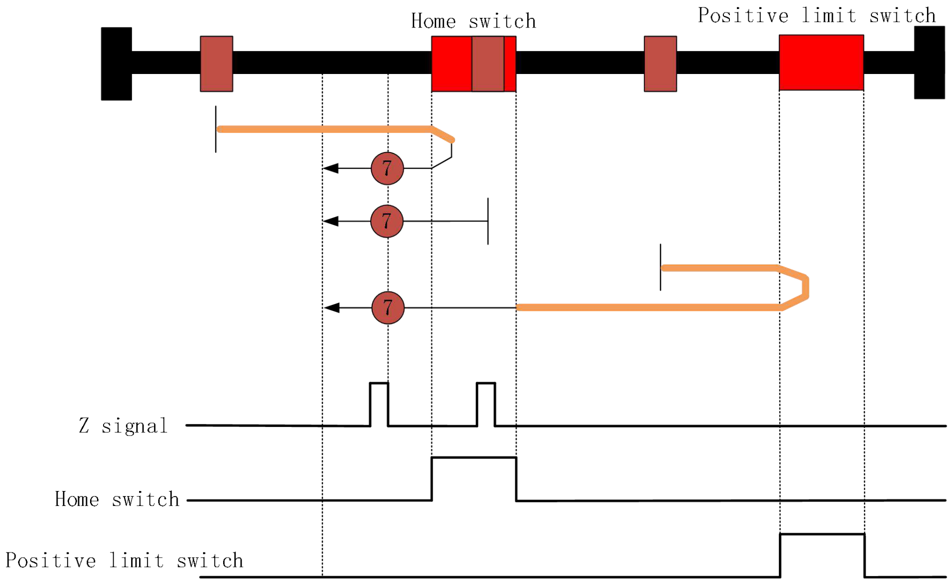


Figure 4-47 Method 7

- **When homing starts, if the home switch signal is low level,** the motor moves at a high speed in the positive direction.
  - ✓ After the home switch signal becomes high level, the motor decelerates to 0 with the homing deceleration (609A), and moves in the negative direction with the homing acceleration (609A) to accelerate to a low speed (6099-02), and keeps moving in the negative direction at the low speed. After the home switch signal becomes low level and the first Z signal shows, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.
  - ✓ After the positive limit switch signal becomes high level, the motor decelerates to 0 with the homing deceleration (609A), and moves in the negative direction with the homing acceleration (609A) to accelerate to a high speed (6099-01), and keeps moving in the negative direction at the high speed. After the home switch signal becomes high level, the motor decelerates with homing deceleration (609A) to a low speed (6099-02), and keeps moving in the negative direction at the low speed. After the home switch signal becomes low level and the first Z signal shows, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.

- **When homing starts, if the home switch signal is high level,** the motor moves at a low speed (6099-02) in the negative direction. After the home switch signal becomes low level and the first Z signal shows, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.

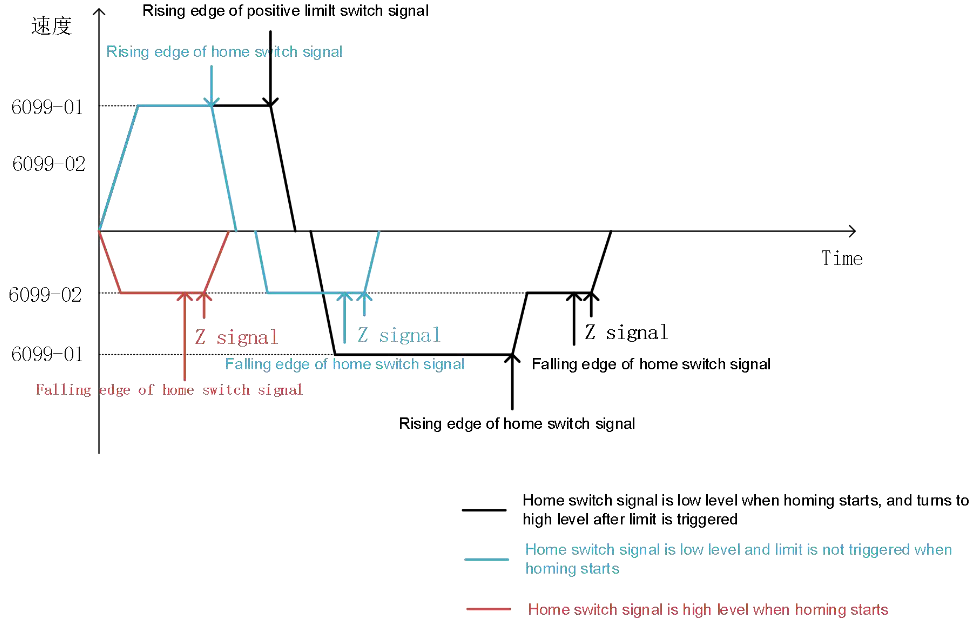


Figure 4-48 Speed-time curve of method 7

4.5.3.9 Method 8: Homing on positive home switch (rising edge) and index pulse-positive limit switch detection

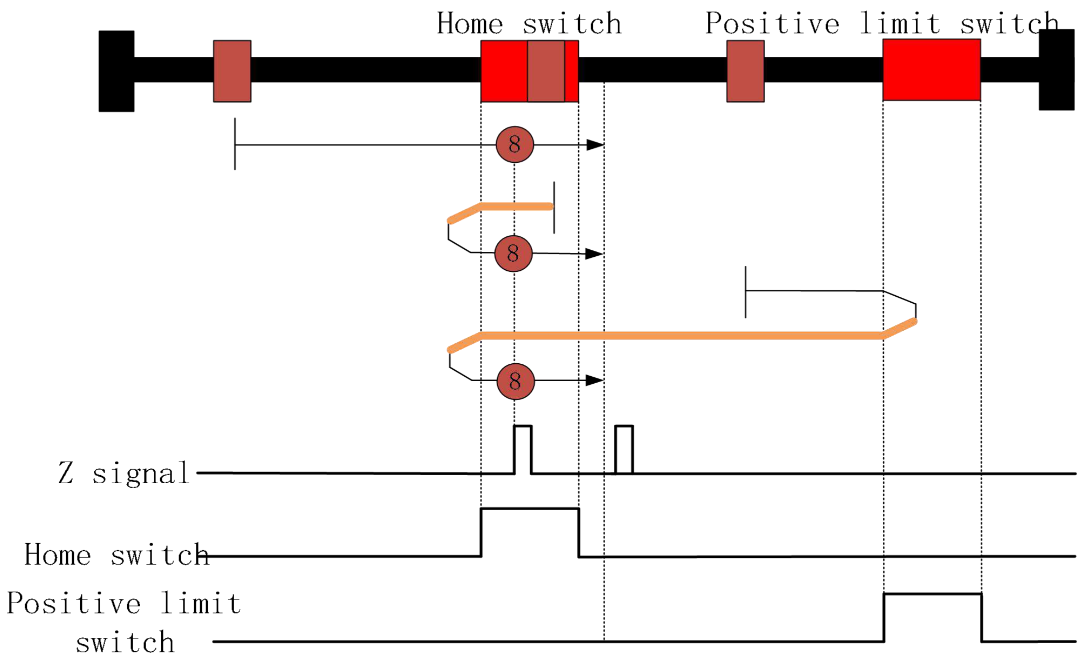


Figure 4-49 Method 8

- **When homing starts, if the home switch signal is low level,** the motor moves at a low speed (6099-02) in the positive direction.
  - ✓ After the home switch signal becomes high level, the motor keeps moving at a low speed (6099-02) in the positive direction. After the home switch signal becomes high level and the first Z signal shows, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.
  - ✓ After the positive limit switch signal becomes high level, the motor decelerates to 0 with the homing deceleration (609A), and moves in the negative direction with the homing acceleration (609A) to accelerate to a high speed (6099-01), and keeps moving in the negative direction at the high speed. After the home switch signal becomes low level, the motor decelerates to 0 with the homing deceleration (609A), move in the positive direction to accelerate to a low speed (6099-02) with the homing acceleration (609A), and keeps moving in the positive direction at the low speed. After the home switch signal becomes high level and the first Z signal shows, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.
- **When homing starts, if the home switch signal is high level,** the motor moves at a high speed (6099-01) in the negative direction. After the home switch signal becomes low level, the motor decelerates to 0 with the homing deceleration (609A), and moves in the positive direction with the homing acceleration (609A) to accelerate to a low speed (6099-02), and keeps moving in the positive direction at the low speed. After the home switch signal becomes high level and the first Z signal shows, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.

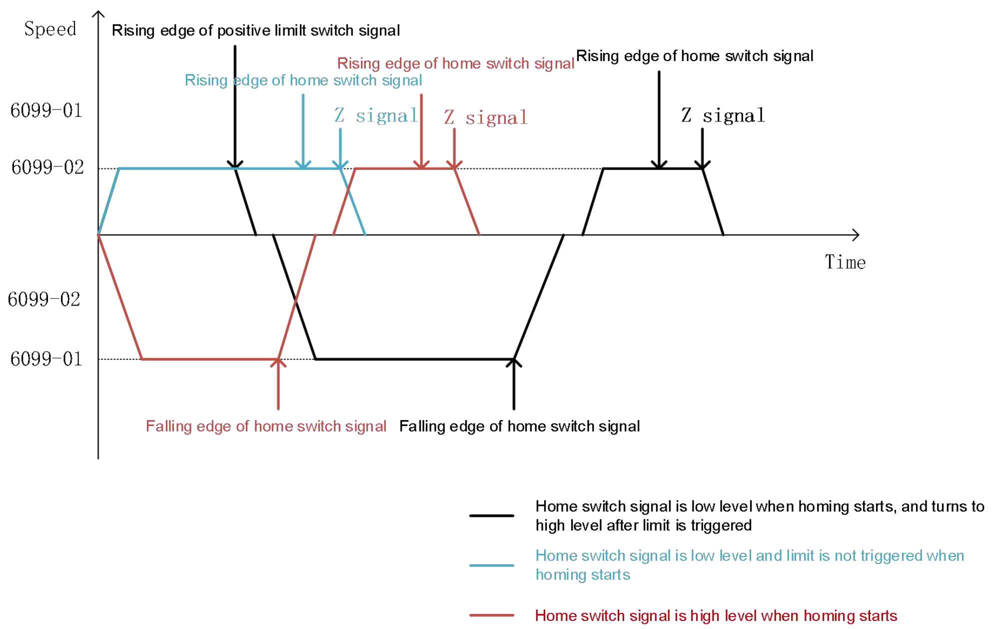


Figure 4-50 Speed-time curve of method 8

4.5.3.10 Method 9: Homing on negative home switch (rising edge) and index pulse-positive limit switch detection

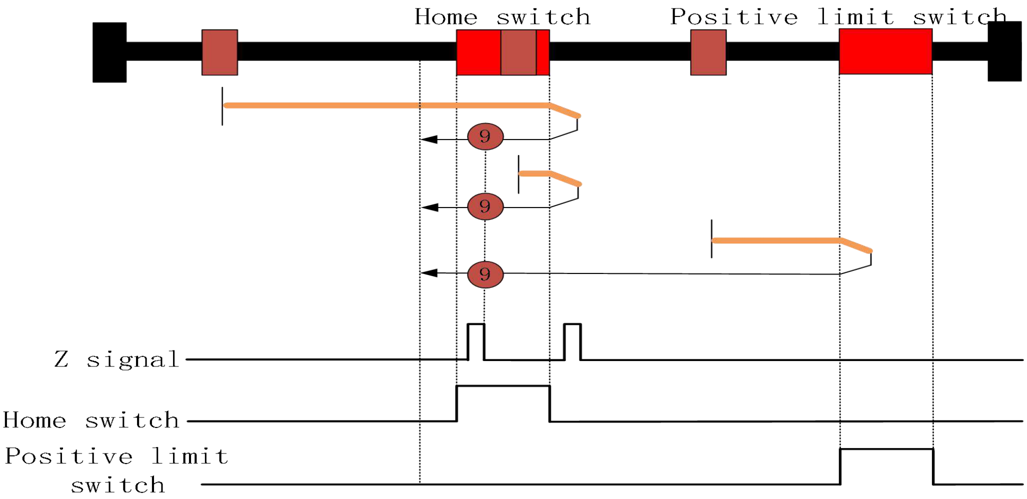


Figure 4-51 Method 9

- **When homing starts, if the home switch signal is low level,** the motor moves at a high speed (6099-01) in the positive direction.
  - ✓ After the home switch signal becomes high level, the motor keeps moving at a high speed (6099-01) in the positive direction. After the home switch signal becomes low level, the motor decelerates to 0 with the homing deceleration (609A), and moves in the negative direction with the homing acceleration (609A) to accelerate to a low speed (6099-02), and keeps moving in the negative direction at the low speed. After the home switch signal changes from low level to high level and the first Z signal shows, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.
  - ✓ After the positive limit switch signal becomes high level, the motor decelerates to 0 with the homing deceleration (609A), and moves in the negative direction with the homing acceleration (609A) to accelerate to a low speed (6099-02), and keeps moving in the negative direction at the low speed. After the home switch signal becomes high level and the first Z signal shows, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.

- **When homing starts, if the home switch signal is high level,** the motor moves at a high speed (6099-01) in the positive direction. After the home switch signal becomes low level, the motor decelerates to 0 with the homing deceleration (609A), and moves in the negative direction with the homing acceleration (609A) to accelerate to a low speed (6099-02), and keeps moving in the negative direction at the low speed. After the home switch signal becomes high level and the first Z signal shows, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.

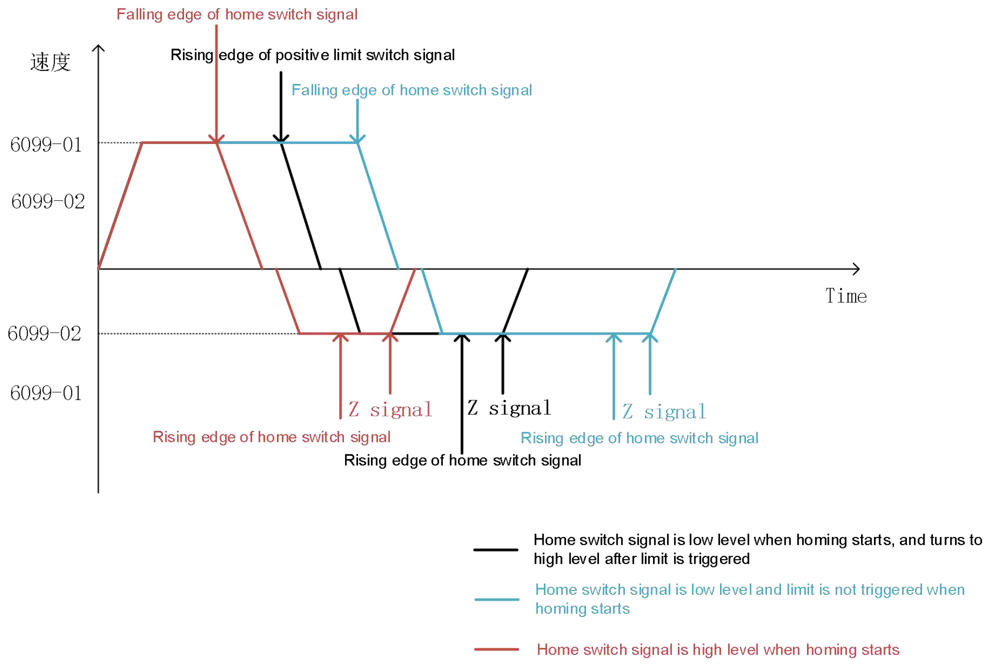


Figure 4-52 Speed-time curve of method 9

4.5.3.11 Method 10: Homing on positive home switch (falling edge) and index pulse-positive limit switch detection

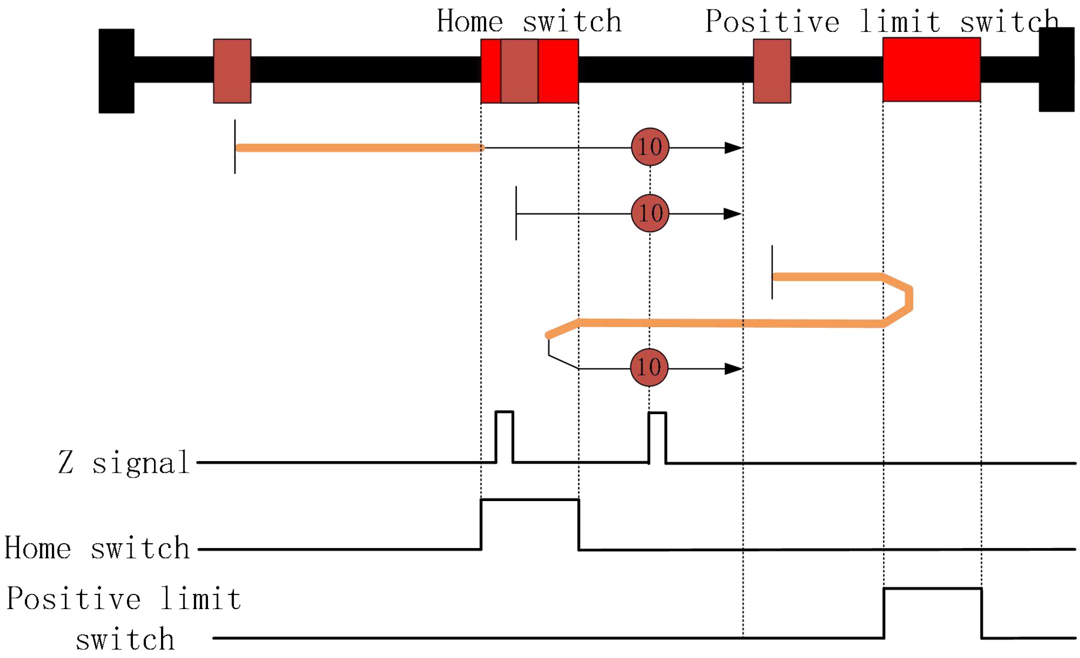


Figure 4-53 Method 10

- **When homing starts, if the home switch signal is low level,** the motor moves at a high speed (6099-01) in the positive direction.
  - ✓ After the home switch signal becomes high level, the motor moves with the homing deceleration (609A) to decelerate to a low speed (6099-02), and keeps moving at the low speed in the positive direction. After the home switch signal becomes low level and the first Z signal shows, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.
  - ✓ After the positive limit switch signal becomes high level, the motor decelerates to 0 with the homing deceleration (609A), and moves in the negative direction with the homing acceleration (609A) to accelerate to a high speed (6099-01), and keeps moving in the negative direction at the high speed. After the home switch signal becomes high level, the motor decelerates to 0 with the homing deceleration (609A), move in the positive direction to accelerate to a low speed (6099-02) with the homing acceleration (609A), and keeps moving in the positive direction at the low speed. After the home switch signal becomes low level and the first Z signal shows, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.

- **When homing starts, if the home switch signal is high level,** the motor moves at a low speed (6099-02) in the positive direction. After the home switch signal becomes low level and the first Z signal shows, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.

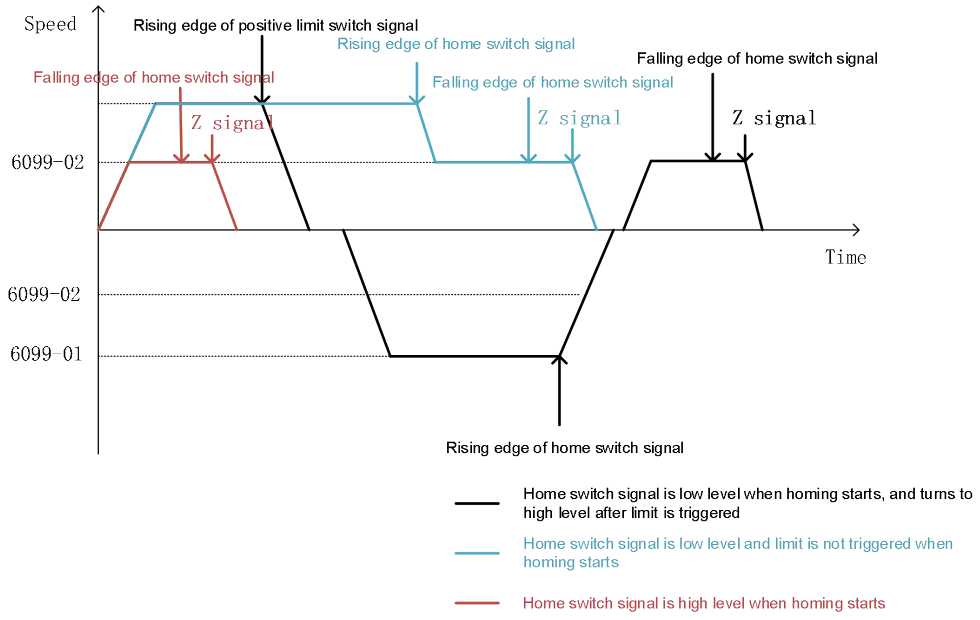


Figure 4-54 Speed-time curve of method 10

4.5.3.12 Method 11: Homing on positive home switch (falling edge) and index pulse-negative limit switch detection

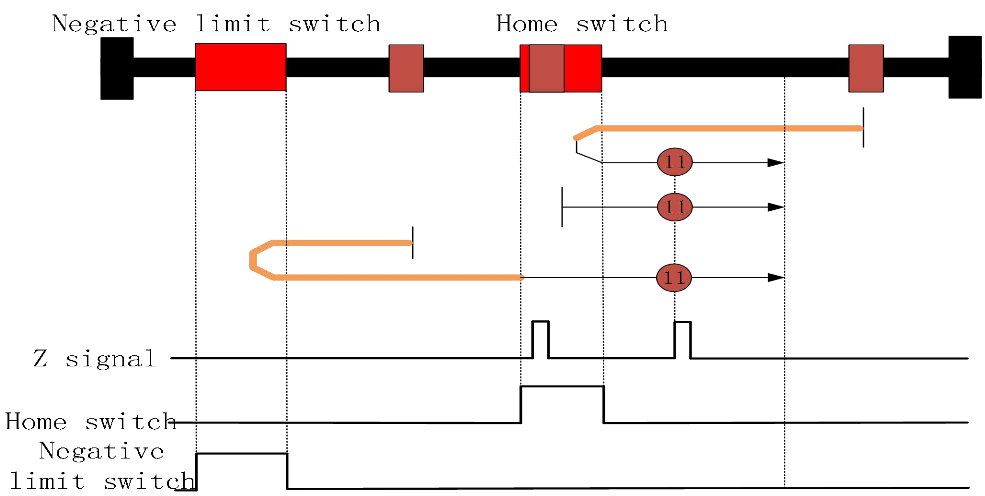


Figure 4-55 Method 11

- **When homing starts, if the home switch signal is low level,** the motor moves at a high speed (6099-01) in the negative direction.
  - ✓ After the home switch signal becomes high level, the motor moves with the homing deceleration (609A) to decelerate to 0, and moves in the positive direction to accelerate to a low speed (6099-02) with the homing acceleration (609A), and keeps moving at the low speed in the positive direction. After the

home switch signal becomes low level and the first Z signal shows, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.

- ✓ After the negative limit switch signal becomes high level, the motor decelerates to 0 with the homing deceleration (609A), and moves in the positive direction with the homing acceleration (609A) to accelerate to a high speed (6099-01), and keeps moving in the positive direction at the high speed. After the home switch signal becomes high level, the motor decelerates to a low speed (6099-02) with the homing deceleration (609A), and keeps moving in the positive direction at the low speed. After the home switch signal becomes low level and the first Z signal shows, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.

- **When homing starts, if the home switch signal is high level,** the motor moves at a low speed (6099-02) in the positive direction. After the home switch signal becomes low level and the first Z signal shows, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.

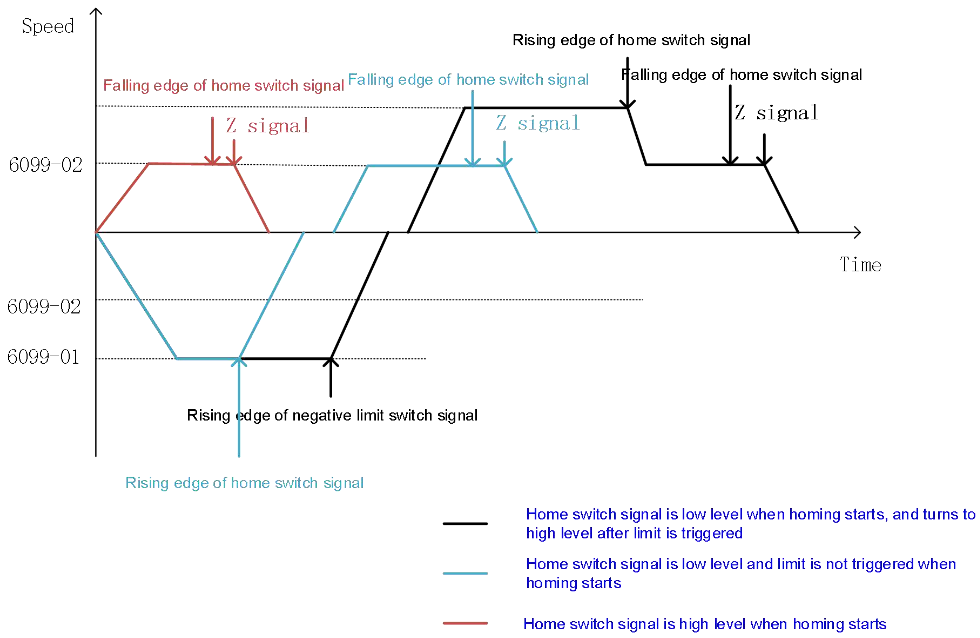


Figure 4-56 Speed-time curve of method 11

4.5.3.13 Method 12: Homing on negative home switch (rising edge) and index pulse-negative limit switch detection

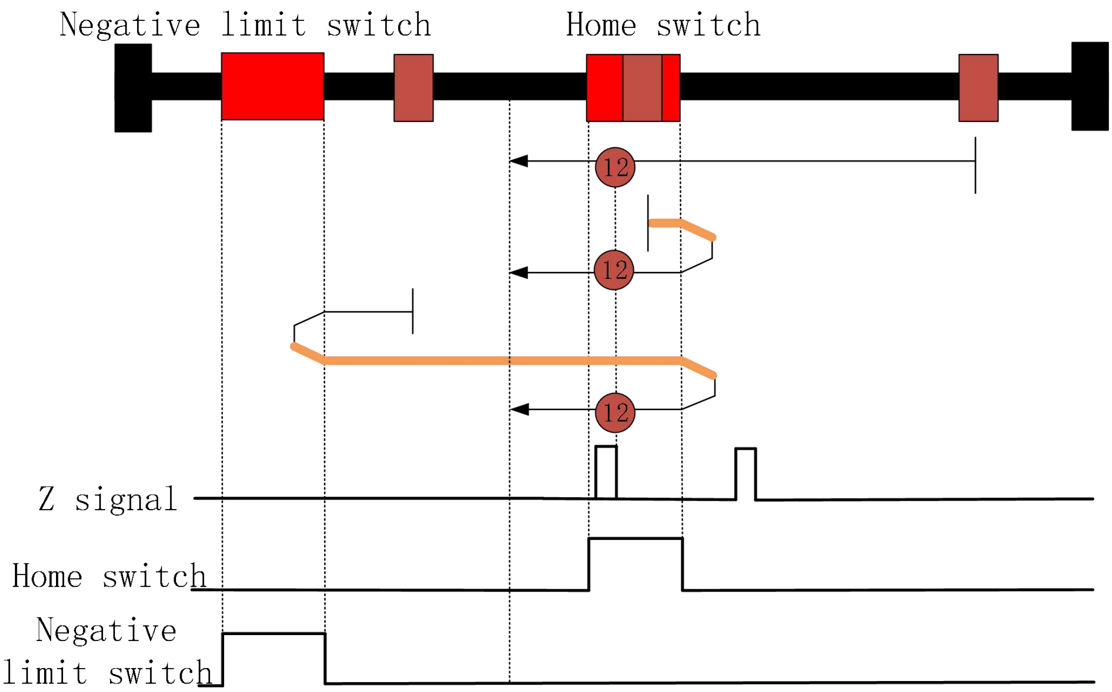


Figure 4-57 Method 12

- **When homing starts, if the home switch signal is low level,** the motor moves at a low speed (6099-02) in the negative direction.
  - ✓ After the home switch signal becomes high level, the motor keeps moving in the negative direction at a low speed (6099-02). After the home switch signal becomes high level and the first Z signal shows, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.
  - ✓ After the negative limit switch signal becomes high level, the motor decelerates to 0 with the homing deceleration (609A), and moves in the positive direction with the homing acceleration (609A) to accelerate to a high speed (6099-01), and keeps moving in the positive direction at the high speed. After the home switch signal becomes low level, the motor decelerates to 0 with the homing deceleration (609A), moves in the negative direction to accelerate to a low speed (6099-02) with the homing acceleration (609A), and keeps moving in the negative direction at the low speed. After the home switch signal becomes high level and the first Z signal shows, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.

- **When homing starts, if the home switch signal is high level,** the motor moves at a high speed (6099-01) in the positive direction. After the home switch signal becomes low level, the motor decelerates to 0 with the homing deceleration (609A), and moves in the negative direction with the homing acceleration (609A) to accelerate to a low speed (6099-02), and keeps moving in the negative direction at the low speed. After the home switch signal becomes high level and the first Z signal shows, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.

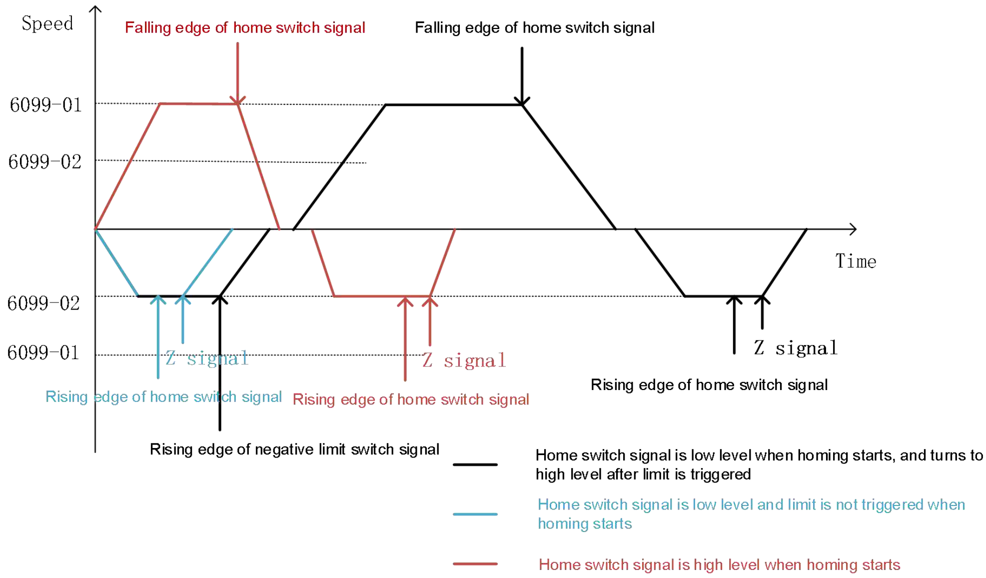


Figure 4-58 Speed-time curve of method 12

4.5.3.14 Method 13: Homing on positive home switch (rising edge) and index pulse-negative limit switch detection

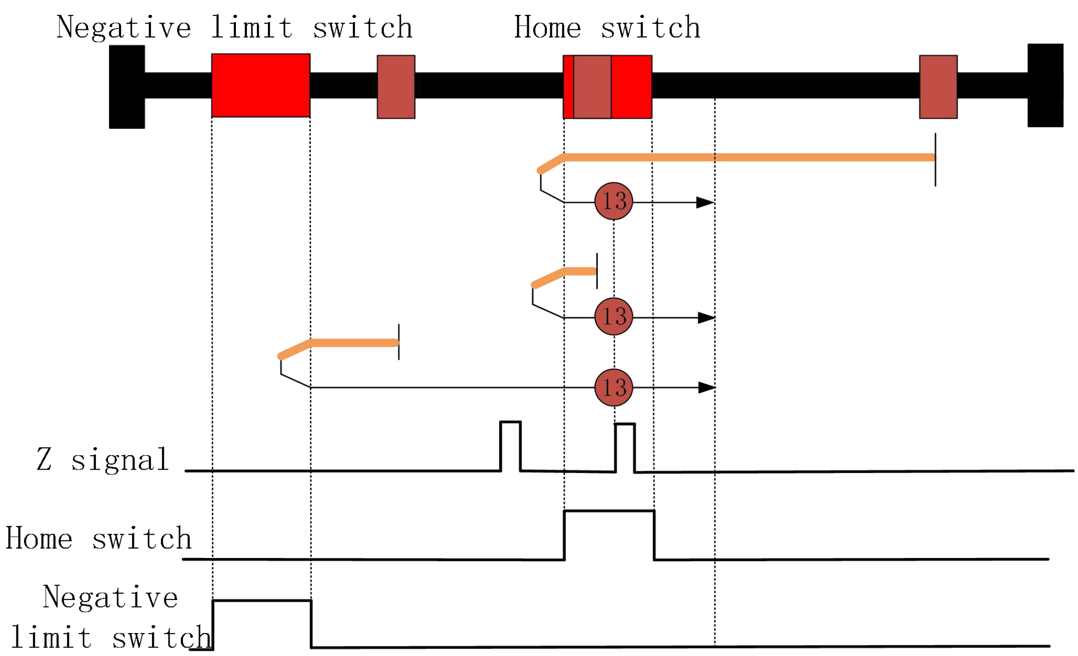


Figure 4-59 Method 13

- **When homing starts, if the home switch signal is low level,** the motor moves at a high speed (6099-01) in the negative direction.
  - ✓ After the home switch signal becomes high level, the motor keeps moving in the negative direction at the high speed. After the home switch signal becomes low level, the motor moves with the homing deceleration (609A) to decelerate to 0, and moves in the positive direction to accelerate to a low speed (6099-02) with the homing acceleration (609A), and keeps moving at the low speed in the positive direction. After the home switch signal becomes high level and the first Z signal shows, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.
  - ✓ After the negative limit switch signal becomes high level, the motor decelerates to 0 with the homing deceleration (609A), and moves in the positive direction with the homing acceleration (609A) to accelerate to a low speed (6099-02), and keeps moving in the positive direction at the low speed. After the home switch signal becomes high level and the first Z signal shows, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.
- **When homing starts, if the home switch signal is high level,** the motor moves at a high speed (6099-01) in the negative direction. After the home switch signal becomes low level, the motor decelerates to 0 with the homing deceleration (609A), and moves in the positive direction with the homing acceleration (609A) to accelerate to a low speed (6099-02), and keeps moving in the positive direction at the low speed. After the home switch signal becomes high level and the first Z signal shows, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.

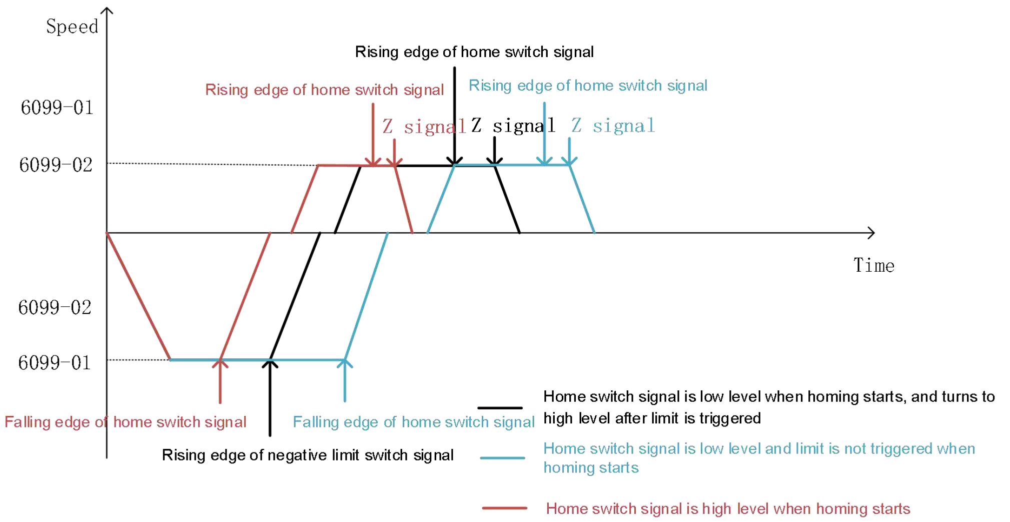


Figure 4-60 Speed-time curve of method 13

4.5.3.15 Method 14: Homing on negative home switch (falling edge) and index pulse-negative limit switch detection

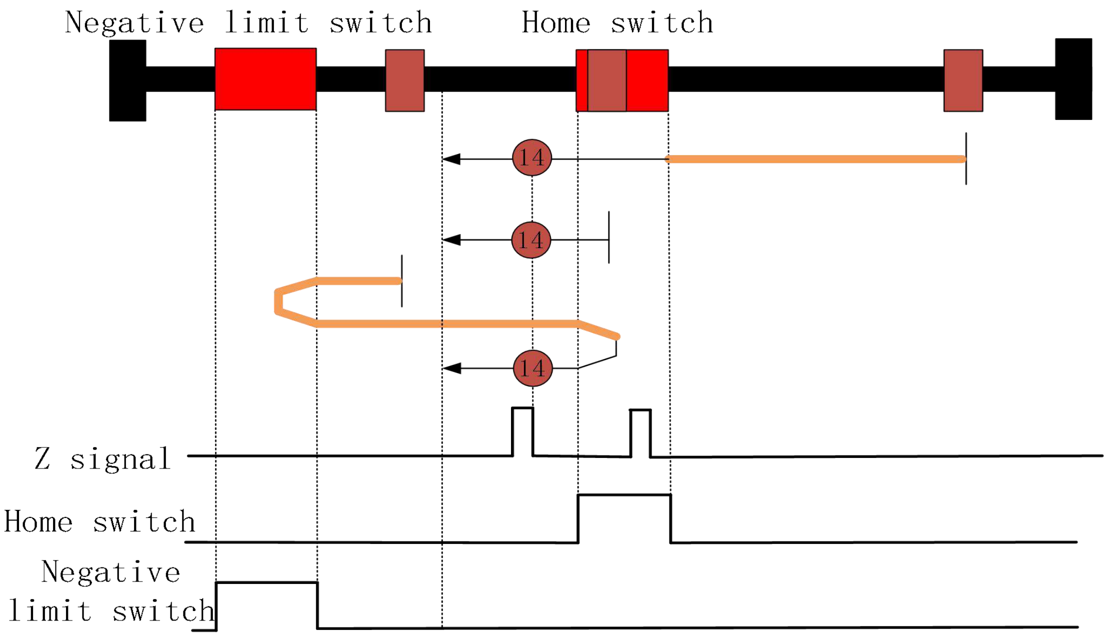


Figure 4-61 Method 14

- **When homing starts, if the home switch signal is low level,** the motor moves at a high speed (6099-02) in the negative direction.
  - ✓ After the home switch signal becomes high level, the motor moves with the homing deceleration (609A) to decelerate to a low speed (6099-02), and keeps moving in the negative direction at a low speed. After the home switch signal becomes low level and the first Z signal shows, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.
  - ✓ After the negative limit switch signal becomes high level, the motor decelerates to 0 with the homing deceleration (609A), and moves in the positive direction with the homing acceleration (609A) to accelerate to a high speed (6099-01), and keeps moving in the positive direction at the high speed. After the home switch signal becomes high level, the motor decelerates to 0 with the homing deceleration (609A), moves in the negative direction to accelerate to a low speed (6099-02) with the homing acceleration (609A), and keeps moving in the negative direction at the low speed. After the home switch signal becomes low level and the first Z signal shows, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.
- **When homing starts, if the home switch signal is high level,** the motor moves at a low speed (6099-01) in the negative direction. After the home switch signal becomes low level and the first Z signal shows, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.

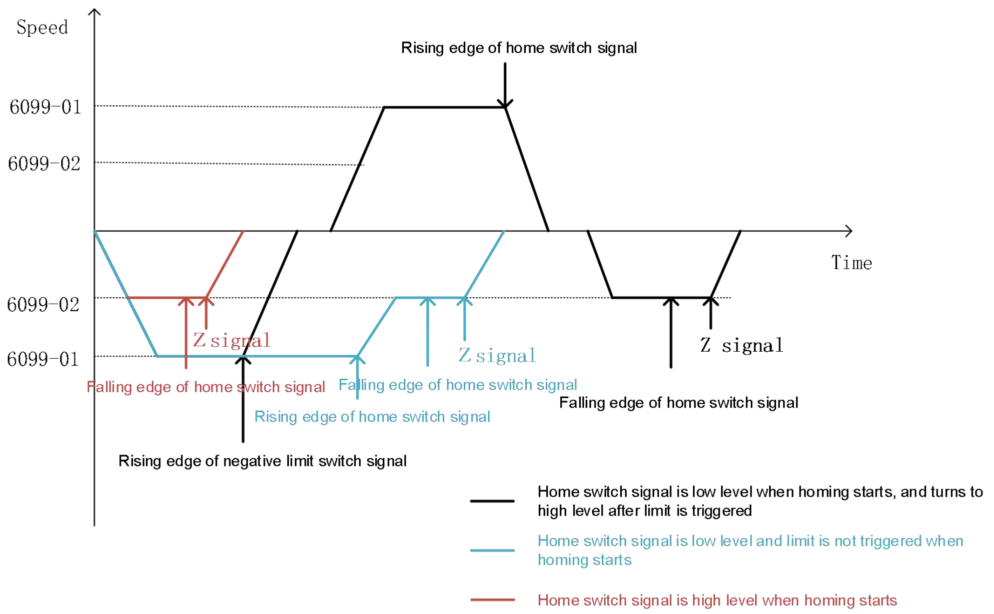


Figure 4-62 Speed-time curve of method 14

4.5.3.16 Method 15: Reserved

4.5.3.17 Method 16: Reserved

4.5.3.18 Method 17: Homing on negative limit switch (falling edge)

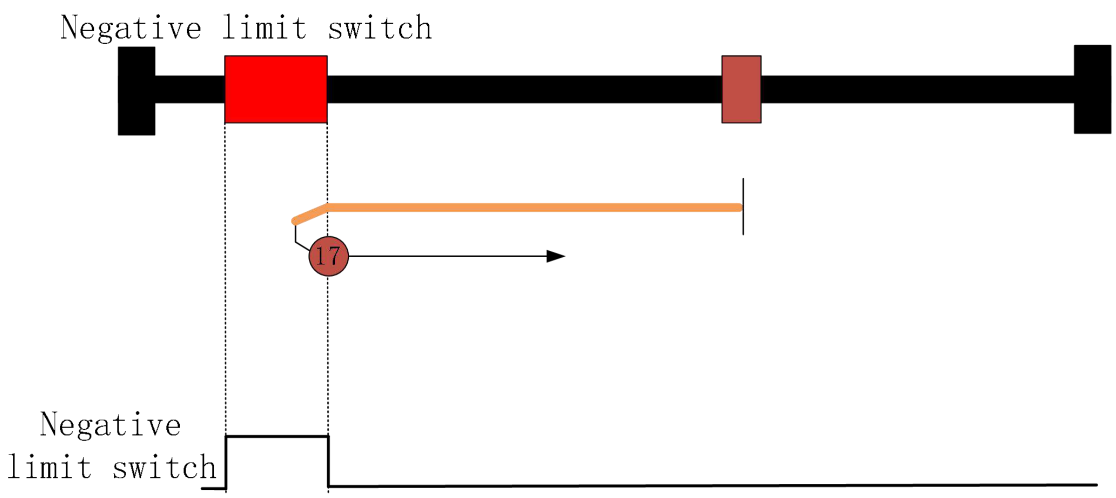


Figure 4-63 Method 17

When homing starts, the motor moves at a high speed (6099-01) in the negative direction. When the negative limit switch signal becomes high level, the motor decelerates to 0 with the homing deceleration (609A), moves in the positive direction with the homing acceleration (609A) to accelerate to a low speed (6099-02), and keeps moving in the positive direction at the low speed. After the negative limit switch signal becomes low level, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.

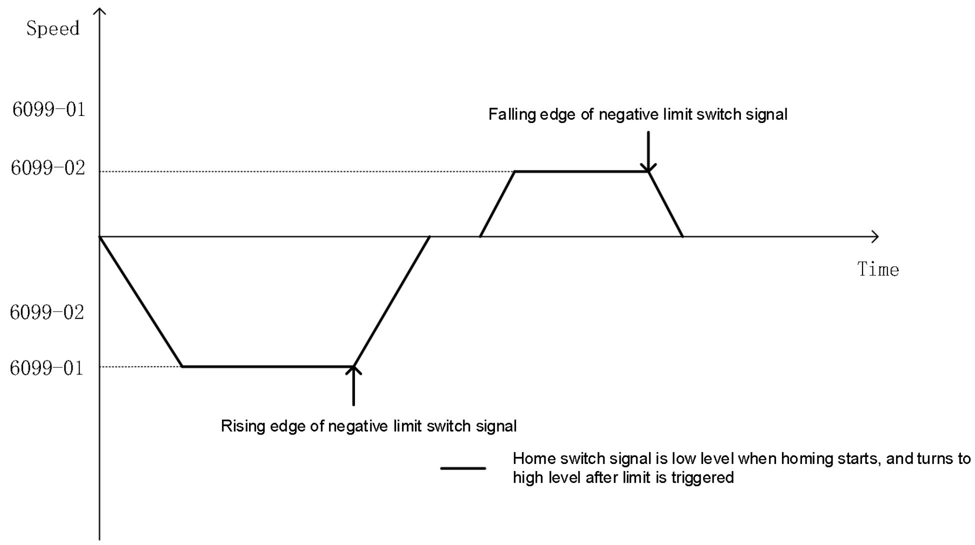


Figure 4-64 Speed-time curve of method 17

4.5.3.19 Method 18: Homing on positive limit switch (falling edge)

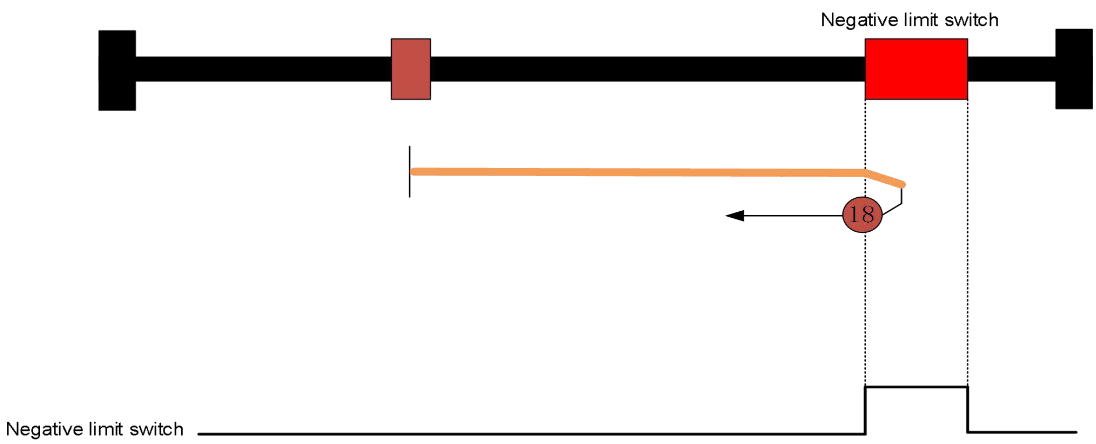


Figure 4-65 Method 18

When homing starts, the motor moves at a high speed (6099-01) in the positive direction. When the positive limit switch signal becomes high level, the motor decelerates to 0 with the homing deceleration (609A), moves in the negative direction with the homing acceleration (609A) to accelerate to a low speed (6099-02), and keeps moving in the negative direction at the low speed. After the positive limit switch signal becomes low level, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.

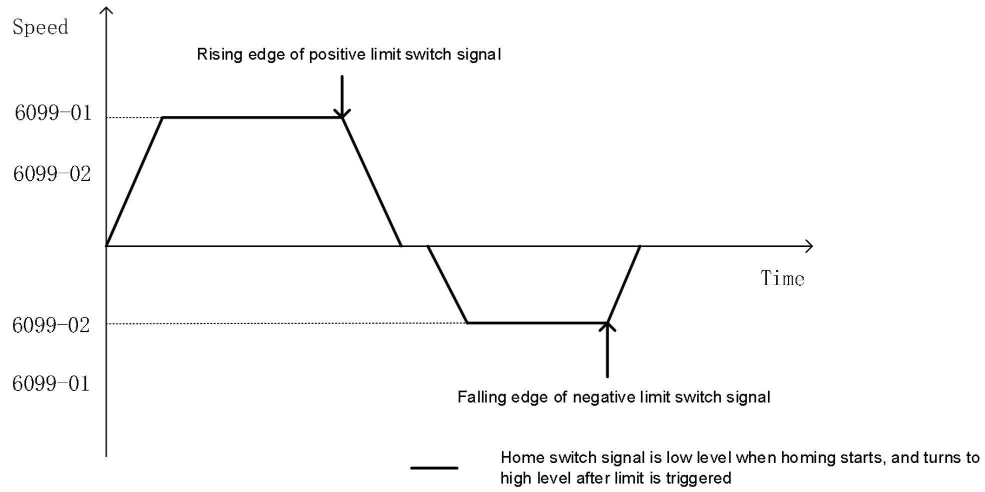


Figure 4-66 Speed-time curve of method 18

4.5.3.20 Method 19: Homing on negative home switch (falling edge)

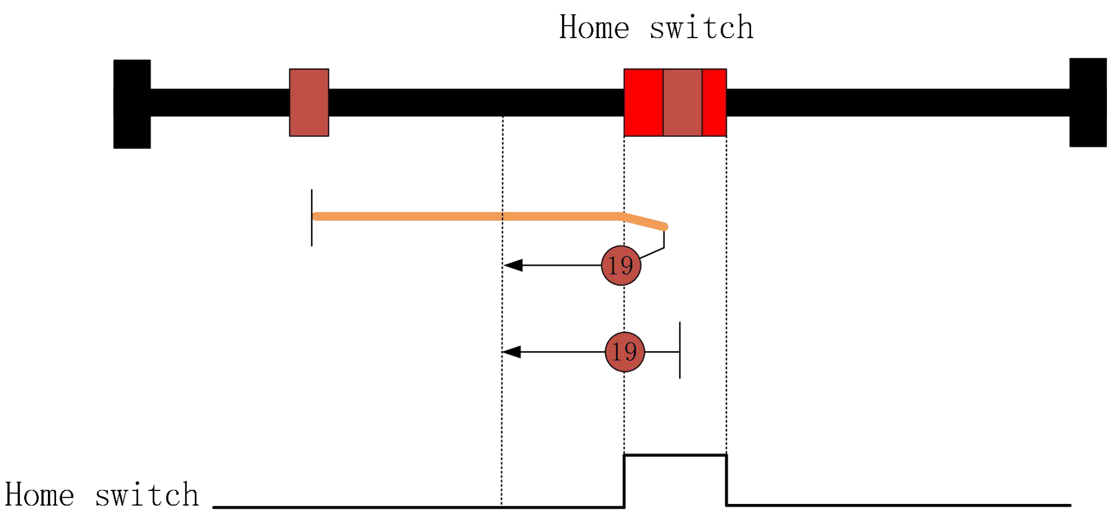


Figure 4-67 Method 19

- **When homing starts, if the home switch signal is low level,** the motor moves at a high speed (6099-01) in the positive direction. After the positive home switch signal becomes high level, the motor decelerates to 0 with the homing deceleration (609A), and moves in the negative direction with the homing acceleration (609A) to accelerate to a low speed (6099-02), and keeps moving in the negative direction at the low speed. After the positive home switch signal becomes low level, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.
- **When homing starts, if the home switch signal is high level,** the motor moves at a low speed (6099-02) in the negative direction. After the home switch signal becomes low level, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.

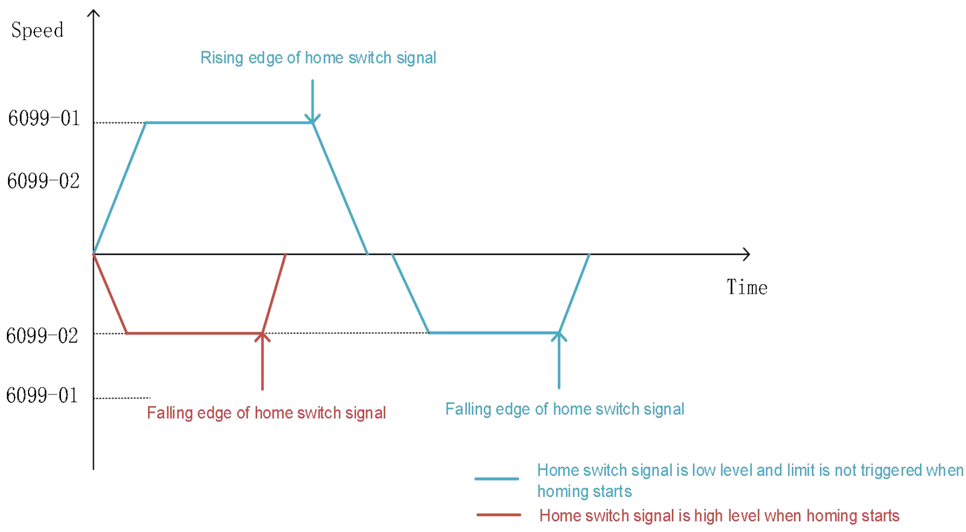


Figure 4-68 Speed-time curve of method 19

4.5.3.21 Method 20: Homing on positive limit switch (rising edge)

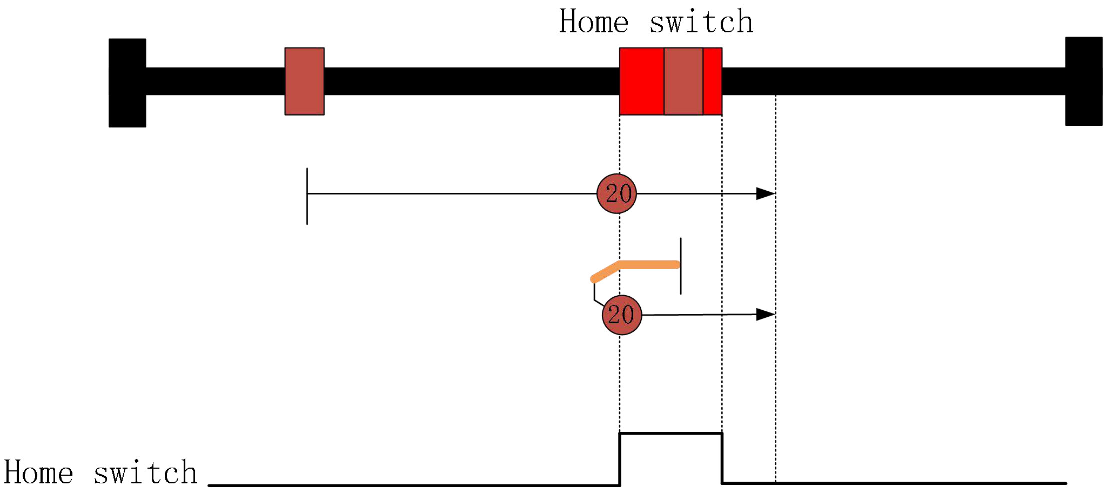


Figure 4-69 Method 20

- **When homing starts, if the positive home switch signal is low level,** the motor moves at a low speed (6099-02) in the positive direction. After the positive home switch signal becomes high level, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.
- **When homing starts, if the positive home switch signal is high level,** the motor moves at a high speed (6099-01) in the negative direction. After the positive home switch signal becomes low level, the motor decelerates to 0 with the homing deceleration (609A), and moves in the positive direction with the homing acceleration (609A) to accelerate to a low speed (6099-02), and keeps moving in the positive direction at the low speed. After the positive home switch signal becomes high level, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.

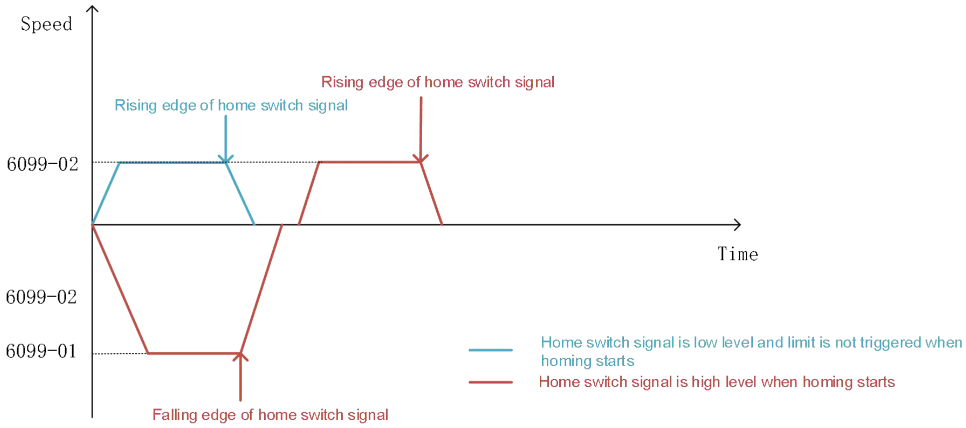


Figure 4-70 Speed-time curve of method 20

4.5.3.22 Method 21: Homing on negative home switch (falling edge)

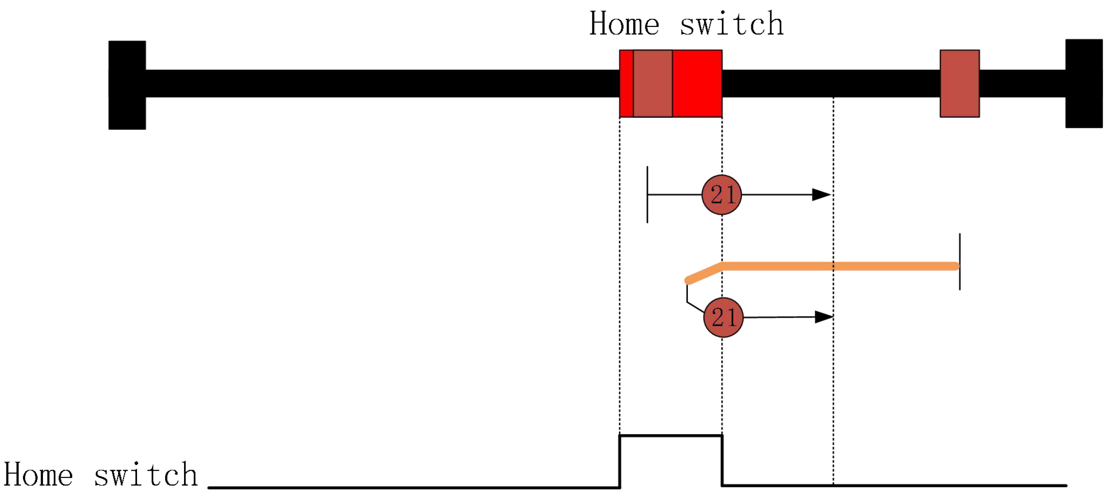


Figure 4-71 Method 21

- **When homing starts, if the home switch signal is low level,** the motor moves at a high speed (6099-01) in the negative direction. After the home switch signal becomes high level, the motor decelerates to 0 with the homing deceleration (609A), and moves in the positive direction with the homing acceleration (609A) to accelerate to a low speed (6099-02), and keeps moving in the positive direction at the low speed. After the home switch signal becomes low level, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.
- **When homing starts, if the home switch signal is high level,** the motor moves at a low speed (6099-02) in the positive direction. After the home switch signal becomes low level, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.

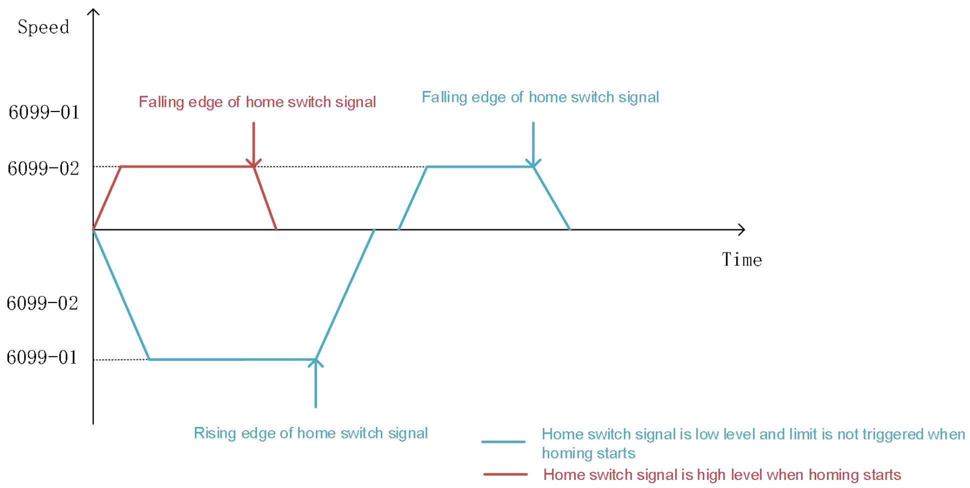


Figure 4-72 Speed-time curve of method 21

4.5.3.23 Method 22: Homing on negative home switch (rising edge)

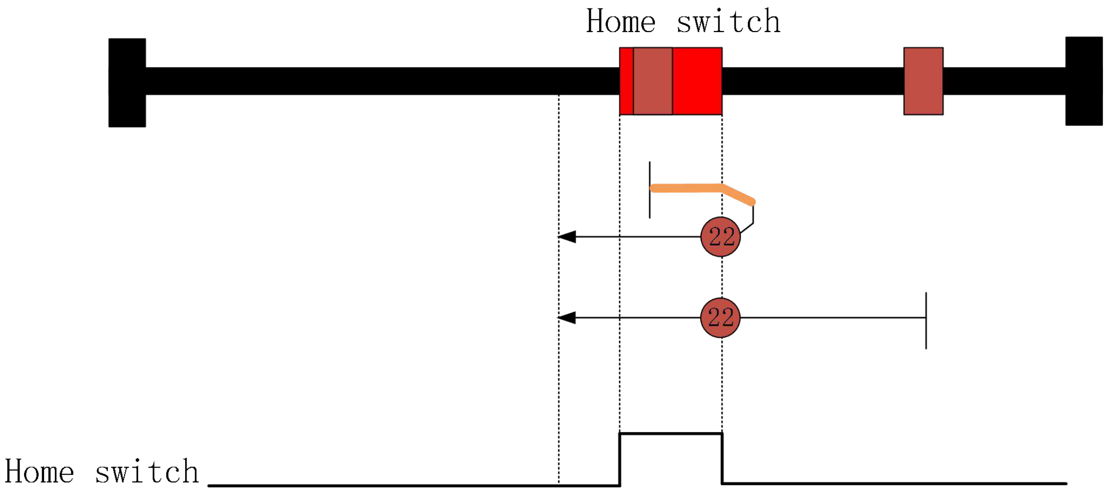
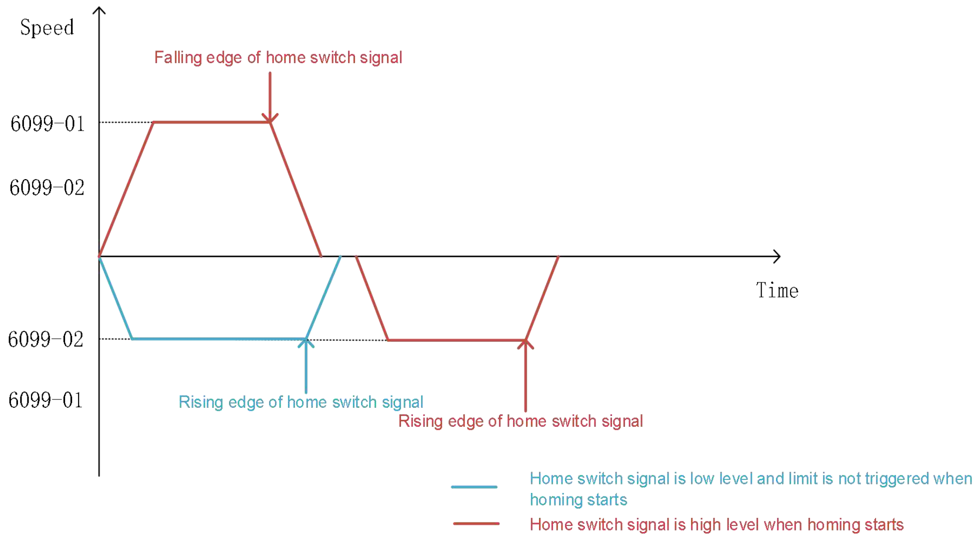


Figure 4-73 Method 22

- **When homing starts, if the home switch signal is low level,** the motor moves at a low speed (6099-02) in the negative direction. After the home switch signal becomes high level, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.
- **When homing starts, if the home switch signal is high level,** the motor moves at a high speed (6099-01) in the positive direction. After the home switch signal becomes low level, the motor decelerates to 0 with the homing deceleration (609A), and moves in the negative direction with the homing acceleration (609A) to accelerate to a low speed (6099-02), and keeps moving in the negative direction at the low speed. After the home switch signal becomes high level, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.



●

Figure 4-74 Speed-time curve of method 22

4.5.3.24 Method 23: Homing on negative home switch (falling edge) -positive limit switch detection

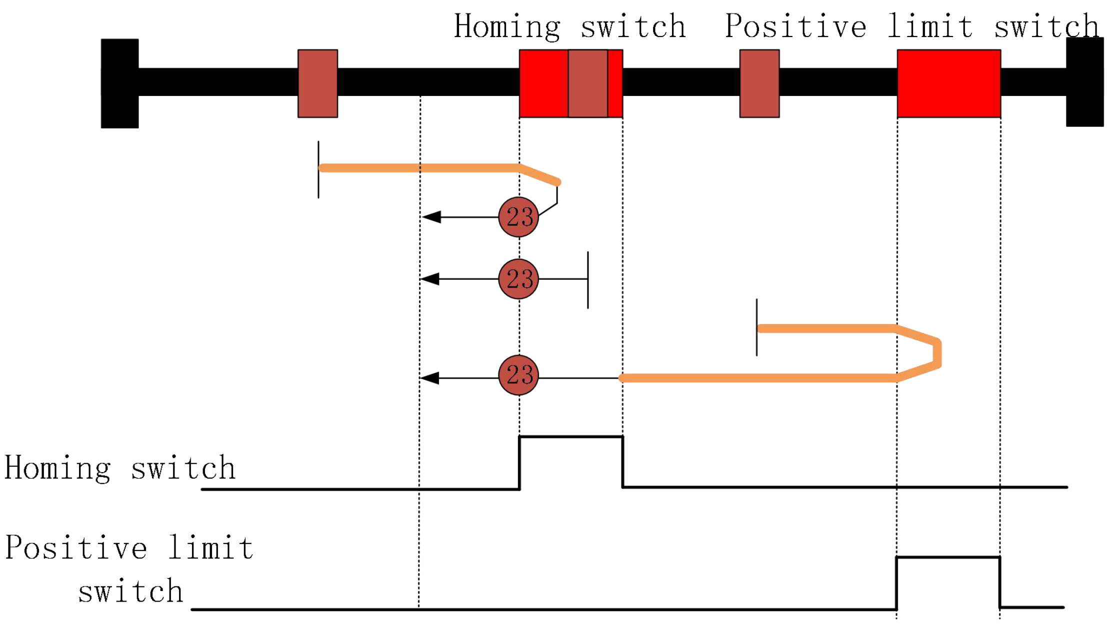


Figure 4-75 Method 23

- **When homing starts, if the home switch signal is low level,** the motor moves at a high speed (6099-01) in the positive direction.
  - ✓ After the home switch signal becomes high level, the motor decelerates to 0 with the homing deceleration (609A), and moves in the negative direction with the homing acceleration (609A) to accelerate to a low speed (6099-02), and keeps moving in the negative direction at the low speed. After the home switch signal becomes low level, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target

reached is set to 1 when the motor stops.

✓ After the positive limit switch signal becomes high level, the motor decelerates to 0 with the homing deceleration (609A), and moves in the negative direction with the homing acceleration (609A) to accelerate to a high speed (6099-01), and keeps moving in the negative direction at the high speed. After the home switch signal becomes high level, the motor decelerates to a low speed (6099-01) with the homing deceleration (609A), and keeps moving in the negative direction at the low speed. After the home switch becomes low level, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.

- **When homing starts, if the home switch signal is high level,** the motor moves at a low speed (6099-02) in the negative direction. After the home switch signal becomes low level, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.

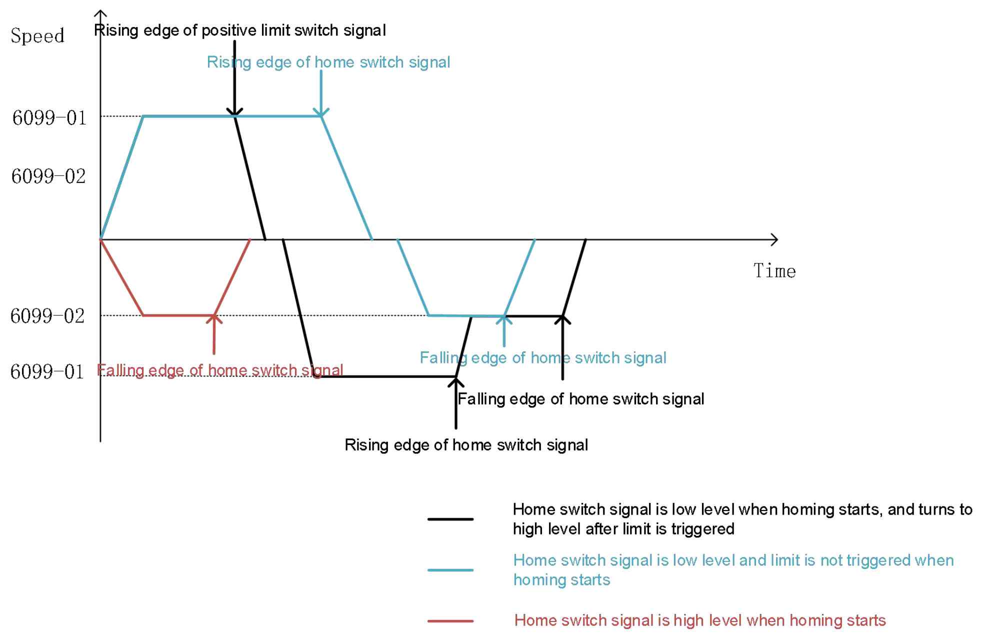


Figure 4-76 Speed-time curve of method 23

4.5.3.25 Method 24: Homing on positive home switch (rising edge)-positive limit switch detection

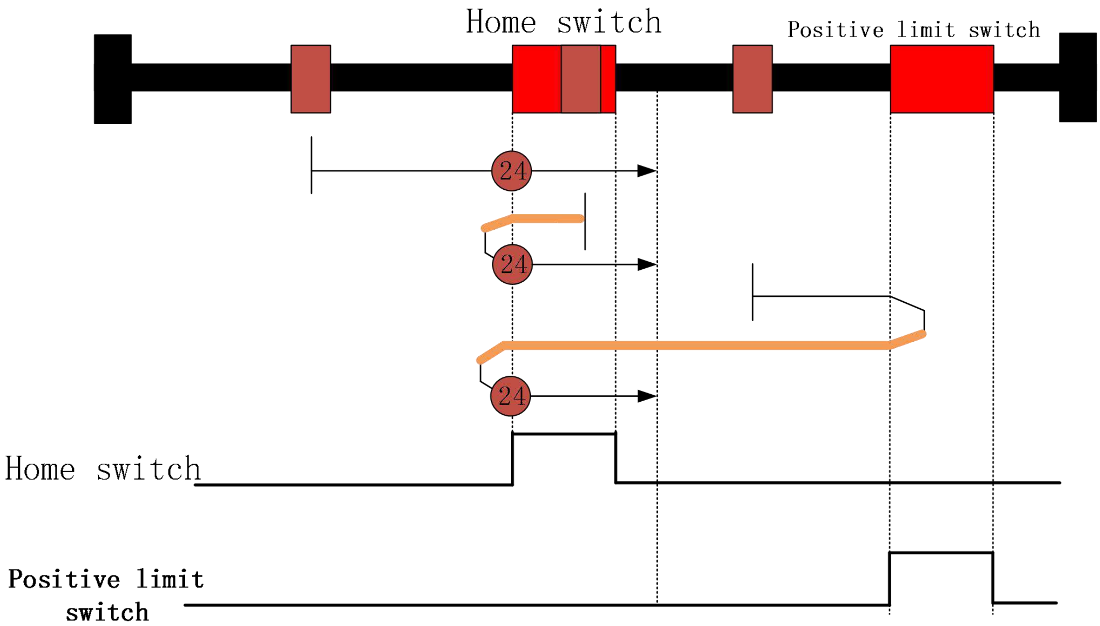


Figure 4-77 Method 24

- **When homing starts, if the home switch signal is low level,** the motor moves at a low speed (6099-02) in the positive direction.
  - ✓ After the home switch signal becomes high level, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.
  - ✓ After the positive limit switch signal becomes high level, the motor decelerates to 0 with the homing deceleration (609A), and moves in the negative direction with the homing acceleration (609A) to accelerate to a high speed (6099-01), and keeps moving in the negative direction at the high speed. After the home switch signal becomes low level, the motor decelerates to 0 with the homing deceleration (609A), and moves in the positive direction with the homing acceleration (609A) to accelerate to a low speed (6099-02), and keeps moving in the positive direction at the low speed. After the home switch signal becomes high level, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.
- **When homing starts, if the home switch signal is high level,** the motor moves at a high speed (6099-01) in the negative direction. After the home switch signal becomes low level, the motor decelerates to 0 with the homing deceleration (609A), and moves in the positive direction with the homing acceleration (609A) to accelerate to a low speed (6099-02), and keeps moving in the positive direction at the low speed. After the home switch signal becomes high level, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.

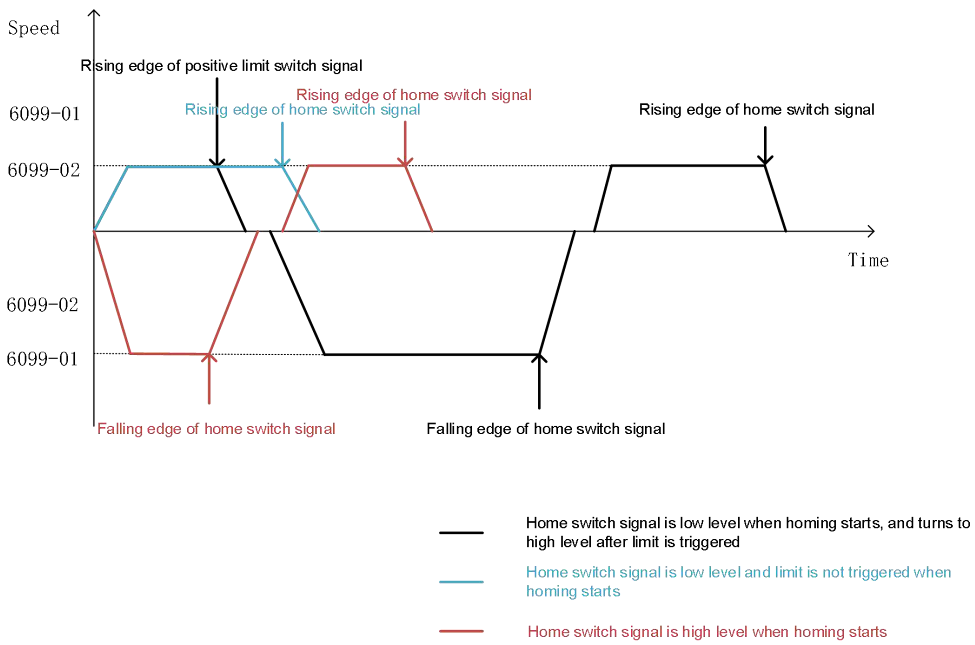


Figure 4-78 Speed-time curve of method 24

4.5.3.26 Method 25: Homing on negative home switch (rising edge)-positive limit switch detection

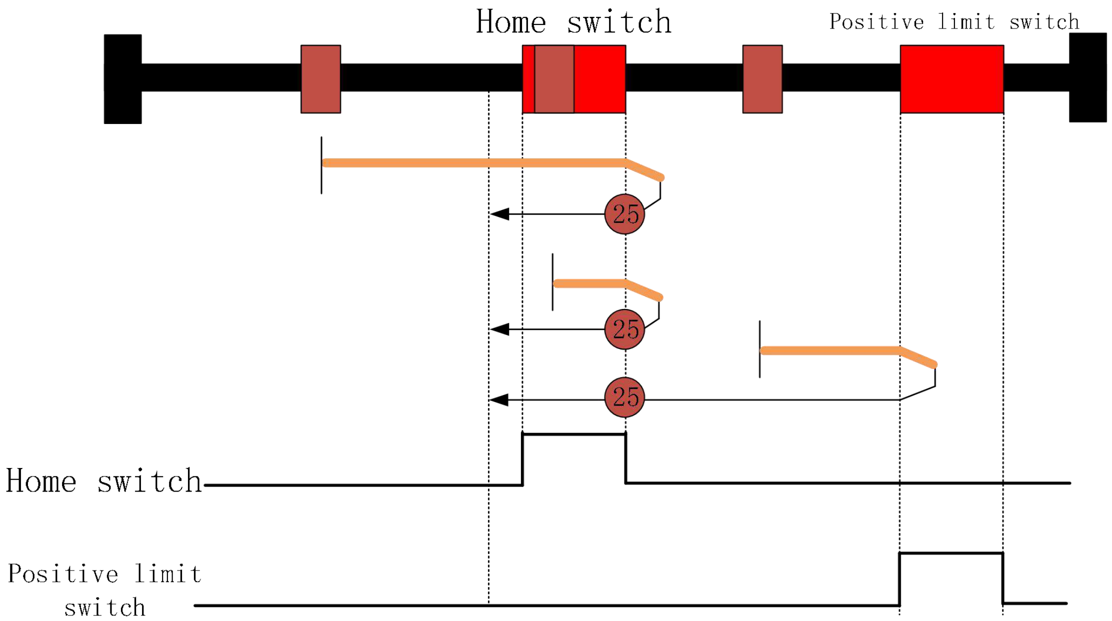


Figure 4-79 Method 25

Regardless of the high level or low level of the home switch signal, the motor moves in the positive direction.

When homing starts, the motor moves in the positive direction at a high speed (6099-01). After the home switch signal becomes low level or the positive limit switch becomes high level, the motor decelerates to 0 with the homing deceleration (609A), and moves in the negative direction with the homing acceleration (609A) to accelerate to a low speed (6099-02), and keeps moving in the negative direction at the low speed. After the home switch signal becomes high level, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.

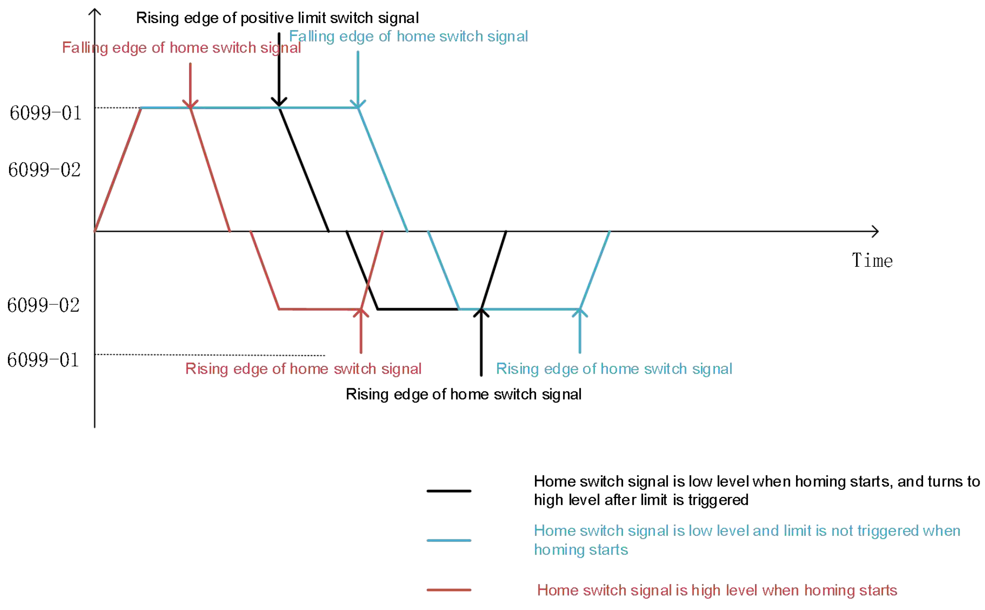


Figure 4-80 Speed-time curve of method 25

4.5.3.27 Method 26: Homing on positive home switch (falling edge)-positive limit switch detection

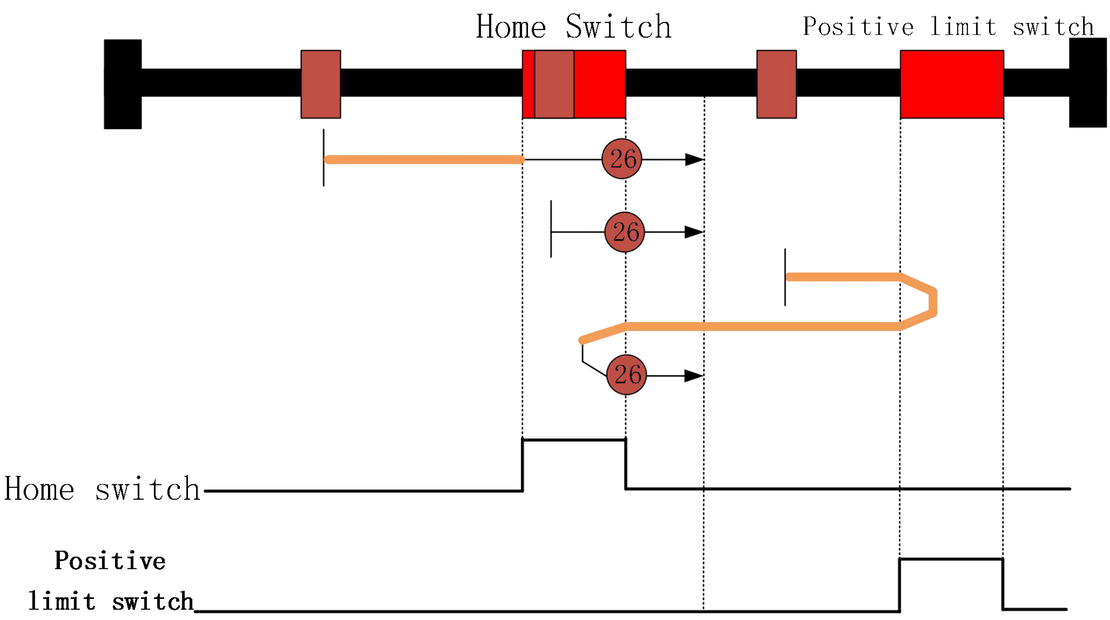


Figure 4-81 Method 26

- **When homing starts, if the home switch signal is low level,** the motor moves at a high speed (6099-01) in the positive direction.
  - ✓ After the home switch signal becomes high level, the motor decelerates to a low speed (6099-02) with the homing deceleration (609A), and keeps moving in the

positive direction at the low speed (6099-02). After the home switch signal becomes low level, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.

✓ After the positive limit switch signal becomes high level, the motor decelerates to 0 with the homing deceleration (609A), and moves in the negative direction with the homing acceleration (609A) to accelerate to a high speed (6099-01), and keeps moving in the negative direction at the high speed. After the home switch signal becomes high level, the motor decelerates to 0 with the homing deceleration (609A), and moves in the positive direction with the homing acceleration (609A) to accelerate to a low speed (6099-02), and keeps moving in the positive direction at the low speed. After the home switch signal becomes low level, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.

- **When homing starts, if the home switch signal is high level,** the motor moves at a low speed (6099-02) in the positive direction. After the home switch signal becomes low level, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.

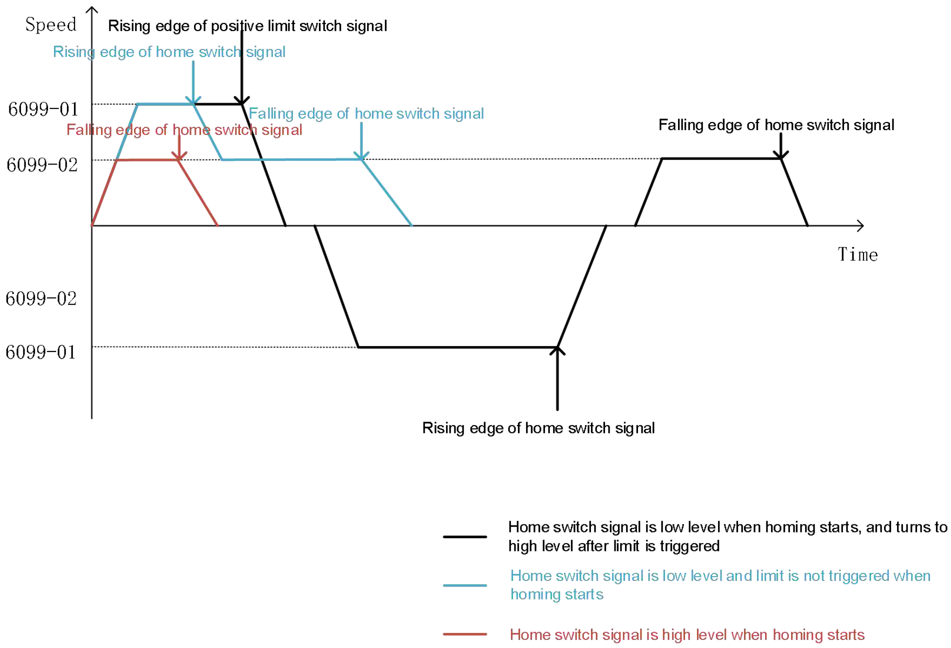


Figure 4-82 Speed-time curve of method 26

4.5.3.28 Method 27: Homing on positive home switch (falling edge)-negative limit switch detection

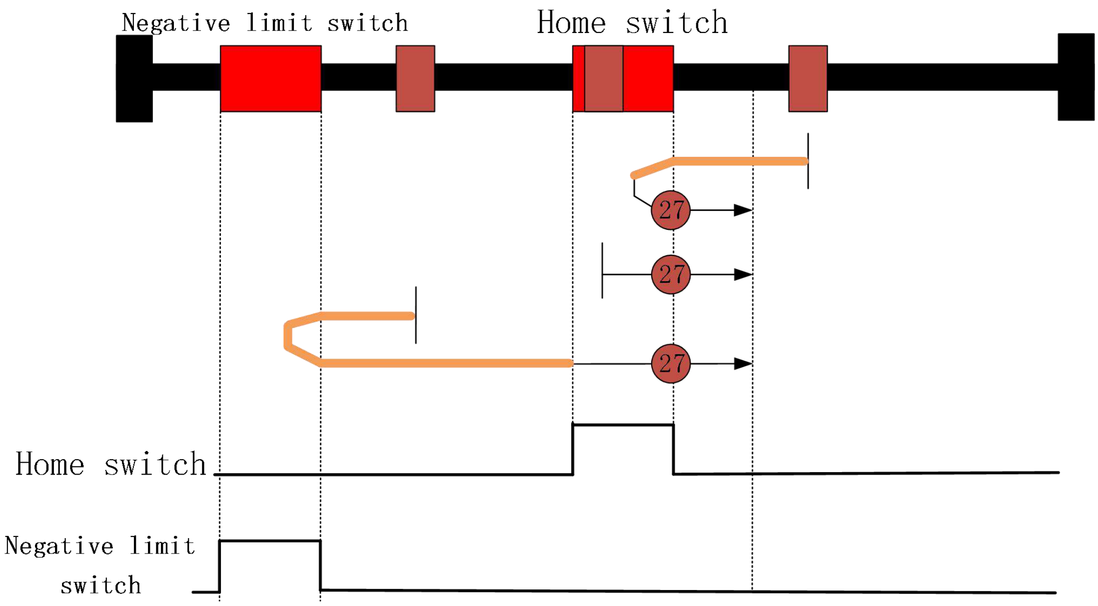


Figure 4-83 Method 27

- **When homing starts, if the home switch signal is low level,** the motor moves at a high speed (6099-01) in the negative direction.
  - ✓ After the home switch signal becomes high level, the motor decelerates to 0 with the homing deceleration (609A), moves in the positive direction with the homing acceleration to accelerate to a low speed (6099-02), and keeps moving in the positive direction at the low speed (6099-02). After the home switch signal becomes low level, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.
  - ✓ After the negative limit switch signal becomes high level, the motor decelerates to 0 with the homing deceleration (609A), and moves in the positive direction with the homing acceleration (609A) to accelerate to a high speed (6099-01), and keeps moving in the positive direction at the high speed. After the home switch signal becomes high level, the motor moves in the positive direction with the homing acceleration (609A) to accelerate to a low speed (6099-02), and keeps moving in the positive direction at the low speed. After the home switch signal becomes low level, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.
- **When homing starts, if the home switch signal is high level,** the motor moves at a low speed (6099-02) in the positive direction. After the home switch signal becomes low level, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.

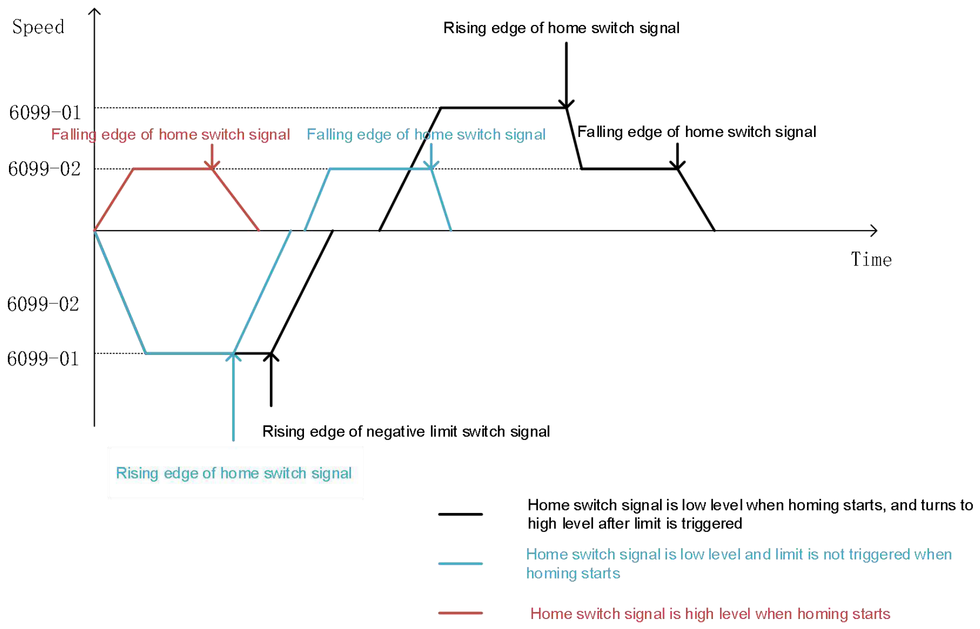


Figure 4-84 Speed-time curve of method 27

4.5.3.29 Method 28: Homing on negative home switch(rising edge)-negative limit switch detection

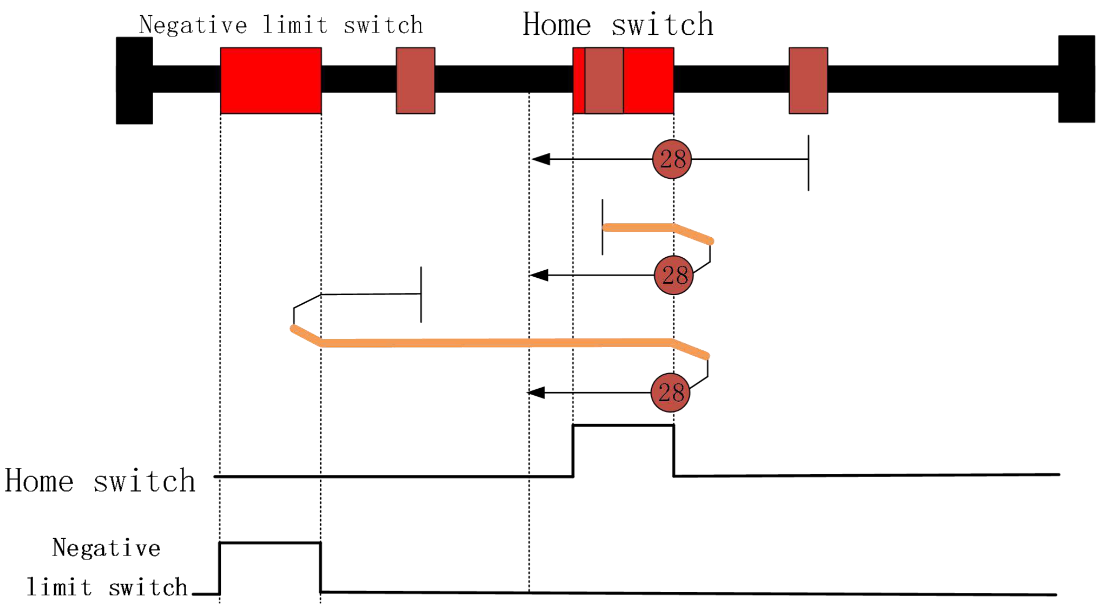


Figure 4-85 Method 28

- **When homing starts, if the home switch signal is low level,** the motor moves at a low speed (6099-02) in the negative direction.
  - ✓ After the home switch signal becomes high level, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.
  - ✓ After the negative limit switch signal becomes high level, the motor decelerates to 0 with the homing deceleration (609A), and moves in the positive direction with the homing acceleration (609A) to accelerate to a high speed (6099-01), and

keeps moving in the positive direction at the high speed. After the home switch signal becomes low level, the motor decelerates to 0 with the homing deceleration (609A), and moves in the negative direction with the homing acceleration (609A) to accelerate to a low speed (6099-02), and keeps moving in the negative direction at the low speed. After the home switch signal becomes high level, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.

- **When homing starts, if the home switch signal is high level,** the motor moves at a high speed (6099-01) in the positive direction. After the home switch signal becomes low level, the motor decelerates to 0 with the homing deceleration (609A), and moves in the negative direction with the homing acceleration (609A) to accelerate to a low speed (6099-02), and keeps moving in the negative direction at the low speed. After the home switch signal becomes high level, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.

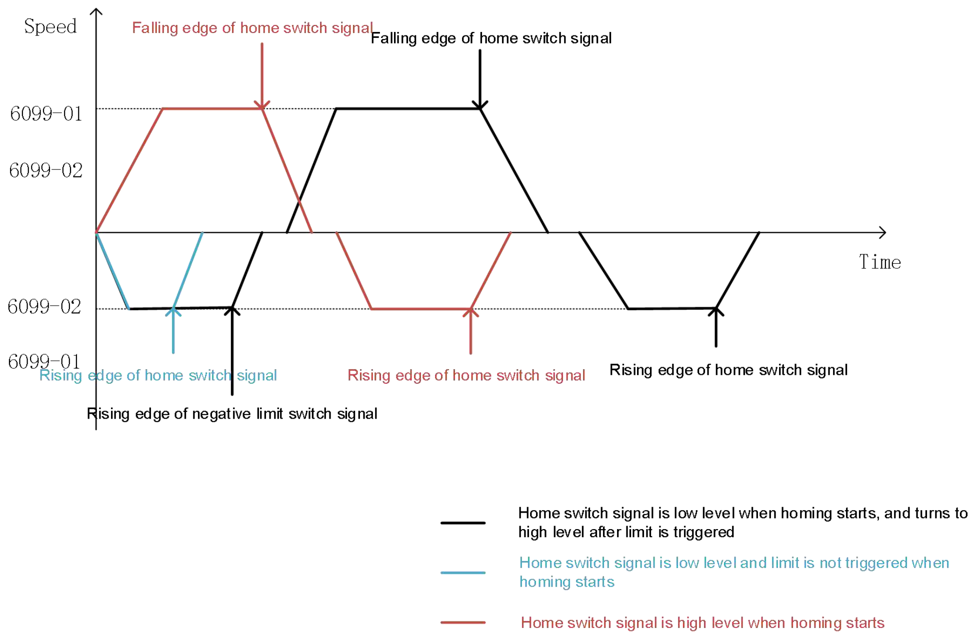


Figure 4-86 Speed-time curve of method 28

4.5.3.30 Method 29: Homing on positive home switch(rising edge)-negative limit switch detection

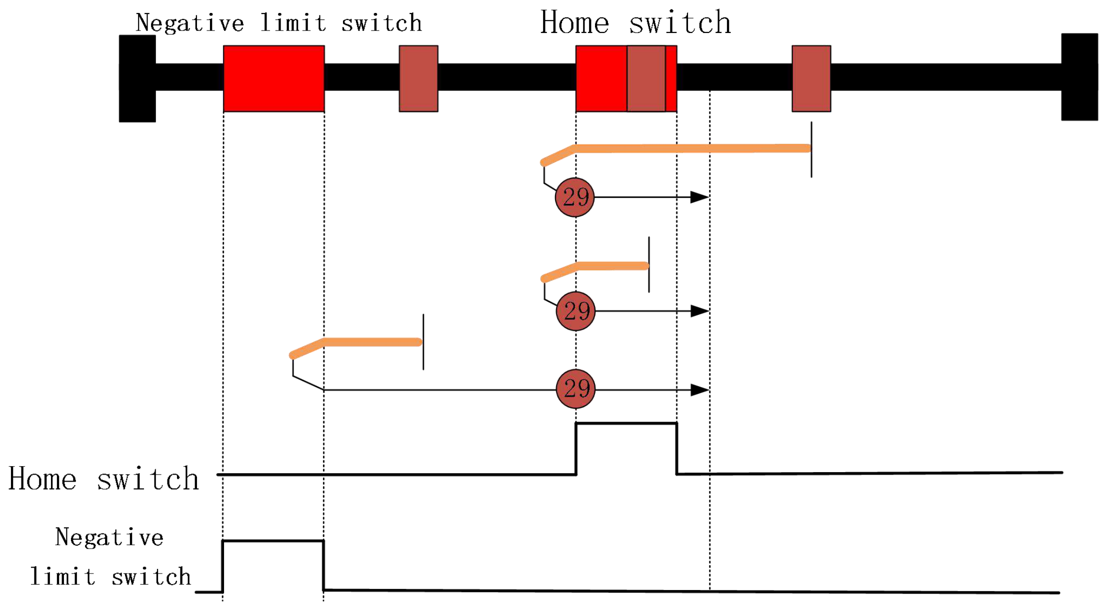


Figure 4-87 Method 29

Regardless of the high level or low level of the home switch signal, the motor moves in the negative direction.
When homing starts, the motor moves in the negative direction at a high speed (6099-01). After the home switch signal becomes low level or the negative limit switch becomes high level, the motor decelerates to 0 with the homing deceleration (609A), and moves in the positive direction with the homing acceleration (609A) to accelerate to a low speed (6099-02), and keeps moving in the positive direction at the low speed. After the home switch signal becomes high level, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.

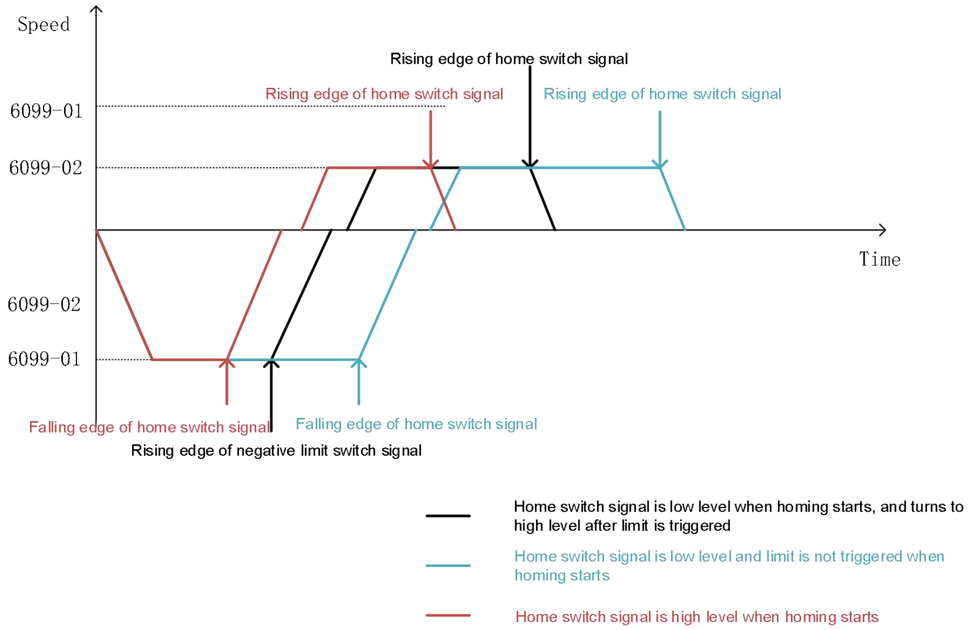


Figure 4-88 Speed-time curve of method 29

4.5.3.31 Method 30: Homing on negative home switch (falling edge)-negative limit switch detection

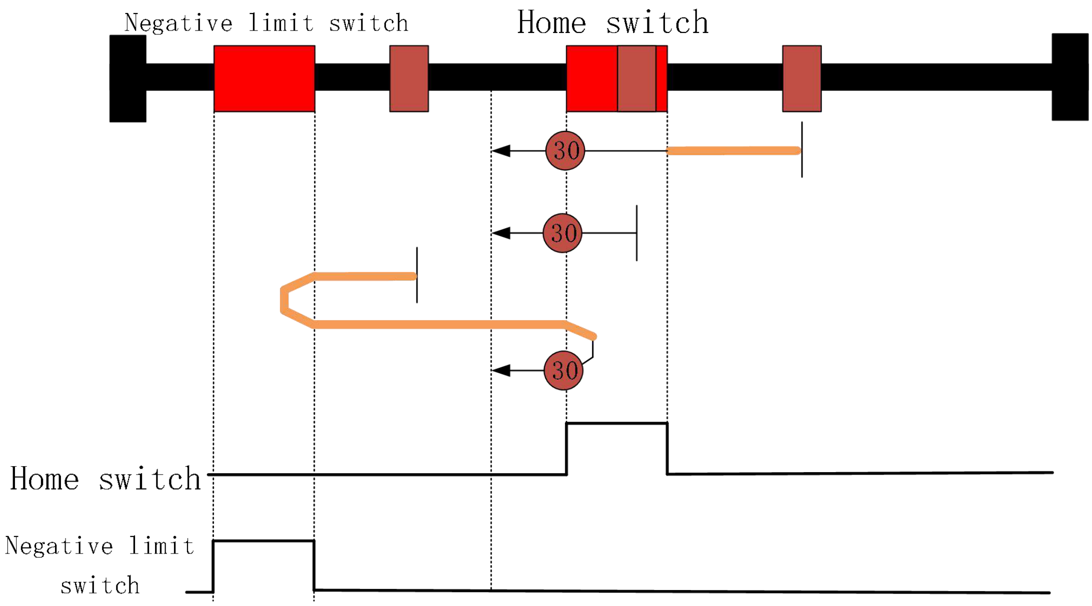


Figure 4-89 Method 30

- **When homing starts, if the home switch signal is low level,** the motor moves at a high speed (6099-01) in the negative direction.
  - ✓ After the home switch signal becomes high level, the motor moves with the homing deceleration (609A) to decelerate to a low speed (6099-02), and keeps moving in the negative direction at the high speed. After the home switch signal becomes low level, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.
  - ✓ After the negative limit switch signal becomes high level, the motor decelerates to 0 with the homing deceleration (609A), and moves in the positive direction with the homing acceleration (609A) to accelerate to a high speed (6099-01), and keeps moving in the positive direction at the high speed. After the home switch signal becomes high level, the motor decelerates to 0 with the homing deceleration (609A), and moves in the negative direction with the homing acceleration (609A) to accelerate to a low speed (6099-02), and keeps moving in the negative direction at the low speed. After the home switch signal becomes low level, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.
- **When homing starts, if the home switch signal is high level,** the motor moves at a low speed (6099-02) in the negative direction. After the home switch signal becomes low level, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.

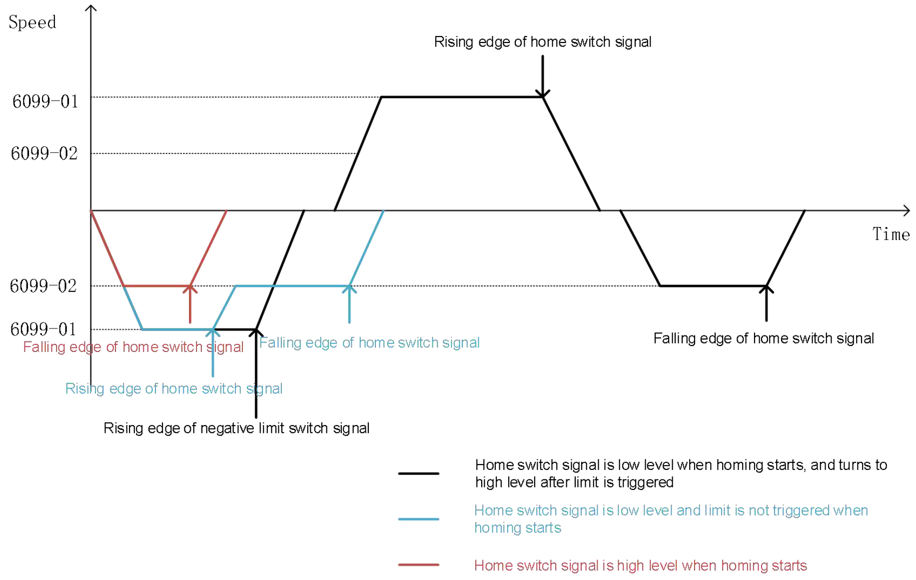


Figure 4-90 Speed-time curve of method 30

4.5.3.32 Method 31: Reserved

4.5.3.33 Method 32: Reserved

4.5.3.34 Method 33: Homing on index pulse in negative direction

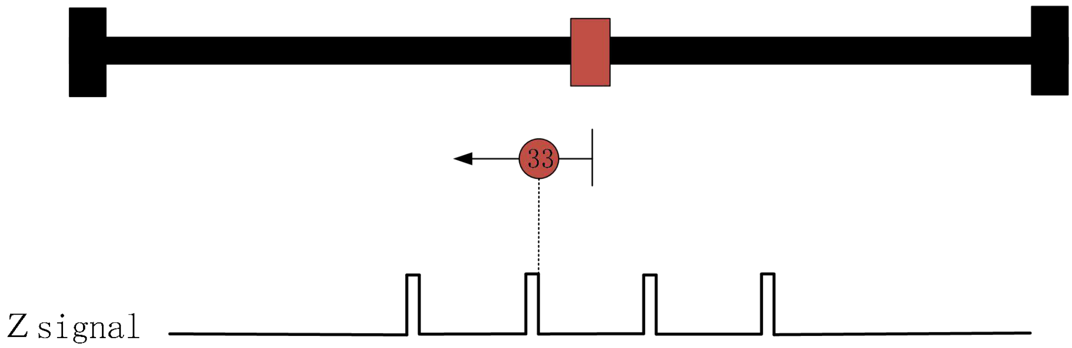


Figure 4-91 Method 33

When homing starts, the motor moves at a low speed (6099-02) in the negative direction. After the first Z signal shows, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.

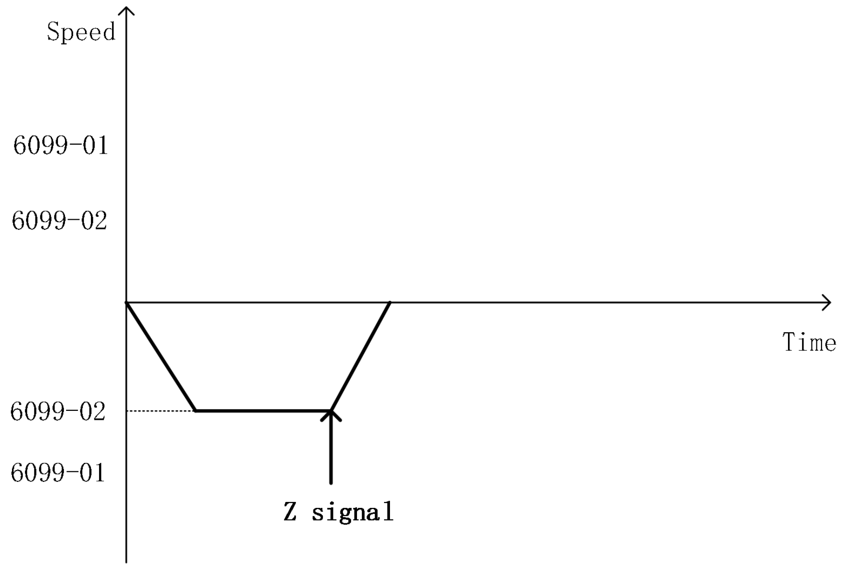


Figure 4-92 Speed-time curve of method 33

4.5.3.35 Method 34: Homing in index pulse in positive direction

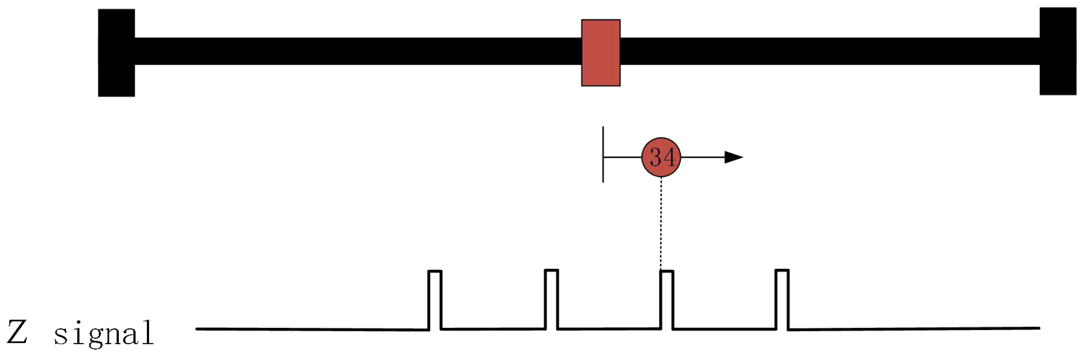


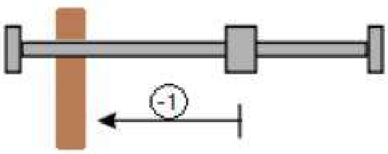
Figure 4-93 Method 34

When homing starts, the motor moves at a low speed (6099-02) in the positive direction. After the first Z signal shows, the status word Homing attained is set to 1, and the motor starts to decelerate with the homing deceleration (609A). The status word Target reached is set to 1 when the motor stops.

4.5.3.36 Method 35: Current position

In this method, the current position shall be taken to the home position.

4.5.3.37 Method -1: Guard position as home point in negative direction

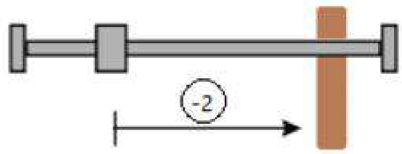


When homing starts, the motor moves in the negative direction, and the locked rotor torque reaches the set value of 0x2138 when the guard is touched. The motor stops after the time set by 0x2137:

✓ If the retraction distance is not set, the current position is set as the home point.

- ✓ If the retraction distance is set, the motor will retract the corresponding distance and set the current position as the home point.

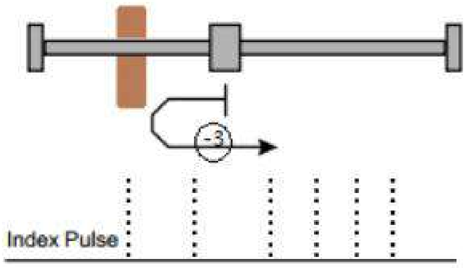
4.5.3.38 Method -2: Guard position as home point in positive direction



When homing starts, the motor moves in the positive direction, and the locked rotor torque reaches the set value of 0x2138 when the guard is touched. The motor stops after the time set by 0x2137:

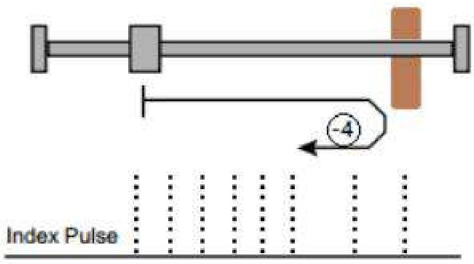
- ✓ If the retraction distance is not set, the current position is set as the home point.
- ✓ If the retraction distance is set, the motor will retract the corresponding distance and set the current position as the home point.

4.5.3.39 Method -3: C pulse as home point after guard is touched in negative direction



When homing starts, the motor moves in the negative direction, and the locked rotor torque reaches the set value of 0x2138 when the guard is touched. After the time set by 0x2137, the motor moves in the positive direction. The first C pulse is home point.

4.5.3.40 Method -4: C pulse as home point after guard is touched in positive direction



When homing starts, the motor moves in the positive direction, and the locked rotor torque reaches the set value of 0x2138 when the guard is touched. After the time set by 0x2137, the motor moves in the negative direction. The first C pulse is home point.

4.5.4 Torque Control Mode

Torque mode is generally used for servo to act as loading.

The process of motion control in torque mode is as follows:

1. Click **"Motion"** in the main menu, and click **"Torque Mode".** The interface of motion control in torque mode shows as in Figure 4-96.

Figure 4- 96 Interface of motion control in torque mode

2. Set the following parameters:
   - ✓ Target torque: the torque output by the motor. (unit: permillage of rated torque)
   - ✓ Torque Ramp: The acceleration when the motor starts to output torque. (unit: permillage of rated torque / second)
3. To enable the servo drive, click **"Enable"**. **Servo Enable** shows in the interface.
4. To control the motor to move with a positive given torque, click “**Forward**”, to control the motor to move with a negative given torque, click “**Reverse**”.

4.5.5 Pulse Control Mode

- **External wiring of the servo drive**

  Currently Diamond Plus series is supported only with CANopen communication mode. You can control it in one of the following pulse modes:

  1. A/B-phase quadrature pulse control: respectively connect pulse signal A/B to J2 (3, 4, 5, 6) of the incremental encoder.
  2. Direction + pulse control: connect direction signal to J2 (5, 6) of the incremental encoder and the position pulse signal to J2 (3, 4).

3	Encoder_2A+	PULSE+
4	Encoder_2A-	PULSE-
5	Encoder_2B+	SIGN+
6	Encoder_2B-	SIGN-

Note: The voltage of differential pulse input signal is ±5 V. Due to the best anti-noise ability of this signal transmission method, it is recommended to use this connection method first; if the upper unit is 24 V output, it needs to be converted to 5 V input with a conversion module.

- **Configuration of the upper computer**

  To configure the upper computer, do the following:

  1. Correctly set the motor and encoder parameters.
  2. Carry out debugging for the motor and PID parameters.
     Please refer to the relevant debugging manual for details.
  3. Open the upper computer, click "**Tools**" → "**Parameter Editor**" → "**AI, pulse control parameters**", and set the following parameters:

No.	Name	Description	Set Value
0x2023	PulseControlEnable	Pulse control enable	1
0x2024	InputResolution	Input resolution: ✓ Rotary motors: it corresponds to one rotation of the motor. ✓ Linear / voice coil motors: it corresponds to a magnetic pole pitch.	1000
0x2025	PositionControlLPFFreq	-	0
0x2026	PulseControlMode	Selection of pulse control mode: ✓ 1: A/B-phase quadrature pulse control. ✓ 2: Direction + pulse control ✓ 3: not support.	1

After the above parameters are set and the motor debugging is finished, it can directly receive the pulse input signal for position control.

4.5.6 Analog Control Mode

The servo drive supports receiving analog quantities to control the position, speed and current of the motor. The relevant parameters are as follows:

No.	Name	Description
0x201B	Analog control mode	✓ 0: not used ✓ 1: position control ✓ 2: velocity control ✓ 3: current control ✓ 4: position feedback ✓ 5: speed feedback ✓ 6: current feedback
0x 201C	Analog input offset	Set according to the initial 0 drift
0x201D	AI 1 input dead-time	Default: 0
0x201E	AI1 input low-pass filter cutoff frequency	Default: 3000
0x201F	AI1 control gain	Unit: position-cnt/V, speed-rpm/V, current-mA/V. (Please set it according to specific control range.)
0x2413	AI1 analog input value	Unit: mv (Currently this group is used by default)
0x2414	AI2 analog input value	Unit: mv (Reserved)

Please refer to the relevant documents of analog control for details.

4.6 Troubleshooting

If an error occurs during debugging, please troubleshoot the error by following the error description, possible causes and troubleshooting methods displayed by the upper computer software, as shown in Figure 4-98.

Figure 4- 96 Fault Display

After the error is successfully troubleshooted, click "**Clear Alarm**" in the toolbar.

Debugging can be continued after the system shows no error.

Note: If you have any questions during debugging, please seek technical support. Please do not arbitrarily modify the parameters, so as to avoid damages to personnel and property.

5 Troubleshooting

When an error occurs, the LED on the Diamond Plus servo panel will flash red in rhythm. After connecting the upper computer software, the error code based on the CiA402 standard will be displayed in the error handling interface.

When the servo alarms, please refer to the following table to check the servo, and solve the servo error according to the corresponding solution.

Table 5-1 Fault description

Error Code	Name	Cause	Solution
0x2230	Bus overcurrent	1. DC bus with excessive voltage. 2. Short circuit at periphery. 3. Encoder failure. 4. Internal components of the servo are damaged.	1. Check power supply and whether high inertia loads leads to rapid stop without dynamic braking. 2. Check whether the servo and the output wiring are short circuit, whether earthing is short circuit, and whether the braking resistor is short circuit. 3. Check whether the encoder is damaged or the wiring is correct; check whether the shielding layer of the encoder cable is well grounded, and whether there is strong interference near the cable.
0x2310	U-phase overcurrent	1. U-phase output is short circuit. 2. High load. 3. Cable insulation is damaged. 4. Poor motor insulation. 5. Failure of U-phase current detecting circuit.	1. Check U-phase wiring. 2. Lower the load. 3. Check U-phase cable and replace it if necessary. 4. Measure the motor insulation, repair and replace it if necessary; 5. Repair or replace the drive.
0x2311	V-phase overcurrent	1. V-phase output is short circuit; 2. High load. 3. Cable insulation is damaged. 4. Poor motor insulation. 5. Failure of V-phase current detecting circuit.	1. Check V-phase wiring. 2. Lower the load. 3. Check V-phase cable and replace it if necessary. 4. Measure the motor insulation, repair and replace it if necessary. 5. Repair or replace the drive.

Error Code	Name	Cause	Solution
0x2320	Hardware short circuit	1. DC bus with excessive voltage. 2. Short circuit at periphery. 3. Encoder failure. 4. Internal components of the servo are damaged.	1. Check power supply and whether high inertia loads leads to rapid stop without dynamic braking. 2. Check whether the servo and the output wiring are short circuit, whether earthing is short circuit, and whether the braking resistor is short circuit. 3. Check whether the encoder is damaged or the wiring is correct; check whether the shielding layer of the encoder cable is well grounded, and whether there is strong interference near the cable.
0x3220	Servo undervoltage	1. Low input voltage of the power circuit. 2. Poor insulation of DC bus. 3. High load. 4. Poor insulation of the driver cable. 5. Failure of DC bus undervoltage detecting circuit. 6. Basic power module failure.	1. Check the power circuit. 2. Check the DC bus insulation. 3. Lower the load. 4. Check the drive cable. 5. Repair or replace the drive. 6. Repair or replace the basic power module.
0x3210	Servo overvoltage	1. Insufficient capacity of brake circuit. 2. Insufficient capacity of braking resistor. 3. Basic power module failure	1. Reduce the start-stop frequency; increase the acceleration/deceleration time constant; lower the load inertia; increase the drive and motor capacity. 2. Increase the power of the braking resistor. 3. Repair or replace the basic power module;
0x4110	Ambient temperature overheating	1. High ambient temperature. 2. Abnormal cooling system. 3. Temperature detecting circuit failure.	1. Lower the ambient temperature and strengthen ventilation and heat dissipation. 2. Check the cooling fan speed and air volume. If they are abnormal, replace the fan with the same model. 3. Check whether the servo cooling channel is blocked by foreign objects.

Error Code	Name	Cause	Solution
0x4120	Ambient temperature underheating	1. Low ambient temperature. 2. Temperature detecting circuit failure.	1. Check whether the ambient temperature is too low; 2. Check the value of parameter minimum ambient temperature.
0x4310	Power module overheating	1. High ambient temperature. 2. Abnormal cooling system. 3. Temperature detecting circuit failure.	1. Lower the ambient temperature and strengthen ventilation and heat dissipation. 2. Check the cooling fan speed and air volume. If they are abnormal, replace the fan with the same model. 3. Check whether the servo cooling channel is blocked by foreign objects.
0x8482	Exceed maximum speed	1. Motor run away. 2. Wrong encoder parameters. 3. Encoder failure 4. Instruction error 5. Load mutation	1. Check the phase sequence of the motor power cable. 2. Check the settings of encoder parameter. 3. Check whether the encoder is damaged or the wiring is correct; check whether the shielding layer of the encoder cable is well grounded, and whether there is strong interference near the cable. 4. Check the position / speed / torque command. 5. Check whether the load is mutated and related cause. 6. Correct the phase zero again. 7. Adjust PID parameters.
0x8483	Large speed tracking error	1. The encoder wiring is wrong or the connector is in poor contact. 2. The gain does not match. 3. Large external load fluctuations or interference.	1. Check the encoder wiring; 2. Adjust the servo gain again. 3. Increase anti-interference measures.

Error Code	Name	Cause	Solution
0x8611	Large position deviation	1. The encoder wiring is wrong or the connector is in poor contact. 2. The gain does not match. 3. Large external load fluctuations or interference.	1. Check the encoder wiring; 2. Adjust the servo gain again. 3. Increase anti-interference measures.
0x7380	Encoder connection error	1. Wrong encoder parameters. 2. Encoder cable failure. 3. The encoder cable is not connected. 4. The internal components of the servo are damaged.	1. Check the settings of encoder parameters. 2. Check the line sequence of encoder cable. 3. Connect the encoder cable.
0x7383	Encoder multi-turn info error	Internal encoder error.	Power off and restart the servo. If the fault cannot be cleared, replace the encoder.
0x7385	Encoder count error	Internal encoder error.	Power off and restart the servo. If the fault cannot be cleared, replace the encoder.
0x7389	Encoder count overflow error	Internal encoder error.	Clear the encoder multi-turn value, power off and restart the servo. If the fault cannot be cleared, replace the encoder.
0x738A	Encoder communication CRC error	1. Wrong encoder parameters. 2. Encoder cable failure.	1. Check the settings of encoder parameters. 2. Check whether the encoder is damaged or the wiring is correct; check whether the shielding layer of the encoder cable is well grounded, and whether there is strong interference near the cable.
0x738B	Encoder delimiter error	Internal encoder error.	Power off and restart the servo. If the fault cannot be cleared, replace the encoder.

Error Code	Name	Cause	Solution
0x3221	PWM drive abnormal	PWM drive +15 V undervoltage.	Check whether the control power +24V is connected properly.
0x8612	Exceed position limit	Given position or actual position exceeds position limit.	1.Check the setting of limit position. 2.Check the settings of given position. 3.Check whether the limit switch is triggered.
0x7384	Encoder overheating	The working temperature of the encoder exceeds 95℃.	1. Test again after the motor has cooled down. 2. Improve the heat dissipation conditions and check whether the motor overheats during running. 3. Internal encoder error.
0x6280	Wrong profile value	There is a zero value in the set value of the profile track, which makes the planned track unsuccessful.	1. Make sure the set speed is not zero. 2. Make sure the set acceleration is not zero.
0x6281	Termination speed setting error	The termination speed is greater than the profile speed, which makes the planned track unsuccessful.	1. The set termination speed must be less than or equal to the profile speed.
0x6282	Termination speed setting error	The target position is too close to the current position to reach the termination speed.	1. Check whether the set termination speed is too large.
0x6283	Software limit setting error	When the minimum / maximum software limit is not set to 0, the minimum value is greater than or equal to the maximum value; or exceed the position limit.	1. Set the minimum value greater than the maximum when the minimum / maximum software limit is not set to 0. 2. Check whether the maximum value is too large. 3. Check whether the minimum value is too small.

Error Code	Name	Cause	Solution
0x6284	Wrong position limit	When the minimum / maximum position limit is not set to 0, the minimum value is greater than or equal to the maximum value.	1. Set the minimum value greater than the maximum when the minimum / maximum position limit is not set to 0.
0x6285	Wrong planned curve type	The set planned curve type is not supported.	1. Set the planned curve type to 0 (Linear ramp)
0x6286	Wrong planned curve type	The set planned curve type is not supported.	1. Set the planned curve type to 0 (Linear ramp) or 3 (Jerk-limited ramp).
0x6287	Wrong planned torque curve	The set planned torque type is not supported.	1. Set the planned torque curve type to 0 (Linear ramp).
0x6288	Wrong homing method	The limit switch was accidentally triggered.	1. Start homing again after setting a suitable homing method.
0x6289	Wrong homing method	The set homing method is not supported.	1. Start homing again after setting a suitable homing method.
0x628B	Homing process timed out	The zero point was not found during homing.	1.Check the lower limit switch or the origin switch. 2.Set a suitable homing method.
0x628C	Initial speed not zero when planning Jerk-limited ramp	When the planned curve type is Jerk-limited ramp, the initial speed is not zero.	1. Make sure the motor is still before enabling the curve planning of the Jerk-limited ramp.
0x6180	Execution time of planned curve less than 0	The settings of position, speed, or acceleration / deceleration are incorrect.	Reset position, speed, acceleration and deceleration.

Error Code	Name	Cause	Solution
0x6181	Stop speed greater than initial speed	Stop speed is not set to 0	Set stop speed to 0.
0x6182	Position, speed, acceleration and deceleration not set for continuous motion of multiple points	The position, speed, acceleration and deceleration are not set for continuous motion of multiple points	Reset the target position, speed, acceleration and deceleration.
0x6184	Internal state transition error in homing	Jump exception of the internal homing state.	Execute homing again.
0x7124	Motor overtheating	The motor temperature is detected by the external temperature sensor and then connected to the servo through the DI port, and its upper limit is determined by the external temperature sensor.	1. High load. 2. Lack of phase. 3. Fault related to motor machinery, including lack of lubricating grease, improper assembly of bearings and end caps, eccentricity of inner holes, etc.

Error Code	Name	Cause	Solution
0x3130	Lack of phase	UVW phases have open circuit.	Check the wiring of UVW phases.
0x8700	Sync error	Bus synchronization error.	Restart the servo.
0x738C	Hall error	Hall signal is disconnected.	Check the wring of Hall.
0x6551	Wrong target speed	The target speed is 0 in position control.	Check the value of 0x6081 and make sure it is not 0.
0x6552	Wrong acceleration and deceleration in position and velocity control	The track planning is unsuccessful when acceleration and deceleration is set to 0.	Make sure acceleration or deceleration is not 0.
0x6553	Wrong position track planning period	The position track planning is set to 0.	Make sure the set period is not 0.
0x7320	Z pulse repetition positioning position error	The difference of adjacent Z pulses exceeds 0x2001.	1. Check the scale installation or accuracy. 2. Check the Z pulse positioning deviation.
0x8620	Failed to enable auto calibration	Failed to enable automatic calibration.	1. Check whether the motion control mode is 0. 2. Check whether the device is stuck, the frictional resistance increases or the load is abnormal, etc. 3. Check whether there is an open circuit or short circuit in the three-phase wiring. 4. Check whether the settings of 0x2105 and 0x2402 are proper. 5 Check whether the phase sequence of the UVW wiring and the setting of 0x2002 are correct. 6. Check the encoder wiring.

Error Code	Name	Cause	Solution
0x6542	Planned deceleration or quick stop deceleration in the position mode is 0	The planned deceleration or quick stop deceleration in the position mode is 0.	Check the deceleration or quick stop deceleration and make sure it is not 0.
0x6572	Planned deceleration or quick stop deceleration in the position mode is 0	The planned deceleration or quick stop deceleration in the position mode is 0.	Check the deceleration or quick stop deceleration and make sure it is not 0.
0x9100	DI external input alarm	DI external input condition triggers an alarm.	Check the external input conditions.
0x8900	I2T protection alarm	Exceed the I2T setting threshold	1. Adjust limiter protection peak current. 2. Adjust limiter protection peak current duration. Note: The alarm takes effect when 0x2017 bit1 is set to 1.
0x8901	Alarm of no calibration	Operation is enabled without performing angle identification.	Enable operation after If Hall is connected and angle identification is finished.
0xB010	Position feedback jitter during angle identification	Wrong encoder wiring. Abnormal load or external disturbance.	Check the encoder wiring. Check the load or external disturbance.
0xB020	Rotor not moving during angle identification	Parameter settings such as current are incorrect. High load. The machine is stuck, or the wiring is wrong.	Set appropriate parameter values. Check device, load and wiring.

Error Code	Name	Cause	Solution
0xB030	Large action of angle identification	Large setting of current. Fault, including device, load, wiring (phase sequence), etc.	Set appropriate parameter values. Check device, load and wiring (phase sequence).
0xB040	Angle identification timed out	Software exception	Check the upper computer software, M3 and C28. Check each parameter setting. Check device, load and wiring.
0xB102	Motor hardly rotates during phase sequence detection	Wrong encoder wiring. High load or friction. Problem with current loop configuration The commutation current ratio 1 is too small.	Check the encoder wiring. Increase commutation current ratio 1.
0xB104	Hall status feedback abnormal	Wrong wiring of Hall sensor. Wrong Hall mode.	Check the wring of Hall Sensor. Make sure 0x2103 is set to 0.

6 Debugging Software ISMC

Stone Motion Control (ISMC) is a servo debugging software independently developed by our company. Through USB serial communication, you can configure and modify servo parameters, debug controller parameters, realize motion control, monitor system status in real time, diagnose faults, check error logs, and realize update and maintenance.
This chapter only focuses on software download and setup. For software operations, please refer to "***Servo Debugging Software ISMC User Manual***".

6.1 Software Download

6.1.1 System Requirements

System environment requirements:

- Memory: 1 GB or more (1.5 GB or more for running on a virtual machine)
- Display: above 800x600
- System type: 32-bit or 64-bit Windows 7 / Windows 8 / Windows 10
- Processor: above 1.6 GHZ

6.1.2 Software Installation

The setup process of SMC is as follows:

1. Download the setup package from the official website.
2. Double click the .exe application file, and wait for the decompression. After decompression, the setup wizard pops up, as shown in Figure 6-1.

Figure 6-1 SMC installation wizard

3. Click "**Next**". The agreement dialog box pops up, as shown in Figure 6-2.

Figure 6- 2 Installation agreement

4. Select “**I Agree**”, click "**Next**" and enter the user information, as shown in Figure 6-3.

Figure 6-3 User information

5. Click "**Next**", and select the setup type, as shown in Figure 6-4.
   Generally, please use the default type.

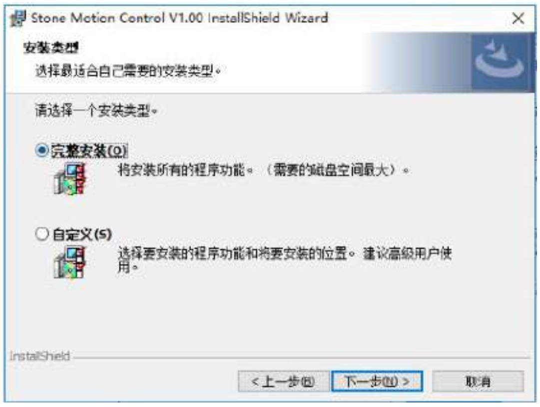


Figure 6-4 Setup type

6. Click “**Next**”, and click "**Finish**" after the setup is finished to exit the setup interface and complete the setup, as shown in Figure 6-5.

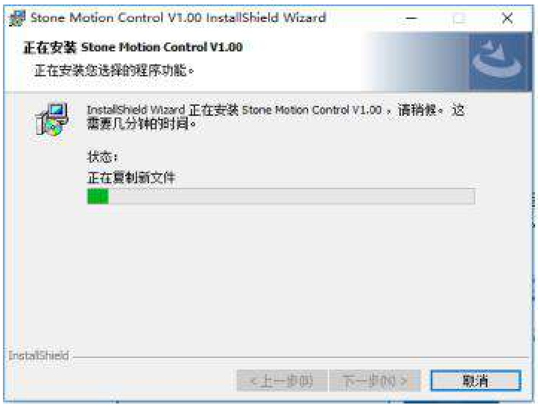


Figure 6-5 Installing

7. After setup, check the shortcut of SMC software by accessing "**Desktop**" → "**Start**" →"**All Programs**" on your computer, as shown in Figure 6-6.

Figure 6-6 Start menu

6.2 Driver Installation

When using USB communication for the first time, you need to install the USB driver.

Note: The driver will be automatically installed on Windows 10 after connecting the USB data cable.

Taking Windows 7 as an example, the process of driver installation is as follows:

1. Connect the upper computer and the servo drive via the USB data cable. A prompt that the driver cannot be installed automatically, as shown in Figure 6-7.

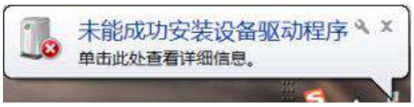


Figure 6-7 Failed to install the driver

2. Open the Windows main menu, and right click "**Computer**".

Figure 6-8 Windows main menu

3. To open the computer management, click "**Manage**":

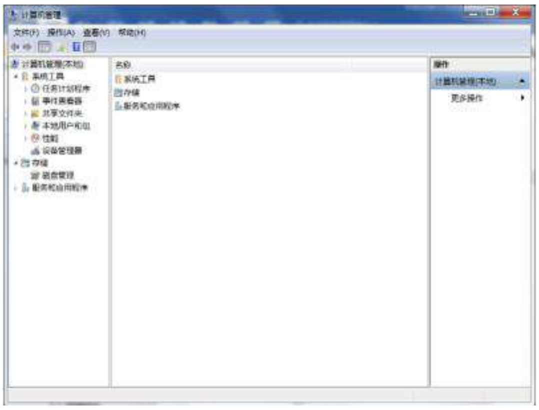


Figure 6-9 Computer management window

4. Select "**Device Manager**" → "**Others**", and find the unrecognized device **Virtual COM Port**.

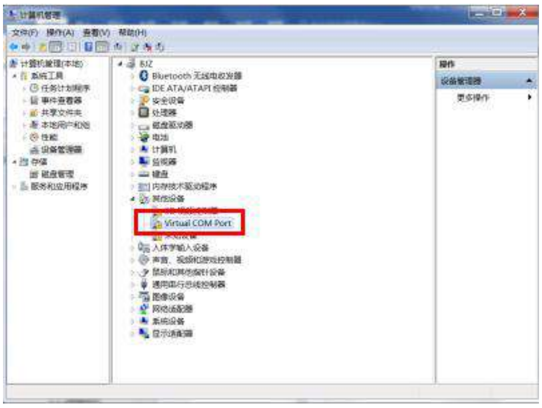


Figure 6-10 Device manager

5. Right click "**Virtual COM Port**", and select **"Update Driver Software**".

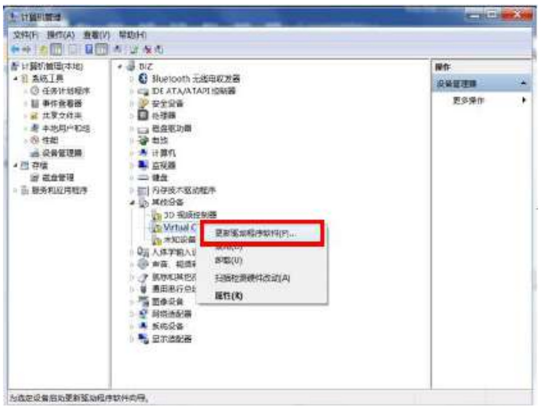


Figure 6-11 Update driver software

6. Select "**Browse my computer for driver software**".

Figure 6-12 Find driver software

7. Click **“Browse",** find and select the driver folder “**windows_drivers”** in the SMC installation directory.
   Default path: C:\Program Files(x86)\SMC\Files\windows_drivers.

Figure 6-13 Browse the driver installation path

8. Click **"Next"** to start installation, and select "**Always install this driver software**" in the pop-up security warning window.

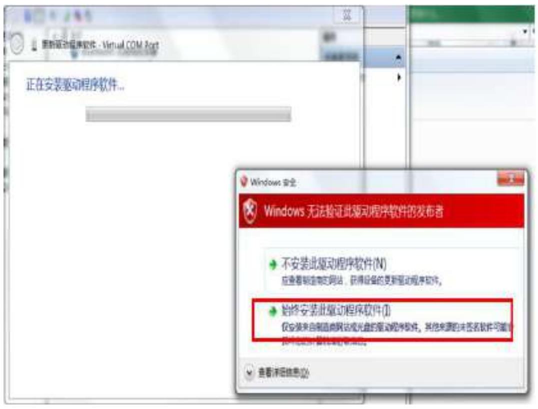


Figure 6-14 Security warning pop-up

9. Finish driver installation.

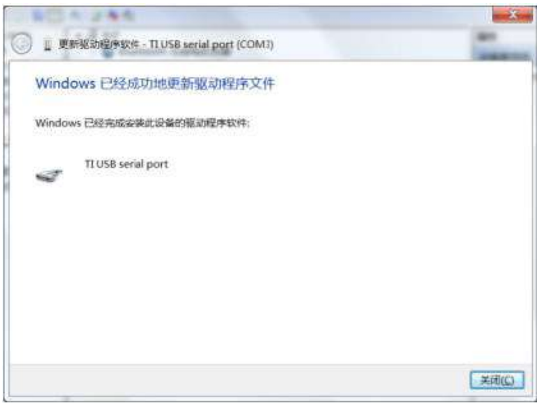


Figure 6-15 Finish driver installation

Note: If the driver fails to be installed successfully, please contact the technician.

6.3 Firmware Upgrade

You can burn and upgrade the M3 and C28 files in the servo through SMC.

The process of firmware upgrade is as follows:

1. Select **"Help"** in the main menu, and click **"Firmware Upgrade"** to open the firmware upgrade interface, as shown in Figure 6-16.

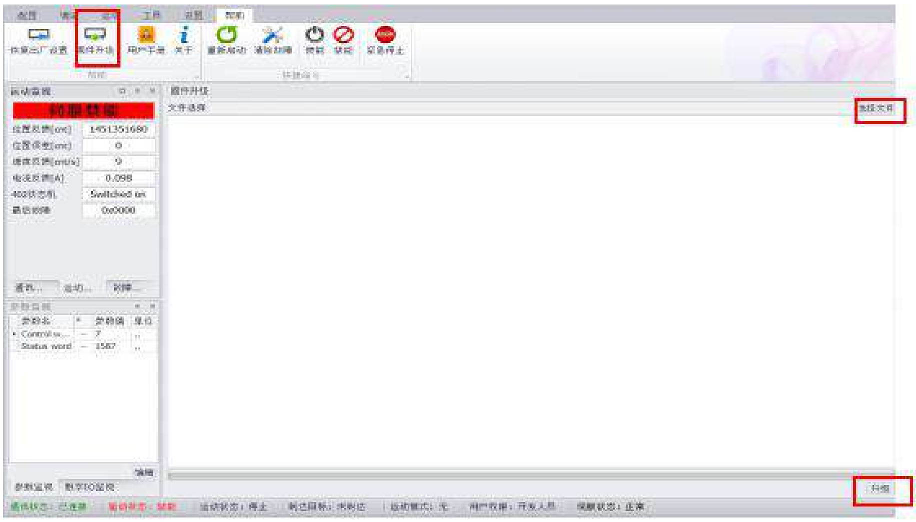


Figure 6-16 Firmware upgrade

2. Click **"Select File"** to open the folder and select the M3 or C28 file to be upgraded.
3. Click **"Upgrade"** to start upgrading. After the upgrade is successful, SMC software and the servo are restarted.
4. After reconnecting, repeat the above steps to upgrade the next program until all programs are upgraded.

Note:

1. For firmware upgrade, please contact our technical team.
2. Servo DC+/DC- power supply is required when upgrading, and 24 V power supply is recommended.
3. The names of the upgraded files are fixed, i.e. C28-APP.bin and M3-APP.bin.
4. For the first time, please flash M3-APP.bin first, and then flash C28-APP.bin.

7 Communication

The Stone servo drive supports both CANopen and EtherCAT. This chapter introduces the principles, usage and cases of these two communication modes. For details, please refer to "**User Manual of Stone Servo Drive (CANopen)**" and "**User Manual of Stone Servo Drive (EtherCAT)**".

7.1 CANopen Communication

7.1.1 CANopen Protocol

CANopen is a high-level communication protocol and device profile specification that is based on the CAN (Controller Area Network) protocol, following the ISO/OSI standard model. Different devices in the communication network exchange data with each other through the object dictionary, wherein the master node can access or modify the data in the object dictionary list of other nodes through process data object (PDO) or service data object (SDO).

7.1.1.1 Object Dictionary

The object dictionary is an ordered set of parameters and variables, including all parameters of the device description and device network status. A set of objects can be accessed through a network in an ordered and predefined manner.
The CANopen protocol uses an object dictionary, identified with a 16-bit index and an 8-bit subindex. The structure of the object dictionary is shown in Table 7-1.

Table 7-1 Object dictionary structure

Index	Object
0x0001-0x0FFF	Data type definition
0x1000-0x1029	Communication parameter object (such as CiA-301 protocol parameter)
0x1200-0x12FF	SDO object
0x1400-0x15FF	RPDO object
0x1600-0x17FF	RPDO mapping
0x1800-0x19FF	TPDO object
0x1A00-0x1BFF	TPDO mapping
0x1C00-0x1FFF	Other communication parameters
0x2000-0x5FFF	Manufacturer specific sub-protocol object
0x6000-0x9FFF	Standard device sub-protocol object (such as DSP-402 protocol parameter)
0xA000-0xFFFF	Reserved

7.1.1.2 CANopen Communication Object

7.1.1.2.1 Network Management (NMT)

The network management system (NMT) is responsible for initializing, starting and stopping the network and the devices in the network, and belongs to the master-slave system. There is only one NMT master in the CANopen network, which can configure the CANopen network including itself. Network management objects include Boot-up messages, Heartbeat protocols and NMT messages. Based on the master-slave communication mode, NMT is used to manage and monitor each node in the network, mainly including node status control, error control and node startup.

1. NMT service

CANopen performs conversion according to the state machine specified by the protocol. Among them, some are internal automatic conversions, and some must be converted by NMT messages sent by the NMT master. Conversion by the state machine is as shown in Figure 2-2.

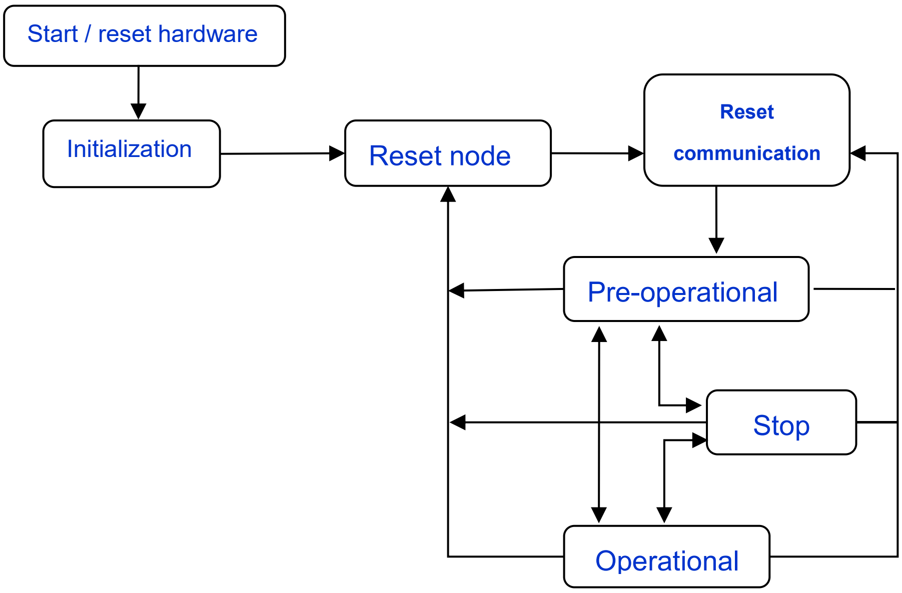


Figure 7-1 Working of NMT state machine

Part of the conversion in Figure 7-1 is achieved by NMT messages, and only the NMT master can send NMT messages. The message format is shown in Table 7-2.

Table 7-2 NMT module ControlNMT message format

COB-ID	RTR	Data/byte	
		0	1
0x000	0	Command word	Node_ID

COB-ID is fixed to "**0X000**".

The data consists of two bytes:

- Command word: indicating the control function of the frame, as shown in Table 7-3.
- CANopen node address: when Data1 is 0, this message is broadcast to all slaves in the network.

Table 7-3 Command word of NMT Module ControlNMT message

Command word	Jump state	Description
0X01	Pre-operational -> Operational	Start remote node
0X02	Operational/Pre-operational -> Stop	Stop remote node
0X80	Operational -> Pre-operational	Enter pre-operation state
0X81	Operational -> Reset node	Reset node
0X82	Operational -> Reset communication	Reset communication

After the device is powered on, it will automatically enter the initialization state, including the following:

- Initializing: During Initializing, the device starts up and initializes its internal parameters.
- Reset node: During Reset Application all parameters in the object dictionary range from 2000_h to $9FFF_h$ are set to the power-on or default values.
- Reset communication: the parameters of the communication profile (Index range $1xxx_h$) are set to their power-on/default values.

After the device initialization is finished, the device automatically transits to pre-operational state and indicates this transition by transferring the boot-up message. After the configuration, the node needs the NMT master to send NMT messages to enter the operational state in which each module should work normally.

When the NMT master sends a stop node message, the device enters the stopped state, and only the NMT module works normally in CANopen communication.

The CANopen services supported in various NMT states are shown in Table 7-4.

Table 7-4 Services supported by various NMT states

Serve	Pre-operational	Operational	stop
Process Data Object (PDO)	No	Yes	No
Service Data Object (SDO)	Yes	Yes	No
Synchronization Object (SYNC)	Yes	Yes	No
Emergency message (EMCY)	Yes	Yes	No
Network Management (NMT)	Yes	Yes	Yes
Error control	Yes	Yes	Yes

- **NMT error control protocols**

  Error control protocols enable the monitoring of a CANopen network. They comprise the Heartbeat-, Node-/Life-Guarding-, as well as the Boot-up protocol.

  a. Node-/Life guarding

  Node guarding is used to monitor the non-central peripheral modules, while they themselves can use Life Guarding to detect the failure of the guarding master. Guarding involves the master transferring remote frames (remote transmit requests) to the guarding identifier of the slaves that are to be monitored. These reply with the guarding message. This contains the slave's status code and a toggle bit that has to change after every message. If either the status or the toggle bit do not agree with that expected by the NMT master, or if there is no answer at all, the master assumes that there is a slave fault.

  The objects related to node-/life guarding are guard time 100Ch and live time factor 100Dh. 100Ch is the remote frame interval of node guarding (unit: ms). Object 100Ch multiplied by object 100Dh determines the latest time for master query. Generally, node guarding can be achieved. Lifetime guarding is activated when 100Ch and 100D are both non-zero and a node guarding request frame is received.

  The master station sends a node guarding remote frame every 100Ch. The slave station must respond, otherwise it assumes that the slave station has failed; If the slave fails to receive a message request from the master within the time of 100Ch X 100Dh, it assumes that the master station has failed.

The format of remote frames is as shown in Table 7-5 below.

Table 7 - 5 Remote frame format

COB-ID	RTR
0x700+Node_ID	1

The response message returned by NMT node is a one-byte status word, as shown in Table 7-6 below.

Table 7-6 Response message

COB-ID	RTR	Data
0x700+Node_ID	0	status word

Bits of the status word are shown in Table 7-7 below.

Table 7-7 Bits of status word

Data bits	Description
bit7	It must be alternately set to "0" or "1" each time
Bit6 - bit0	4-Stopped state 5-Operational state 127 - Pre-operational state

b. Heartbeat protocol

With heatbeat messages, nodes can be configured as heartbeat producers or heartbeat consumers. The CANopen device can send heartbeat messages according to 1017h Producer heartbeat time (unit: ms). The node with the consumer heartbeat function in the network monitors the producer according to 1016h Consumer time. If a consumer does not see an expected heartbeat within its expected time, it signals an error.

After configuring 1017h Producer heartbeat time, the node heartbeat function is activated and starts to send heartbeat messages. After configuring 1016h Consumer heartbeat, the monitoring will start after receiving a frame of heartbeat sent by the corresponding node.

The master sends a heartbeat message according to the producer heartbeat time, and if the slave monitoring the master does not receive a heartbeat message within the consumer time, it assumes that the master has failed. The slave sends a heartbeat message every 1017h. If the master monitoring the salve does not receive a heartbeat message within the consumer time, it assumes that the slave has failed.

The format of the heartbeat message is shown in Table 7-8. The data segment has only one byte, the highest bit is fixed at 0, and the rest are the same as the node guarding response message.

Table 7-8 Heartbeat message format

COB-ID	RTR	Data
0x700+Node_ID	0	status word

7.1.1.2.2 Service Data Object (SDO)

It includes receiving SDO-SDO (Rx) and transferring SDO-SDO (Tx). Through indexes and sub-indexes, SDO enables access to all entries of a CANopen object dictionary. SDO is implemented through the CMS object of the multi-field in the CAL, allowing transmitting data of any length, and splitting it into several messages when the data is out of 4 bytes. This is a confirmed communication service that generates a reply for each message. SDO request and response messages always contain 8 bytes.

1. SDO transfer framework

   With an SDO, a peer-to-peer client-server communication between two CANopen devices can be established on the broadcast medium CAN. The owner of the accessed object dictionary acts as a SDO server. The device that accesses the object dictionary of the other device is the SDO client. Therefore, the data exchange between SDOs requires at least two CAN messages.

2. SDO transfer message

   SDO transfer is divided into transfer not greater than 4 bytes (expedited SDO transfer) and transfer greater than 4 bytes (segmented SDO transfer).

   SDO transfer message consists of COB-ID and data segment, as shown in Table 7-9 below.

Table 7-9 Format of SDO transfer message

<table>
<tr><th>COB-ID</th><th colspan="8">Data</th></tr>
<tr><td rowspan="2">580h+Node_ID/
600h+Node_ID</td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td></tr>
<tr><td>command code</td><td colspan="2">index</td><td>sub-index</td><td colspan="4">data</td></tr>
</table>

Among them, command code indicates the transfer type and transfer data length, index and sub-index indicate object position in the list, and data is the value of the object.

Expedited SDO transfer (write)

If there is 4 bytes or less than, the transfer can be expedited and all the data sent within the command or response message. Due to different read and write methods and data length, the transfer messages are different, as shown in Table 7-10.

Table 7-10 Format of expedited SDO transfer (write)

<table>
<tr><th colspan="2">Transfer</th><th>COB-ID</th><th>0</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th></tr>
<tr><td colspan="2" rowspan="4">Client</td><td rowspan="4">600h+Node_ID</td><td>23h</td><td colspan="2" rowspan="4">index</td><td rowspan="4">sub-index</td><td colspan="4">data</td></tr>
<tr><td>27h</td><td colspan="3">data</td><td>-</td></tr>
<tr><td>2bh</td><td colspan="2">data</td><td>-</td><td>-</td></tr>
<tr><td>2fh</td><td>data</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td rowspan="2">Server</td><td>Normal</td><td rowspan="2">580h+Node_ID</td><td>60h</td><td colspan="2" rowspan="2">index</td><td rowspan="2">Sub-index</td><td>-</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td>Abnormal</td><td>80h</td><td colspan="4">abort code</td></tr>
</table>

"-" indicates that there is data but it is not considered, it is recommended to write 0 when writing data, the same below.

Example:

The station number of the slave is 2, the write value of 60FFh-00 in expedited SDO transfer is 1000, that is 0x3E8, and the message sent by the master is as follows. (all in hexadecimal)

Table 7-11 Examples of message sent by the master

COB-ID	0	1	2	3	4	5	6	7
602	23	FF	60	00	E8	03	00	00

Table 7-12 Example of the message returned by the slave

COB-ID	0	1	2	3	4	5	6	7
582	60	FF	60	00	00	00	00	00

If the write data type does not match, the error code 0x06070010 will appear, and the message is as follows:

Table 7-13 Example of message when write data type mismatches

COB-ID	0	1	2	3	4	5	6	7
582	80	FF	60	00	10	00	07	06

Expedited SDO transfer (read)

If there is 4 bytes or less than, the transfer can be expedited and all the data sent within the command or response message. The expedited SDO transfer (read) is as shown in Table 7-10.

Table 7-14 SDO Format of expedited SDO transfer (read)

<table>
<tr><th colspan="2">Transfer</th><th>COB-ID</th><th>0</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th></tr>
<tr><td colspan="2">Client</td><td>600h+Node_ID</td><td>40h</td><td colspan="2">index</td><td>sub-index</td><td>-</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td rowspan="5">Server</td><td rowspan="4">Normal</td><td rowspan="5">580h+Node_ID</td><td>43h</td><td colspan="2" rowspan="5">index</td><td rowspan="5">sub-index</td><td colspan="4">data</td></tr>
<tr><td>47h</td><td colspan="3">data</td><td>-</td></tr>
<tr><td>4bh</td><td colspan="2">data</td><td>-</td><td>-</td></tr>
<tr><td>4fh</td><td>data</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td>Abnormal</td><td>80h</td><td colspan="4">abort code</td></tr>
</table>

Example:

The station number of the slave is 2, the read value of 6061h-00 in expedited SDO transfer is 3, that is 0x03, and the message sent by the master is as follows. (all in hexadecimal)

Table 7-15 Examples of message sent by the master

COB-ID	0	1	2	3	4	5	6	7
602	40	61	60	00	00	00	00	00

Table 7-1 6 Examples of message returned by the slave

COB-ID	0	1	2	3	4	5	6	7
582	4f	61	60	00	03	00	00	00

If the write data type does not match, the error code 0x05040001 will appear, and the message is as follows:

Table 7-17 Example of message when read data type mismatches

COB-ID	0	1	2	3	4	5	6	7
582	80	61	60	00	01	00	04	05

Segmented SDO transfer (read)

If more than 4 bytes of data is required to be transferred, a segmented transfer is used. Its message and initial transfer frame is the same with the expedited transfer.

Table 7-18 Format of initial transfer message

Transfer		COB-ID	0	1	2	3	4	5	6	7
Client		600h+Node_ID	40h	index		sub-index	-	-	-	-
Server	Normal	580h+Node_ID	41h	index		sub-index	Data length			
	Abnormal		80h				abort code			

The trigger bit (bit6) alternately sends 0 and 1 during transfer process. The format of the process message is shown in Table 7-19.

Table 7-19 Format of process message

Transfer		COB-ID	0	1	2	3	4	5	6	7
Client		600h+Node_ID	60h			-	-	-	-	-
Server	normal	580h+Node_ID	00h	Data length						
	abnormal		80h	index		sub-index	abort code			
Client		600h+Node_ID	70h	-	-	-	-	-	-	-
Server	normal	580h+Node_ID	10h	Data length						
	abnormal		80h	index		sub-index	abort code			

The end frame includes the end frame marker and the valid data length. Its format is shown in Table 7-20.

Table 7-20 Format of end message

Transfer		COB-ID	0	1	2	3	4	5	6	7
Client		600h+Node_ID	60h/0X70h	index		sub-index	-	-	-	-
Server	normal	580h+Node_ID	01h/11h	data						
			03h/13h	data						
			05h/15h	data						
			07h/17h	data						
			09h/19h	data						
			0Bh/1Bh	data						
			0Dh/1Dh	data						
	abnormal		80h	index		sub-index				

7.1.1.2.3 Process Data Object (PDO)

PDO includes receive PDO (RPDO) and transmit PDO (TPDO), which is used to transmit real-time data and is the most important data transmission method in CANopen. Data is passed from a creator to one or more receivers. Since PDO transmission does not require a response, and the PDO length can be less than 8 bytes, the transmission speed is fast. Each CANopen device contains 8 default PDO channels (4 transmit PDO channels and 4 receive PDO channels). PDO includes two transmission modes, synchronous and asynchronous, which are determined by PDO communication parameters. The content of the PDO message is predefined and determined by the PDO mapping parameters.

1. PDO object

   According to the difference between receive and transmit, PDO can be divided into RPDO and TPDO. The final transmission method and content of PDO is determined by communication parameters and mapping parameters. The servo drive can use at most 4 groups of RxPDO and 4 groups of TxPDO to realize PDO transmission. The related object list is as shown in table 7-21.

Table 7-21 Format of PDO message

Name		COB-ID	Communication Object	Mapping Object
RxPDO	1	200h+Node_ID	1400h	1600h
	2	300h+Node_ID	1401h	1601h
	3	400h+Node_ID	1402h	1602h
	4	500h+Node_ID	1403h	1603h
TxPDO	1	180h+Node_ID	1800h	1A00h
	2	280h+Node_ID	1801h	1A01h
	3	380h+Node_ID	1802h	1A02h
	4	480h+Node_ID	1803h	1A03h

2. PDO communication parameters

   CAN identifier

   The CAN identifier (also known as COB-ID) includes control bit and identification data, and determines the CAN priority.

   .COB-ID is located on sub-index 01 of communication parameters (RxPDO: 1400h~1403h and TxPDO: 1800h~1803h), and the highest bit determines whether the PDO is valid.

   Transmission type

   The transmission type is located on sub-index 02 of communication parameters (RxPDO: 1400h~1403h and TxPDO: 1800h~1803h), which determines which transmission method the PDO follows. Transmission Type represents different transmission types, and defines the method for triggering TxPDO transmitt or RxPDO receive, as shown in Table 7-22.

Table 7-22 PDO communication type

Communication Type	Synchronous		Asynchronous
	Cyclic	Acyclic	

0		√	
1~240	√		
241~253	-		
254, 255			√

- When the transmission type of TxPDO is 0, if the mapping data changes and a synchronization frame is received, TxPDO is transmitted.
- When the transmission type of TxPDO is 1~240, TxPDO is transmitted when a corresponding number of synchronization frames are received.
- When the transmission type of the TxPDO is 254 or 255, TxPDO is transmitted when the event timer is triggered.
- When the transmission type of RxPDO is 0~240, as long as one synchronization frame is received, the latest data of RxPDO is applied.
- When the transmission type of RxPDO is 254 or 255, the received data is directly applied.

Inhibit time

The inhibit time (unit: 100us) is set for TxPDO, which is located on sub-index 03 of communication parameters (1800h~1803h) to prevent the CAN network from being continuously occupied by PDO with lower priority. The transmission interval of the same TxPDO shall not be less than the inhibit time.

Event timer

For asynchronous TxPDO, an event timer is defined, which is located on sub-index 05 of communication parameters (1800h~1803h).

PDO mapping parameters

PDO mapping refers to mapping of the application objects (real time data) from the object directory to the process data objects, including index, sub-index and object length. The data length of each PDO can be up to 8 bytes, and one or more objects can be mapped at the same time. The sub-index 0 records the number of objects mapped by the PDO, and the sub-index 1~8 is the mapping content, as shown in table 7-23.

Table 7-23 PDO mapping parameters

Bit	31	......	16	15	......	8	7	......	0
Definition	index			sub-index			object length		

The index and sub-index jointly determine the position of the object in the object dictionary, and the object length indicates the specific bit length, expressed in hexadecimal.

Table 7-24 Relationship between object length and bit length

Object Length	Bit Length
08h	8 bits
10h	16 bit
20h	32 bit

SYNC Object

A synchronization (SYNC) object provides synchronous communication on a CANopen network. You can configure devices to transmit synchronous PDOs when these devices receive an SYNC message.

SYNC object is a special mechanism that controls the coordination and synchronization between transmit and receive of multiple nodes. Its transmission follows producer-consumer model. An SYNC producer broadcasts SYNC messages in the network periodically and all other nodes in the CAN network can receive the synchronization frames as consumers without feedback. Only one synchronization generator is allowed in the same CAN network.

The relationship of synchronous PDOs and synchronization frame is as follows:

- For synchronous RxPDO, as long as the PDO is received, the received RxPDO will be applied at the next SYNC.
- For synchronous TxPDO, it can be divided into the following:
  - Cyclic synchronous: when the PDO transmission type is 1~240, and only when the SYNC specified by the transmission type is reached, no matter whether the data has changed or not, this TxPDO needs to be sent.
  - Acyclic synchronous: when the PDO transmission type is 0 and the content of the PDO mapping object changes, this TxPDO needs to be sent at the next SYNC.

7.1.1.2.4 Emergency (EMCY) Object Service

A CANopen device generates an emergency (EMCY) message when a device-internal error occurs. The EMCY message follows the producer-consumer model. After it is sent, other nodes in the CAN network can choose to handle it.

When a node fails, regardless of whether the EMCY object is activated, the error registers and predefined error fields need to be updated. The format of the EMCY message is as follows:

Table 7-25 Format of EMCY message

COB-ID	0	1	2	3	4	5	6	7
80h+Node_ID	error code		error register	Reserved	auxiliary byte			

For the definition of error codes and auxiliary bytes, please refer to”.

7.1.1.3 CANopen frame COB-ID

CANopen 2.0A defines 11-bit CAN-ID, the first 4 bits equal a function code and the next 7 bits contain the node ID.

To facilitate the networking between devices on the bus, CANopen defines communication object identifier (COB-ID) according to CAN-ID: specifies the priority of the object during communication and the identification of the communication object. The COB-ID corresponds to the CAN-ID. The 11-bit COB-ID consists of two parts, which are 4-bit object function and 7-bit node ID.

Table 7 - 26 COB-ID and CAN-ID

10	9	8	7	6	5	4	3	2	1	0
Function Code				Node-ID						

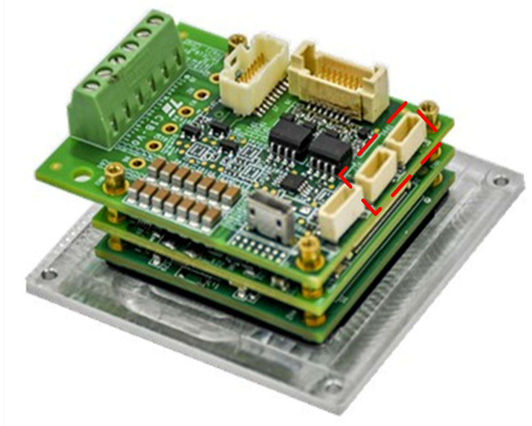
Each communication object of CANopen has a default COB-ID, to identify the devices and communication objects on the bus. The smaller the function code, the higher the priority of the communication object. The communication objects and their COB-IDs are shown in Table 7-27.

Table 7 - 27 Communication objects and their COB-IDs

Communication Object	COB-ID(hex)	Function Code	Related communication parameters in object dictionary
NMT Network Control (Broadcast)	000	0000b	-
Sync object (broadcast)	080	0001b	0x1005, 0x1006, 0x1007
EMCY object	080+Node-ID	0001b	0x1014, 0x1015
TPDO1	180+Node-ID	0011b	0x1800
RPDO1	200+Node-ID	0100b	0x1400
TPDO2	280+Node-ID	0101b	0x1801
RPDO2	300+Node-ID	0110b	0x1401
TPDO3	380+Node-ID	0111b	0x1802
RPDO3	400+Node-ID	1000b	0x1402
TPDO4	480+Node-ID	1001b	0x1803
RPDO4	500+Node-ID	1010b	0x1403
SDO(Tx)	580+Node-ID	1011b	0x1200
SDO(Rx)	600+Node-ID	1100b	0x1200
NMT Network Error Control	700+Node-ID	1110b	0x1016, 0x1017

7.1.2 CANopen

7.1.2.1 Communication Interface



Pin	Definition
1	NC
2	GND
3	CAN_L
4	CAN_H
5	PE

Figure 7-2 Definition of communication interface

7.1.2.2 Communication wiring

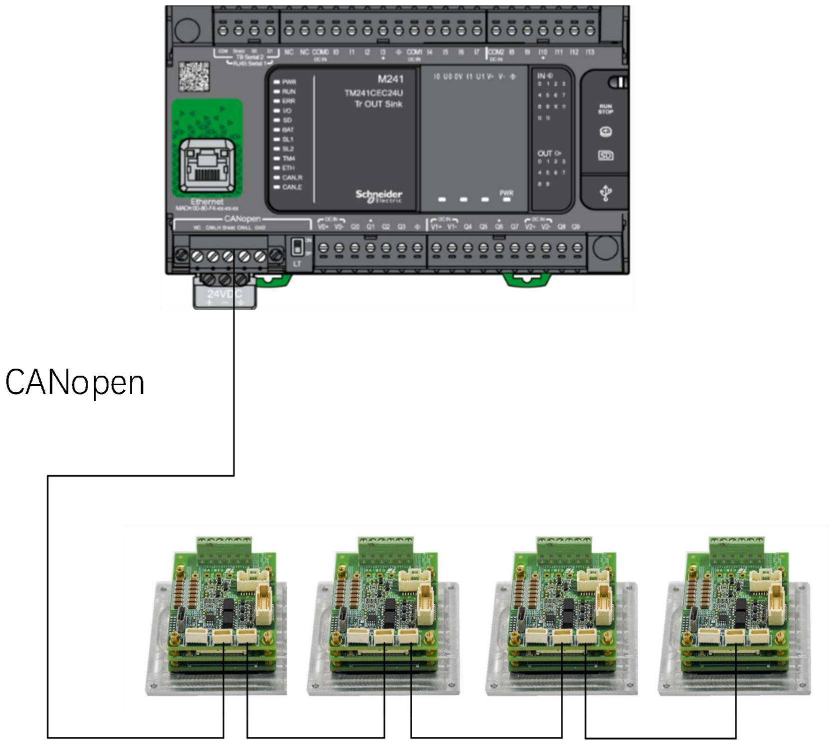


Figure 7-3 CANopen communication wiring

7.1.2.3 Software settings

The whole process to start communication is as follows:

1. In the upper computer software, import the EDS file.
2. Scan the Stone servo.
3. Set the communication parameters.

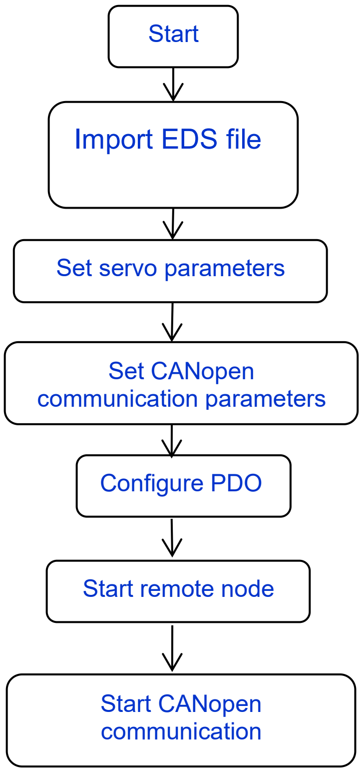


Figure 7-4 Software Settings

Before using the CANopen servo drive, please set the following parameters:

Note: The two parameters are effective after the servo restarts. After modifying them, please power on again or "restart" the servo in the SMC software.

1. To set the baud rate of CAN communication, modify parameter 0x2004 CAN baud rate via the SMC software.

   The baud rate of the master and slave must be same. Otherwise, communication fails.

   The relationship between the value of baud rate and the length of the bus communication cable is as shown in Tables 7-28.

Table 7-28 Relationship between baud rate and communication cable length

Baud Rate (bit/s)	Communication Cable Length (m)
1000000	25
500000	100
250000	250
125000	500
50000	1000
20000	1000

2. To set communication node ID of each slave, modify parameter 0x2401 Node ID via the SMC software.

The node ID of each servo slave cannot be the same with CNC or PLC of the master. And the node ID of each servo slave cannot be the same as well.

7.1.3 Communication with Schneider PLC

1. Connect hardware and check configuration

Refer to 7.1.2 and complete the hardware and basic configuration between the servo and PLC.

2. Create a project

a. Run Schneider SoMachine software, and create a new blank project named test_CAN.

The following takes software version 4.3 as an example:

Figure 7-5 Run SoMachine software

b. Double-click "**Delete and Add Devices**" to add the master, and click "OK".

Here takes deviceTM241CEC2U as an example:

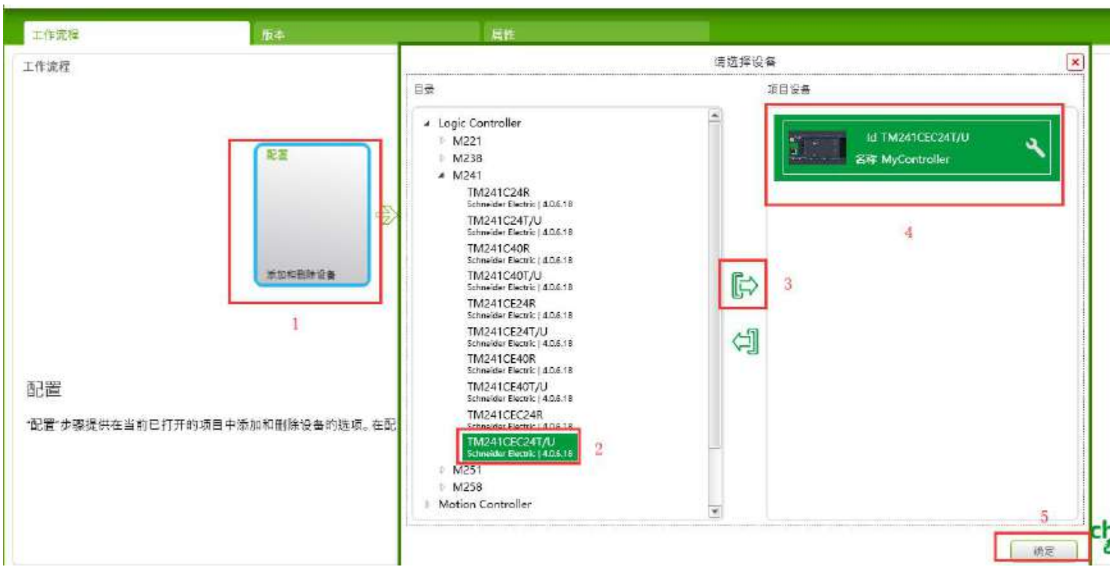


Figure 7-6 Add a master in SoMachine

c. Double click “**Controller**”, and add the app design:

Figure 7-7 Create a project in SoMachine

3. Import the EDS file of the slave

a. In the menu bar, click “**Tool**” --> “**Device Library**”.

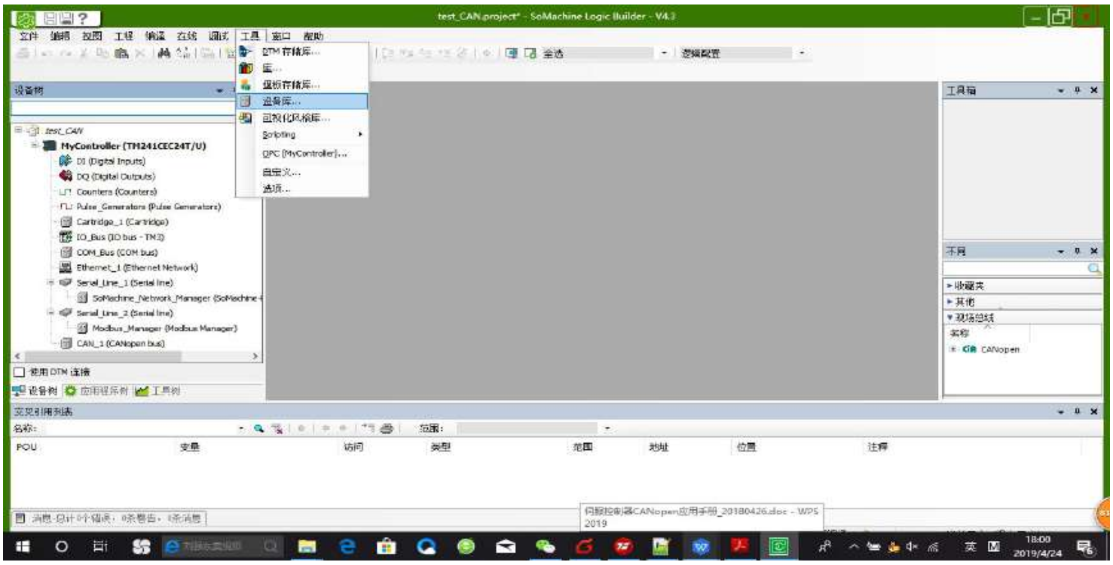


Figure 7-8 Open Device Library in SoMachine

b. Click “**Install**”, select the EDS file "**Stone.eds**" in the target path, and click “**Open**”.

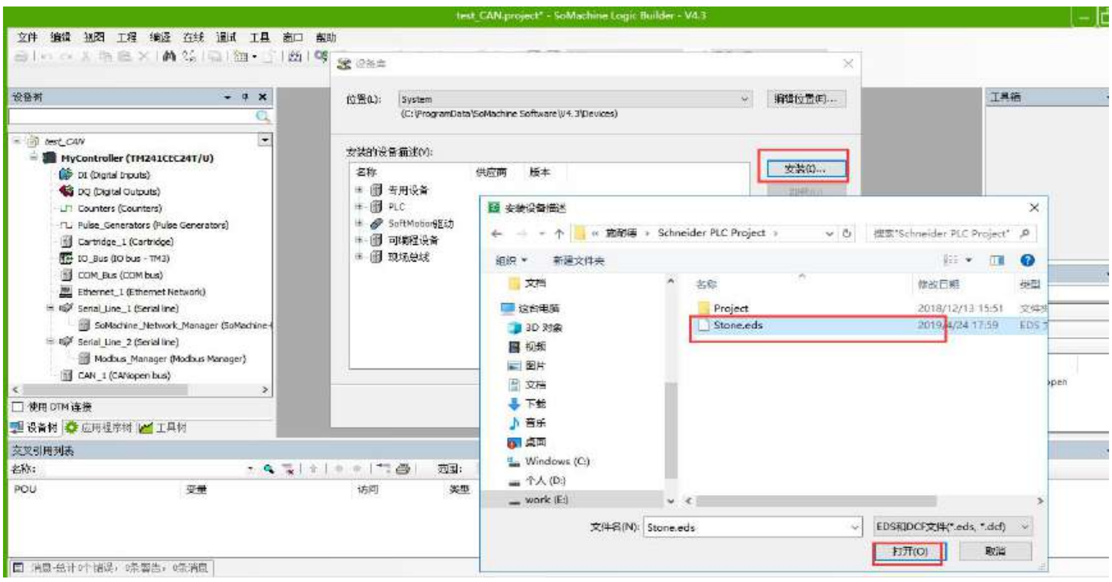


Figure 7-9 Import EDS file into SoMachine

4. Add CANopen gateway and slave

a. To add a device, click “**+**” under the device tree “**CANopen bus**”, select “**Schneider Electric**” as the supplier, select “**CANopen Performance**”, and click “**Add**”.

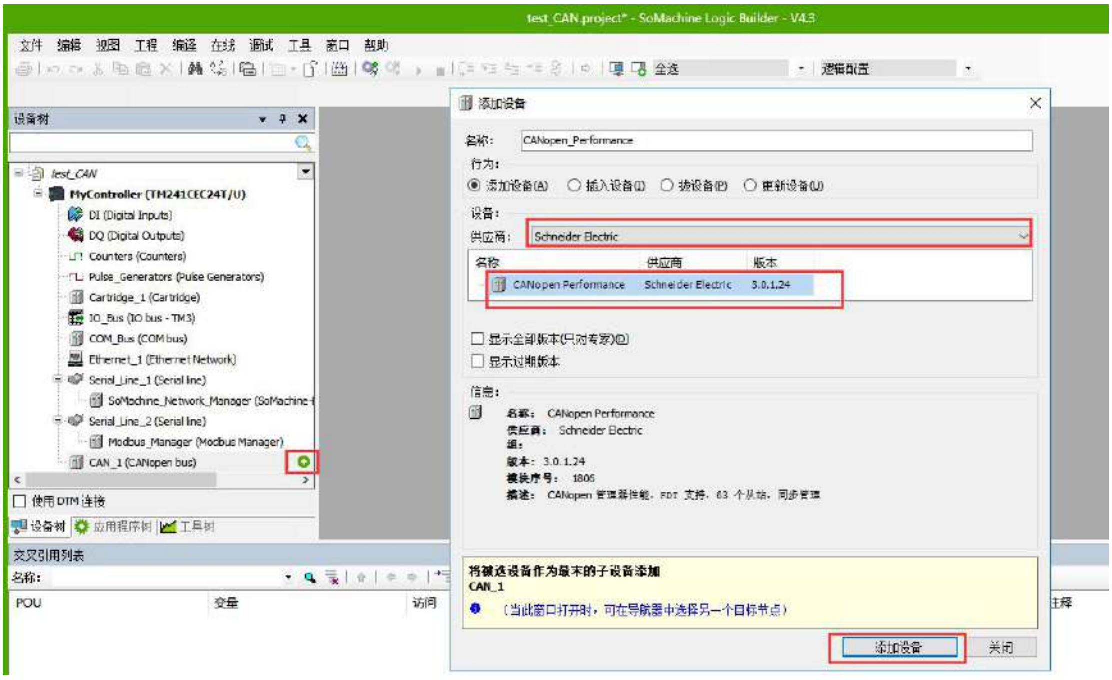


Figure 7-10 Add CANopen to SoMachine

b. Right click “**CANopen_Performance**” to add a slave, select “**Stone Motion Control**” as the supplier and "**Diamond Plus**" as the device name.

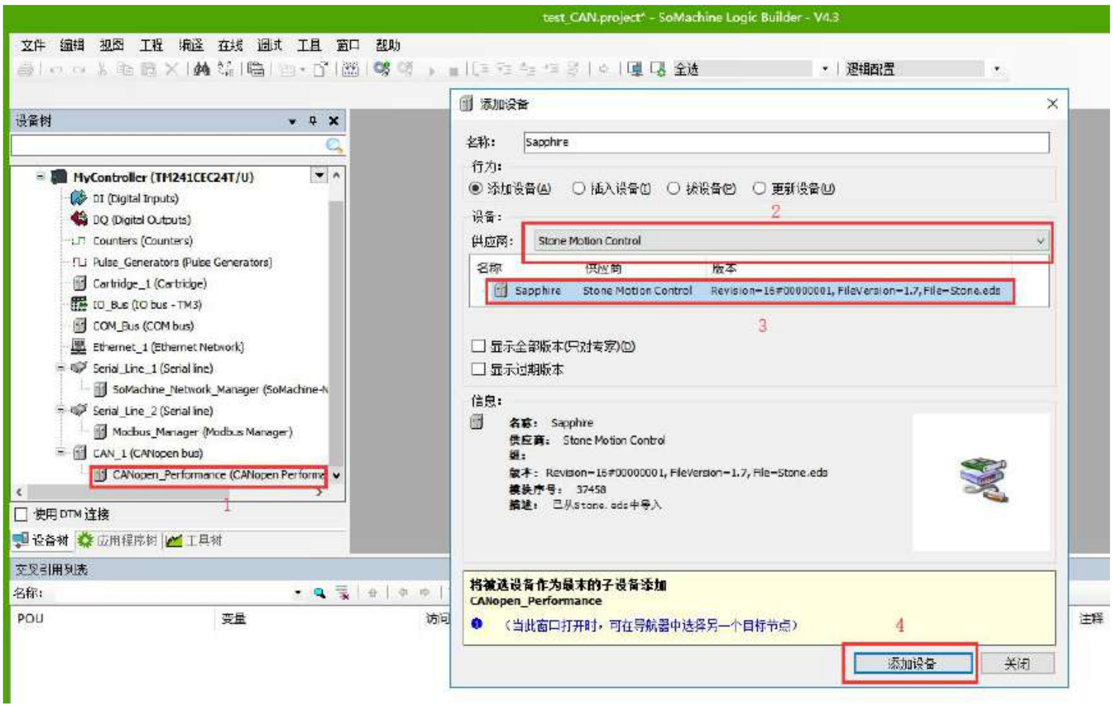


Figure 7-11 Add CANopen slave to SoMachine

5. **Configure CANopen for master and slave**

   a. Set the CAN baud rate of the master to 1 M/s.

   The default baud rate of the servo is 1 M/s.

   You can also change the value according to requirements. But please make sure that the baud rate of the servo is the same as that of the master.

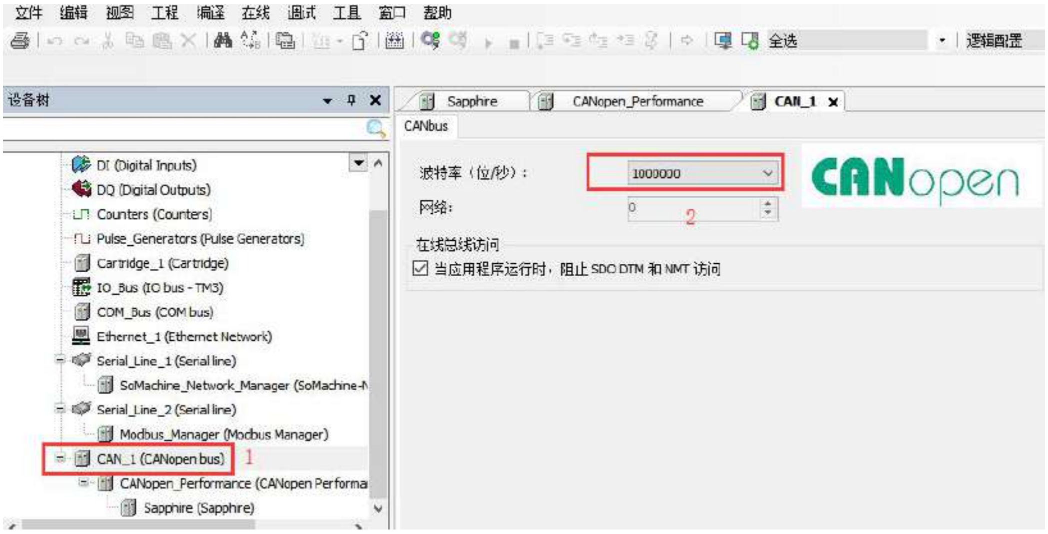


Figure 7-12 Set baud rate of master in SoMachine

   b. Set the node ID of the slave to 1

   The default node ID of the servo to 1.

   You can also change the value according to requirements. But please make sure that the node ID of the servo is the same as that of the master.

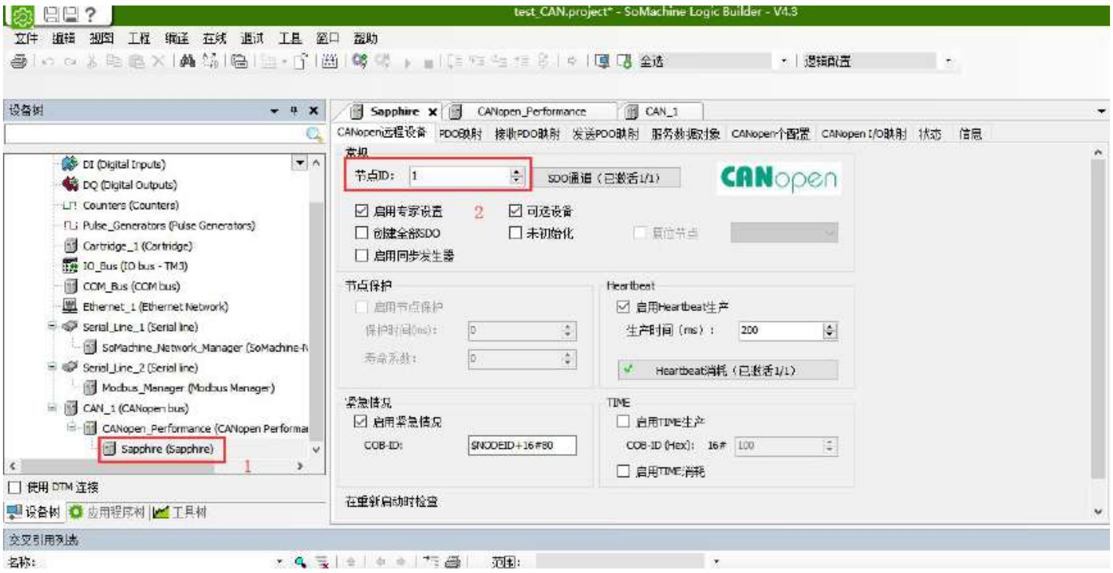


Figure 7-13 Set node ID of slave in SoMachine

   c. To set the baud rate of CAN communication, modify parameter 0x2004 CAN baud rate via the SMC software.

   The baud rate of the master and slave must be same. Otherwise, communication fails.

   The relationship between the value of baud rate and the length of the bus communication cable is as shown in Tables 7-29.

Table 7-29 Relationship between baud rate and communication cable length

Baud rate (bit/s)	Communication cable length (m)
1000000	25
500000	100
250000	250
125000	500
50000	1000
20000	1000

d. To set communication node ID of each slave, modify parameter 0x2401 Node ID via the SMC software.

The node ID of each servo slave cannot be the same with CNC or PLC of the master. And the node ID of each servo slave cannot be the same as well.

Note: 0x2004 and 0x2401 are effective after the servo restarts. After modifying them, please power on again or "restart" the servo in the SMC software.

6. Configure PDO mapping and SDO

Here takes position mode as an example, for details, please refer to "Stone Servo Drive CANopen Application Manual". In the position mode, PDO configuration is as follows.

Table 7-30 PDO configuration in position mode

PDO	Object	Meaning	Bit length
RPDO1	6040h-00h	Control word	16
	6060h-00h	Control mode	8
	607Ah-00h	Target position	32
RPDO2	6081h-00h	Planned speed	32
TPDO1	6041h-00h	Status word	16
	6061h-00h	Control mode display	8
	6064h-00h	Position feedback	32
TPDO2	606Ch-00h	Velocity feedback	32
	6078h-00h	Current feedback	16

a. Add PDO mapping:

Start expert setup, configure PDO mapping, and select 2 groups to receive PDOs, and 2 groups to send PDOs.

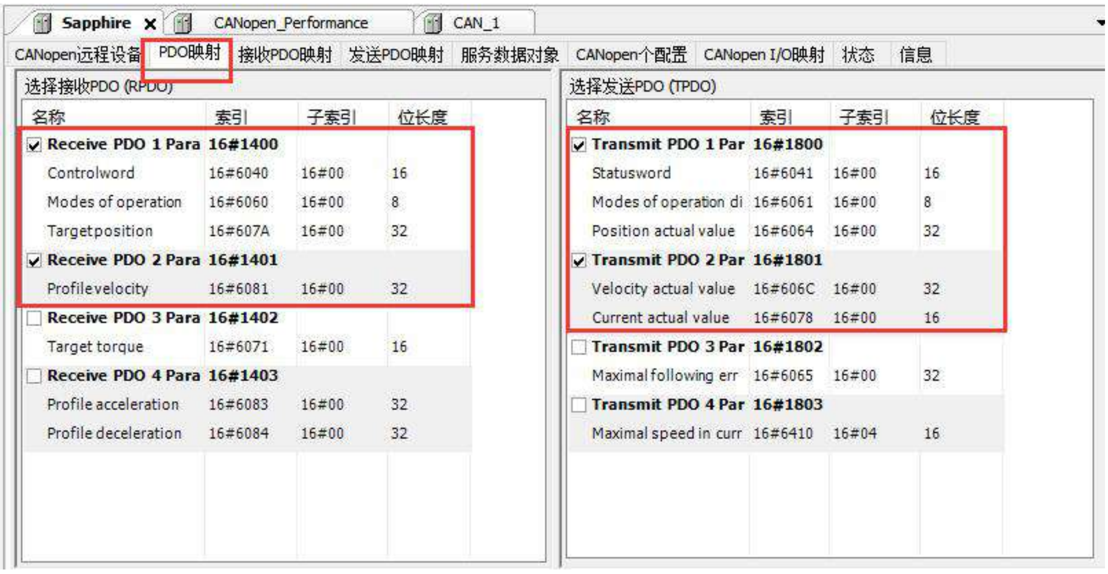


Figure 7-14 Add RPDO and TPDO mapping

When the default RPDO and TPDO do not contain a control word, please separately add PDO mapping as follows:

1) Set mapping parameters for receiving PDO1 and PDO2 as follows, and set the transmission type to 255 Asynchronous mode.

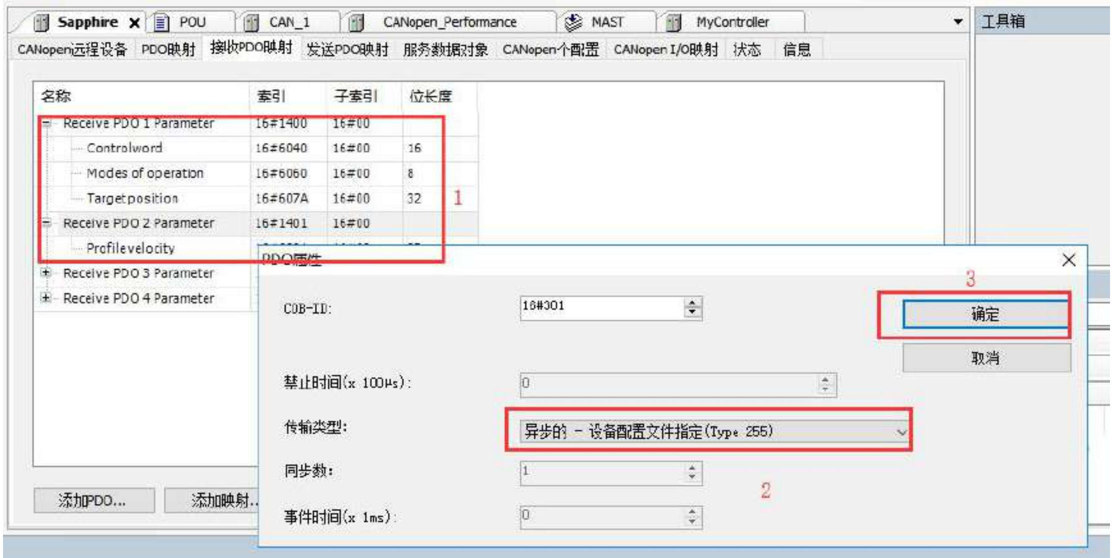


Figure 7-15 Add RPDO mapping

2) Set mapping parameters for transferring PDO1 and PDO2 as follows, and set the transmission type to 255 asynchronous mode.

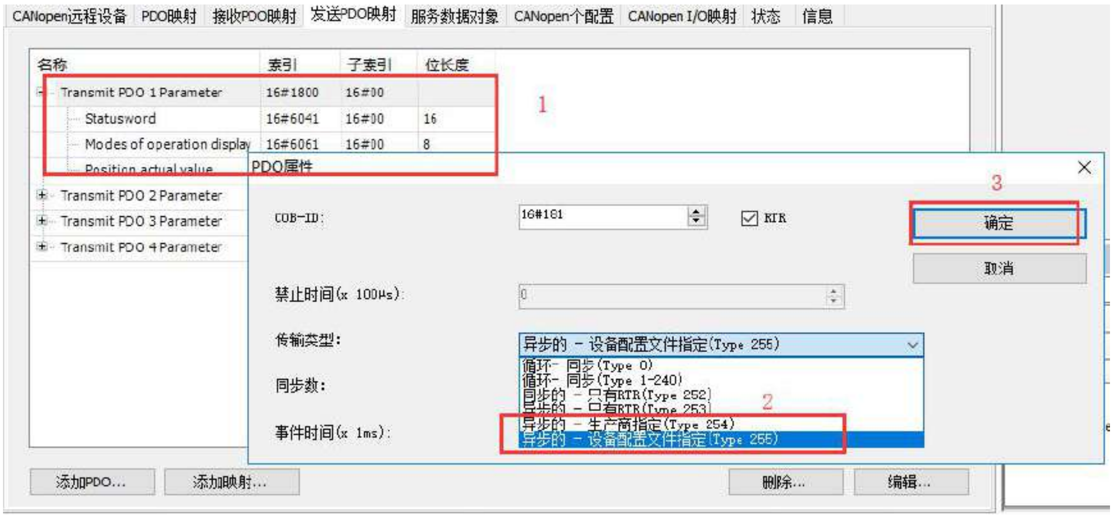


Figure 7-16 Add TPDO mapping

b. Configure SDO data: use SDO to configure acceleration 6083h and deceleration in position mode.

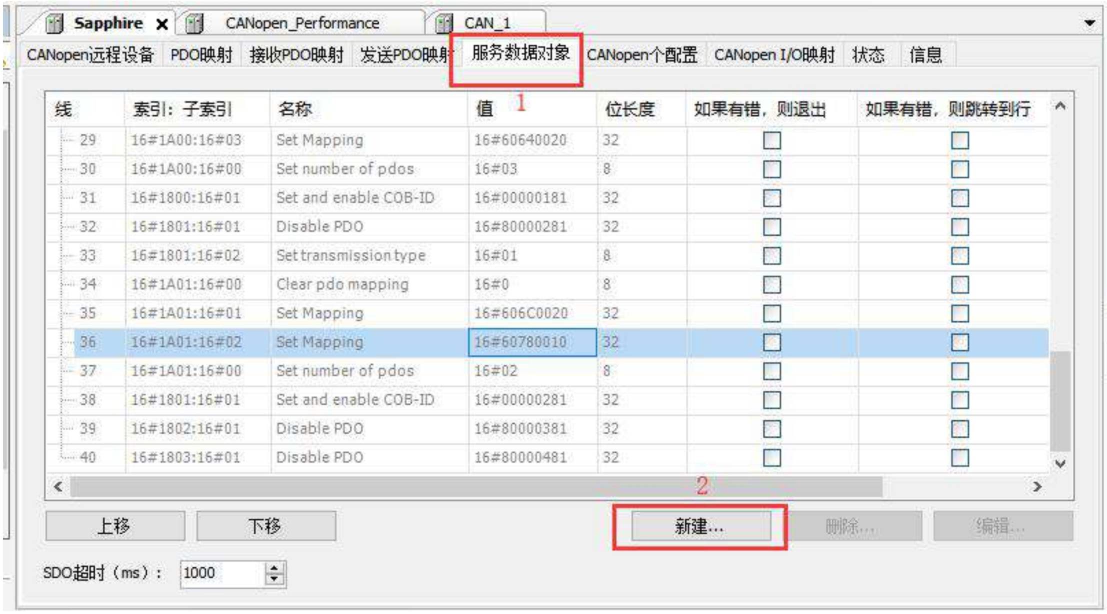


Figure 7-17 Add SDO (1)

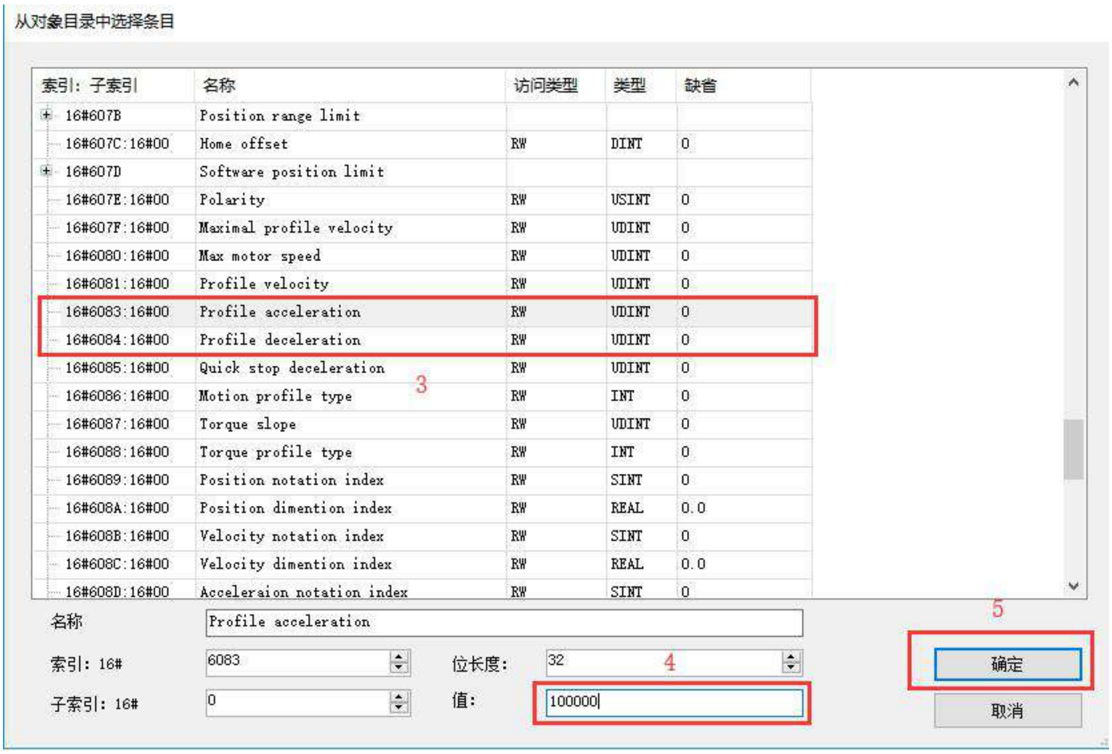


Figure 7-18 Add SDO (2)

7. Create a POU program

a. Under the application device tree, select “**Application**”, add a POU program, select the program language, and click “**Add**”.

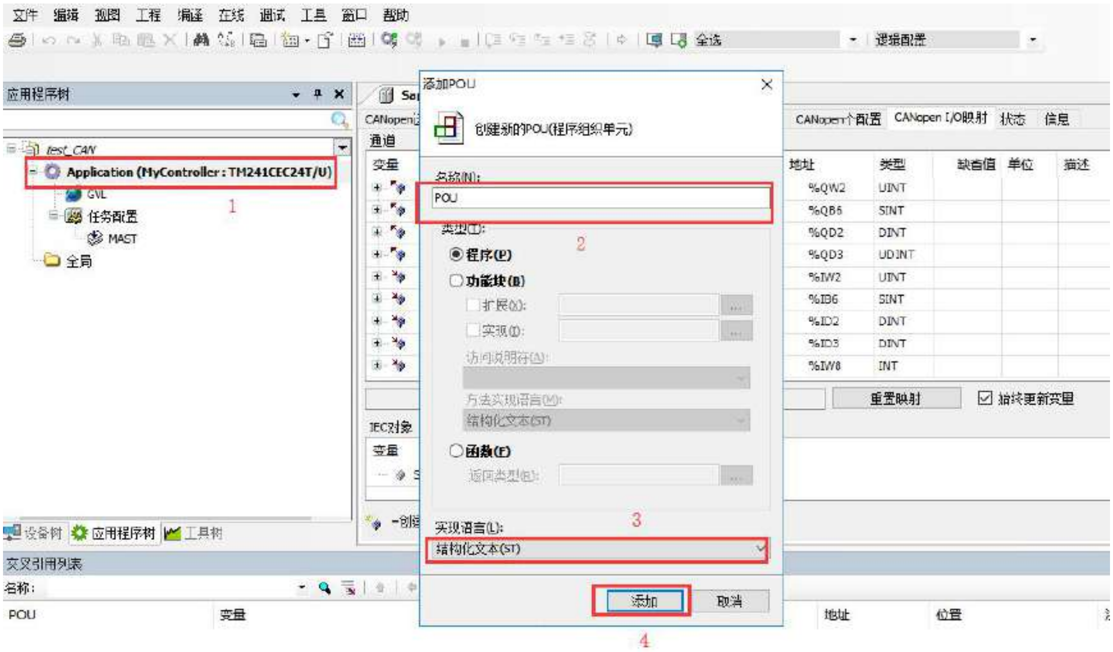


Figure 7-19 Create a POU

b. Double click "**POU**", define the variable in area 2, write the logic program in 3 area, click "**Compile**", confirm that there is no error, and proceed to the next step, as shown in Figure 7-20.

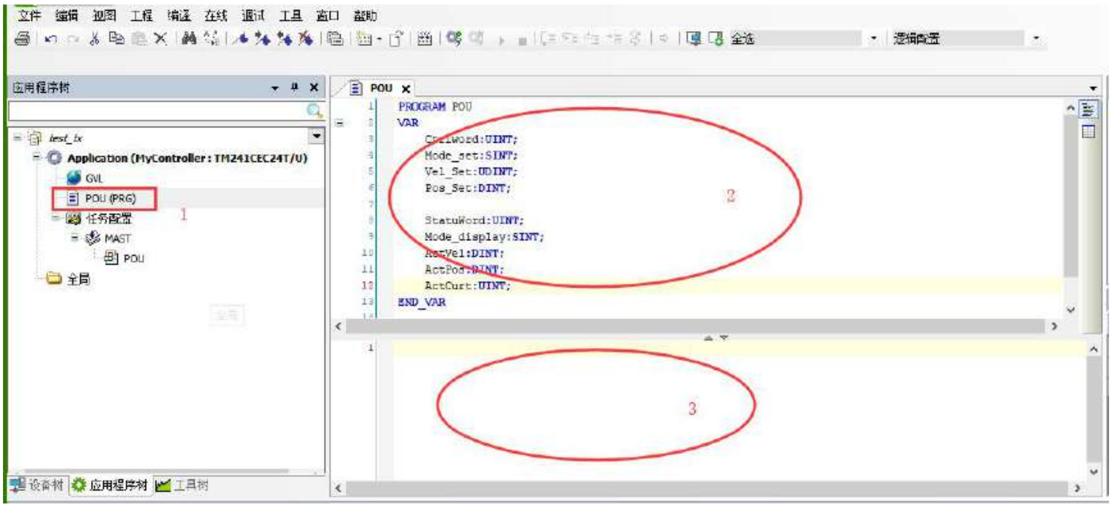


Figure 7-20 Write POU program

c. Double click "**Master**", and set the program cycle.

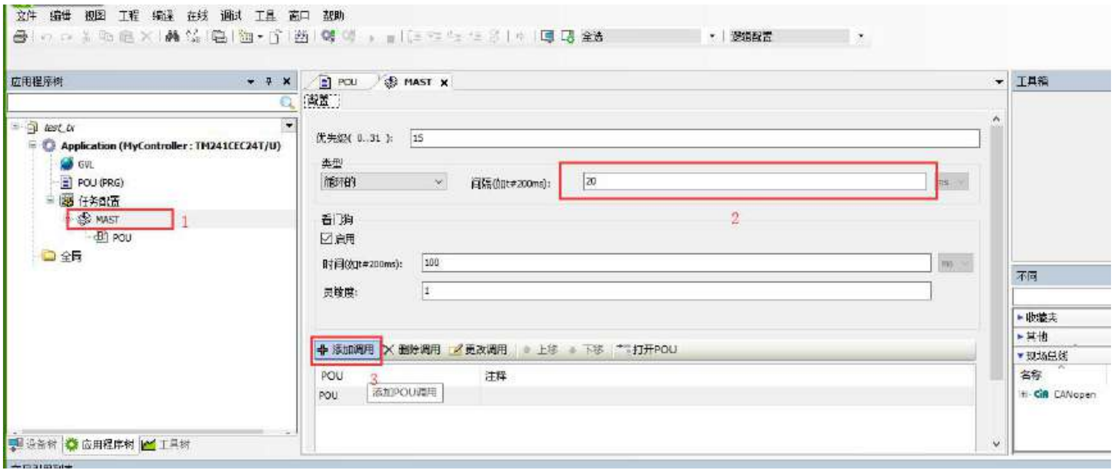


Figure 7-21 Set program cycle

d. Double click "**Master**", and add a POU call.

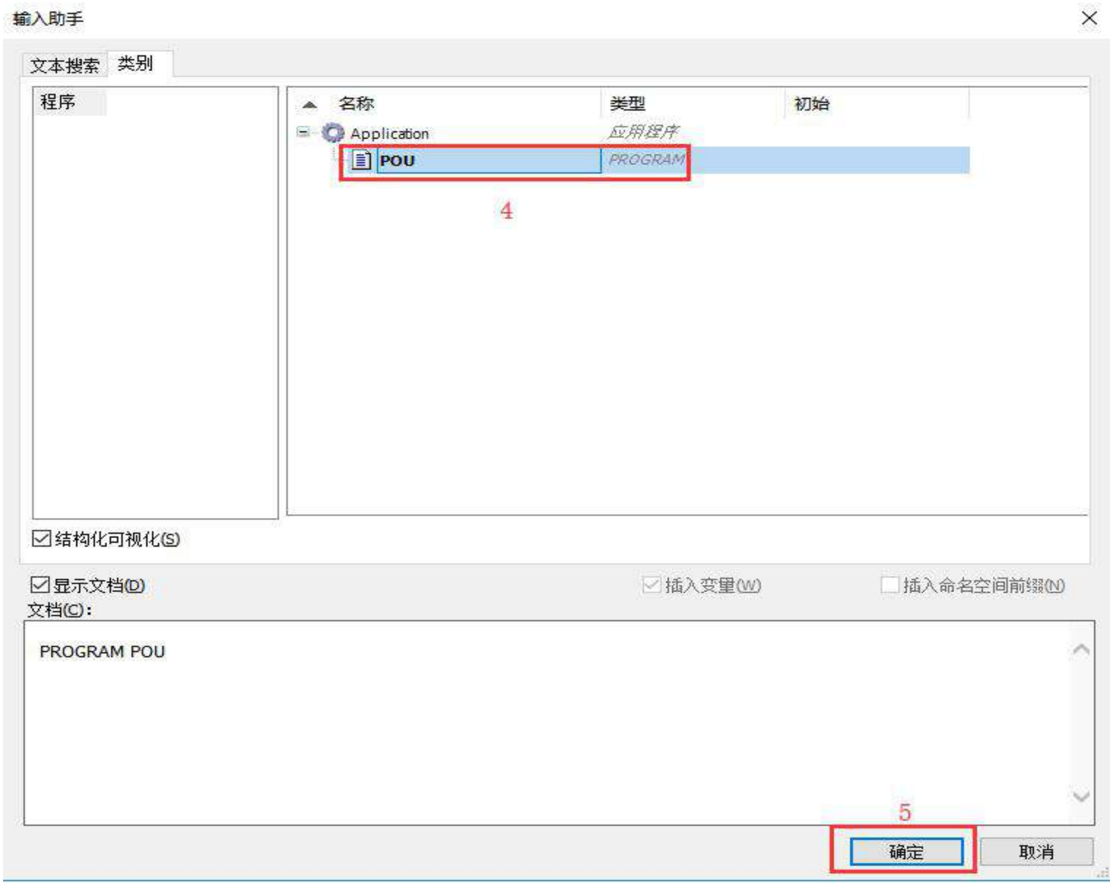


Figure 7-22 Add a POU call

8. Link CANopen I/O mapping to POU variable.

a. Select "**Diamond Plus**", click "**CANopen I/O Mapping**", and link POU variables to PDO parameters.

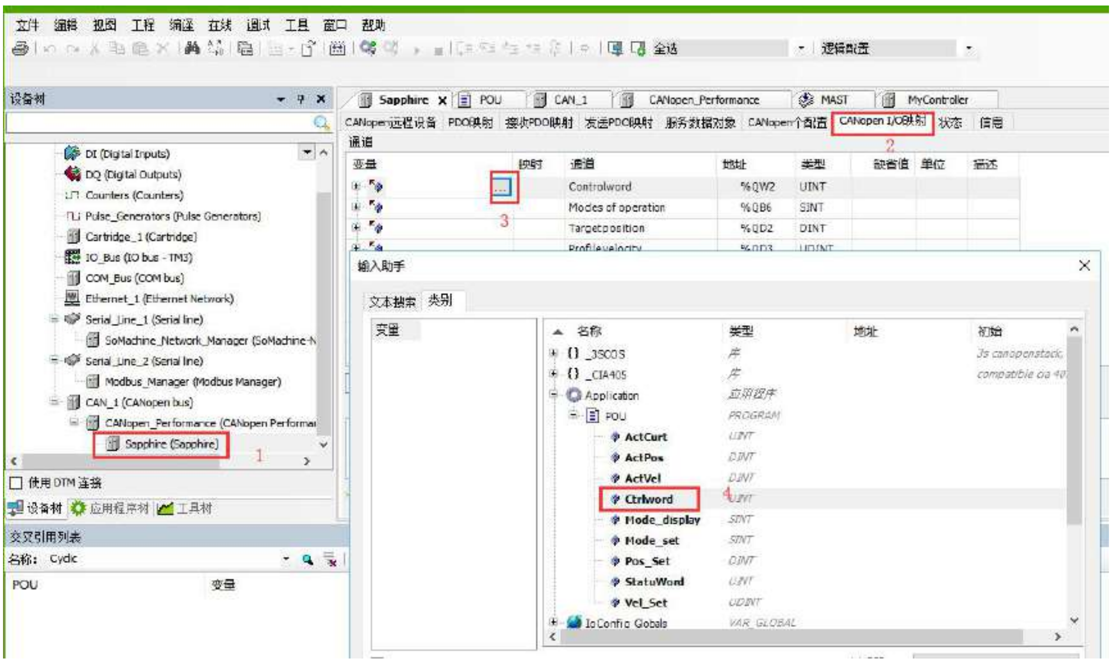


Figure 7-23 CANopen I/O Mapping

b. Link all variables in turn.

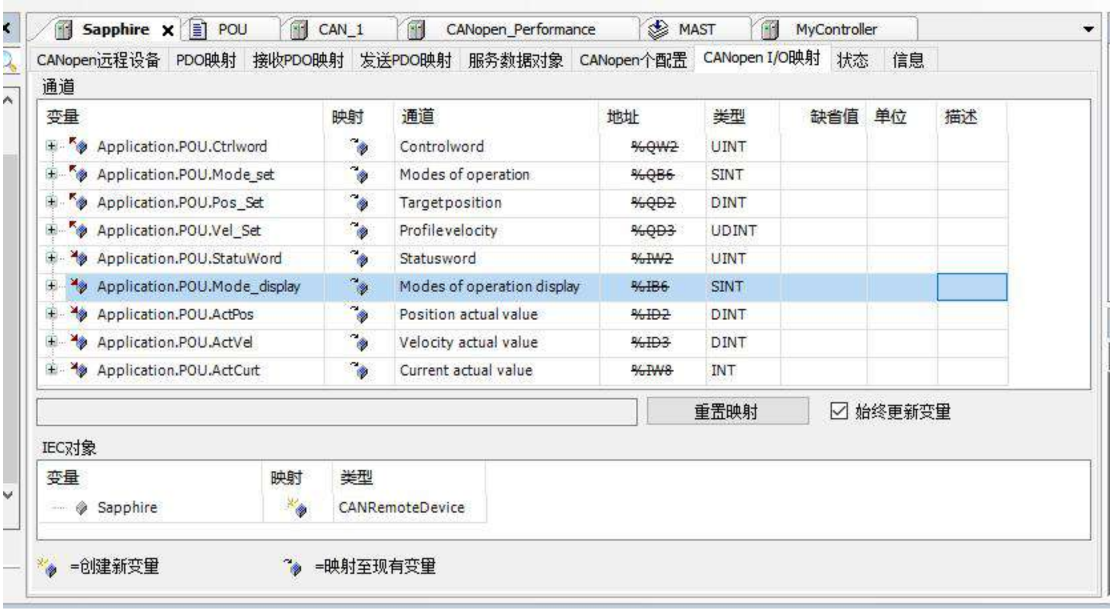


Figure 7-24 Link POU variables to PDO parameters

9. Log in and do debugging

a. Click the login button  or press "**Alt**" and "**F8**" to log in PLC. The following dialog box pops up as shown in 7-25.

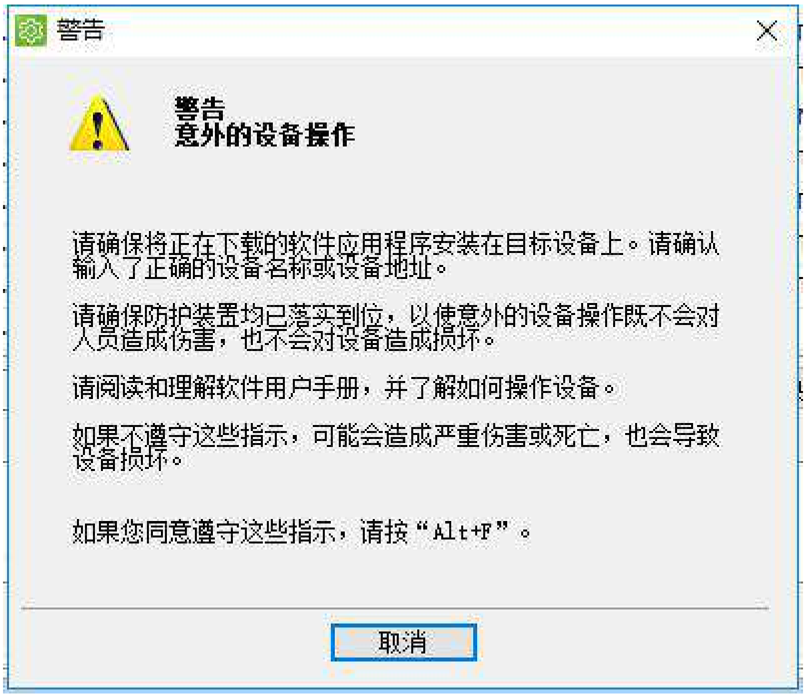


Figure 7-25 After clicking login button

b. Press "**Alt**" and "**7**", and click "**Yes**" to complete PLC login and download.

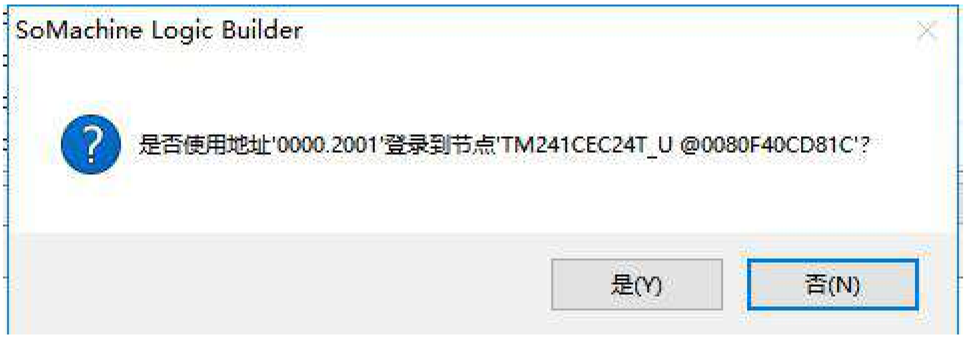


Figure 7-26 Confirm PLC login

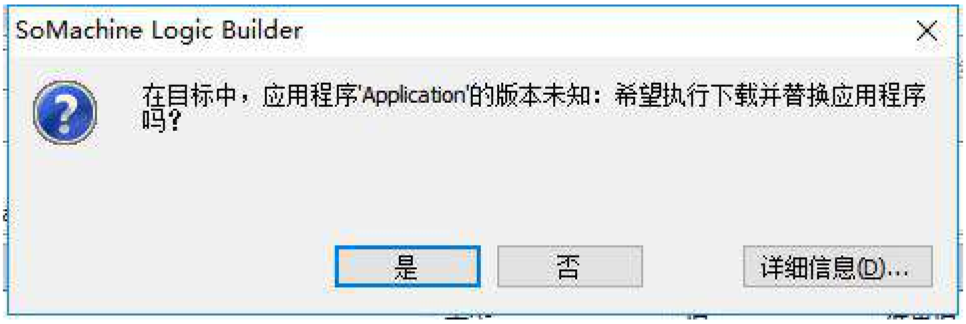


Figure 7-27 Confirm program download

c. Start PLC.

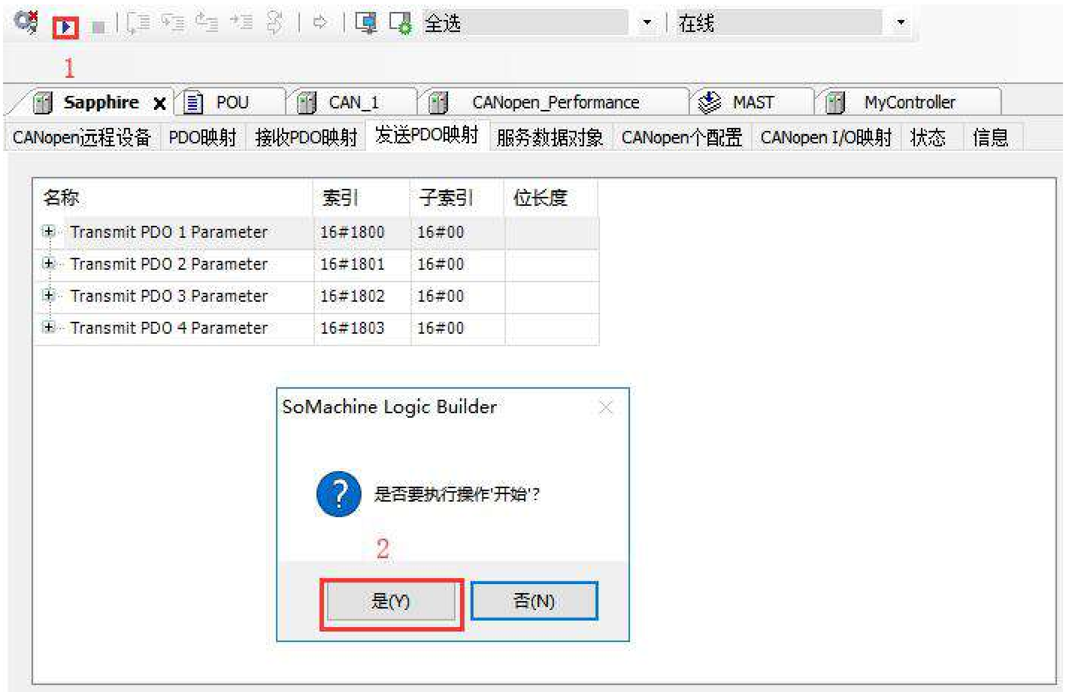


Figure 7-28 Start PLC

d. Do debugging

Forcibly configure the following parameters, and press F7 to force writing

e.g. Run the relative posit ion for 10 turns: to control the motor to run, set the control mode to "1", the target position to 100000 cnt, the planned speed to 1000000cnt/s, and the control word to 0x06 -> 0x07 -> 0x4F -> 0x5F.

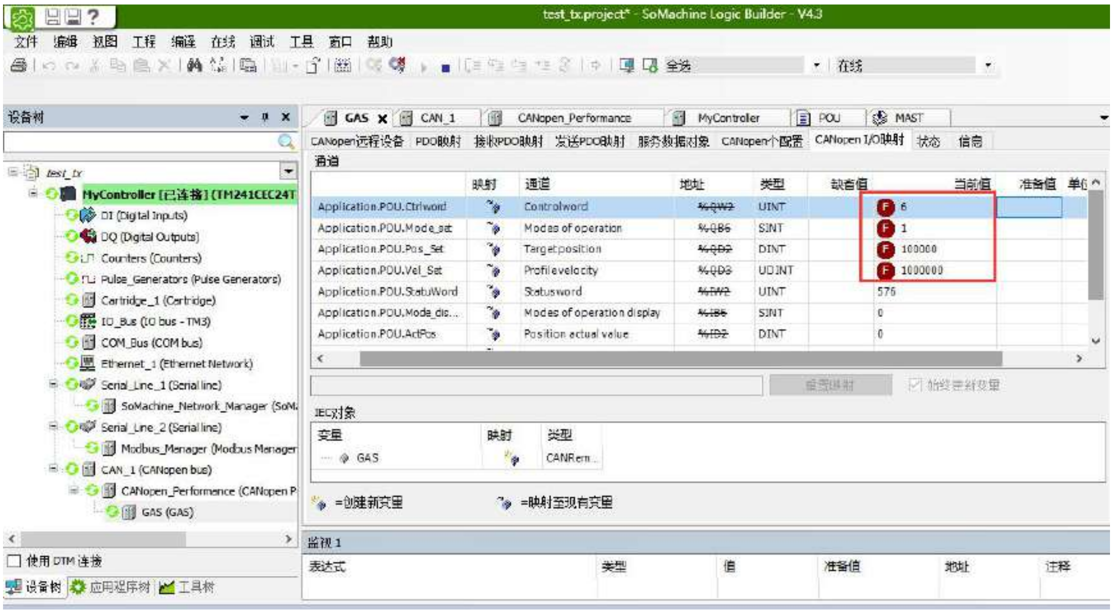


Figure 7-29 Program debugging

7.2 EtherCAT Communication

7.2.1 Principle

7.2.1.1 CoE reference model

The internal CANopen over EtherCAT (CoE) network model of Stone servo is shown in Figure 7-30.

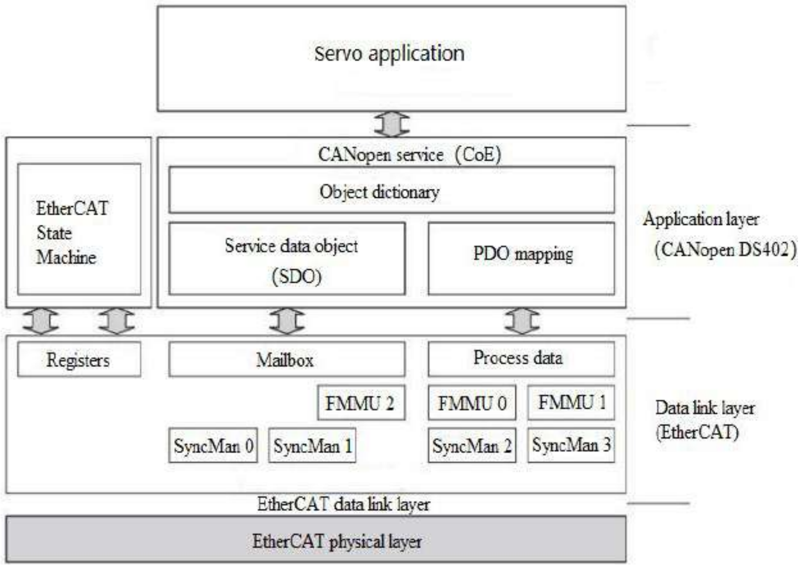


Figure 7-30 COE reference model

The EtherCAT (CoE) network model consists of two parts:

- Data link: mainly responsible for EtherCAT communication protocol
- Application: it embeds the CANopen drive Profile (DS402) communication protocol.

The object dictionary in CoE includes parameters, application data, and PDO mapping information.

Process data object (PDO) consists of objects in the object dictionary that can do PDO mapping, and the content in PDO data is defined by PDO mapping. The read and write of PDO data is periodic with no need to look up the object dictionary; while the mailbox communication (SDO) is non-periodic communication with a need to look up the object dictionary.

7.2.1.2 EtherCAT slave information

The EtherCAT slave information file (XML file) is read by the master and used to construct the configuration of the master and slave. The XML file contains the necessary information for EtherCAT communication. STONE provides the "Stone_E XML.xml" file for the servo drive to construct the configuration of the master and slave.

7.2.1.3 EtherCAT State Machine

It is used to describe the states and state changes of the slave. The state change request is usually initiated by the master and the slave responds. The details is shown in Figure 7-31.

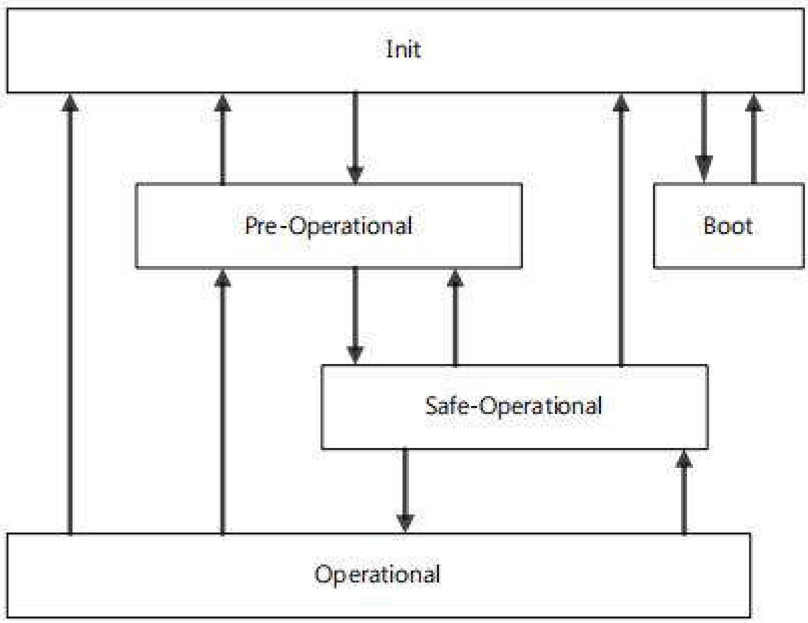


Figure 7-31 EtherCAT state machine

The status of state machine is shown in Table 7-31.

Table 7-31 Status

Status	Description
Boot	◆ Firmware update. ◆ Drive can transit to Init state.
Init	◆ Initialize the communication. ◆ Unable to communicate with SDO and PDO.
Init->Pre-OP	◆ The master configures the link address and SM channel to start mailbox communication. ◆ The master initializes DC clock synchronization. ◆ The master requests a transition to the Pre-Op state. ◆ The master sets the AL control register. ◆ The slave checks if mailbox is normally initialized.
Pre-OP	◆ Mailbox communication is activated. ◆ Unable to communicate with PDO.
Pre-OP->Safe-OP	◆ The master configures the Sync Manager channel and FMMU channel for PDO. ◆ The master configures PDO data mapping and Sync Manager PDO parameters through SDO. ◆ The master requests transition to Safe-Op state. ◆ The slave checks if the Sync Manager responsible for the PDO data is correctly configured, and check the distributed clock when the slave sends a request to start synchronization.
Safe-OP	◆ The slave application will transfer the actual input data and will not respond to the output. ◆ Output is set to "safe state".
Safe-OP->OP	◆ The master sends valid output data. ◆ The master requests a transition to the Op state.
OP	◆ Mailbox communication is available. ◆ PDO communication is available.

7.2.1.4 PDO mapping

STONE servo has 4 configurable PDOs, including 2 RxPDOs (0x1600 and 0x1601) and 2 TxPDOs (0x1A00 and 0x1A01). When you need to change the default PDO mapping, you can change the xml file and configure it into the servo.

Note: When using EtherCAT communication, it is necessary to set the communication cycle of the upper computer to be the same as that of the lower servo (default: 4 ms).

The default PDO mapping of STONE servo is as follows:

RxPDO

Table 7-32 RxPDO

(sub) index	Name	Object Type	Default
0x1600	1st Receive PDO	REC Data type -	-
0x00	Number of mapped objects	UINT8	10
0x01	Mapped object 1	UINT 16	0x6040 Control word
0x02	Mapped object 2	UINT32	0x607A Target position
0x03	Mapped object 3	UINT32	0x60B1 Velocity offset
0x04	Mapped object 4	UINT 16	0x60B2 Torque offset
0x05	Mapped object 5	UINT32	0x60FF Target velocity
0x06	Mapped object 6	UINT 16	0x6071 Target torque
0x07	Mapped object 7	UINT 8	0x6060 Modes of operation
0x08	Mapped object 8	UINT 8	0x0000
0x09	Mapped object 9	UINT32	0x0000
0x0A	Mapped object 10	UINT32	0x0000

Table 7-33 RxPDO

(sub) index	Name	Object Type	Default
0x1601	2st Receive PDO	REC Data type -	-
0x00	Number of mapped objects	UINT8	12
0x01	Mapped object 1	UINT32	0x0000
		UINT32	0x0000
0x0C	Mapped object 12	UINT32	0x0000

TxPDO

Table 7-34 TxPDO

(sub) index	Name	Object Type	Default
0x1A00	1st Transmit PDO	REC Data type -	-
0x00	Number of mapped objects	UINT8	12
0x01	Mapped object 1	UINT 16	0x6041 Statusword
0x02	Mapped object 2	UINT32	0x6064 Position actual value
0x03	Mapped object 3	UINT32	0x 606C Velocity actual value
0x04	Mapped object 4	UINT 16	0x 6077 Torque actual value
0x05	Mapped object 5	UINT 8	0x 6061 Modes of operation

(sub) index	Name	Object Type	Default display
0x06	Mapped object 6	UINT 8	0x0000
0x07	Mapped object 7	UINT32	0x0000
0x08	Mapped object 8	UINT32	0x 0000
0x09	Mapped object 9	UINT32	0x0000
0x0A	Mapped object 10	UINT32	0x 0000
0x0B	Mapped object 11	UINT32	0x0000
0x0C	Mapped object 12	UINT32	0x0000

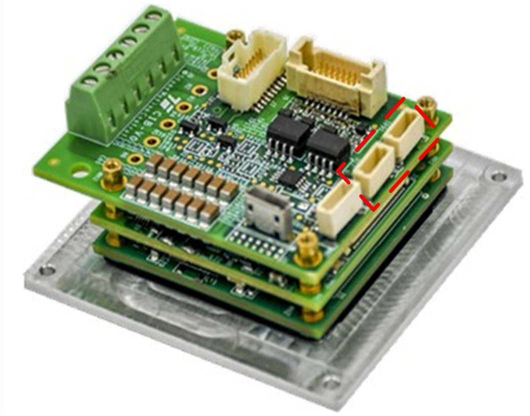
Table 7-35 TxPDO

(sub) index	Name	Object Type	Default
0x1A01	2 st Receive PDO	REC Data type -	-
0x00	Number of mapped objects	UINT8	12
0x01	Mapped object 1	UINT32	0x0000
⋮		UINT32	0x0000
0x0C	Mapped object 12	UINT32	0x0000

Note: You can query detailed PDO mapping information in the xml file.

7.2.2 EtherCAT

7.2.2.1 Communication Interface



Pin	Name	Definition	Direction
1	TX+	Send data+	Output
2	TX-	Send data-	Output
3	RX+	Receive data+	Input
4	RX-	Receive data-	Input
5	PE	Shield	-

Figure 7-32 Definition of EtherCAT communication interface

7.2.2.2 Communication Wiring

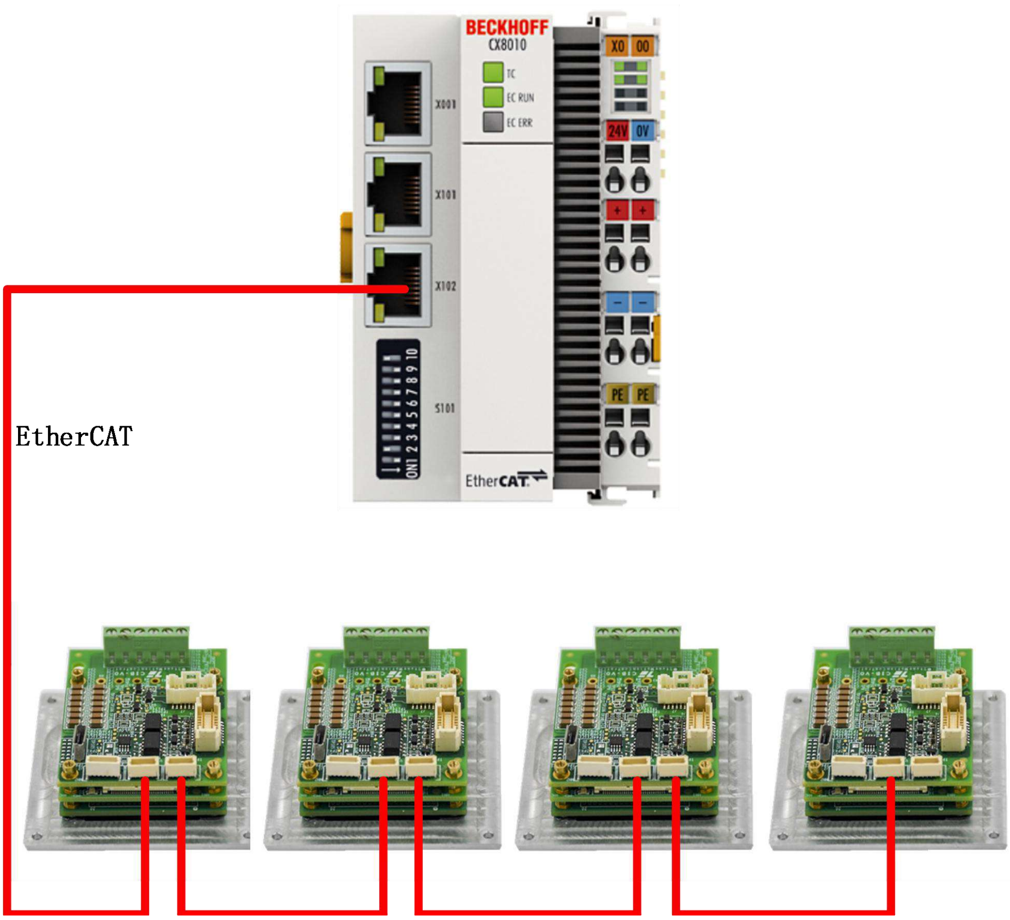


Figure 7-33 Communication wiring

7.2.2.3 Software Settings

1. Configure the motor parameters, and make sure that the servo motor can be operated normally with the SMC software.
   For the trial run, please refer to "***Diamond Plus Servo User Manual***".
2. Select master type (0x2005):
   - ✓ 0: support the 402 state machine of most masters, including Beckhoff.
   - ✓ 1: specially support the 6061 state machine of Omron PLC.
3. Set servo communication cycle, set 0x60C2 to 01.
   Communication cycle range is within 1 - 4 ms (default value: 4 ms).
   The communication cycle of controller and servo should be the same. Otherwise, a synchronization error will occur during running.

The master triggers DC mode in the CSP mode. Otherwise, it will not operate normally.

Note:

1. The transferring and receiving PDOs can be dynamically configured by the master, but the maximum number for each PDO parameter is 10. If the range is exceeded, the slave will be unable to enter the op state.
2. The sequence of network cables is IN → OUT. Otherwise, some nodes may be unable to enter the op state.

7.2.3 Communication with Beckhoff PLC

1. Connect hardware and check configuration

Refer to chapter 1 and 2 and complete the hardware and basic configuration between the servo and PLC.

2. Place the configuration file

Place file Stone_E XML.xml under the TwinCAT directory as follows:

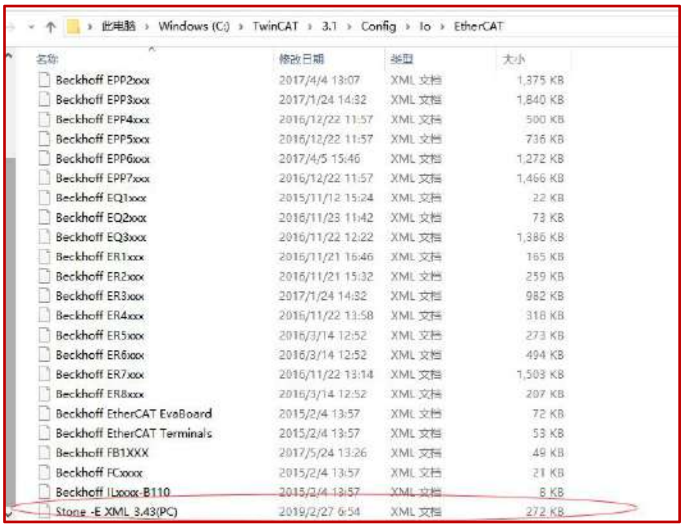


Figure 7-34 Directory of file Stone_E XML.xml

3. Establish project and connection

Run TwinCAT software, create a project, modify the IP address of the computer and the controller in the same local area network, and select the target system to be connected:

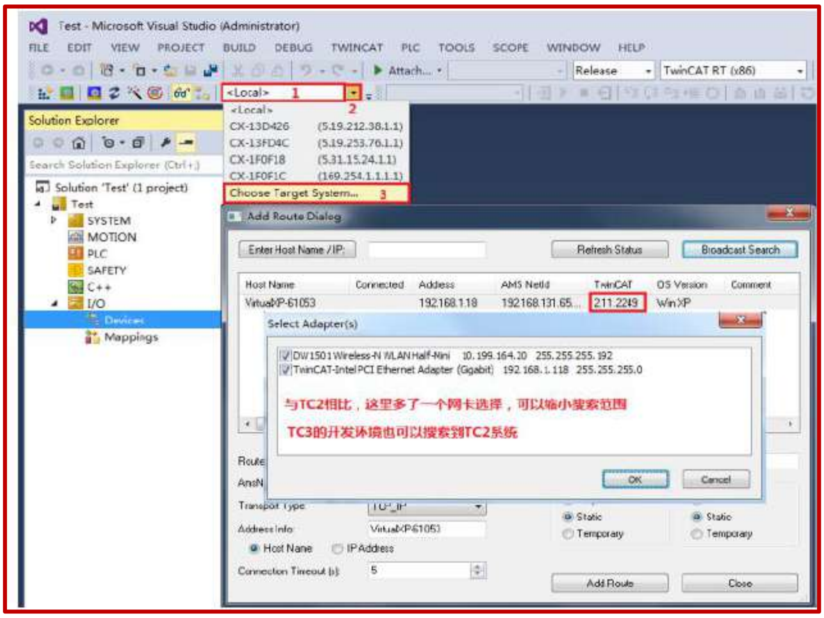


Figure 7-35 Modify IP address in TwinCAT

4. Scan the slave and automatically configure NC axis

Right click on “**I/O**”, select “**Scan**”, scan EtherCAT slave, click “**Scan boxes**” after scanning the slave, and click “**Automatically add NC axis**”.

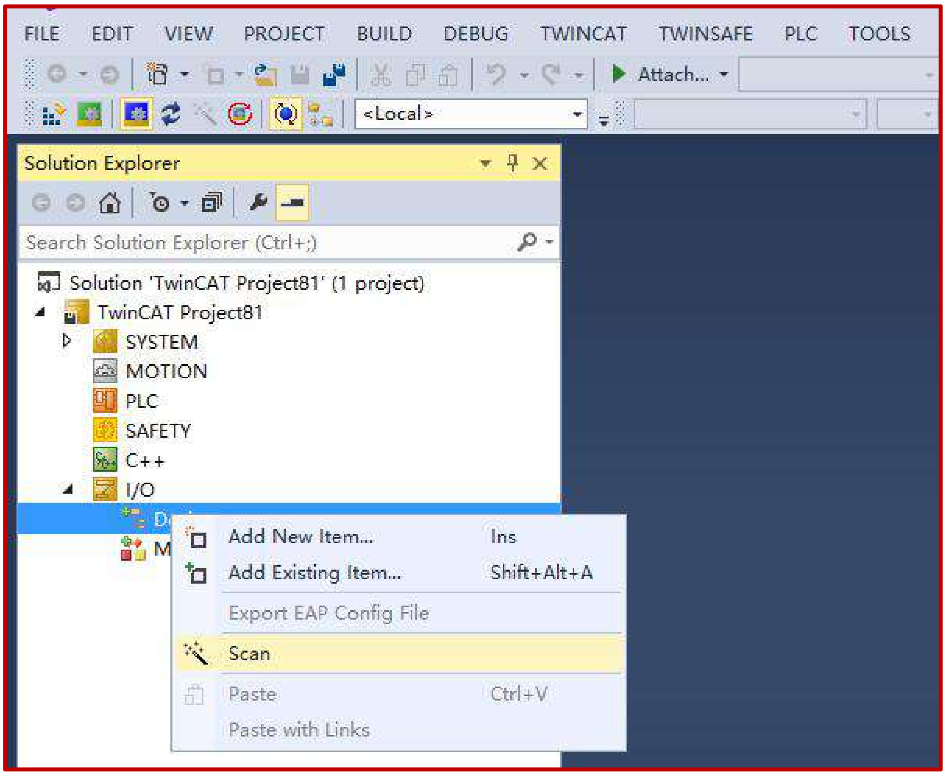


Figure 7-36 Automatic scanning and configuration of slaves in TwinCAT

After scanning is successful, icon StoneLOGO appears as shown in the below, and the servo status is in the OP state.

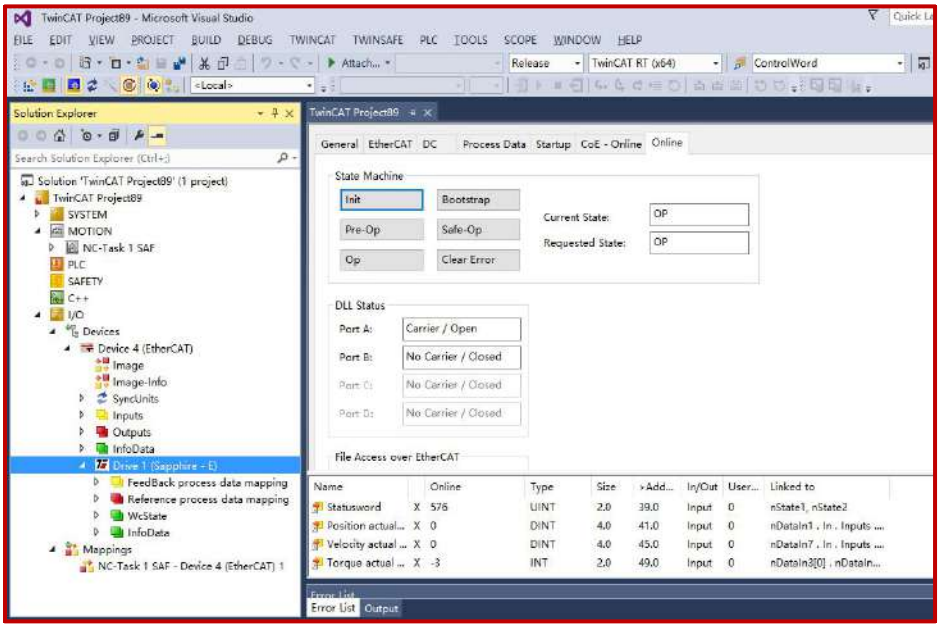


Figure 7-37 Successfully scan stone servo

Note: When scanning slave, please make sure TwinCat is in Config Mode.

5. Read COE data from Stone servo

As shown in Figure 7-39, you can read and write the data of the servo slave through SDO. Or you can call the COE command function module through the EtherCAT function library in the PLC program (Please refer to "TwinCat User Manual" for details).

Note: If the data is configured as PDO, write is invalid.

To check if read and write are successful, you can monitor and compare data through the SMC software.

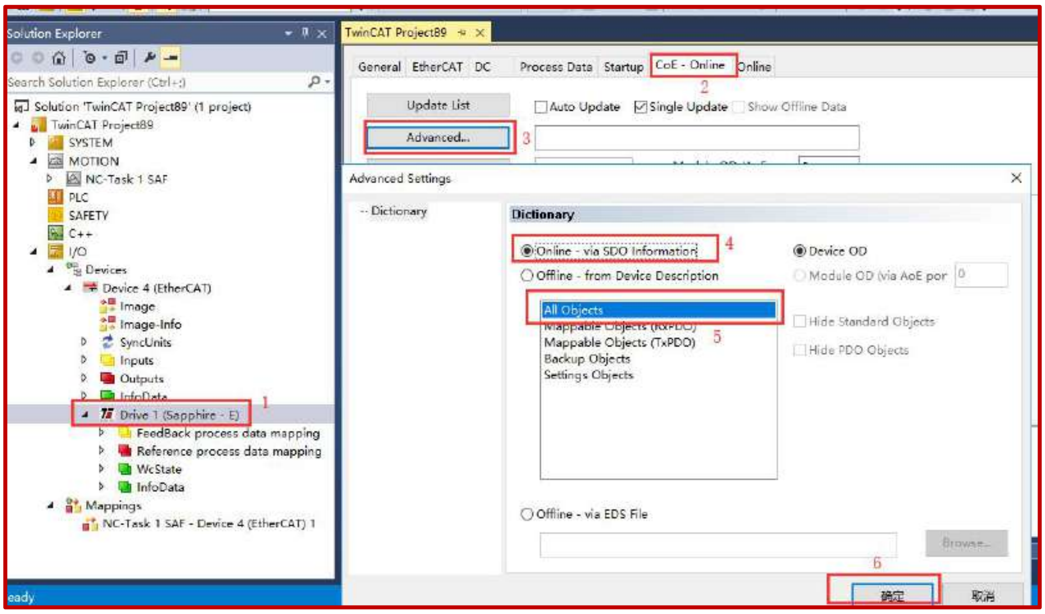


Figure 7-39 Configure SDO in TwinCAT

6. **Read and configure PDO mapping**

TwinCAT will automatically read the default PDO configuration of the lower computer when scanning the XML file of the slave. The default PDO mapping object and configuration are shown in Figure 7-40.

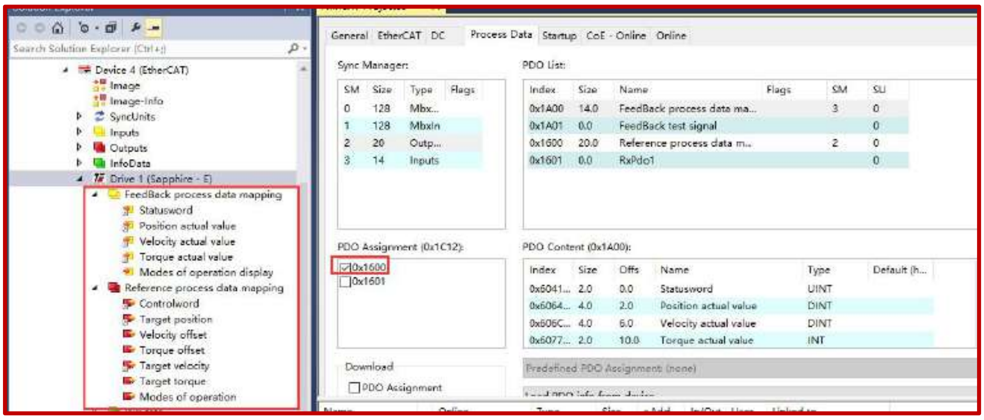


Figure 7-40 Configure PDO in TwinCAT

If the target PDO parameter is not in the default PDO configuration, you can add PDO parameter mapping through TWinCAT. For example, if you would like to add DI (0X2701) status when transferring PDO (1600):

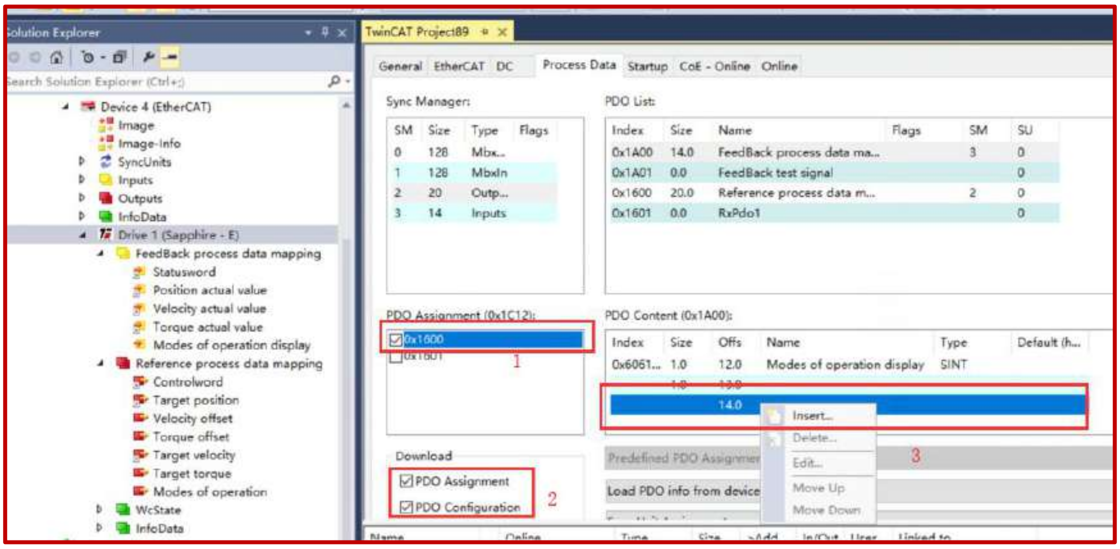


Figure 7-41 Add PDO parameters in TwinCAT

7. **Configure NC control**

a. Configure NC TASK cycle: set “Cycle ticks” in NC-Task 1 SAF to “4” (unit: ms). NC determines generation and calculation of position, velocity, acceleration and the direction.

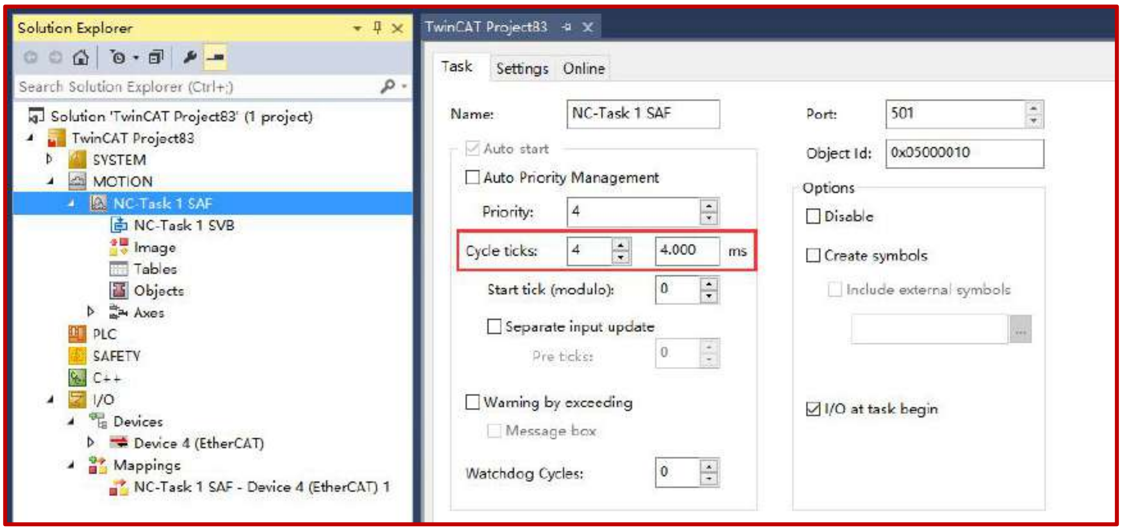


Figure 7-42 a. Configure NC TASK cycle in TwinCAT

b. Configure the synchronization clock: enable distributed clock.

Please note that the setting of Cycle Time should be the same with the synchronization period (4 ms) of the servo drive. Otherwise, the servo may vibrates during running.

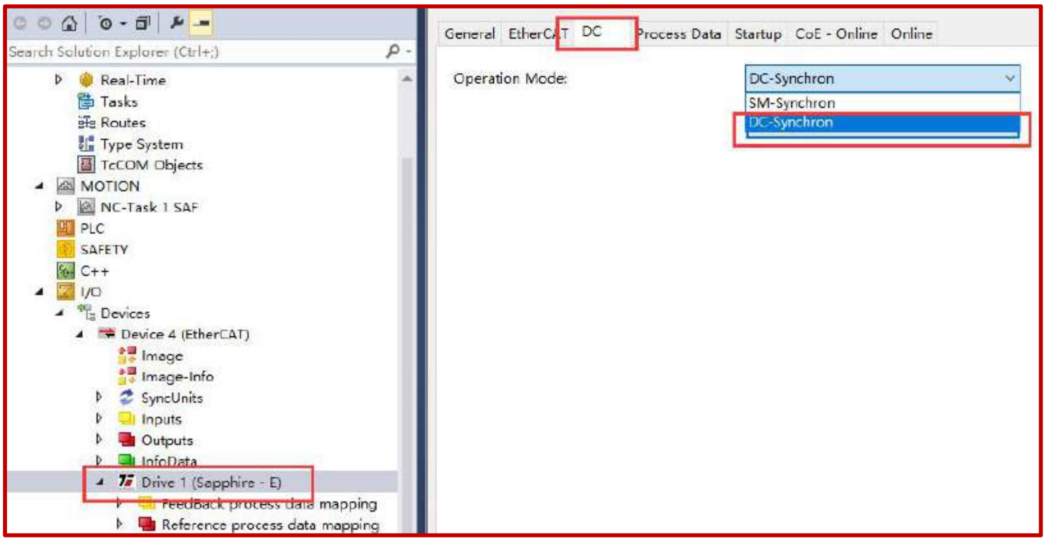


Figure 7-43 Check synchronization clock in TwinCAT

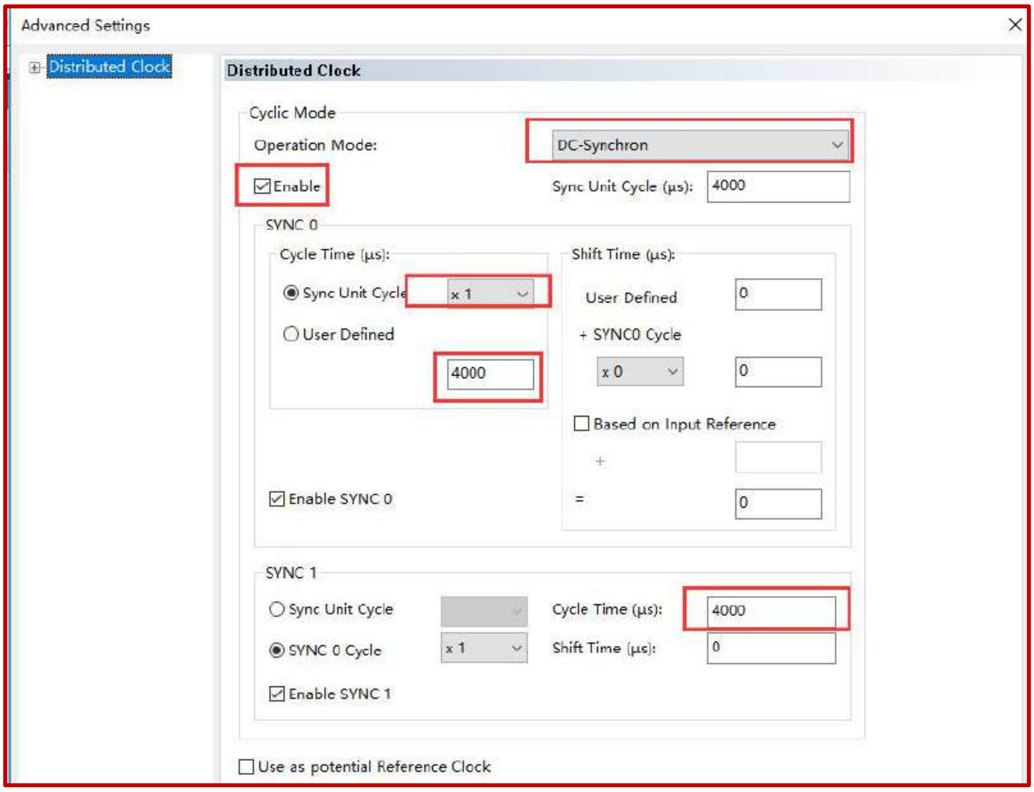


Figure 7-44 Set cycle time in TwinCAT

c. Set the following in the NC axis:

- Set Scaling Factor: the distance corresponding to the encoder pulse of each position feedback.

  e.g. If one turn of the servo motor is 10000 pulses, and each turn is 1 mm, the scaling factor should be set to 1/10000 = 0.0001 mm/Inc; if the target position increases by 10 mm, the actual servo position should increase by 100000 INC. Generally,

- Set the speed of NC control.

  Otherwise, an alarm will occur.

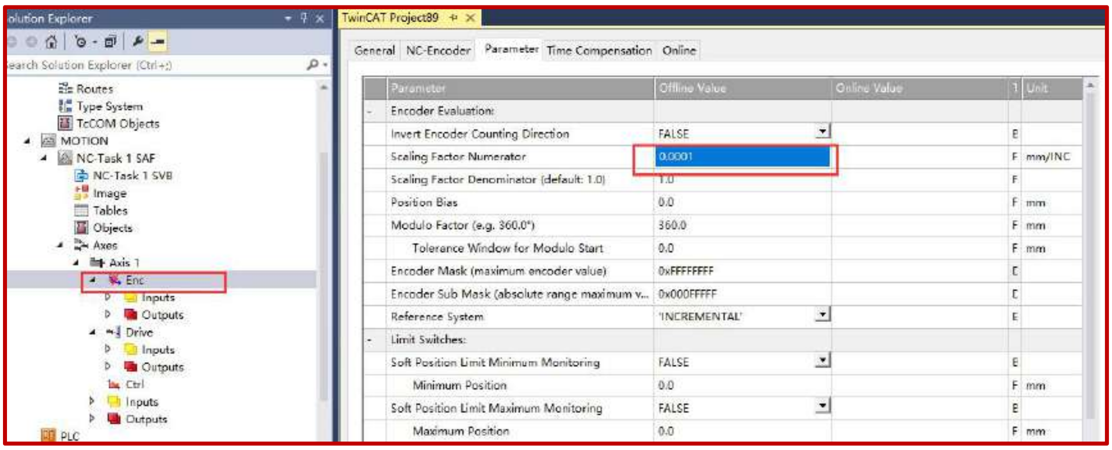


Figure 7-45 Set scaling factor in TwinCAT

d. To prevent the PLC from reporting a following error, set "**Following Error Calculation**" to "**Extern**".

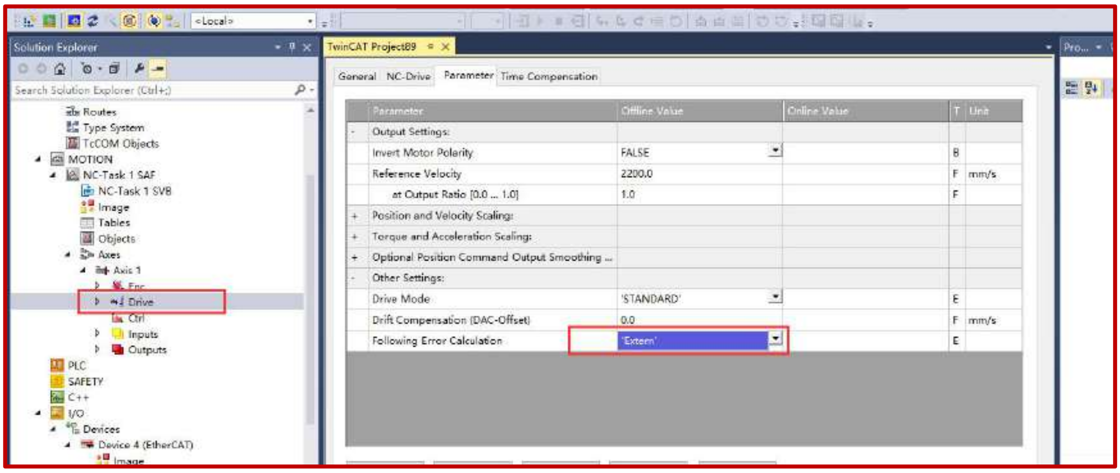


Figure 7-46 Set following error calculation in TwinCAT

e. To reverse the motor control polarity, set "**Invert Encoder Counting Directio**n" to "**TRUE**" and "**Invert Motor Polarity**" to "**TRUE**".

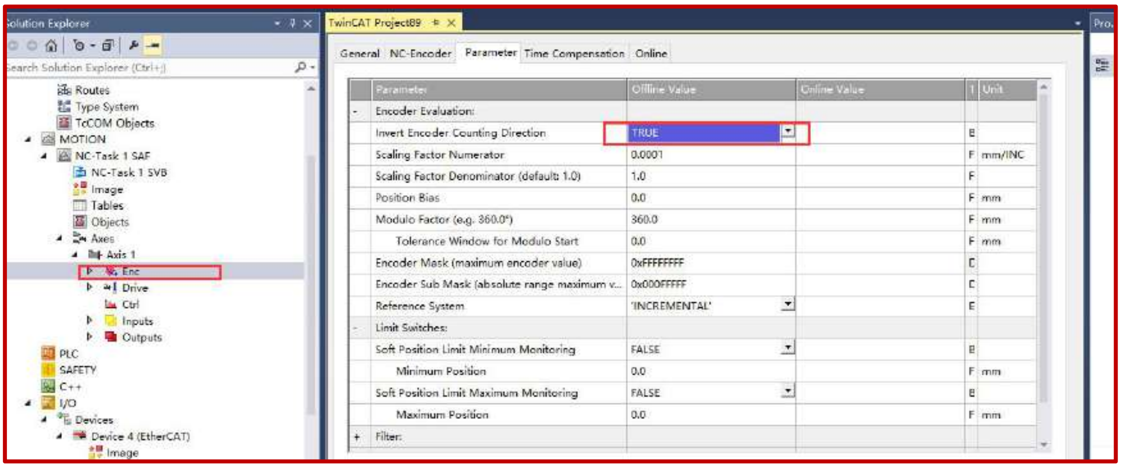


Figure 7-47 Set invert encoder counting direction in TwinCAT

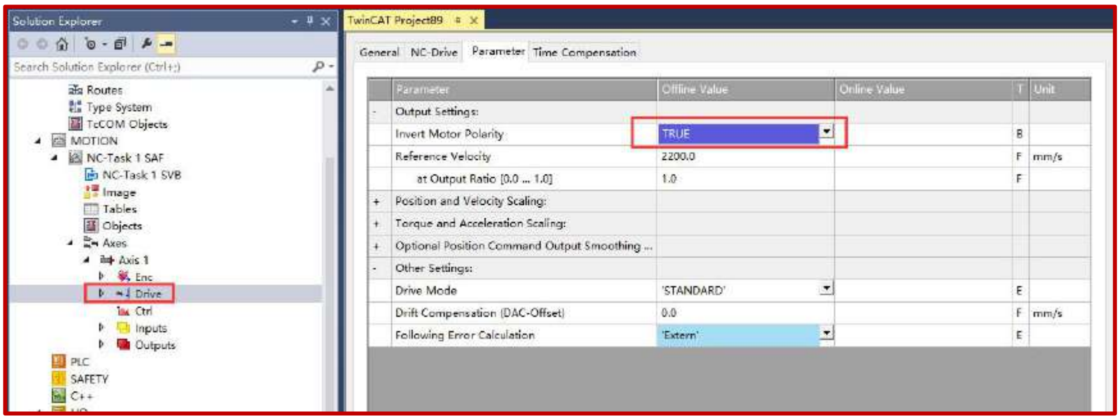


Figure 7-48 Set invert motor polarity in TwinCAT

f. Activate the configuration, control the servo working via the NC debugging interface, use Online function to simulate the servo working in the running mode (Make the servo lock the shaft, and click the button to make the servo run).

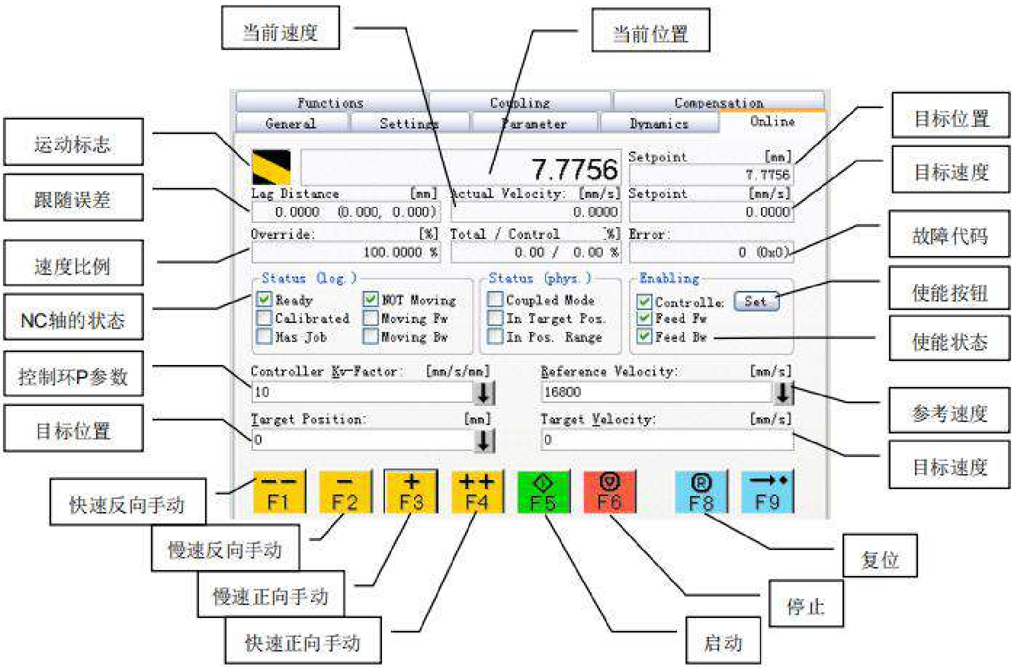


Figure 7-49 NC debugging interface in TwinCAT

当前速度：Current speed
当前位置：Current position
目标位置：Target position
目标速度：Target speed
故障代码：Error code
使能按钮：Enable button
使能状态：Enable status
参考速度：Reference speed
目标速度：Target speed
复位：Reset
停止：Stop
启动：Start
快速正向手动：Fast positive debugging
慢速正向手动：Slow positive debugging
慢速反向手动：Slow negative debugging
快速反向手动：Fast negative debugging
目标位置：Target position
控制环 P 参数：Control loop P parameter
NC 轴的状态：NC axis Status
速度比例：Speed rate
跟随误差：Follow-up error
运动标志：Movement sign

8. Create a PLC project

a. Create a new PLC project.

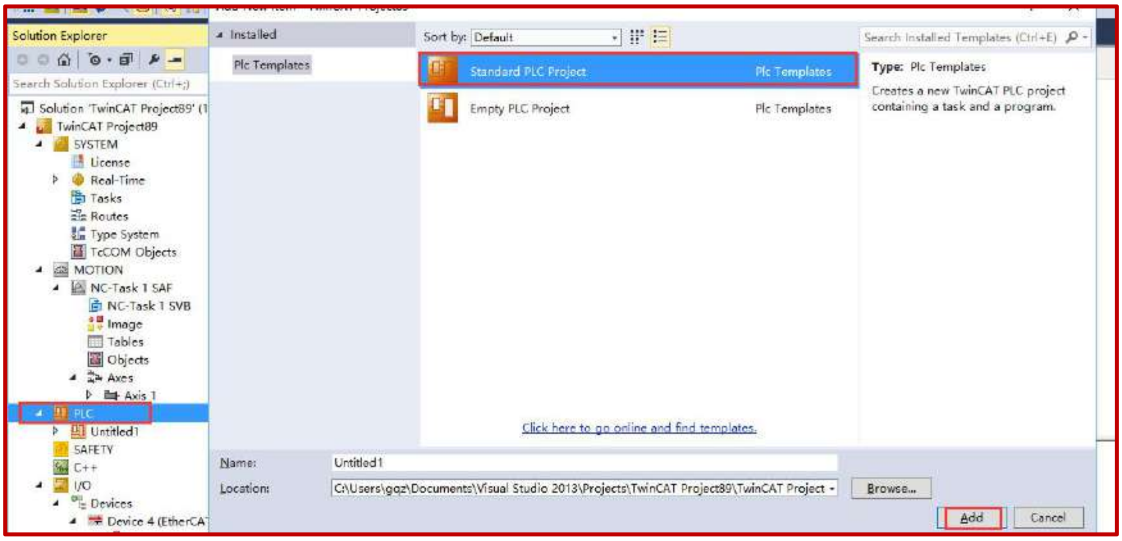


Figure 7-50 Create a new PLC project in TwinCAT

b. Set “**Cycle ticks**” of PLC Task to 4 ms.

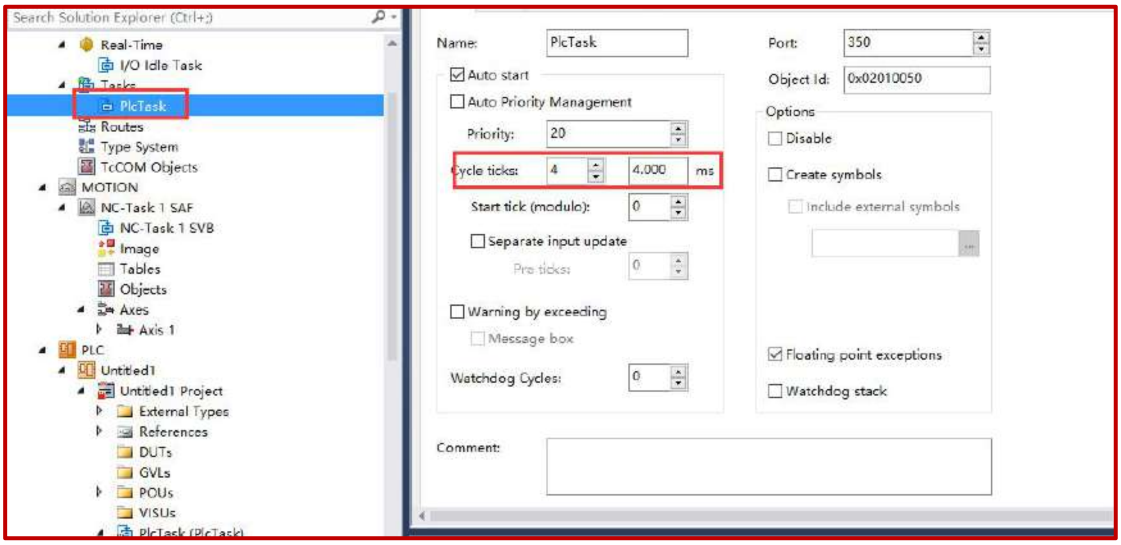


Figure 7-51 Set cycle ticks of PLC task in TwinCAT

9. Use of CoeSDO

CoeSDO is similar to SDO in CANOPEN. It can be used to read and write some objects whose exchange is not frequent or that are not supported by PDO communication. The steps are as follows:

a. Add "**Tc2_EtherCAT.lib**" in TwinCAT PLC library manager.

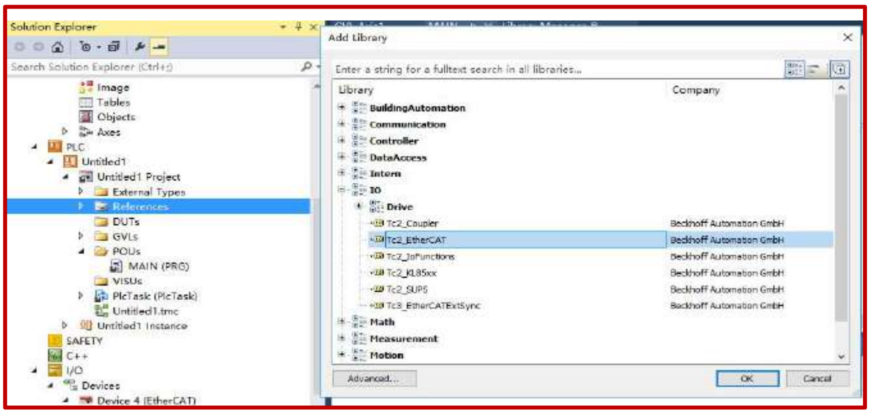


Figure 7-52 Add "Tc2_EtherCAT.lib" to the TwinCAT PLC library manager

b. After adding, declare the CoeSDO read and write in the program.

Taking read of status word 60410010 and write of homing mode 60980008 as examples, both of which have no symbols.

```
PROGRAM MAIN
VAR
    sNetId      : T_AmsNetId := '169.254.110.127.5.1';
    bExecute    : BOOL:=FALSE;
    bExe    : BOOL:=FALSE;
    nSlaveAddr : UINT := 1001;
    nIndex      : WORD := 16#6041;
    nSubIndex   : BYTE :=0;
    nIndex1      : WORD := 16#6098;
    nSubIndex1   : BYTE :=0;
    bError      : BOOL;
    nErrId      : UDINT;
    fbSdoRead   : FB_EcCoESdoRead;
    fbSdoWrite  : FB_EcCoESdoWrite;
    statuword: UINT;
    Homing_mode: INT:= 7;

END_VAR
```

Figure 7-53 Add CoeSDO function to program in TwinCAT

c. Set the T_AmsNetId of the EtherCAT master.

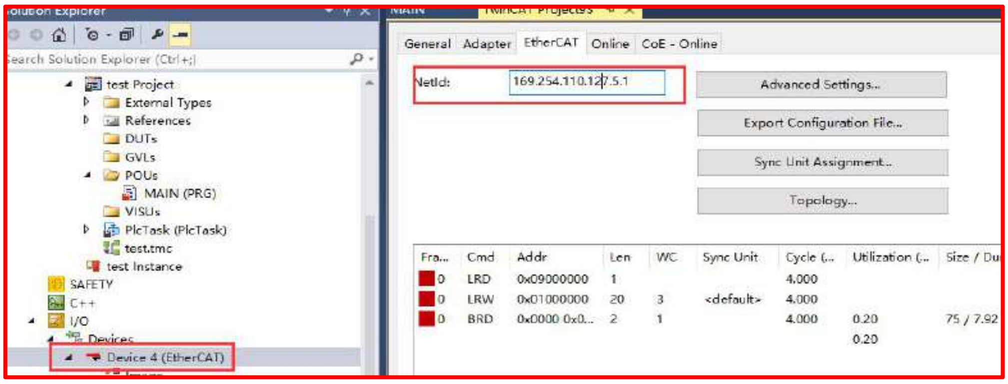


Figure 7-54 Set T_AmsNetId of EtherCAT master in TwinCAT

d. Set the slave's address SlaveAddr.

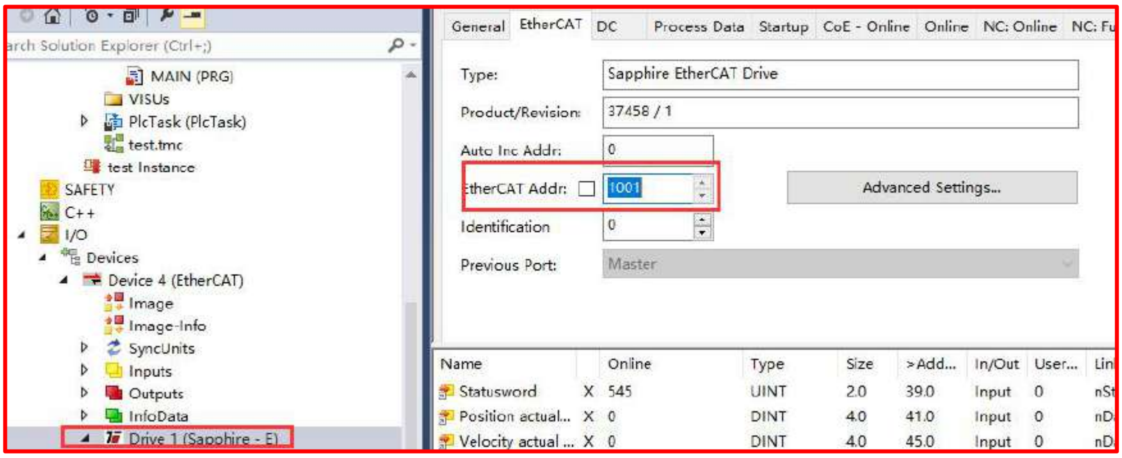


Figure 7-55 Set slave's address SlaveAddr in TwinCAT

e. Call the read and write function in the program, e.g. trigger read 0X6041 status word as 545, and write homing mode 0X6098 as 7.

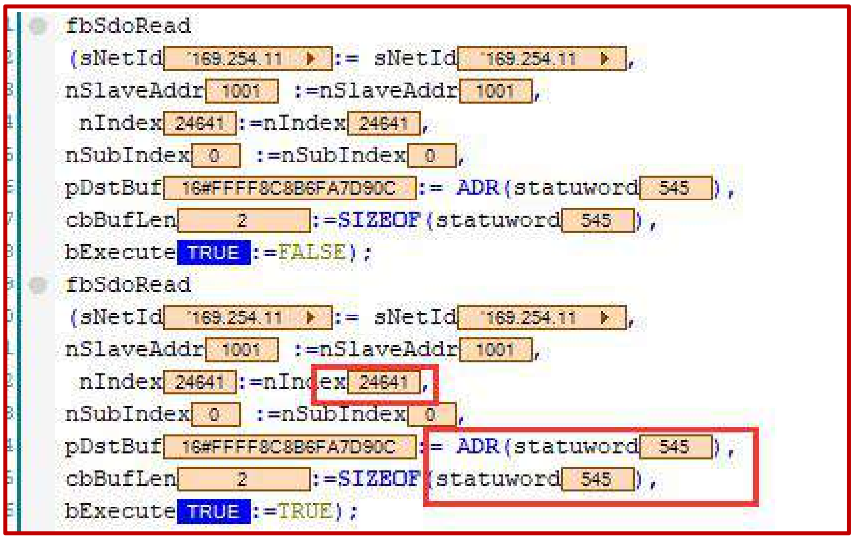


Figure 7-56 Trigger read of 0X6041 as 545 in TwinCAT

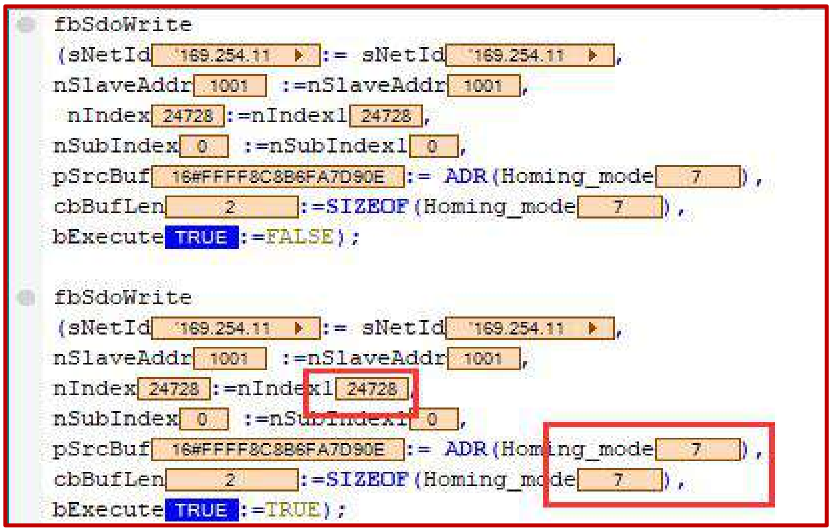


Figure 7-57 Trigger write of homing mode 0X6098 as 7 in TwinCAT

7.2.4 Communication with Omron PLC

1. Connect hardware

Please refer to 7.22 for the hardware connection diagram.

2. Establish communication

a. Communication between PC and PLC

The PC and PLC are connected in a 1:1 manner with no need to specify an IP address and a connected device.

Click "Online" in Sysmac Studio to judge whether the communication is on by observing the status of the controller:

- ✓ The first green light indicates that communication between PC and PLC is established.
- ✓ The second green light indicates that the communication between PLC and the servo drive is established.

Figure 7-59 Controller status

b. Communication between PLC and the servo drive

Import an XML file

Right click “**EtherCAT**” under Sysmac Studio “Configuration and Settings, and click “**Show ESI Library”**, place file Store Stone -E XML_3.43 (PC).XML in this path.

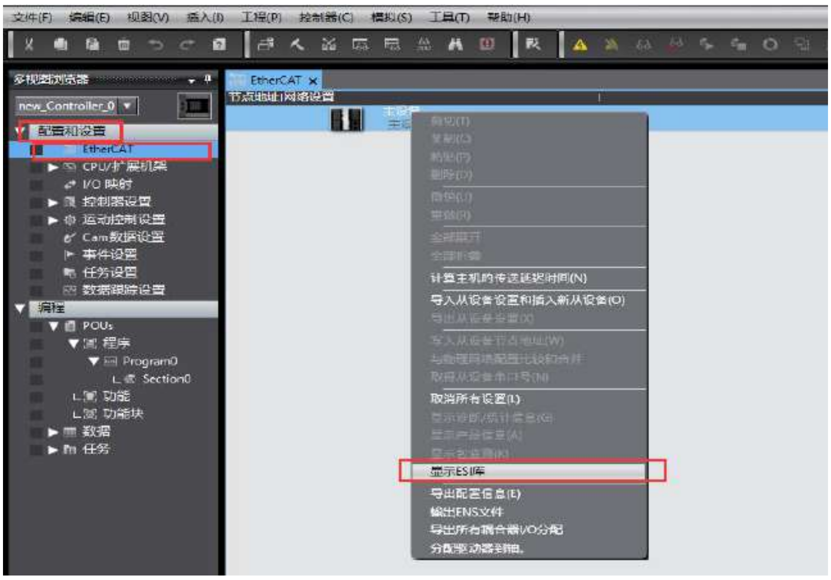


Figure 7-60 Import an XML file

Assign node

1) Create a project in Sysmac Studio, and log in online after compiling.
   The default node address of the servo drive is 0.
   Before establishing communication or when the device is replaced to merge the physical network and the application, the following warning shows, and node assignment is required.

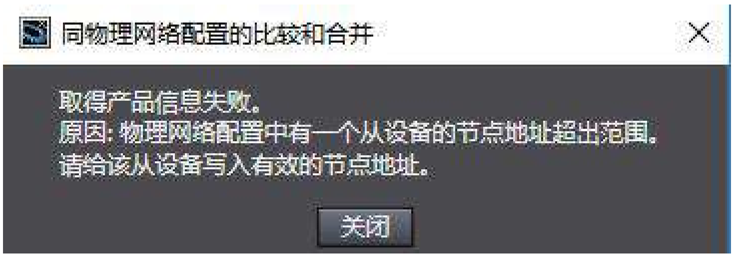


Figure 7-61 Node address out of range

2) In the left tree window, select "**Configuration and Settings**-**EtherCAT"**, right click the master, select "**Write slave node address"**, modify it to an address greater than 1 in the following interface, press "**Enter**" to write, Press "**Write**" after writing**,** and restart the servo drive.
   The node address of different servo drives cannot be the same.

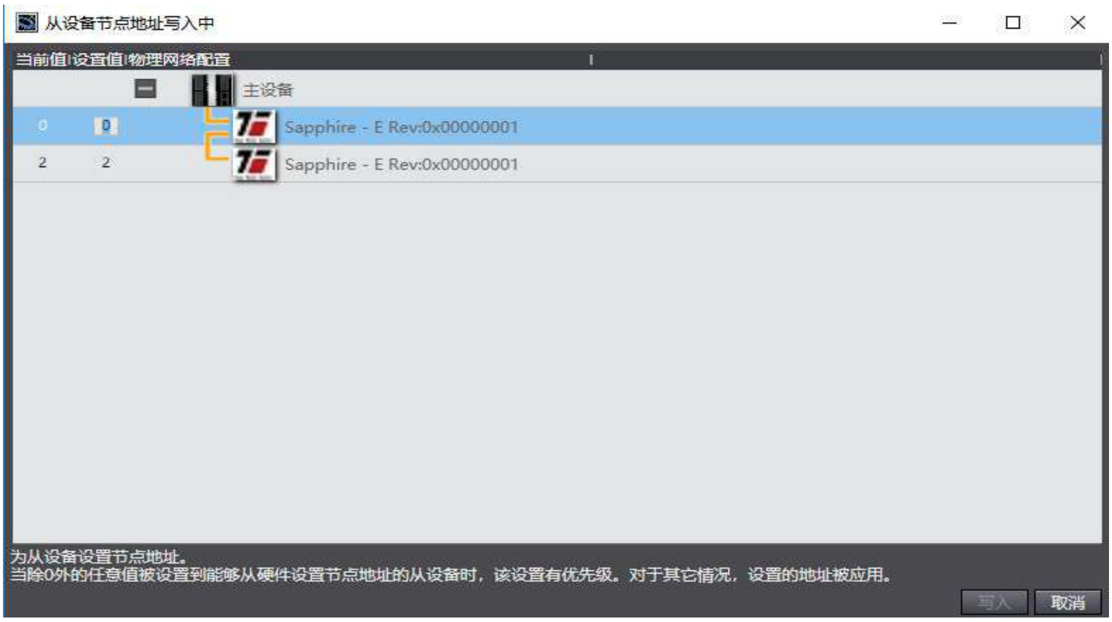


Figure 7-62 Modify node address

Establish communication

1) In the left tree window, select "**Configuration and Settings-EtherCAT"**, right click the master, and select "**Compare and Merge with Physical Network Configuration**".

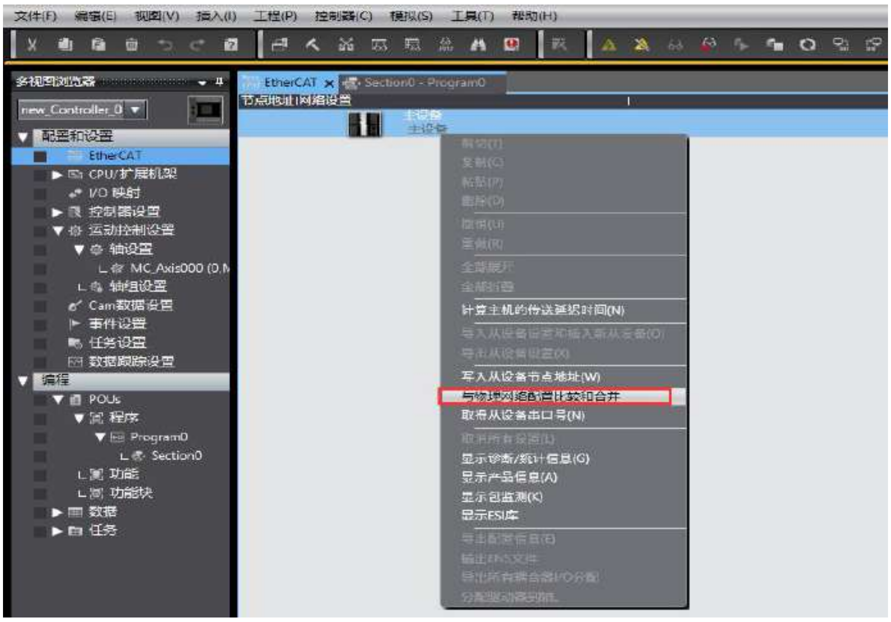


Figure 7-63 Compare and merge interface with physical network configuration

2) Click "**Apply physical network configuration**" in the pop-up to complete the communication between the PLC and the servo drive.

3) Click  to download the program to the PLC.

The two yellow lights of the servo network port are on and the green lights are flashing, indicating that the communication between the PLC and the drive is successful.

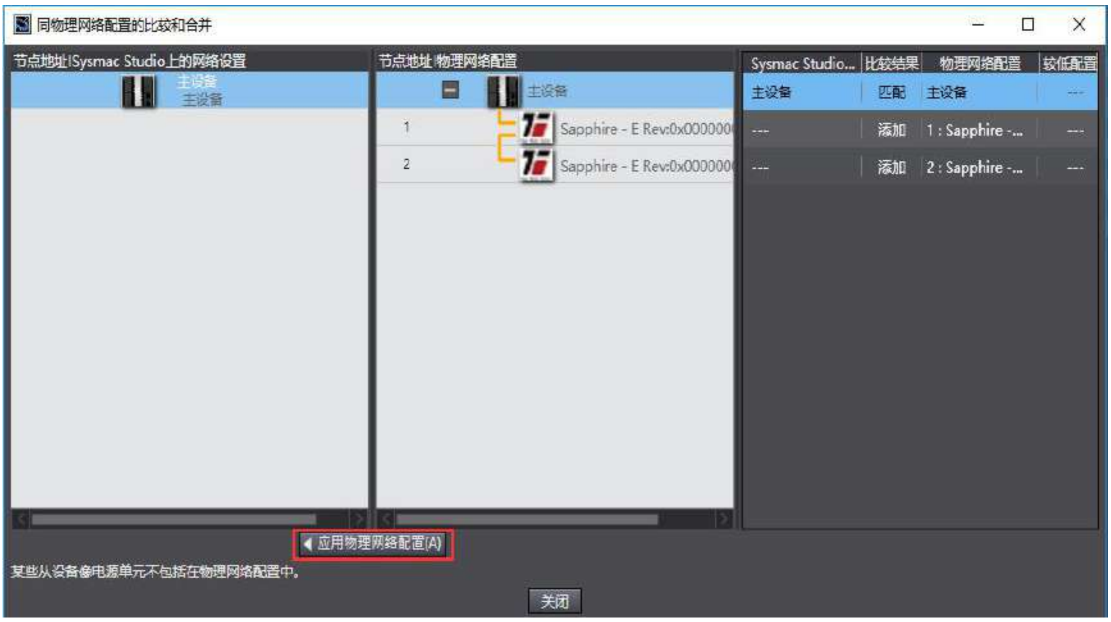


Figure 7-64 Apply physical network configuration

3. Configure communication

a. Configure the servo drive

In the left tree window, select "**Configuration and Settings-EtherCAT**", click "Drive", and configure the servo drive.

Settings of PDO mapping and distributed clock settings are required.

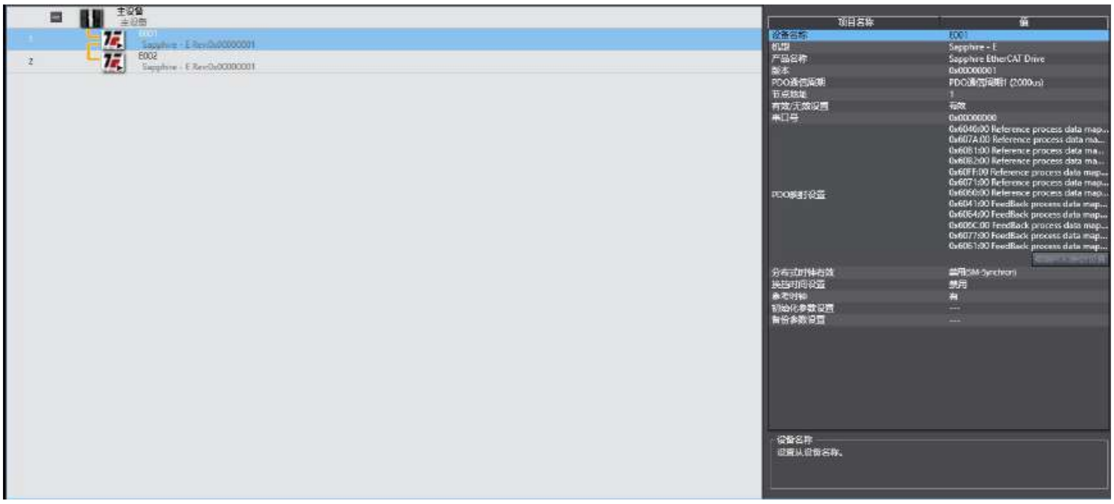


Figure 7-65 Modify PDO and distributed clock

PDO mapping: use default settings.

Distributed clock: please enable DC distributed clock. Otherwise the program cannot run normally.

b. Configure axis

To add a control axis, in menu "**Motion Control Settings**", right click "**Axis Settings**" → "**Add**" → "**Motion Control Axis**".

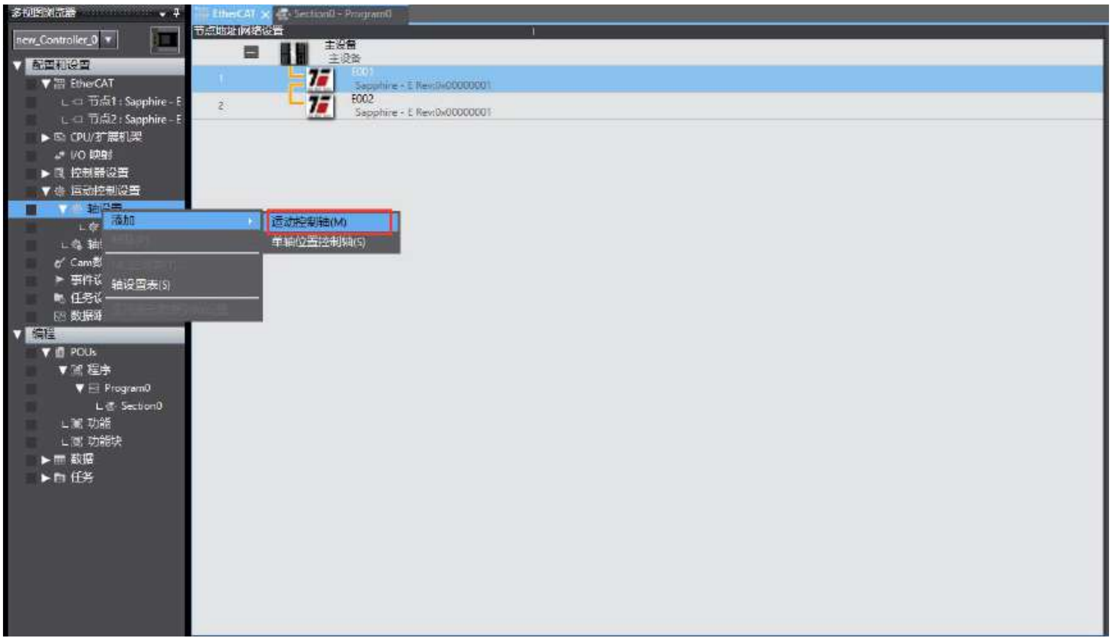


Figure 7-66 Add a motion control axis

Basic settings of axis

✓ Axis Use: the axis to use.

✓ Axis type: servo axis.

✓ Output device 1: to select the servo drive of the corresponding node.

✓ Detailed settings: set PDO for axes. Each axis configures the corresponding PDO according to the used function. If there is a digital input, please set it.

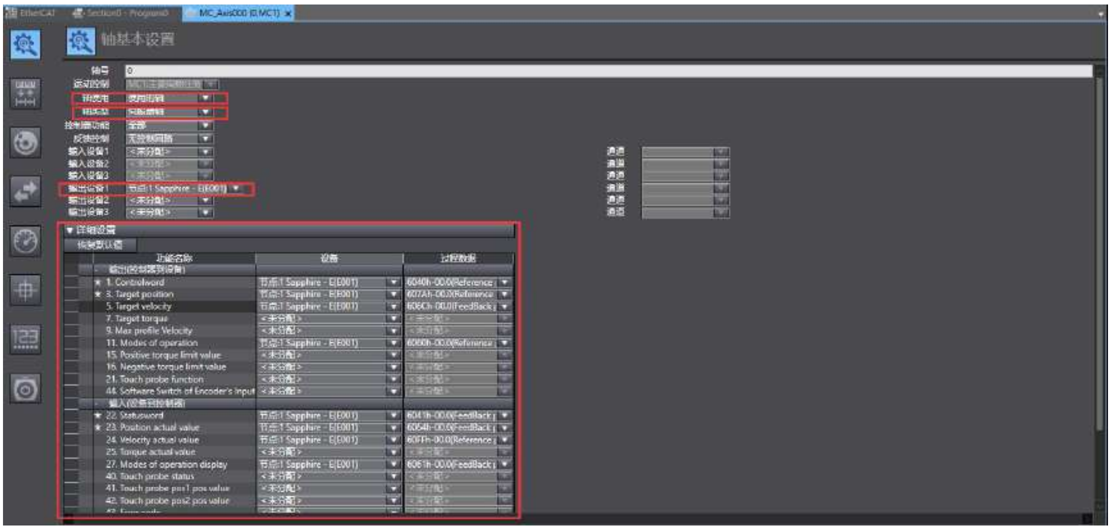


Figure 7-67 Basic axis settings

Unit settings

Unit: set the required unit according to actual needs.

Stroke distance:

- The number of command pulses in one revolution of the motor: please set it as the resolution of the encoder.
- To determine whether to use the gearbox according to whether there is a reducer.
- The stroke of the motor for one revolution: please set it according to the actual load movement distance for one revolution of the motor.

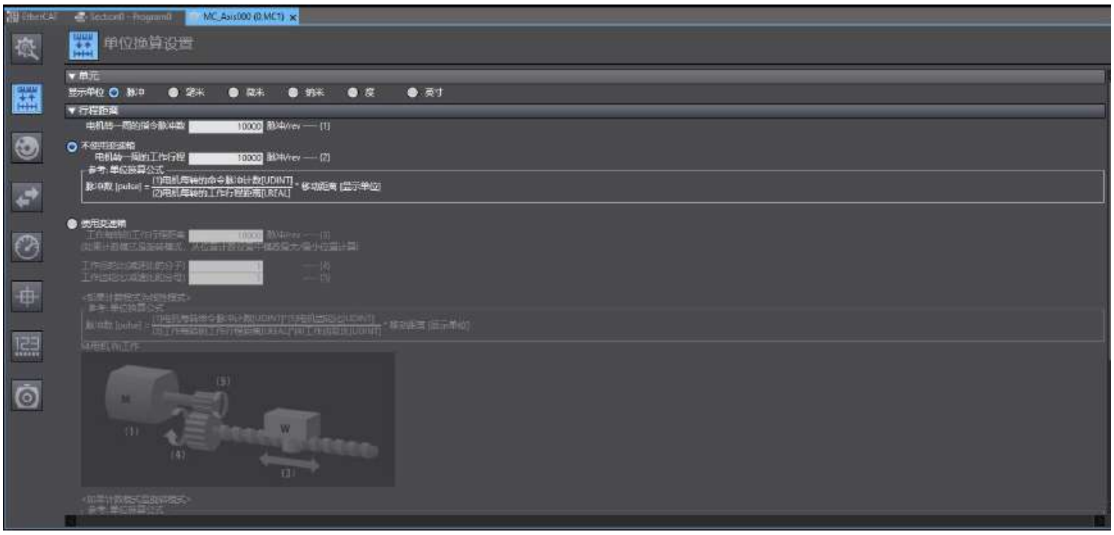


Figure 7-68 Set unit

Homing setting

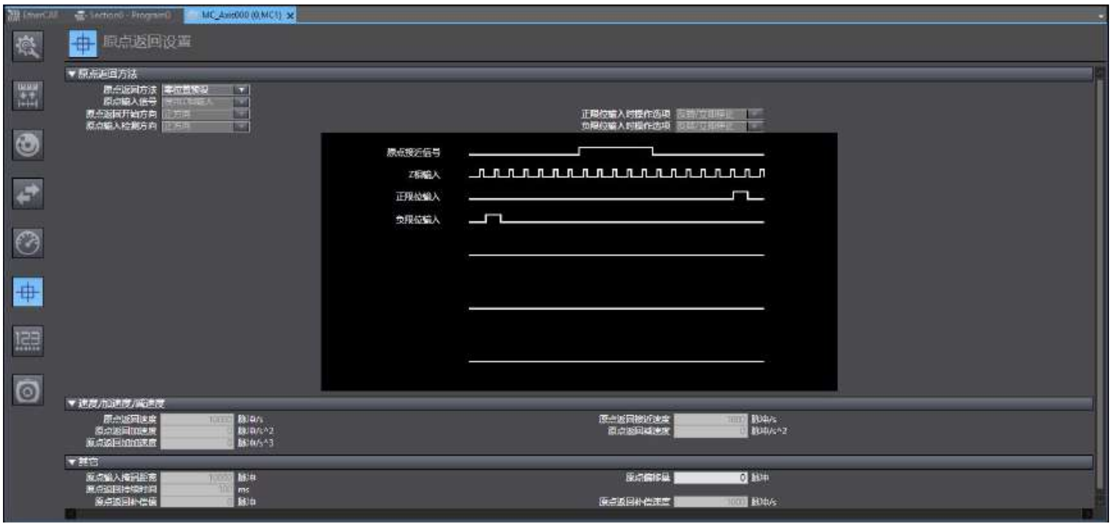


Figure 7-69 Zero return setting

Please select an appropriate homing method as required.

If you choose to use the external origin input, please connect the origin input signal to the digital input of the servo, and configure it in "**Axis-basic settings**" → "**Details**" → "**Digital Input**".

Figure 7-70 Configure digital input

- Home switch: Home approaching signal of servo input.
- Positive limit switch: positive limit signal of servo input
- Negative limit switch: negative limit signal of servo input

Other settings

Note: it is the unit of the actuator. Units are important and set in the unit settings.

- Set the maximum acceleration.

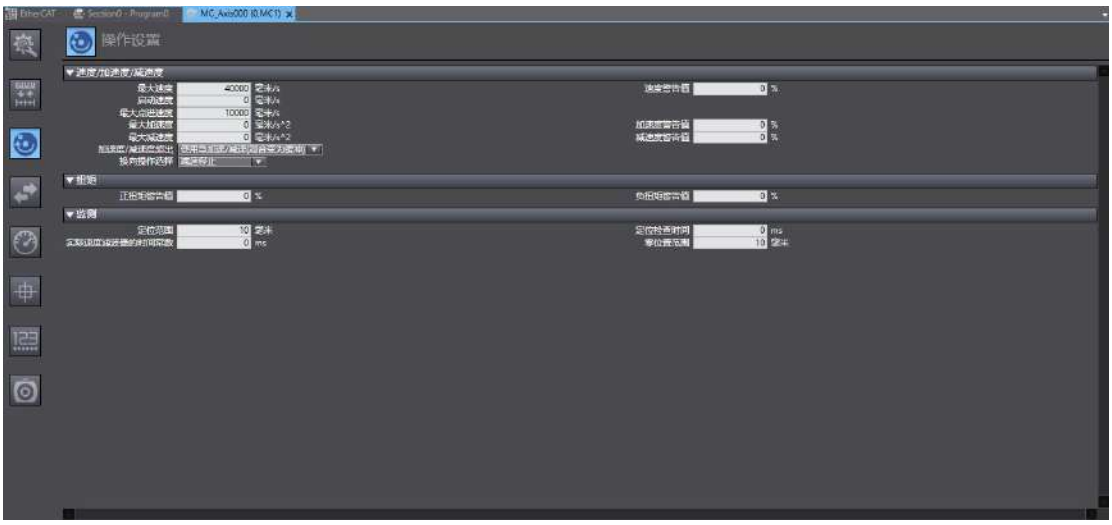


Figure 7-71 Set the maximum acceleration

- Set the maximum torque

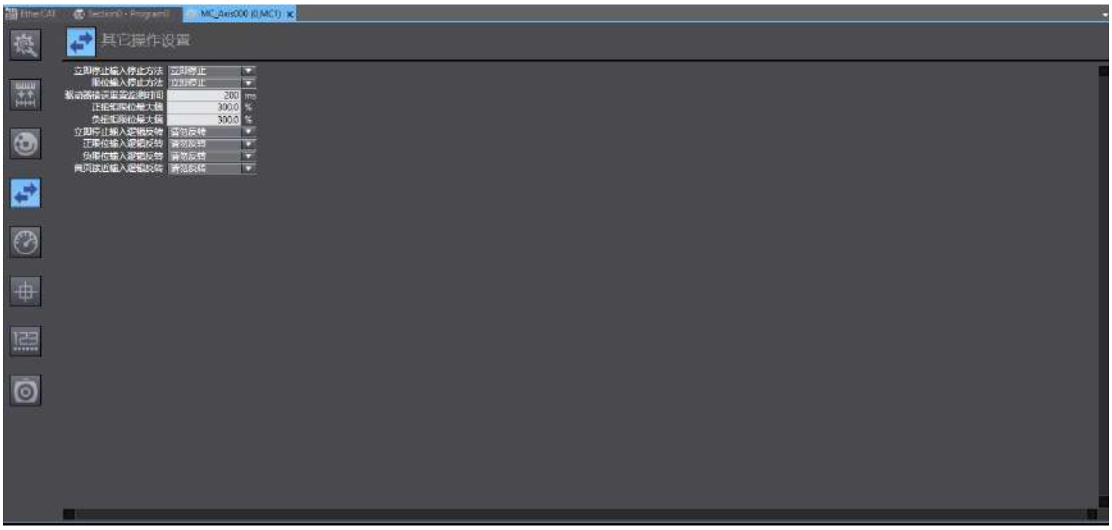


Figure 7-72 Set the maximum torque

- Set other settings.

c. Set task cycle to 4 ms (default value)

It should be the same with synchronization cycle of the lower computer.

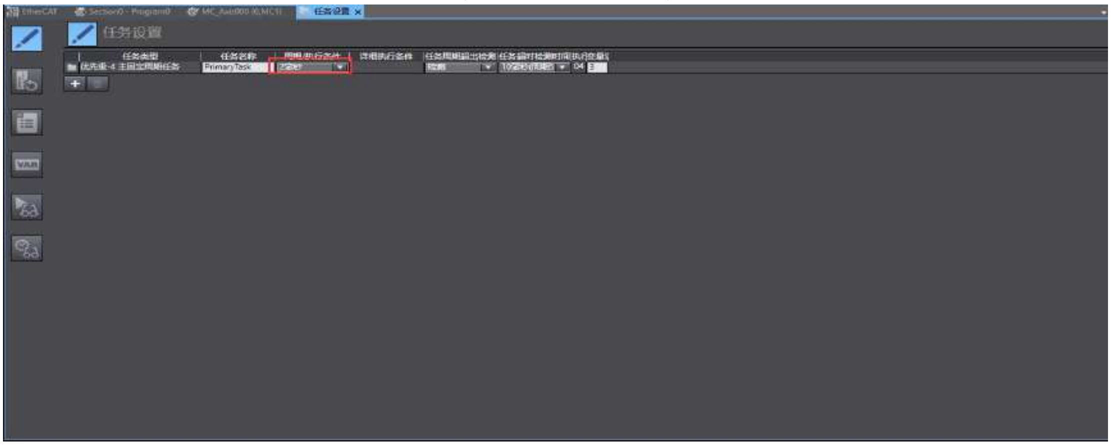


Figure 7-72 Set task cycle

ISMC (China)

Addr: Room 0108, Block A, No. 418, Guiping Road, Xuhui District, Shanghai, China
Hotline: 021-64030375
Email: sales@istomemc.com
Web: www.istomemc.com