INOVANCE

Application Guide - CANopen Communication

SV660C Servo Drive

Application Guide

A02
Data code 19011302

Preface

Thank you for purchasing SV660C series servo drives equipped with CANopen fieldbus function.

The SV660C series servo drive is integrated with CANopen communication function, which covers all the products of this series. The CANopen communication function enables the SV660C series servo drive to be connected to the high-speed CANopen communication network to achieve fieldbus control.

This user guide presents CANopen-related applications only. For other general functions, see *SV660P Series Servo Drive Advanced User Guide*. Contact our technical support if you have any question concerning use of the CANopen function.

Documents provided by Inovance are subject to change without notice due to continuous product improvement.

Safety Instructions
◆ The drawings in the user guide are sometimes shown without covers or protective guards. Remember to install the covers or protective guards as specified first, and then perform operations in accordance with the instructions described in the user guide. ◆ The drawings in the user guide are shown for descriptions only and may not match the product you purchased. ◆ This user guide is subject to change without notice due to product upgrade, specification modifications as well as efforts to improve the accuracy and convenience of the user guide. ◆ If the user guide is damaged or lost, contact our regional agents or customer service centers to order the user guide. ◆ Contact our customer service centers for concerns during use.

Unpacking Inspection

Check the following items upon unpacking.

Items	Description
Check whether the delivered product complies with your order.	Check whether the product model and specifications shown on the packing box comply with your order.
Check whether the product is intact.	Check whether the overall appearance of the product is intact. If there is any part missing or damaged, contact Inovance or your supplier immediately.

Revision History

Date	Version	Description
April 2020	A00	First release
August 2020	A01	Deleted the service hotline.
September 2020	A02	◆ Improved contents in section 2.3. ◆ Deleted descriptions for units in section 4.2.

Safety Instructions

Safety Precautions

1) Before installing, using, and maintaining this equipment, read the safety information and precautions thoroughly, and comply with them during operations.
2) To ensure the safety of humans and equipment, follow the signs on the equipment and all the safety instructions in this user guide.
3) "CAUTION", "WARNING", and "DANGER" items in the user guide do not indicate all safety precautions that need to be followed; instead, they just supplement the safety precautions.
4) Use this equipment according to the designated environment requirements. Damage caused by improper usage is not covered by warranty.
5) Inovance shall take no responsibility for any personal injuries or property damage caused by improper usage.

Safety Levels and Definitions

	Indicates that failure to comply with the notice will result in severe personal injuries or even death.
	Indicates that failure to comply with the notice may result in severe personal injuries or even death.
	Indicates that failure to comply with the notice may result in minor or moderate personal injuries or equipment damage.

Safety Instructions

Unpacking

◆ Check whether the packing is intact and whether there is damage, water seepage, damp, and deformation. ◆ Unpack the package by following the package sequence. Do not hit the package with force. ◆ Check whether there are damage, rust, or injuries on the surface of the equipment or equipment accessories. ◆ Check whether the number of packing materials is consistent with the packing list.

◆ Do not install the equipment if you find damage, rust, or indications of use on the equipment or accessories. ◆ Do not install the equipment if you find water seepage, component missing or damage upon unpacking. ◆ Do not install the equipment if you find the packing list does not conform to the equipment you received.

Storage and Transportation
CAUTION ◆ Store and transport this equipment based on the storage and transportation requirements for humidity and temperature. ◆ Avoid transporting the equipment in environments such as water splashing, rain, direct sunlight, strong electric field, strong magnetic field, and strong vibration. ◆ Avoid storing this equipment for more than three months. Long-term storage requires stricter protection and necessary inspections. ◆ Pack the equipment strictly before transportation. Use a sealed box for long-distance transportation. ◆ Never transport this equipment with other equipment or materials that may harm or have negative impacts on this equipment.
WARNING ◆ Use professional loading and unloading equipment to carry large-scale or heavy equipment. ◆ When carrying this equipment with bare hands, hold the equipment casing firmly with care to prevent parts falling. Failure to comply may result in personal injuries. ◆ Handle the equipment with care during transportation and mind your step to prevent personal injuries or equipment damage. ◆ Never stand or stay below the equipment when the equipment is lifted by hoisting equipment.
Installation
WARNING ◆ Thoroughly read the safety instructions and user guide before installation. ◆ Do not modify this equipment. ◆ Do not rotate the equipment components or loosen fixed bolts (especially those marked in red) on equipment components. ◆ Do not install this equipment in places with strong electric or magnetic fields. ◆ When this equipment is installed in a cabinet or final equipment, protection measures such as a fireproof enclosure, electrical enclosure, or mechanical enclosure must be provided. The IP rating must meet IEC standards and local laws and regulations.
DANGER ◆ Equipment installation, wiring, maintenance, inspection, or parts replacement must be performed by only professionals. ◆ Installation, wiring, maintenance, inspection, or parts replacement must be performed by only experienced personnel who have been trained with necessary electrical information. ◆ Installation personnel must be familiar with equipment installation requirements and relevant technical materials. ◆ Before installing equipment with strong electromagnetic interference, such as a transformer, install an electromagnetic shielding device for this equipment to prevent malfunctions.

Wiring
 DANGER ◆ Equipment installation, wiring, maintenance, inspection, or parts replacement must be performed by only professionals. ◆ Never perform wiring at power-on. Failure to comply will result in an electric shock. ◆ Before wiring, cut off all equipment power supplies. Wait at least 15 minutes before further operations because residual voltage exists after power-off. ◆ Make sure that the equipment is well grounded. Failure to comply will result in an electric shock. ◆ During wiring, follow the proper electrostatic discharge (ESD) procedures, and wear an antistatic wrist strap. Failure to comply will result in damage to internal equipment circuits.
 WARNING ◆ Never connect the power cable to output terminals of the equipment. Failure to comply may cause equipment damage or even a fire. ◆ When connecting a drive with the motor, make sure that the phase sequences of the drive and motor terminals are consistent to prevent reverse motor rotation. ◆ Wiring cables must meet diameter and shielding requirements. The shielding layer of the shielded cable must be reliably grounded at one end. ◆ After wiring, make sure that no screws are fallen and cables are exposed in the equipment.
Power-on
 DANGER ◆ Before power-on, make sure that the equipment is installed properly with reliable wiring and the motor can be restarted. ◆ Before power-on, make sure that the power supply meets equipment requirements to prevent equipment damage or even a fire. ◆ At power-on, unexpected operations may be triggered on the equipment. Therefore, stay away from the equipment. ◆ After power-on, do not open the cabinet door and protective cover of the equipment. Failure to comply will result in an electric shock. ◆ Do not touch any wiring terminals at power-on. Failure to comply will result in an electric shock. ◆ Do not remove any part of the equipment at power-on. Failure to comply will result in an electric shock.
Operation
 DANGER ◆ Do not touch any wiring terminals during operation. Failure to comply will result in an electric shock. ◆ Do not remove any part of the equipment during operation. Failure to comply will result in an electric shock. ◆ Do not touch the equipment shell, fan, or resistor for temperature detection. Failure to comply will result in heat injuries. ◆ Signal detection must be performed by only professionals during operation. Failure to comply will result in personal injuries or equipment damage.

- Prevent metal or other objects from falling into the device during operation. Failure to comply may result in equipment damage.
- Do not start or stop the equipment using the contactor. Failure to comply may result in equipment damage.

Maintenance

- Equipment installation, wiring, maintenance, inspection, or parts replacement must be performed by only professionals.
- Do not maintain the equipment at power-on. Failure to comply will result in an electric shock.
- Before maintenance, cut off all equipment power supplies and wait at least 15 minutes.

- Perform daily and periodic inspection and maintenance for the equipment according to maintenance requirements and keep a maintenance record.

Repair

- Equipment installation, wiring, maintenance, inspection, or parts replacement must be performed by only professionals.
- Do not repair the equipment at power-on. Failure to comply will result in an electric shock.
- Before inspection and repair, cut off all equipment power supplies and wait at least 15 minutes.

- Require for repair services according to the product warranty agreement.
- When the equipment is faulty or damaged, require professionals to perform troubleshooting and repair by following repair instructions and keep a repair record.
- Replace quick-wear parts of the equipment according to the replacement guide.
- Do not operate damaged equipment. Failure to comply may result in worse damage.
- After the equipment is replaced, perform wiring inspection and parameter settings again.

Disposal

- Dispose of retired equipment by following local regulations or standards. Failure to comply may result in property damage, personal injuries, or even death.
- Recycle retired equipment by following industry waste disposal standards to avoid environmental pollution.

Safety Signs

■ Description of safety signs in the user guide


Read the user guide before installation and operation.


Reliably ground the system and equipment.


Danger!


High temperature!


Prevent personal injuries caused by machines.


High voltage!

Wait 15 minutes before further operations.

■ Description of safety signs on the equipment

For safe equipment operation and maintenance, comply with safety signs on the equipment, and do not damage or remove the safety labels. The following table describes the safety signs.

Safety Sign	Description
	Never fail to connect the Protective Earth (PE) terminal. Read the user guide and follow the safety instructions before use. To prevent the risk of electric shock, do not touch terminals within 15 minutes after cutting off the power supply. To prevent the risk of burning, do not touch the heatsink when the power supply is ON.

Contents

1 Product Information

1.1 Nameplate and Model Number

SV 660 C S 2R8 I

Code	Product Family
SV	Servo

Code	Product Series
660	660 series

Code	Product Type
P	Pulse type
A	CANlink
C	CANopen

Code	Voltage Class
S	220 V
T	380 V

Code	Installation Mode
I	Baseplate installation (standard)

Code	Rated Current
1R6	1.6 A
2R8	2.8 A
3R5	3.5 A
5R4	5.4 A
5R5	5.5 A
7R6	7.6 A
8R4	8.4 A
012	12 A
017	17 A
021	21 A
026	26 A

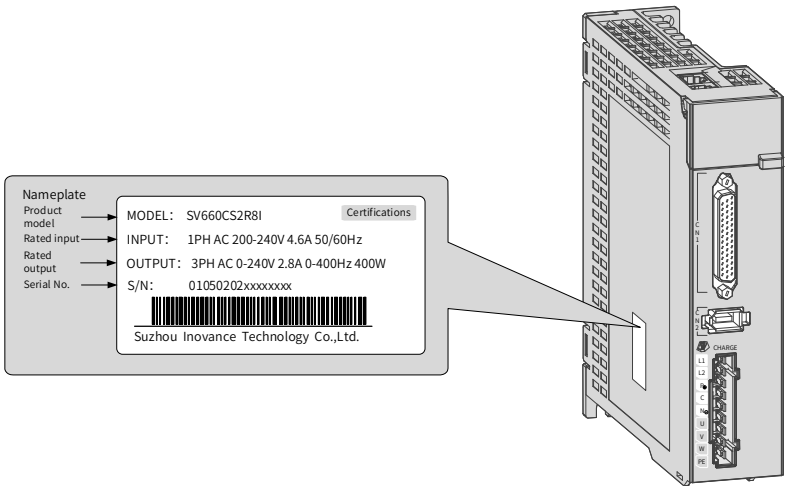


Figure 1-1 Servo drive nameplate and model number

1.2 Comprehensive Performance Parameters

Table 1-1 Comprehensive performance parameters

Item	Description
Link layer protocol	CAN bus
Application layer protocol	CANopen protocol
CAN-ID type	11bit-CAN2.0A
Baud rate	500 Kbit/s (default) 1 Mbit/s, 250 Kbit/s, 125 Kbit/s, 100 Kbit/s, 50 Kbit/s, and 20 Kbit/s
Maximum station No.	63
CAN frame length	0 bytes to 8 bytes
CAN frame types in the application layer	Data frame, remote frame

Item	Description
Termination resistor	120 Ω
Supported profiles	◆ CiA-301 V4.02: CANopen application layer and communication profile ◆ DSP-402 V3.0: drive and motion control profile
Available services	◆ NMT: Network Management system ◆ SDO: Service Data Object ◆ PDO: Process Data Object ◆ Device monitoring: including node guarding and heartbeat ◆ SYNC: including synchronous generator and synchronous reception, applied to PDO transmission
PDO transmission types	Time trigger, event trigger, synchronous trigger
Number of supported PDOs	4 RPDOs, 4 TPDOs
SDO transmission modes	Expedited SDO transmission, segmented SDO transmission
Supported operation modes	◆ Profile position mode ◆ Profile velocity mode ◆ Profile torque mode ◆ Homing mode ◆ Interpolation mode

The CANopen communication function supports the following seven kinds of baud rates. The communication distance is related to the baud rate and the communication cable.

Table 1-2 Supported baud rates

Baud Rate (bps)	1 M	500 K	250 K	125 K	100 K	50 K	20 K
Length (m)	25	100	250	500	500	1000	1000

Table 1-3 Relation among CAN communication transmission distance, baud rate, and nodes

No.	Transmission Distance	Rate	Number of Nodes	Cable Diameter
1	25 m	1 Mbps	64	0.205 mm^2
2	95 m	500 Kbps	64	0.34 mm^2
3	560 m	100 Kbps	64	0.5 mm^2
4	1100 m	50 Kbps	64	0.75 mm^2

The cable diameter, which may has little impact on the transmission distance in CAN communication, must be as large as possible. The following table lists the transmission distance between two nodes under different cable diameters and baud rates.

Table 1-4 Relation between cable diameter and transmission distance

Cable Diameter	500 Kbps	1 Mbps
3x0.3 mm^2	95 m	30 m
3x0.5 mm^2	95 m	30 m
3x0.75 mm^2	100 m	30 m

2 Wiring

The two RJ45 terminals of the SV660C series servo drive, as shown in the following figure, are CANopen communication ports and also serve as CN3 and CN4 ports of general-purpose products.

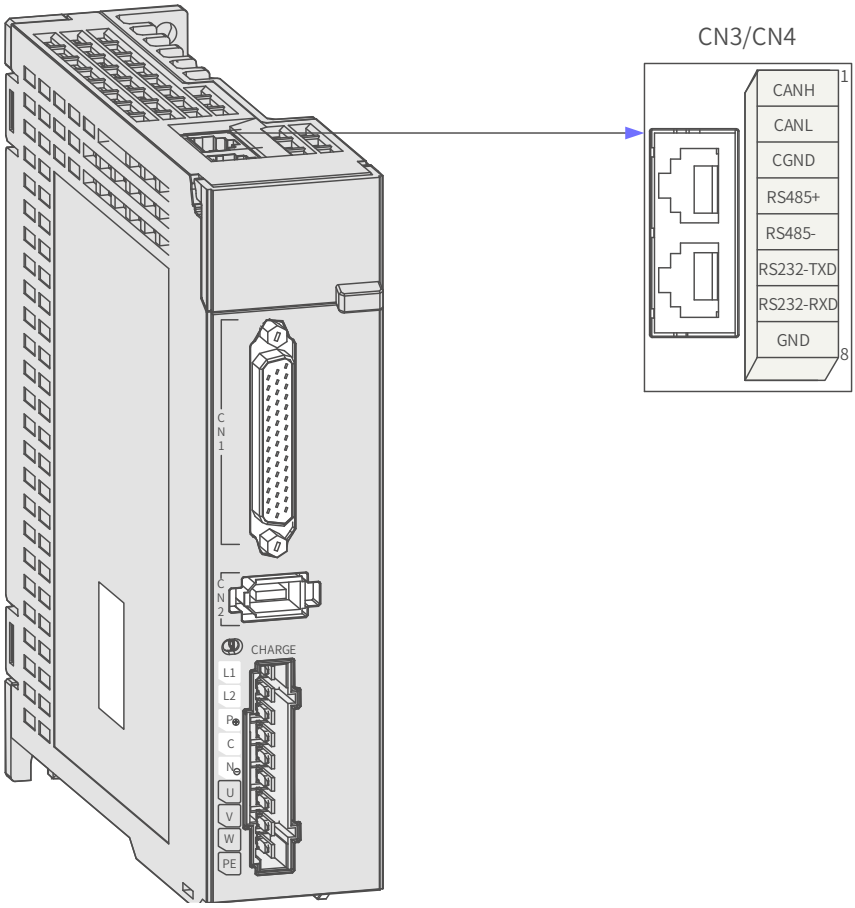
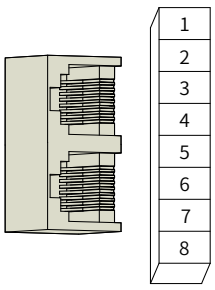


Figure 2-1 Position of CANopen communication ports

Pins of these two ports are internally connected. The following table describes definitions of the pins. CAN interface connectors are configured with the CANH, CANL, and CGND pins at least.

Table 2-1 Pin definition of communication signal connector

Pin No.	Signal Name	Definition	Terminal Pin Layout
1	CANH	CAN communication port	
2	CANL		
3	CGND	CAN communication ground	
4	RS485+	RS485 communication port	
5	RS485-		
6	RS232-TXD	RS232 transmission end, connected to the receiving end of the host controller	
7	RS232-RXD	RS232 receiving end, connected to the transmitting end of the host controller	
8	GND	Ground	
Enclosure	PE	Shield	

NOTE

The connection to CGND greatly improves the anti-interference performance of the CAN interface.

2.1 CAN Communication Cable

1) CAN communication cable for PLC

The following figure shows the cable (model: S6-L-T02-2.0) connecting the servo drive and the PLC during CAN communication networking.

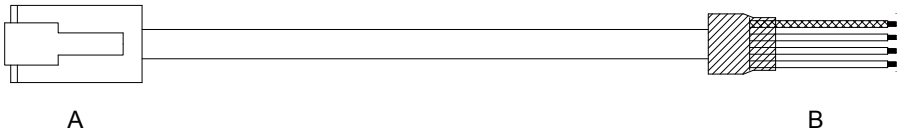


Figure 2-2 Outline drawing of the communication cable (model: S6-L-T02-2.0) between the servo drive and PLC

Table 2-2 Pin connection relation of the communication cable (model: S6-L-T02-2.0) between the servo drive and PLC

RJ45 on the Servo Drive Side (A)			PLC Side (B)		
Communication Type	Signal Name	Pin No.	Communication Type	Signal Name	Pin No.
CAN	CANH	1	CAN	CANH	1
	CANL	2		CANL	2
	CGND	3		CGND	3
-	PE (shield)	Enclosure	-	PE (shield)	Enclosure

2) CAN communication cable for multiple servo drives connected in parallel

The following figure shows the cable (model: S6-L-T01-0.3) for multiple servo drives connected in parallel during CAN communication networking.

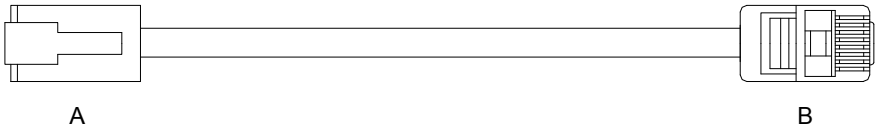


Figure 2-3 Outline drawing of the communication cable (model: S6-L-T01-0.3) for multiple drives connected in parallel

Table 2-3 Pin connection relation of the communication cable (model: S6-L-T01-0.3) for multiple servo drives connected in parallel (pins in CAN group used only)

RJ45 on the Servo Drive Side (A)			RJ45 on the Servo Drive Side (B)		
Communication Type	Signal Name	Pin No.	Communication Type	Signal Name	Pin No.
CAN	CANH	1	CAN	CANH	1
	CANL	2		CANL	2
	CGND	3		CGND	3
-	PE (shield)	Enclosure	-	PE (shield)	Enclosure

2.2 CAN Communication Bus and Multi-node Connection Mode

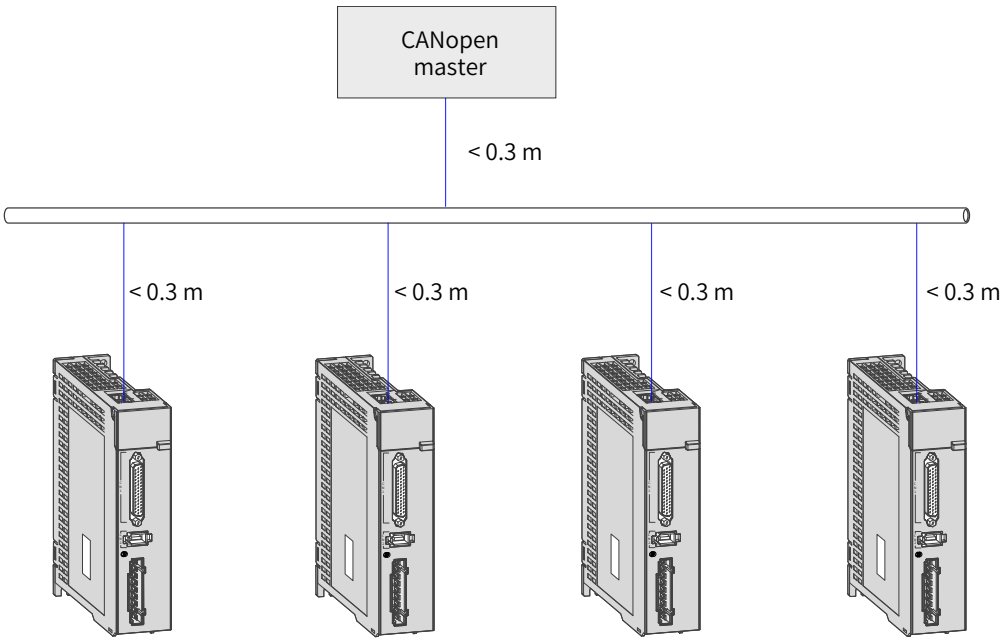


Figure 2-4 CAN communication network topology

The CAN communication network is connected in the bus mode, as shown in Figure 2-4.

CAN transceivers are mounted on the bus. Each branch must be shorter than 0.3 m. Otherwise, reflection may occur, causing communication errors.

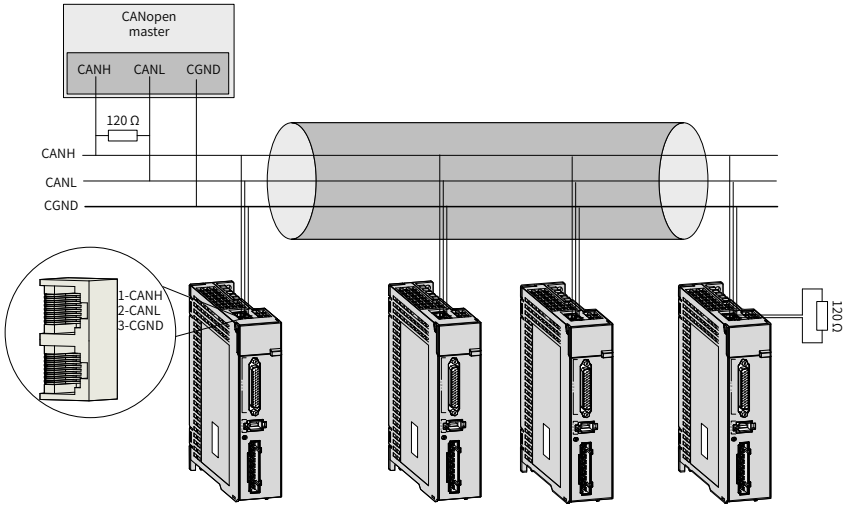


Figure 2-5 Wiring of CANopen

- It is recommended to use shielded twisted pairs for connection. Connect two 120 Ω termination resistors to each end of the bus to prevent signal reflection. Single-point grounding is generally used for the shield.
- Use a multimeter to measure the resistance between CANH and CANL to check whether the termination resistor is proper. The normal resistance is around 60 Ω (two resistors connected in parallel).
- Up to 64 devices can be mounted.
- When CAN devices communicate over a long distance, CGND of different CAN circuits must be mutually connected to ensure the reference potential of different communication devices is the same.

2.3 Recommended Twisted Pairs for CAN Communication

■ The following figure shows the CAN bus topology in a daisy chain mode. It is recommended to use shielded twisted pairs for CAN bus connection. Connect a 120-Ω termination resistor on each end of the bus to prevent signal reflection. The reference grounds of CAN signals of all the nodes (up to 64 nodes) are connected together.

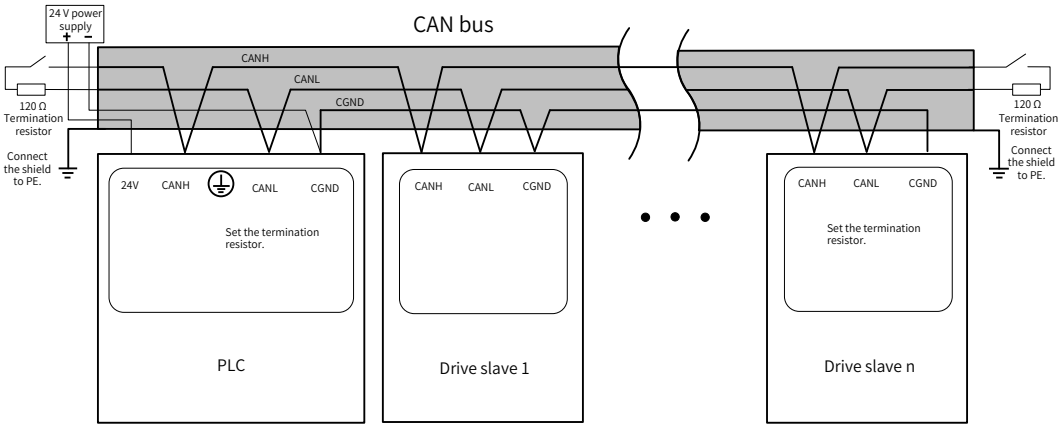


Figure 2-6 CAN bus topology

■ The transmission distance of the CAN bus is directly related to the baud rate and the communication cable, as shown in the following table.

No.	Transmission Distance	Baud Rate	Number of Nodes	Cross Sectional Area
1	25 m	1 Mbps	64	0.205 mm^2
2	95 m	500 kbps	64	0.34 mm^2
3	560 m	100 kbps	64	0.5 mm^2
4	1100 m	50 kbps	64	0.75 mm^2

2.4 Recommended Connection Modes for Different Cables

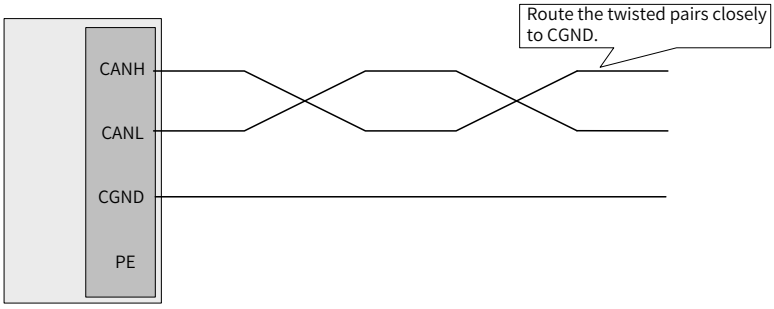


Figure 2-7 Recommended connection mode 1

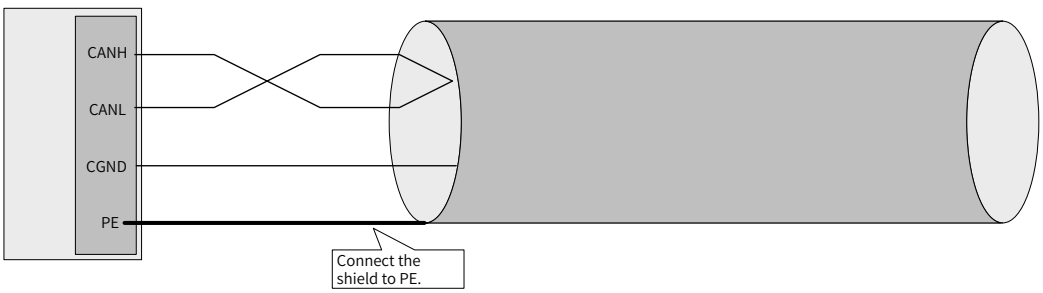


Figure 2-8 Recommended connection mode 2

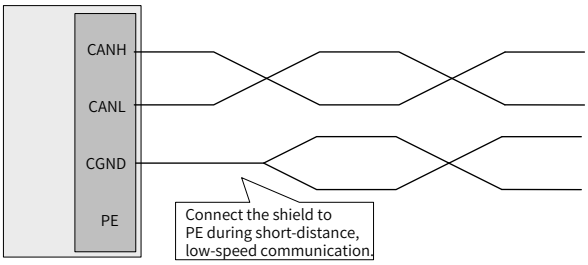


Figure 2-9 Recommended connection mode 3

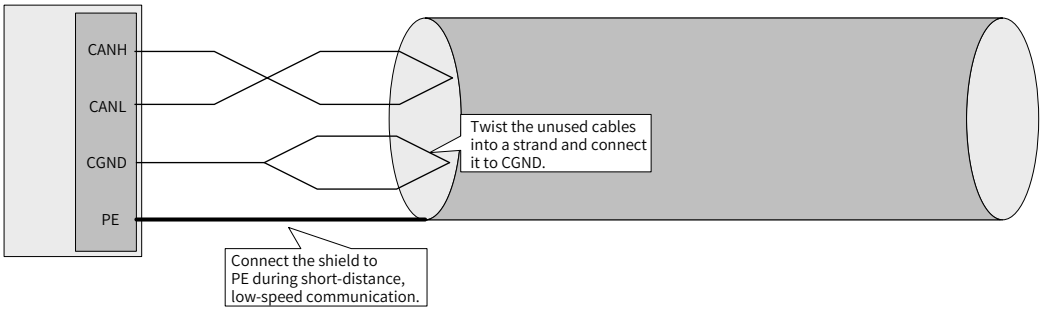


Figure 2-10 Recommended connection mode 4

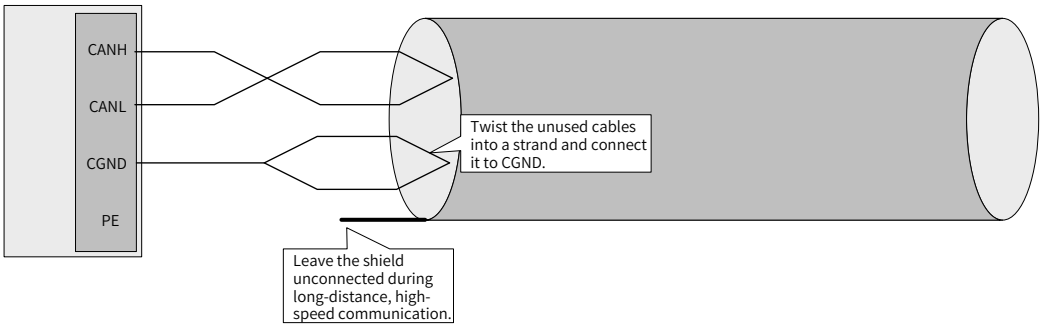


Figure 2-11 Recommended connection mode 5

2.5 Precautions for Grounding during CAN Communication

When CAN communication is used, the CGND terminal of the host controller must be connected to the CGND terminal of the servo drive, as shown below.

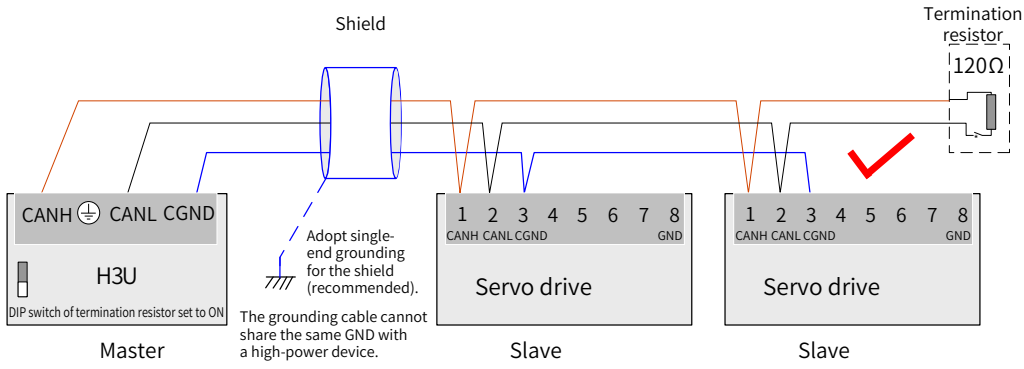


Figure 2-12 Correct CAN connection mode

CAUTION
◆ The PLC carries built-in CAN communication termination resistor and the corresponding DIP switch must be set to "ON". ◆ It is recommended to adopt single-end grounding for the shield. ◆ Do not connect the CGND terminal of the host controller to the GND terminal of the servo drive. Failure to comply will damage the device.

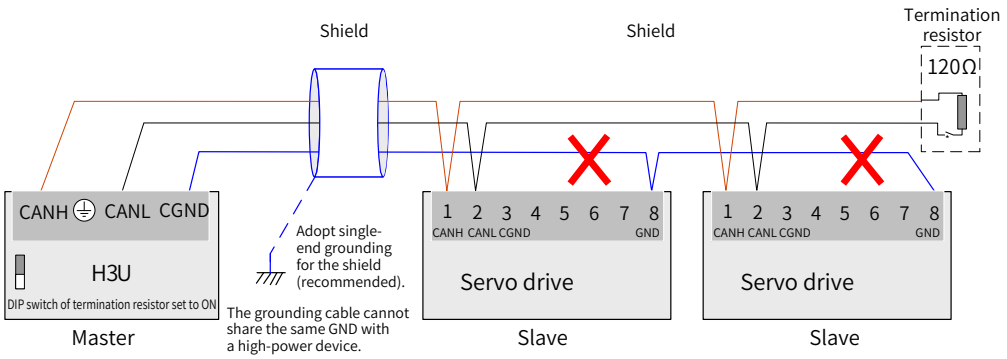


Figure 2-13 Wrong CAN connection mode

2.6 Wiring of Other Devices Without External CGND Port

2.6.1 Non-isolated CAN Devices Sharing GND or COM Port with Other Signals

Connect the GND or COM port of the device to the CGND of Inovance devices, as shown below.

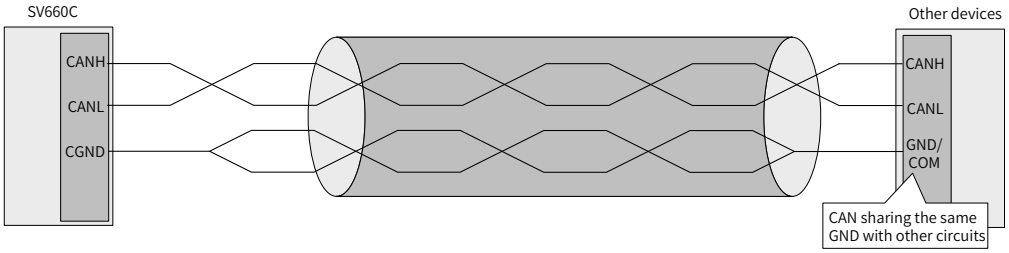


Figure 2-14 Connection mode for sharing the same GND with other circuits

2.6.2 No CGND for Device CAN and Other Ports

Do not connect CGND to any cable. Use an extra cable of AWG12 and above to connect the PE of each device. This extra cable must be laid more than 5 cm away from the CAN communication cable, as shown below.

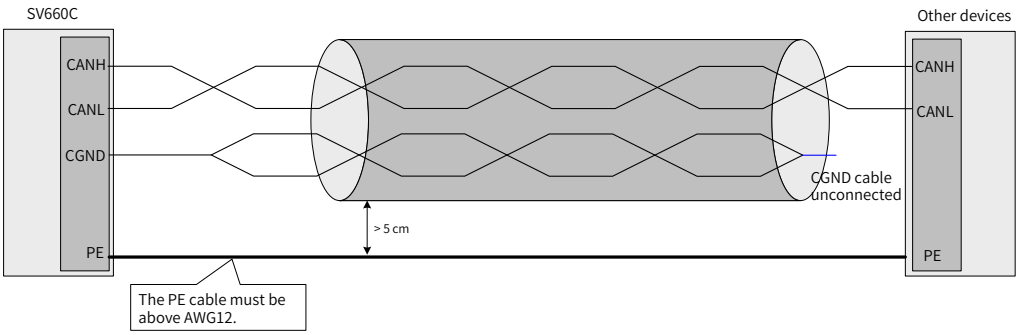


Figure 2-15 No external GND port of other device CAN

2.7 Recommended Routing of CAN Communication Cables

CAN communication is susceptible to interference. Route the CAN communication cable away from interference sources to prevent interference.

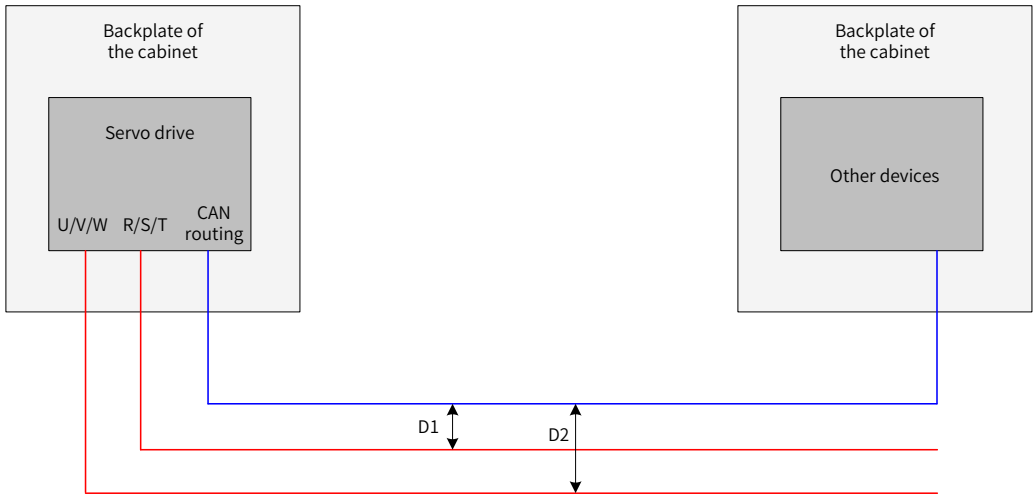


Figure 2-16 Recommended routing

- Route the interference cables and CAN cables in a vertical direction. During parallel routing, the distance D1 between the R/S/T cable and the CAN signal cable must be above 20 cm; the distance D2 between the U/V/W cable and the CAN signal cable must be above 50 cm. If interference cables are routed closely along the backplate of the cabinet, the distance between the CAN communication cable and the cabinet backplate must be above 1 cm.
- The R/S/T power cables, U/V/W power cables, and CAN communication cables, after passing through the cabinet, are routed in three cable troughs respectively. The distance among cable troughs must be above 20 cm. When interference cables and CAN communication cables are routed in the same cable trough, the routing clearance must also be above 20 cm.

3 Communication Network Configuration

The following figure shows the process for setting CANopen.

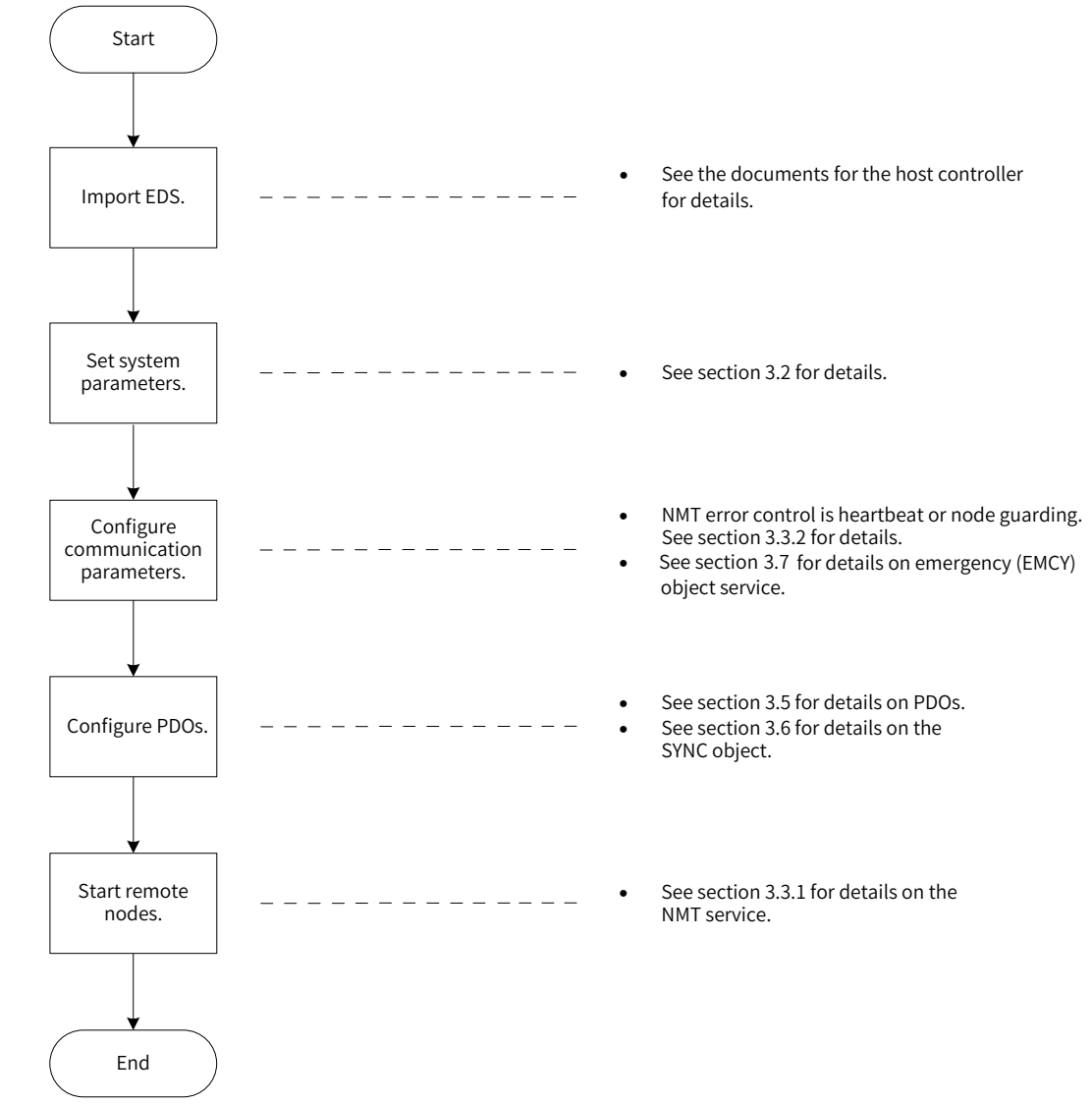


Figure 3-1 CANopen setting process

NOTE

For details on how to use SDOs, see *"3.4 Service Data Object (SDO)"*.

3.1 Overview of the CANopen Protocol

CANopen is an application layer protocol of a network transmission system based on CAN serial bus. It complies with the ISO/OSI standard model. Devices in the network exchange data through the object dictionary or objects. The master obtains or modifies data in the object dictionaries of other nodes through PDOs or SDOs. The CANopen device model is shown in the following figure.

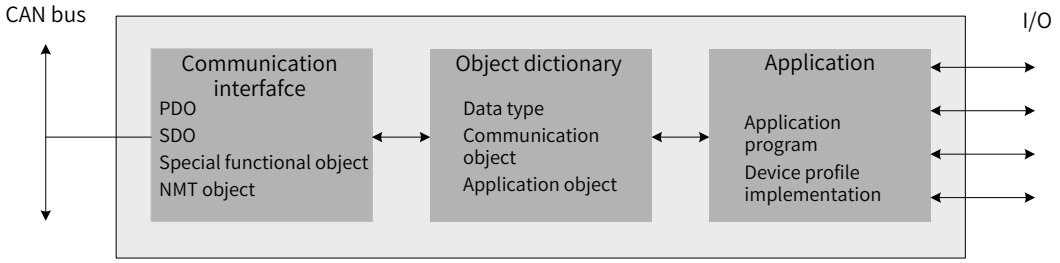


Figure 3-2 CANopen device model

3.1.1 Object Dictionary

The object dictionary is the most important part in device specifications. It is an ordered set of parameters and variables that contain all parameters of device descriptions and device network status. You can access a group of objects in an ordered and pre-defined way through the network.

The CANopen protocol adopts the object dictionary with 16-bit indexes and an 8-bit subindexes. The structure of the object dictionary is shown in the following table.

Table 3-1 Structure of the object dictionary

Index	Object
000	Unused
0001h–001Fh	Static data type (standard data type, such as Boolean and Integer16)
0020h–003Fh	Complex data type (predefined structure composed of simple types, such as PDOCommPar and SDOParmeter)
0040h–005Fh	Complex data type specified by the manufacturer
0060h–007Fh	Static data type specified by the device profile
0080h–009Fh	Complex data type specified by the device profile
00A0h–0FFFh	Reserved
1000h–1FFFh	Communication profile area (such as device type, error register, and number of supported PDOs)
2000h–5FFFh	Profile area specified by the manufacturer (such as parameter mapping)
6000h–9FFFh	Standard device profile area (such as DSP-402 protocol)
A000h–FFFFh	Reserved

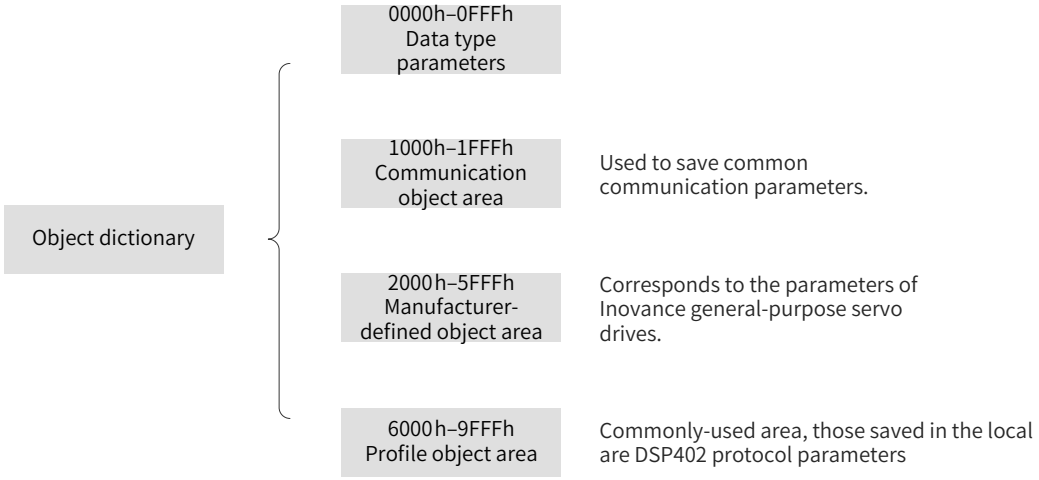


Figure 3-3 Structure of CANopen object dictionary

The mapping relation between the parameter and the object dictionary is as follows:

Object dictionary index = 0x2000 + Parameter group No.

Object dictionary subindex = Hexadecimal offset within the parameter group + 1

Example:

H02-10 corresponds to the object 0x2002-0B in the object dictionary.

Each object in the object dictionary is described based on types.

Example:

The object 607Dh used to limit the software position describes the minimum and maximum position limits. The object is defined as follows:

Table 3-2 Example of object dictionary description based on types

Index	Subindex	Name	Meaning
607Dh	00h	Number of subindexes for software position limit	Quantity of the object data, excluding the object itself
607Dh	01h	Min. position limit	Min. position limit (in absolute position mode)
607Dh	02h	Max. position limit	Max. position limit (in absolute position mode)

3.1.2 Common Communication Objects

1) Network Management (NMT)

An NMT object includes the Boot-up message, heartbeat protocol, and NMT message. Based on the master/slave communication mode, NMT is used to manage and monitor each node in the network to implement node status control, error control, and node startup.

2) Servo Data Object (SDO)

- The SDO includes the receive SDO (R_SDO) and the transmit SDO (T_SDO).
- The SDO enables clients to access entries in the device object dictionary through indexes and subindexes.
- The SDO is implemented through multi-domain CMS objects in the CAL and allows transmission of data in any length. Data with more than four bytes will be segmented into several messages.

- The protocol is used to confirm the service type and generate a response for each message. An SDO request and a response message always contain eight bytes.

3) Process Data Object (PDO)

- The PDO includes the receive PDO (RPDO) and the transmit PDO (TPDO).
- The PDO is used to transmit real-time data from one creator to one or multiple receivers. The length of the data transmitted ranges from one to eight bytes.
- Each CANopen device offers eight default PDO channels, including four TPDO channels and four RPDO channels.
- PDOs support synchronous and asynchronous transmission modes. The transmission mode depends on the communication parameters corresponding to the PDO.
- The content of a PDO message is pre-defined by PDO mapping parameters.

4) Synchronization (SYNC) object

The SYNC object is a message periodically broadcasted by the CANopen master to the CAN bus. It is used to provide basic network clock signals. Each device determines whether to use the event to synchronize with other network devices based on its own configurations.

5) Emergency (EMCY) message

The EMCY message is sent in the case of a communication failure or application failure.

3.1.3 Communication Object Identifier

A communication object identifier (COB-ID) specifies the priority of an object during communication and identifies the communication object. A COB-ID corresponds to a 11-bit frame of CAN 2.0A. The 11-bit COB-ID consists of two parts, a 4-bit object function code and a 7-bit node address, as shown in the following table.

Table 3-3 Structure of a COB-ID

10	9	8	7	6	5	4	3	2	1	0
Function code				Node address						

Each CANopen communication object has its default COB-ID, which can be read through SDOs. Some COB-IDs can be modified through SDOs. See the following table for the COB-ID list.

Table 3-4 List of COB-IDs

Communication Object	Function Code	Node Address	COB-ID	Object Index
NMT	0000b	0	0h	-
SYNC object	0001b	0	80h	1005h, 1006h
EMCY message object	0001b	1 to 127	80h + Node ID	1014h
TPDO1	0011b	1 to 127	180h + Node ID	1800h
RPDO1	0100b	1 to 127	200h + Node ID	1400h
TPDO2	0101b	1 to 127	280h + Node ID	1801h
RPDO2	0110b	1 to 127	300h + Node ID	1401h
TPDO3	0111b	1 to 127	380h + Node ID	1802h

Communication Object	Function Code	Node Address	COB-ID	Object Index
RPDO3	1000b	1 to 127	400h + Node ID	1402h
TPDO4	1001b	1 to 127	480h + Node ID	1803h
RPDO4	1010b	1 to 127	500h + Node ID	1403h
T_SDO	1011b	1 to 127	580h + Node ID	1200h
R_SDO	1100b	1 to 127	600h + Node ID	1200h
NMT error control	1110b	1 to 127	700h + Node ID	1016h, 1017h

Example:

The COB-ID of TPDO2 of No. 4 slave is 284h (280h+4).

3.2 System Settings

Related parameters of the SV660C series servo drive must be set properly so that the servo drive can be connected to the CANopen fieldbus network correctly.

Table 3-5 System setting parameter table

Parameter		Name	Value Range	Default
H02	00	Control mode selection	0: Speed mode 1: Position mode 2: Torque mode 3: Speed mode - Torque mode 4: Position mode - Speed mode 5: Position mode - Torque mode 6: Position mode - Speed mode - Torque mode 8: CANopen mode	8
H0C	00	Servo axis address	1 to 127	1
H0C	08	CAN communication rate	0: 20 Kbps 1: 50 Kbps 2: 100 Kbps 3: 125 Kbps 4: 250 Kbps 5: 500 Kbps 6: 1 Mbps 7: 1 Mbps	5
H0C	13	Saved to EEPROM	0: Not save 1: Save	1

3.3 Network Management (NMT) System

The NMT system initializes, starts, and stops a network and devices in the network. It belongs to the master/slave system. There is only one NMT master in the CANopen network. The CANopen network, including the master itself, can be configured.

3.3.1 Network Management Service

CANopen works according to the state machine specified by the protocol. Some data are converted automatically and some data must be converted through NMT messages transmitted by the NMT master, as shown below.

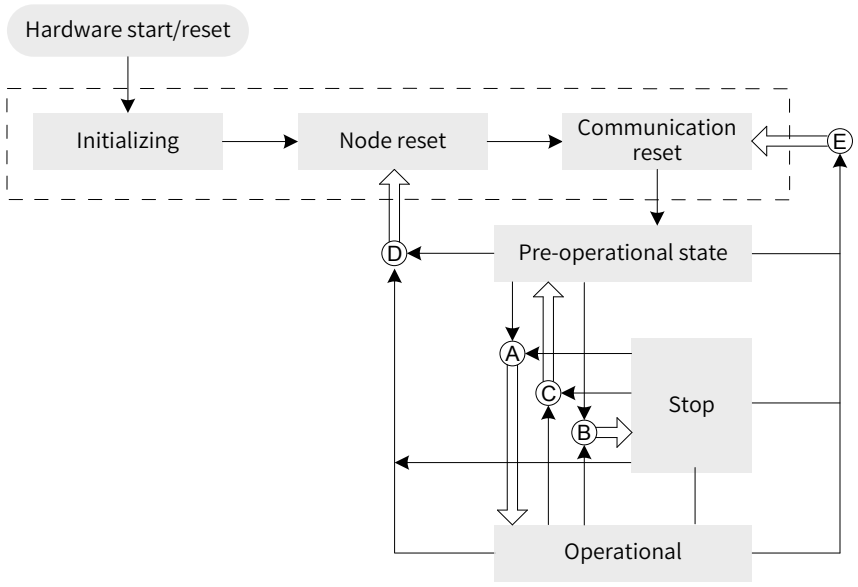


Figure 3-4 Execution process of the NMT state machine

In the above figure, conversions marked with a letter are implemented through NMT messages and only the NMT master can transmit NMT control messages in the format shown below.

Table 3-6 NMT message format

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

The COB-ID of an NMT message is permanently "0x000".

The data area contains two bytes. The first byte is a command word indicating this frame is for control purpose. See Table 3-7 for details.

The second byte is the CANopen node address. The byte value 0 indicates it is a broadcast message and all slaves in the network are active.

Table 3-7 NMT message command

Command Word	Conversion Code	Description
0x01	A	Instruction for starting a remote node

Command Word	Conversion Code	Description
0x02	B	Instruction for stopping a remote node
0x80	C	Instruction for entering the pre-operational state
0x81	D	Instruction for resetting a node
0x82	E	Instruction for resetting communication

After power-on, the device automatically enters the initialization state, including initializing, node reset, and communication reset. During initializing, parameters of each mode are loaded. During node reset, the manufacturer-defined area and profile area of the object dictionary are restored to values saved last time. During communication reset, the communication parameters in the object dictionary are restored to values saved last time.

Next, the device sends Boot-up and enters the pre-operational state, which is the state of the main configuration node.

After configurations are done, the node can enter the operational state only after the NMT master sends the NMT message. When CANopen is working properly, it is in the operational state. All modules should work properly.

When the NMT master sends a stop message, the device enters the stop state. In CANopen communication, only the NMT module works properly.

Table 3-8 lists CANopen services supported in different NMT states.

Table 3-8 Services supported in different NMT states

Service	Pre-operational	Operational	Stop
PDO	No	Yes	No
SDO	Yes	Yes	No
SYNC object	Yes	Yes	No
EMCY message	Yes	Yes	No
NMT system	Yes	Yes	Yes
Error control	Yes	Yes	Yes

3.3.2 NMT Error Control

NMT error control is used to detect whether devices in the network are online and the device state, including node guarding, life guarding, and heartbeat.

NOTE

- Life guarding and heartbeat cannot be used at the same time.
- Set the node guarding, life guarding, and heartbeat time to large values to prevent excessive network load.

1) Node/Life guarding

In the node guarding, the NMT master periodically checks the NMT slave state. In life guarding, the slave monitors the master state indirectly through the remote frame interval used to monitor the slave. The node guarding complies with the master/slave model. A response must be provided for each remote frame.

Objects related to node/life guarding include the guarding time 100Ch and life factor 100Dh.

The value of 100Ch is the remote frame interval (ms) for node guarding under normal conditions. The result of 100Ch multiplied by 100Dh determines the latest time for check by the master. When 100Ch and 100D of a node are set to non-zero values and a node guarding request frame is received, the life guarding will be activated.

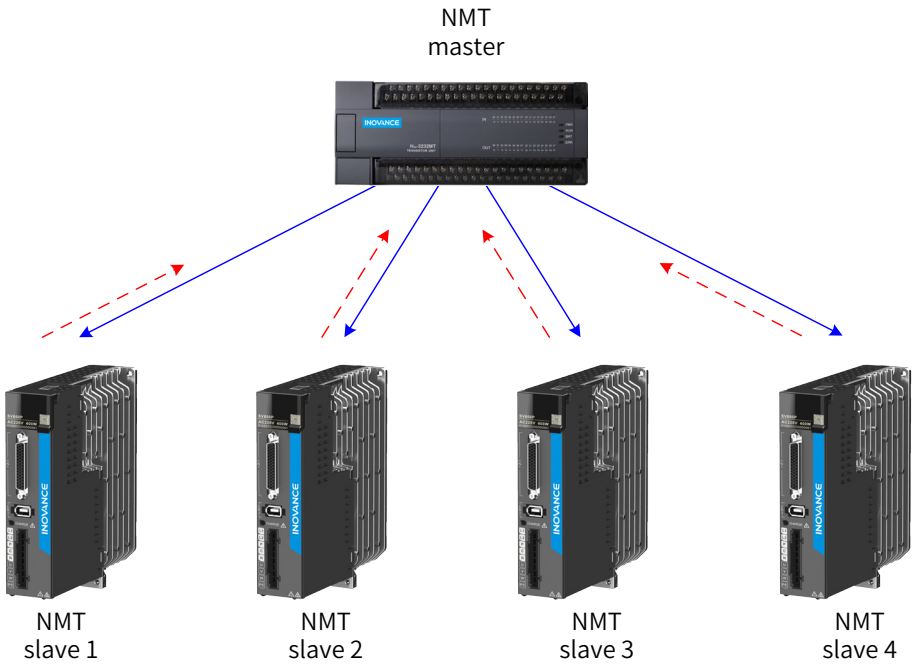


Figure 3-5 Description of node guarding

As shown in the above figure, the master sends a node guarding remote frame at the interval defined by 100Ch, and the slave must respond to the remote frame. Otherwise, the slave is considered to be offline.

If the slave does not receive the node guarding remote frame within 100Chx100Dh, the master is considered to be offline.

The following table describes the remote frame sent by the NMT master.

Table 3-9 Node guarding remote frame message

COB-ID	RTR
0x700+Node_ID	1

The following tables describe the response message returned by the slave. The data segment is a status word consisting of one byte, as described in Table 3-11.

Table 3-10 Node guarding response message

COB-ID	RTR	Data
0x700+Node-ID	0	Status word

Table 3-11 Description of response message state

Data bit	Description
bit7	Set to 0 or 1 alternatively.
bit6 to bit0	4: Stop 5: Operational state 127: Pre-operational state

NOTE

It is recommended that the guarding time 100C be longer than 10 ms and the life factor be greater than or equal to 2.

2) Heartbeat

The heartbeat mode adopts the producer-consumer model. The CANopen device can send heartbeat messages based on the interval (ms) defined by the producer heartbeat interval object (1017h). There is always a node configured with the consumer heartbeat function in a network, which monitors the producer based on the consumer time defined by 1016h. Once the producer heartbeat is not received from the corresponding node within the consumer heartbeat time, a node fault occurs.

After the producer heartbeat interval (1017h) is configured, the node heartbeat will be activated and a heartbeat message is generated. Monitoring will be started after a valid subindex is configured for consumer heartbeat (1016h) and a heartbeat frame is received from the corresponding node.

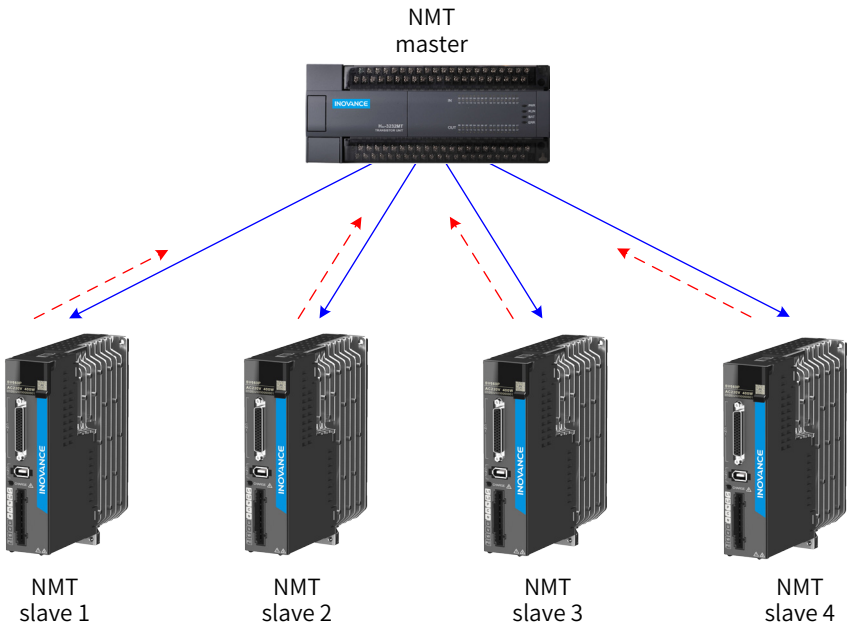


Figure 3-6 Heartbeat

The master sends a heartbeat message based on the producer time. If the slave that monitors the master does not receive the heartbeat message within the time defined by 1016h subindex, the master is considered to be offline. The time defined by the subindex of 1016h must be greater than or equal to the master producer time multiplied by 1.8. Otherwise, a false report indicating the master is offline may occur.

The slave sends a heartbeat message at the interval defined by 1017h. If the master (or other slaves) that monitors the slave does not receive the heartbeat message within the consumer time, the slave is considered to be offline. If the value of 1017h (producer heartbeat interval) multiplied by 1.8 is smaller than or equal to the consumer time of the master (or other slaves) that monitors the slave, a false report indicating the slave is offline may occur.

Table 3-12 describes the format of a heartbeat message. The data segment contains only one byte. The most significant bit is fixed to 0 and other bits are consistent with the state of the response message.

Table 3-12 Heartbeat message

COB-ID	RTR	Data
0x700+Node-ID	0	Status word

The SV660C servo drive is both a heartbeat producer and a heartbeat consumer. It can serve as the heartbeat consumer of up to five nodes. It is recommended that the heartbeat producer time be set to a value greater than or equal to 20 ms and the consumer heartbeat time be set to a value not smaller than 40 ms but greater than the producer heartbeat time multiplied by 1.8.

3.4 Service Data Object (SDO)

The SDO is linked to the object dictionary through the object index and subindex. You can view the object content in the object dictionary or modify the object data through the SDO if allowed.

3.4.1 SDO Transmission Framework

The SDO transmission mode complies with the client-server mode, that is, one initiates a request and the other responds to the request. An SDO client in the CAN bus network initiates a request and the SDO server responds to the request. Therefore, data exchange between SDOs requires at least two CAN messages and the CAN identifiers of these two CAN messages must be different. Figure 3-7 shows the SDO transmission model.

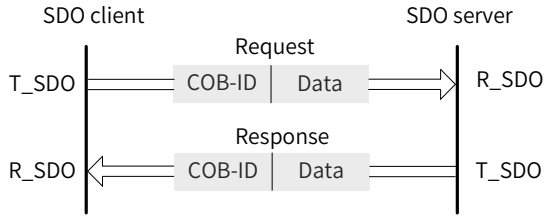


Figure 3-7 Object word in the SDO server read/written by the SDO client

3.4.2 SDO Transmission Message

The SDO can be transmitted using data with no more than four bytes or data with more than four bytes. For the former, the expedited SDO transmission mode is adopted; for the latter, the segmented or block transmission mode is adopted. The SV660C servo drive only supports expedited SDO transmission and segmented transmission.

The SDO transmission message is comprised of the COB-ID and data segment. As described in Table 3-4, the COB-IDs of T_SDO and R_SDO messages are different.

The data segment adopts the little endian mode, in which the least significant bits are arranged in front of the most significant bits. The data segment of the SDO message must contain eight bytes. Table 3-13 describes the format of an SDO transmission message.

Table 3-13 Description of SDO transmission message format

COB-ID	Data							
580h+Node_ID/ 600h+Node_ID	0	1	2	3	4	5	6	7
	Command code	Index		Subindex	Data			

The command code specifies the transmission type and transmission data length of the SDO. The index and subindex indicate the position of the object in the list. The data indicates the value of the object.

1) Messages written in the expedited mode

Accelerated SDO transmission is adopted for writing/reading a message with no more than four bytes. The transmission message varies with the read/write mode and the data length. Table 3-14 describes a message written in the expedited mode.

Table 3-14 Example of a message written in the expedited mode

		COB-ID	0	1	2	3	4	5	6	7
Client →		600h+Node_ID	23h	Index		Sub-index	Data			
			27h				Data			-
			2bh				Data		-	-
			2fh				Data	-	-	-
← Server	Normal	580h+Node_ID	60h	Index		Sub-index	-	-	-	-
	Abnormal		80h				Abort code			

NOTE

"-" indicates the data exists but is not considered, which is recommended to be filled with the value 0 during data writing. The same rule applies to the following descriptions in this section.

Example:

If the slave station No. is 4, write the velocity value (60FFh-00) in the velocity mode by using SDOs. The value written is 1000, namely 0x3E8. The message sent by the master is as follows (in hexadecimal).

Table 3-15 Example of a message sent by the master

COB-ID	0	1	2	3	4	5	6	7
604	23	FF	60	00	E8	03	00	00

If the write operation is normal, the servo drive returns the following message:

Table 3-16 Example of a message returned by the servo drive

COB-ID	0	1	2	3	4	5	6	7
584	60	FF	60	00	00	00	00	00

If the type of the data written does not match, the fault code 0x06070010 is returned. The message is as follows.

Table 3-17 Example of a message returned upon mismatch of the written data type

COB-ID	0	1	2	3	4	5	6	7
584	80	FF	60	00	10	00	07	06

2) Messages read in the expedited mode

The expedited mode is adopted for reading a message with no more than four bytes. Table 3-18 describes the message read in the expedited mode.

Table 3-18 Format of a message read in the expedited mode

		COB-ID	0	1	2	3	4	5	6	7
Client →		600h+Node_ID	40h	Index		Sub-index	-	-	-	-
← Server	Normal	580h+Node_ID	43h	Index		Sub-index	Data			
			47h				Data			-
			4bh				Data		-	-
			4fh				Data	-	-	-
	Abnormal		80h				Abort code			

Example:

If the slave station No. is 4, read the maximum speed limit (H06-07), namely the object 0x2006-08 through the SDO. The message sent by the master is as follows (in hexadecimal).

Table 3-19 Example of a message sent by the master

COB-ID	0	1	2	3	4	5	6	7
604	40	06	20	08	00	00	00	00

The default value of the maximum speed is 6000 RPM, namely 0x1770. Normally, the following message is returned.

Table 3-20 Example of a message returned upon maximum speed

COB-ID	0	1	2	3	4	5	6	7
584	4b	06	20	08	70	17	00	00

If the command word written does not match, an error indicating the command word is invalid will be returned, with fault code 0x05040001. The message is as follows.

Table 3-21 Example of a message returned upon mismatch of the write command

COB-ID	0	1	2	3	4	5	6	7
584	80	06	20	08	01	00	04	05

3) Messages read in the segmented mode

The object with more than 4 bytes is read in the segmented mode. The structure of a message transmitted in the segmented mode is similar to that in the expedited mode. The start frame is the same as that in the expedited mode. Table 3-22 describes the structure of a start message transmitted.

Table 3-22 Structure of a start message transmitted

		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			

During transmission, the trigger bit (bit6) of the command code sends 0 or 1 alternatively. This rule must be observed so that the slave can respond to the message. Table 3-23 describes the message structure during transmission.

Table 3-23 Structure of a message during SDO transmission

		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 response packet of the last frame transmitted in the segmented mode includes the identifier of the last frame and valid data length of the last frame. The transmission message structure is shown in the following table.

Table 3-24 Message structure of the last frame in segmented transmission

		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	Abort code			

3.5 Process Data Object (PDO)

The PDO, as the major data transmission mode in CANopen, is used to transmit real-time data. The PDO transmission is quick and fast as no response is required during PDO transmission and a PDO may consist of less than eight bytes.

Figure 3-8 shows the PDO mapping configuration process.

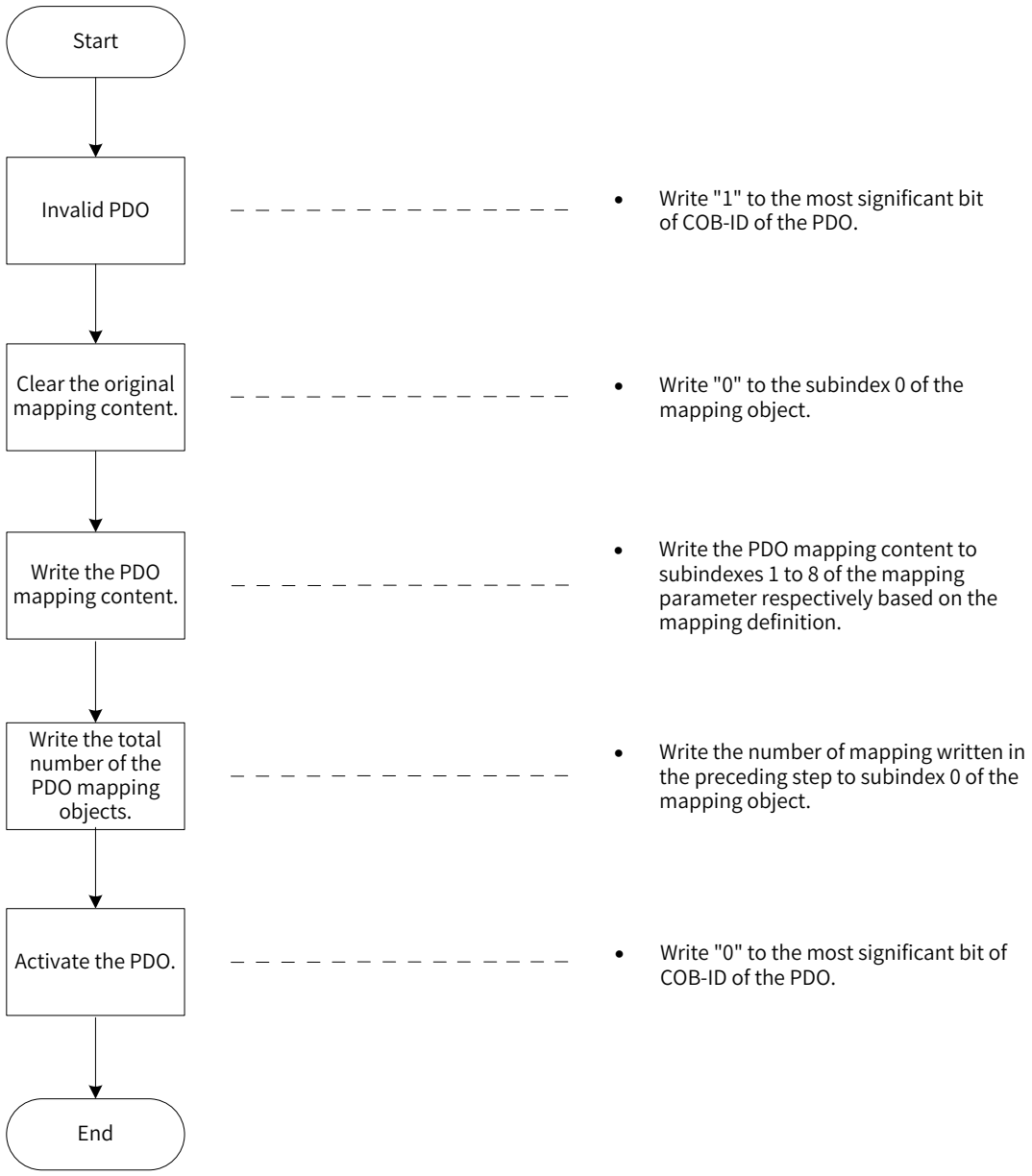


Figure 3-8 PDO mapping configuration process

3.5.1 PDO Transmission Framework

The PDO transmission complies with the producer-consumer model, that is, in the CAN bus network, the TPDO generated by the producer may be received by one or multiple consumers in the network based on the COB-ID. The transmission model is shown in the following figure.

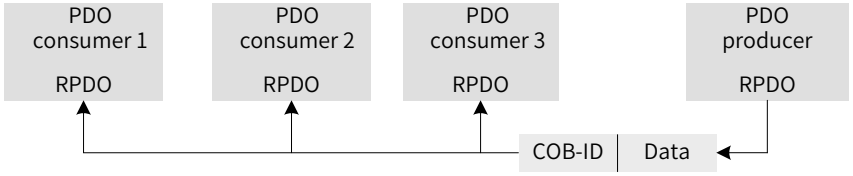


Figure 3-9 PDO transmission model

The CANopen communication of SV660C servo drives only supports point-to-point PDO transmission.

3.5.2 PDOs

PDOs can be divided into RPDOs and TPDOs. The final PDO transmission mode and content are determined by communication parameters and mapping parameters. In SV660C servo drives, the PDO is transmitted by using four RPDOs and four TPDOs. Table 3-25 lists the related PDOs.

Table 3-25 PDOs of SV660C servo drives

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

3.5.3 PDO Communication Parameters

1) CAN identifiers of PDOs

The CAN identifier of a PDO, namely COB-ID of a PDO, includes the control bit and the identifier data. It determines the bus priority of the PDO. The COB-ID is in the subindex 01 of communication parameters (RPDO: 1400h to 1403h; TPDO: 1800h to 1803h). The most significant bit decides whether the PDO is valid.

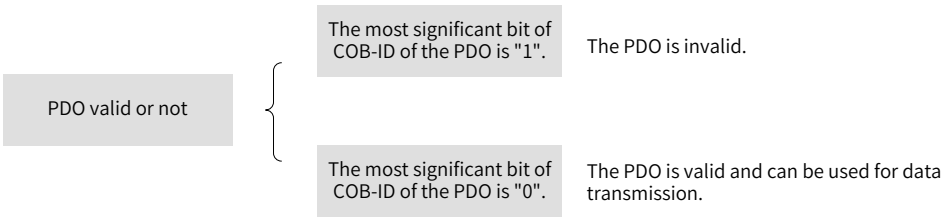


Figure 3-10 Description of the PDO state

The SV660C servo drive only supports point-to-point PDO transmission. Therefore, the seven least significant bits of the COB-ID must be the station No. address of the node.

Example:

For the node whose station No. is 4, when TPDO3 is invalid, its COB-ID should be 80000384h. When 384h is written to the COB-ID, it indicates the PDO is activated.

2) PDO transmission type

The PDO transmission type is in the subindex 02 of communication parameters (RPDO: 1400h to 1403h; TPDO: 1800h to 1803h) and determines the mode in which the PDO is transmitted. For details, see *"4.5 Overview of Operation Modes"*.

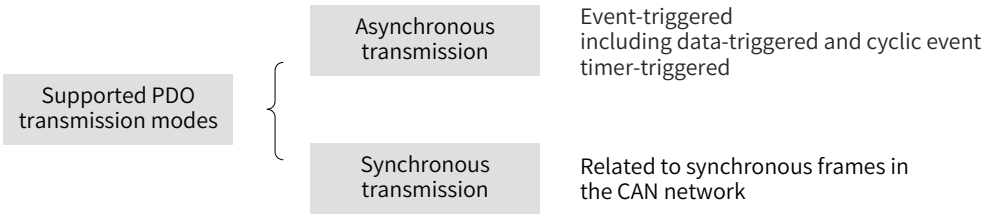


Figure 3-11 Supported PDO transmission modes

The subindex 02 of communication parameters (RPDO: 1400h to 1403h, TPDO: 1800h to 1803h) indicates the transmission type. Different values of the subindex 02 stand for different transmission types and define the methods for triggering TPDO transmission or methods for processing RPDOs. Table 3-26 lists methods for triggering TPDOs and RPDOs.

Table 3-26 Methods for triggering TPDOs and RPDOs

Value of Communication Type	Synchronous		Asynchronous
	Cyclic	Acyclic	
0		√	
1 to 240	√		
241 to 253	-		
254, 255			√

- In TPDO transmission type 0, the TPDO is transmitted when the mapping data changes and the synchronous frame is received.
- In TPDO transmission types 1 to 240, the TPDO is transmitted when corresponding number of synchronous frames are received.
- In TPDO transmission type 254 or 255, the TPDO is transmitted when the mapping data changes or when the event timer expires.
- In RPDO transmission types 0 to 240, the latest data of the RPDO are updated to the application once a synchronous frame is received. In RPDO transmission type 254 or 255, the received data are updated to the application directly.

3) Inhibit time

The inhibit time is set for TPDOs and is saved to the subindex 03 of communication parameters (1800h to 1803h) to prevent the CAN network from being continuously occupied by PDOs with lower priorities. After the inhibit time is set (in 100 us), the transmission interval of a TPDO must be longer than or equal to the inhibit time.

Example:

If the inhibit time of TPDO2 is 300, the transmission interval of the TPDO must be no shorter than 30 ms.

4) Event timer

For TPDOs transmitted in asynchronous mode (transmission type 254 or 255), an event timer is defined in the subindex 05 of communication parameters (1800h to 1803h). The event timer can be considered as a trigger event. It also triggers TPDO transmission. If another event, for example, data change, occurs when the event timer is running, the TPDO is triggered and the event timer is reset immediately.

3.5.4 PDO Mapping Parameters

PDO mapping parameters include pointers of process data corresponding to PDOs to be sent or received, which includes the index, subindex, and mapping object length. The length of each PDO data can be up to eight bytes and one or multiple objects can be mapped. The subindex 0 records the number of objects mapped by the PDO and the subindexes 1 to 8 are the mapping content. Table 3-27 defines the content of mapping parameters.

Table 3-27 Definition of PDO mapping parameters

Bits	31	……	16	15	……	8	7	……	0
Meaning	Index			Subindex			Object length		

The index and subindex jointly decide the location of an object in the object dictionary. The object length indicates the bit length of the object and is expressed in hexadecimal.

Table 3-28 Relation between the object length and object bit length

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

Example:

The mapping parameter of the 16-bit command word 6040h-00 is 60400010h.

The following example describes the mapping relation of PDOs.

Example:

The following three parameters are mapped by RPDO1.

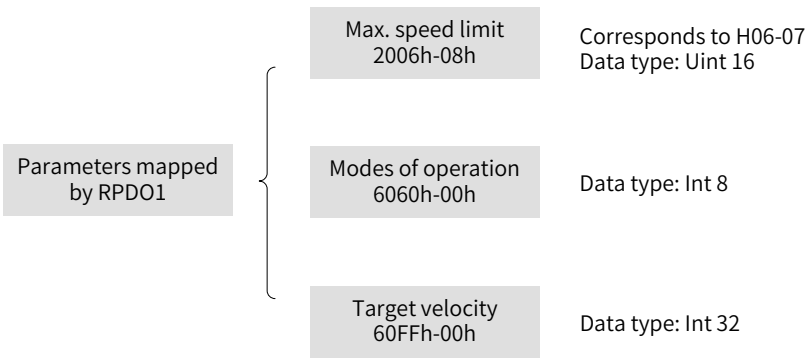


Figure 3-12 Example of RPDO1 mapping relation

The total length of mapping is seven bytes (2+1+4), that is, the RPDO1 data segment has seven bytes during transmission. Figure 3-13 shows the mapping relation.

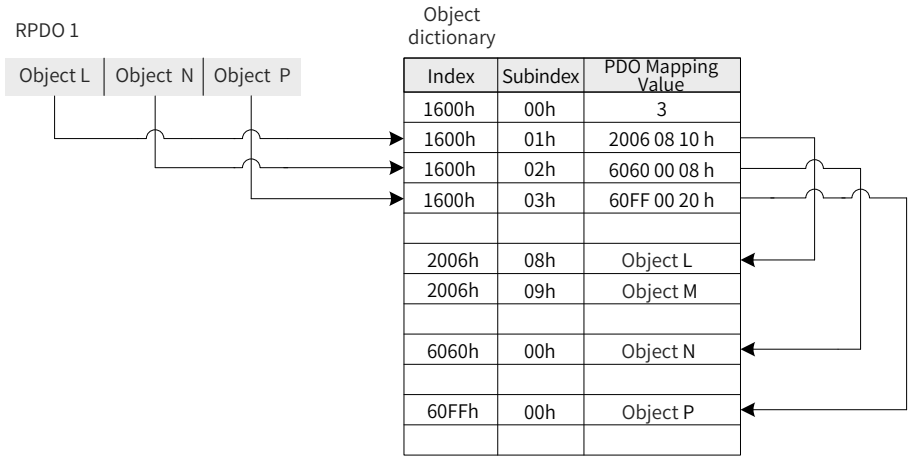


Figure 3-13 Example of RPDO mapping relation

The mapping mode of TPDOs is the same as that of RPDOs, but in the opposite direction. The RPDO decodes the input based on the mapping relation. The TPDO encodes the output based on the mapping relation.

Example:

The following two parameters are mapped by TPDO2.

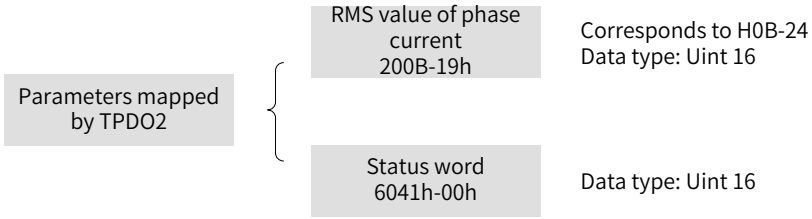


Figure 3-14 Example of TPDO2 mapping relation

The total length of mapping is four bytes (2+2), that is, the TPDO2 data segment has four bytes during transmission. Figure 3-15 shows the mapping relation.

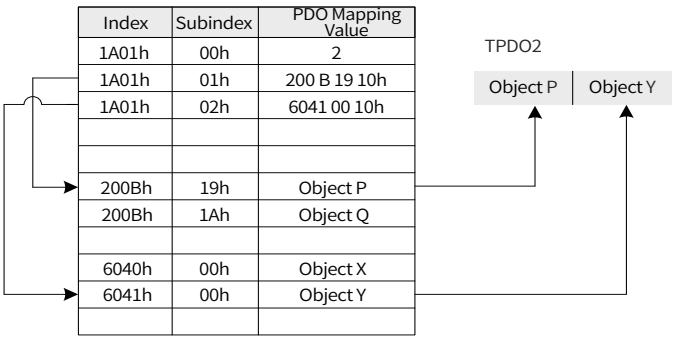


Figure 3-15 Example of TPDO mapping relation

3.6 Synchronization (SYNC) Object

The SYNC object is a special mechanism that controls harmony and synchronization between transmission and reception of multiple nodes. It is used for synchronous transmission of PDOs.

Figure 3-16 shows the configuration process of SYNC generator.

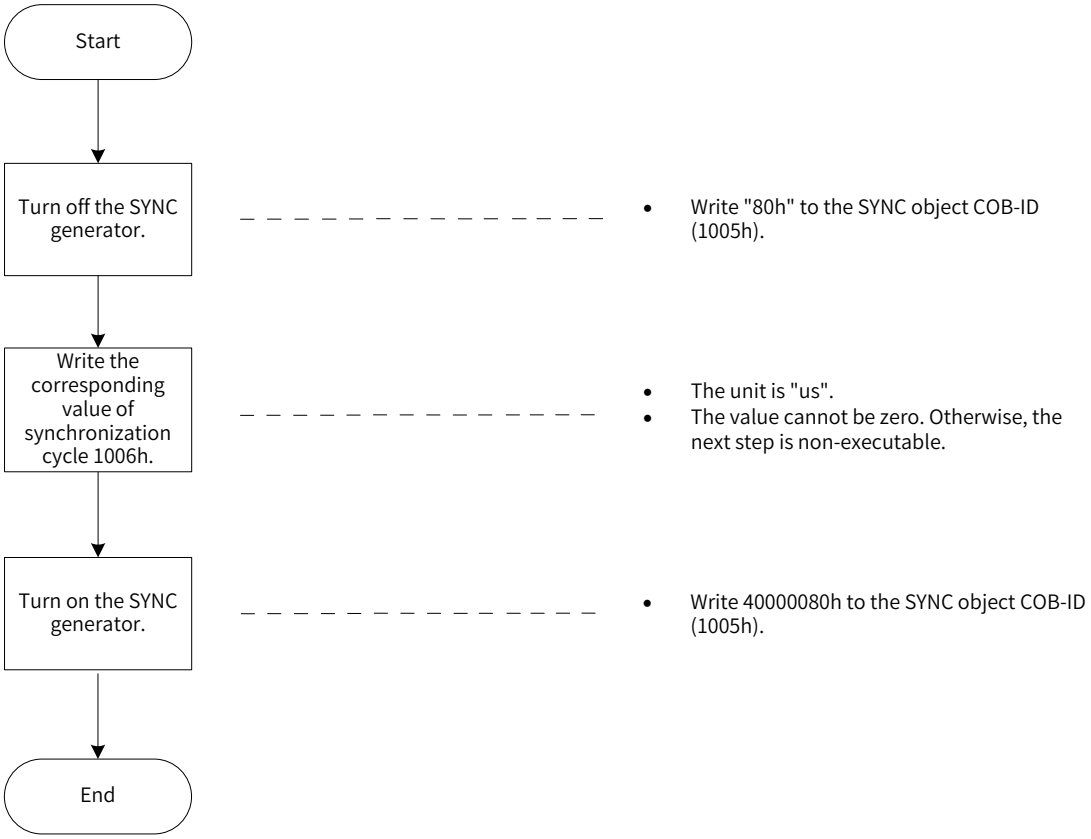


Figure 3-16 SYNC generator configuration process

NOTE

The SYNC generator with a cycle shorter than 500 us is not supported by the SV660C series servo drives. A cycle lower than 1 ms is not recommended.

3.6.1 SYNC Generator

The SV660C servo drive is both a SYNC consumer and a SYNC producer. Synchronization-related objects supported are the synchronization object COB-ID (1005h) and synchronization cycle (1006h).

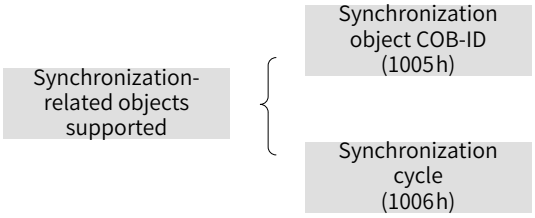


Figure 3-17 Description of synchronization-related objects supported

The second most significant bit of the synchronization object COB-ID decides whether to activate the SYNC generator.

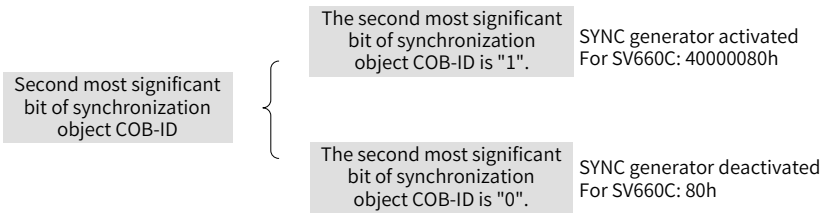


Figure 3-18 Instructions for activating the SYNC generator

The synchronization cycle (in us) is used for the SYNC generator only. It indicates the interval in which a node generates a SYNC object.

3.6.2 SYNC Object Transmission Framework

Similar to PDO transmission, SYNC objects are transmitted based on the producer-consumer model. The SYNC producer sends a synchronous frame, and other nodes in the CAN network receive the synchronous frame as consumers without providing any feedback. In a CAN network, only one SYNC generator can be activated. Figure 3-19 shows the transmission framework of SYNC objects.

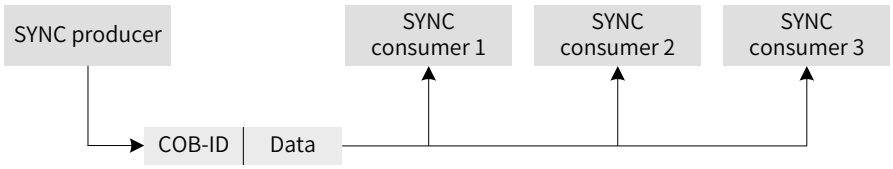


Figure 3-19 Synchronization transmission framework

The transmission of synchronous PDOs is closely related to the synchronous frames.

- For the synchronous RPDO, the PDO received will be updated to the application in the next SYNC.
- The synchronous TPDO can be transmitted in cyclic/acyclic synchronization mode.

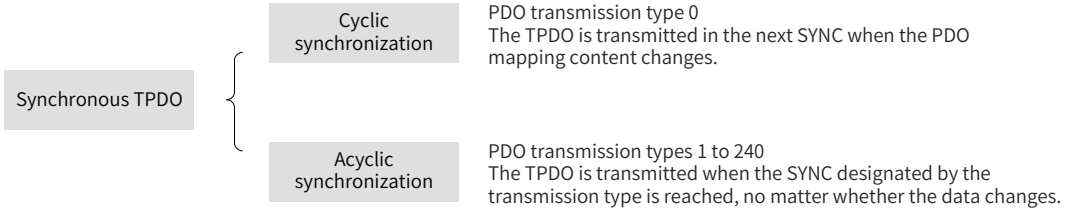


Figure 3-20 Description of synchronous TPDO

Figure 3-21 shows the synchronous transmission model.

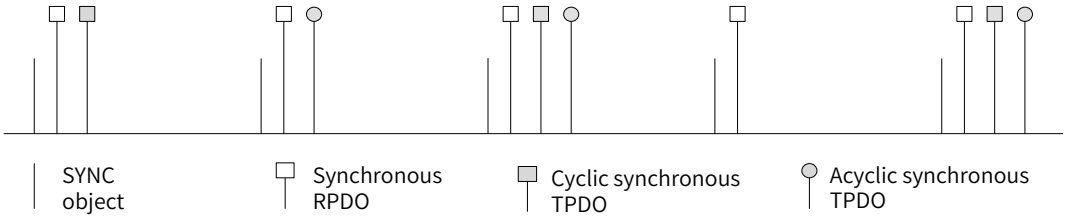


Figure 3-21 Synchronous transmission model

Example:

The transmission types of RPDO1, RPDO2, TPDO1, and TPDO2 are 0, 5, 0, and 20 respectively. Once RPDO1 and RPDO2 receive the PDO, the latest PDO data will be updated to the corresponding application in the next SYNC. TPDO1 will be sent in the next SYNC once the mapping data of TPDO1 changes. PDOs will be sent after TPDO2 experiences 20 SYNCs, no matter whether the data changes .

3.7 Emergency (EMCY) Object Service

When an error occurs in a CANopen node, the node sends an EMCY message according to the standard mechanism. The EMCY message complies with the producer-consumer model. After the node fault is sent, other nodes in the CAN network may handle the fault. As an EMCY message producer, the SV660C servo drive does not process EMCY messages of other nodes.

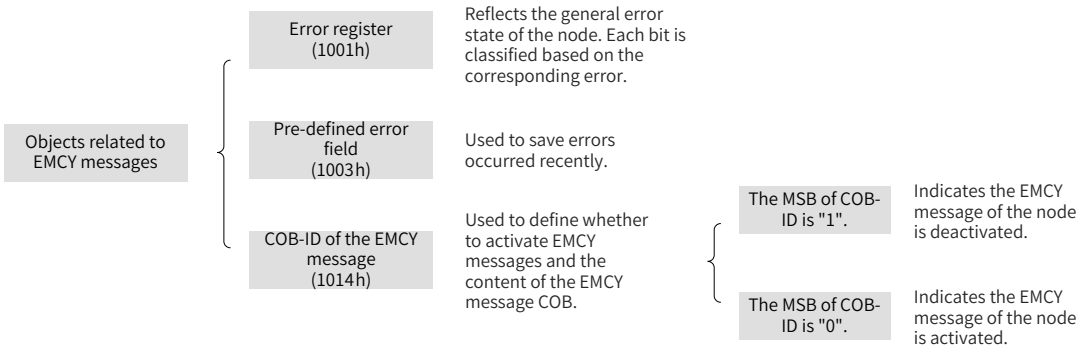


Figure 3-22 Description of objects related to EMCY messages

When a node fault occurs, the error register and the pre-defined error field must be updated no matter whether the EMCY object is activated. The content of the EMCY message complies with the following specifications.

Table 3-29 Specifications of the content of an EMCY message

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

The error register is always consistent with 1001h.

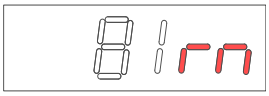
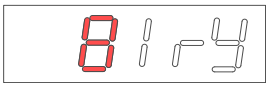
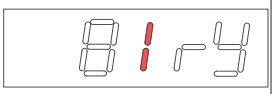
- When a communication error occurs, the error code is consistent with the one required by DS301 and the auxiliary byte is 0.
- When an error described in the DSP402 profile occurs on the servo drive, the error code is consistent with DSP402 requirements and corresponds to the object 603Fh. The auxiliary byte shows the extra description.
- When an error specified by the user occurs on the servo drive, the error code is 0xFF00 and the auxiliary byte shows the error code specified by the user.

For the definitions of the error code and auxiliary byte, see *"5 Troubleshooting"*.

4 Motion Modes

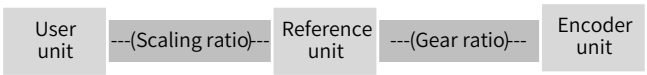
4.1 Keypad Display

Definitions of the CANopen keypad display is shown in the following table. See *SV660P Series Servo Drive User Guide* for details.

Display	Name	Display Condition	Meaning
	Run Servo running	The servo ON (S-ON) signal is active. (S-ON signal turned on)	The servo drive is running. The last digit blinks upon speed reference input.
	1 to 9 Communication state (1st bit)	The CANopen communication is established and the servo drive is ready.	Displays the status of the slave CANopen state machine in the form of characters. 1: Initialization state 2: Pre-operational state 8: Operational state 9: Stop state
	0 to 7 Control mode (2nd bit)	The CANopen communication is established and the servo drive is ready.	Displays present running mode of the servo drive in hexadecimal, without blinking. 0: Local mode 1: Profile position control 3: Profile velocity mode 4: Profile torque mode 6: Homing mode 7: Interpolation mode

4.2 Conversion Factor Setting

- Encoder unit: The direct user of the servo drive is the motor. The motor position is fed back in the form of pulses, and the encoder unit functions as the pulse unit.
- Reference unit: Applied to references controlled by the servo drive and sent under 402 protocol. The reference unit and encoder unit are converted through the gear ratio 6091h.
- User unit: For the sake of convenience, users often use the actual load displacement, speed, and acceleration units. The user unit and reference unit are converted through the user scaling ratio.



When the encoder unit, reference unit, and user unit are inconsistent, a motor error may occur during running. Therefore, set a proper conversion factor before operating the servo drive. The proportional relation between the encoder unit and user unit is established through the conversion factor.

- In the profile position mode, the following formula applies if a 23-bit motor needs to run 100 revolutions (607Ah: 100 x 8388608 p) at 400 RPM (6081: 400 x 8388608/60 p/s) with acceleration rate being 400 RPM/s (6083: 400 x 8388608/60 p/s^2) and deceleration rate being 200 RPM/s (6084: 200 x 8388608/60 p/s^2) under a gear ratio of 1:1:

  Acceleration time t_{up} =Δ6081/Δ6083 =1 (s); Deceleration time t_{down} =Δ6081/Δ6084 =2 (s)

4.2.1 Conversion Factor Setting

1 Gear ratio 6091h

The gear ratio indicates the motor displacement (in encoder unit) corresponding to the load shaft displacement of one reference unit.

The gear ratio is comprised of the numerator 6091-01h and denominator 6091-02h. It can be used to establish a proportional relation between the load shaft displacement (in reference unit) and motor displacement (in encoder unit).

Motor displacement (in encoder unit) = Load shaft displacement (in reference unit) x Gear ratio

The motor is connected to the load through the reducer and other mechanical transmission mechanism. The gear ratio therefore is related to the mechanical reduction ratio, mechanical dimensions and motor resolution. The calculation formula is as follows.

$$\text{Gear ratio} = \frac{\text{Motor resolution}}{\text{Load resolution}}$$

Index 6091h	Name	Gear ratio			Data Structure	ARR	Data Type	Uint32
	Access	RW	Mapping	YES	Data Range	OD data range	Default	OD default value

Used to set the proportional relation between the load shaft displacement designated by the user and the motor shaft displacement.

- ◆ Note: The gear ratio is within the following range:
  0.001 x Encoder resolution/10000 to 4000 x Encoder resolution/10000
- ◆ If the preceding range is exceeded, Er.B03 (Electronic gear ratio overlimit) occurs.
- ◆ The relation between the motor position feedback (in encoder unit) and the load shaft position feedback (in reference unit) is as follows.

  Motor position feedback (in encoder unit) = Load shaft position feedback (in reference unit) x Gear ratio
- ◆ The relation between the motor speed (RPM) and the load shaft speed (p/s) is as follows.

$$\text{Motor speed (RPM)} = \frac{\text{Load shaft speed x Gear ratio (6091h)}}{\text{Encoder resolution}} \times 60$$

- ◆ The relation between the motor acceleration (in RPM/ms) and the load shaft acceleration (in reference unit/s^2) is as follows.

$$\text{Motor acceleration} = \frac{\text{Load shaft acceleration x Gear ratio (6091h)}}{\text{Encoder resolution}} \times 1000/60$$

Sub-index 0	Name	Number of entries			Data Structure	-	Data Type	Uint8
	Access	RO	Mapping	NO	Data Range	2	Default	2

Sub-index 1	Name	Motor revolutions			Data Structure	-	Data Type	Uint32
	Access	RW	Mapping	RPDO	Data Range	0 to 4294967295	Default	1

Sub-index 2	Name	Shaft revolutions			Data Structure	-	Data Type	Uint32
	Access	RW	Mapping	RPDO	Data Range	0 to 4294967295	Default	1

2 Scaling ratio (user ratio)

The scaling ratio indicates the motor displacement (in reference unit) corresponding to the load shaft displacement of one user.

The scaling ratio is set by the host controller user. The proportional relation between the load shaft displacement (in user unit) and motor displacement (in reference unit) can be established through the scaling ratio:

Motor displacement (in reference unit) = Load shaft displacement (in user unit) x Scaling ratio

4.2.2 607Eh: Polarity

607Eh is used to set polarity of position references in the standard position mode and interpolation mode and the polarity of velocity references in the standard velocity mode.

Index 607Eh	Name	Polarity			Data Structure	VAR	Data Type	Uint8
	Access	RW	Mapping	YES	Data Range	OD data range	Default	0

Defines the polarity of position or velocity references.

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Position reference polarity	Velocity reference polarity	Torque reference polarity	N/A	N/A	N/A	N/A	N/A

Bit7 = 1: Indicates the position reference is multiplied by "-1" and the motor direction is reversed in the standard position mode or interpolation mode.

Bit6 = 1: Indicates the speed reference (60FFh) is multiplied by "-1" and the motor direction is reversed in the velocity mode.

Bit5 = 1: Indicates the torque reference (6071h) is multiplied by "-1" and the motor direction is reversed in the torque mode.

N/A: Not defined

4.3 Servo Status Control

4.3.1 CiA402 Servo State Machine

The SV660C CANopen servo drive can run in the specified status only when it is instructed according to the process defined by CiA402 protocol.

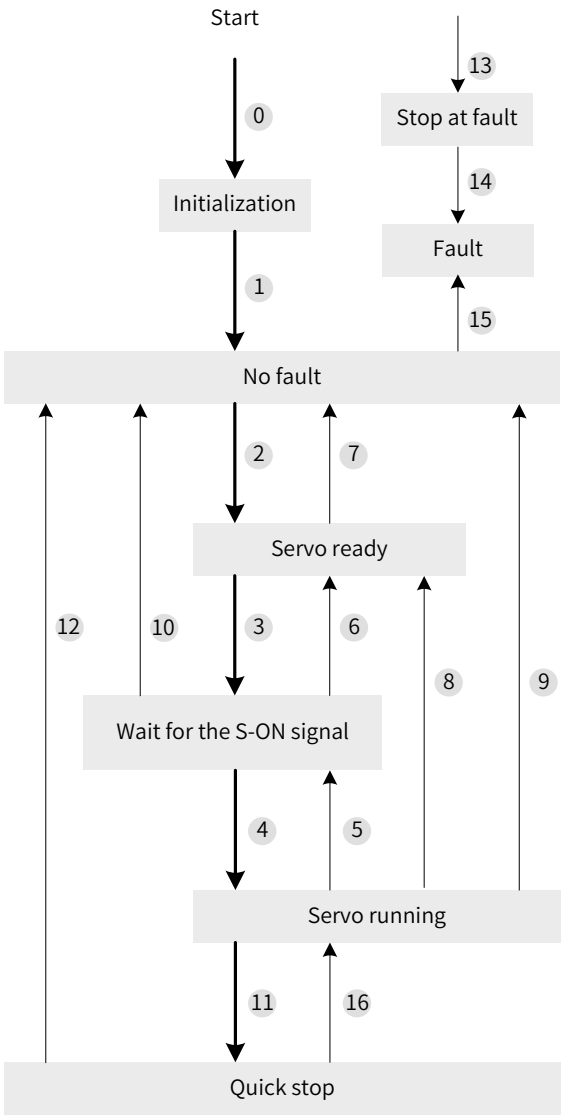


Figure 4-1 State switchover of CiA402 state machine

See the following table for descriptions of different status.

Table 4-1 Descriptions of different status

State	Description
Initialization	Initialization of the servo drive and internal self-check are done. Parameters cannot be set. Functions cannot be executed.
No fault	No fault exists in the servo drive or the fault is cleared. The servo drive parameters can be set.
Servo ready	The servo drive is ready and "rdy" is displayed on the keypad. The servo drive parameters can be set.

State	Description
Wait for the S-ON signal	The servo drive waits for the S-ON signal and the keypad displays "rdy." The servo drive parameters can be set.
Servo running	The servo drive is running properly and a certain operation mode is enabled. The motor is energized and starts running when the speed reference is not 0. Parameters modifiable during running can be set only.
Quick stop	Quick stop is activated and the servo drive is in the process of quick stop. Parameters modifiable during running can be set only.
Stop at fault	A fault occurs and the servo drive is in the process of stop. Parameters modifiable during running can be set only.
Fault	The servo drive stops at fault and all the servo drive functions are inhibited. Parameters can be modified for the convenience of troubleshooting. Reset the resettable faults through writing 0x80 to the control word 6040h after parameter modification.

The relation between status switchover and control commands is as follows.

Table 4-2 Relation between status switchover and control commands

CiA402 Status Switchover		Control Word 6040h	Bit0 to Bit9[1] of Status Word 6041h
0	Power-on → Initialization	Natural transition, no control command required	0x0000
1	Initialization → No fault	Natural transition, no control command required If an error occurs during initialization, the servo drive directly enters status 13.	0x0250
2	No fault → Servo ready	0x06	0x0231
3	Servo ready → Wait for the S-ON signal	0x07	0x0233
4	Wait for the S-ON signal → Servo running	0x0F	0x0237
5	Servo running → Wait for the S-ON signal	0x07	0x0233
6	Wait for the S-ON signal → Servo ready	0x06	0x0231
7	Servo ready → No fault	0x00	0x0250
8	Servo running → Servo ready	0x06	0x0231
9	Servo running → No fault	0x00	0x0250
10	Wait for the S-ON signal → No fault	0x00	0x0250
11	Servo running → Quick stop	0x02	0x0217
12	Quick stop → No fault	Set 605A to a value between 0 to 3. Natural transition applies after stop and no control command is required.	0x0250
13	→ Stop at fault	Once a fault occurs in any state other than "fault", the servo drive automatically switches to the stop-at-fault state.	0x021F
14	Stop at fault → Fault	Natural transition applies after stop at fault, requiring no control command.	0x0218

CiA402 Status Switchover		Control Word 6040h	Bit0 to Bit9[1] of Status Word 6041h
15	Fault → No fault	0x80 The bit7 is rising edge-triggered. The bit7 is kept to 1 and other control commands are invalid.	0x0250
16	Quick stop → Servo running	Set 605A to a value between 5 to 7. 0x0F will be sent after stop.	0x0237

[1] bit10 to bit15 (bit14 is meaningless) of status word 6041h are related to the running state of the servo drive in different modes and are set to 0 in the preceding table. See each operation mode for specific status of the bits.

4.3.2 Control Word 6040h

Index 6040h	Name	Control word					Data Structure	VAR	Data Type	Uint16
	Access	RW	Mapping	YES	Related Mode	All	Data Range	0 to 65535	Default	0

Defines the control command.

Bit	Name	Description
0	Servo ready	0: Invalid 1: Valid
1	Main circuit switch-on	0: Invalid 1: Valid
2	Quick stop	0: Invalid 1: Valid
3	Servo running	0: Invalid 1: Valid
4 to 6	-	Related to the servo drive operation modes.
7	Fault reset	Fault reset is implemented for faults and warnings that can be reset. ◆ The bit7 is rising edge-triggered. ◆ The bit7 is kept to 1 and other control commands are invalid.
8	Halt	0: Invalid 1: Valid
9 to 10	N/A	Reserved
11 to 15	Manufacturer-defined	Reserved, undefined

Note:

◆ All bits in the control word constitute a control command. One bit is meaningless if it is assigned separately.

◆ The meanings of bit0 to bit3 and bit7 are the same in each operation mode. The servo drive can switch to the preset status according to the CiA402 state machine switchover process only when commands are sent in sequence. Each command corresponds to a certain status.

◆ bit4 to bit6 are related to each operation mode (see the control commands in different modes for details).

4.3.3 Status Word 6041h

Index 6041h	Name	Status word					Data Structure	VAR	Data Type	Uint16
	Access	RO	Mapping	TPDO	Related Mode	All	Data Range	0 to 65535	Default	-

Indicates the servo drive status.

Bit	Name	Description
0	Servo ready	1: Valid, 0: Invalid
1	S-ON enable	1: Valid, 0: Invalid
2	Servo running	1: Valid, 0: Invalid
3	Fault	1: Valid, 0: Invalid
4	Main circuit switch-on	1: Valid, 0: Invalid
5	Quick stop	0: Valid, 1: Invalid
6	S-ON disable	1: Valid, 0: Invalid
7	Warning	1: Valid, 0: Invalid
8	Manufacturer-defined	Reserved, undefined
9	Remote control	0: Invalid 1: Valid, control word activated
10	Target reached	0: Target position not reached 1: Target position reached
11	Internal limit active	1: Valid, 0: Invalid
12 to 13	Operation mode-specific	Related to the operation mode.
14	Manufacturer-defined	Undefined
15	Home find	1: Valid, 0: Invalid

Note:

- ◆ All bits in the control word constitute a control command to reflect present servo state. One bit is meaningless if it is read separately.
- ◆ The meanings of bit0 to bit9 are the same in each operation mode. After commands in 6040h are sent in sequence, the servo drive returns a feedback on the servo state.
- ◆ bit12 and bit13 are related to each operation mode (see control commands in different modes for details).
- ◆ The meanings of bit10, bit11, and bit15 are the same in each operation mode. These three bits indicate the servo status after a certain servo mode is executed.

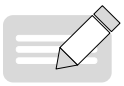
4.3.4 Stop Mode

SV660C CANopen supports the following six stop modes.

- ■ Stop at S-ON OFF
- ■ Stop at fault
- ■ Stop at overtravel
- ■ Emergency stop
- ■ Quick stop
- ■ Halt

1) Stop at S-ON OFF

When the S-ON signal is turned off, the stop mode is defined by H02-05 (object dictionary 2002-06h), which is the same as the standard SV660P servo drive.

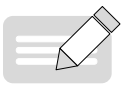


NOTE

For the DO configured with brake, the stop mode is forcibly set to "Zero speed+DB".

2) Stop at fault

When a fault or warning occurs, the servo drive enters the stop-at-fault state automatically. The stop mode is defined by H02-06 (object dictionary 2002-07h), H02-07 (object dictionary 2002-08h), and H02-08 (object dictionary 2002-09h), which is the same as the standard SV660P servo drive.

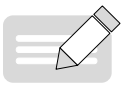


NOTE

For the DO configured with brake, the stop mode is forcibly set to "DB+DB" upon No. 1 fault or "Zero speed+DB" upon No. 2 fault.

3) Stop at overtravel

When overtravel occurs, the stop mode is defined by H02-07 (object dictionary 2002-08h), which is the same as the standard SV660P servo drive.



NOTE

For the DO configured with brake, the stop mode is forcibly set to "Zero speed+Position lock".

4) Emergency stop

Use DI function 34 (FunIN.34: EmergencyStop) to stop at zero speed, keeping position lock state.

5) Quick stop

Execute quick stop through writing 0x02 to the control word 6040h in the non-fault state. You can define the stop mode through 605A.

Index 605Ah	Name	Quick stop option code					Data Structure	VAR	Data Type	Int16
	Access	RW	Mapping	YES	Related Mode	All	Data Range	0 to 7	Default	2

Defines the quick stop mode.

Value	Stop Mode
0	Coast to stop, keeping de-energized state
1	Ramp to stop as defined by 6084h/609Ah (HM), keeping de-energized state after stop
2	Ramp to stop as defined by 6085h, keeping de-energized state after stop
3	Stop at emergency stop torque as defined by 2007-10h (H07-15), keeping de-energized state after stop
4	N/A
5	Ramp to stop as defined by 6084h/609Ah (HM), keeping position lock state after stop
6	Ramp stop as defined by 6085h, keeping position lock state after stop
7	Stop at emergency stop torque as defined by 2007-10h (H07-15), keeping position lock state after stop

Note: When the brake is enabled and 605Ah is set to 0, the stop mode is forcibly set to "Stop at zero speed, keeping de-energized state after stop".

6) Halt

The bit8 (Halt) of 6040h is used to halt the servo drive. You can set the halt mode in the object dictionary 605D.

Index 605Dh	Name	Halt stop option code					Data Structure	VAR	Data Type	Int16
	Access	RW	Mapping	NO	Related Mode	All	Data Range	0 to 3	Default	1

Used to set the deceleration mode of the servo motor from rotating to stop and the servo motor status after halt.

Defines the halt mode.

Value	Stop Mode
1	Ramp to stop as defined by 6084h/6087h (HM: 609Ah), keeping position lock state
2	Ramp to stop as defined by 6085h/6087h, keeping position lock state
3	Stop at the emergency stop torque, keeping position lock state

4.4 Trial Running Procedure

Step	Operation	Description
1	Confirming the installation	Perform installation according to requirements in the appendix (do to install the motor to the machine before trial run). For details, see the *SV660P Series Servo Drive Advanced User Guide*.
2	Confirming cable connections	Connect the encoder cables, motor power cables, and terminal cables according to *"2 Wiring"*. For details, see *SV660P Series Servo Drive Advanced User Guide*.
3	Confirming the supply voltage	Ensure the power input complies with the servo drive specifications.
4	Confirming communication parameter settings	Confirm system settings in *"3.2 System Settings"*.
5	Confirming the motor model	Ensure the motor model matches the servo drive model.
6	Powering on the servo drive	Ensure that no warning is reported during power-on.
7	Setting parameters	Set related objects. For details, see *"4.5 Overview of Operation Modes"*.
8	Performing trial run	Apply the profile velocity mode and ensure a proper forward/reverse run under low-speed references. For detail, see *"4.9 Profile Velocity Mode"*.
9	Adjusting parameters	Adjust gain parameters. Check the waveform through the oscilloscope in the software tool and adjust the gain accordingly.
10	Running	-

4.5 Overview of Operation Modes

SV660C CANopen supports five kinds of operation modes.

The pre-operational mode of the servo drive is set in 6060h. The present operational mode of the servo drive can be viewed in 6061h.

1) Modes of operation (6060h)

Index 6060h	Name	Modes of operation					Data Structure	VAR	Data Type	Int8
	Access	RW	Mapping	YES	Related Mode	All	Data Range	0 to 7	Default	0

Defines the servo drive operation mode.

Bit	Description	Description
0	N/A	Reserved
1	Profile position mode	For parameter settings, see *"4.6 Profile Position Mode"*.
2	N/A	Reserved
3	Profile velocity mode	For parameter settings, see *"4.9 Profile Velocity Mode"*.
4	Profile torque mode	For parameter settings, see *"4.10 Profile Torque Mode"*.
5	N/A	Reserved
6	Homing mode	For parameter settings, see *"4.7 Homing Mode"*.
7	Interpolation mode	For parameter settings, see *"4.8 Interpolation Mode"*.

◆ If an unsupported operation mode is selected through the SDO, an SDO error will be returned.

◆ If an unsupported operation mode is selected through the PDO, the modification on the operation mode will be invalid.

2) Modes of operation display (6061h)

Index 6061h	Name	Modes of operation display					Data Structure	VAR	Data Type	Int8
	Access	RO	Mapping	TPDO	Related Mode	All	Data Range	0 to 7	Default	-

Indicates the actual operation mode.

Bit	Description	Description
0	N/A	Reserved
1	Profile position mode	For parameter settings, see *"4.6 Profile Position Mode"*.
2	N/A	Reserved
3	Profile velocity mode	For parameter settings, see *"4.9 Profile Velocity Mode"*.
4	Profile torque mode	For parameter settings, see *"4.10 Profile Torque Mode"*.
5	N/A	Reserved
6	Homing mode	For parameter settings, see *"4.7 Homing Mode"*.
7	Interpolation mode	For parameter settings, see *"4.8 Interpolation Mode"*.

3) Precautions for mode switchover

■ The servo drive, regardless of the servo state, switches the mode directly without ramp-to-stop. The references non-executed before switchover will be aborted.

4.6 Profile Position Mode

If the profile position mode meets certain conditions, user displacement references can be received in real time. The acceleration time, deceleration time, maximum speed, and displacement of each displacement reference can be controlled independently, and the mode of transition between references can be modified in real time. The profile position mode is often used in point-to-point positioning and the operation curve is planned by the servo drive. The position control, speed control, and torque control are executed by the servo drive.

4.6.1 Control Block Diagram

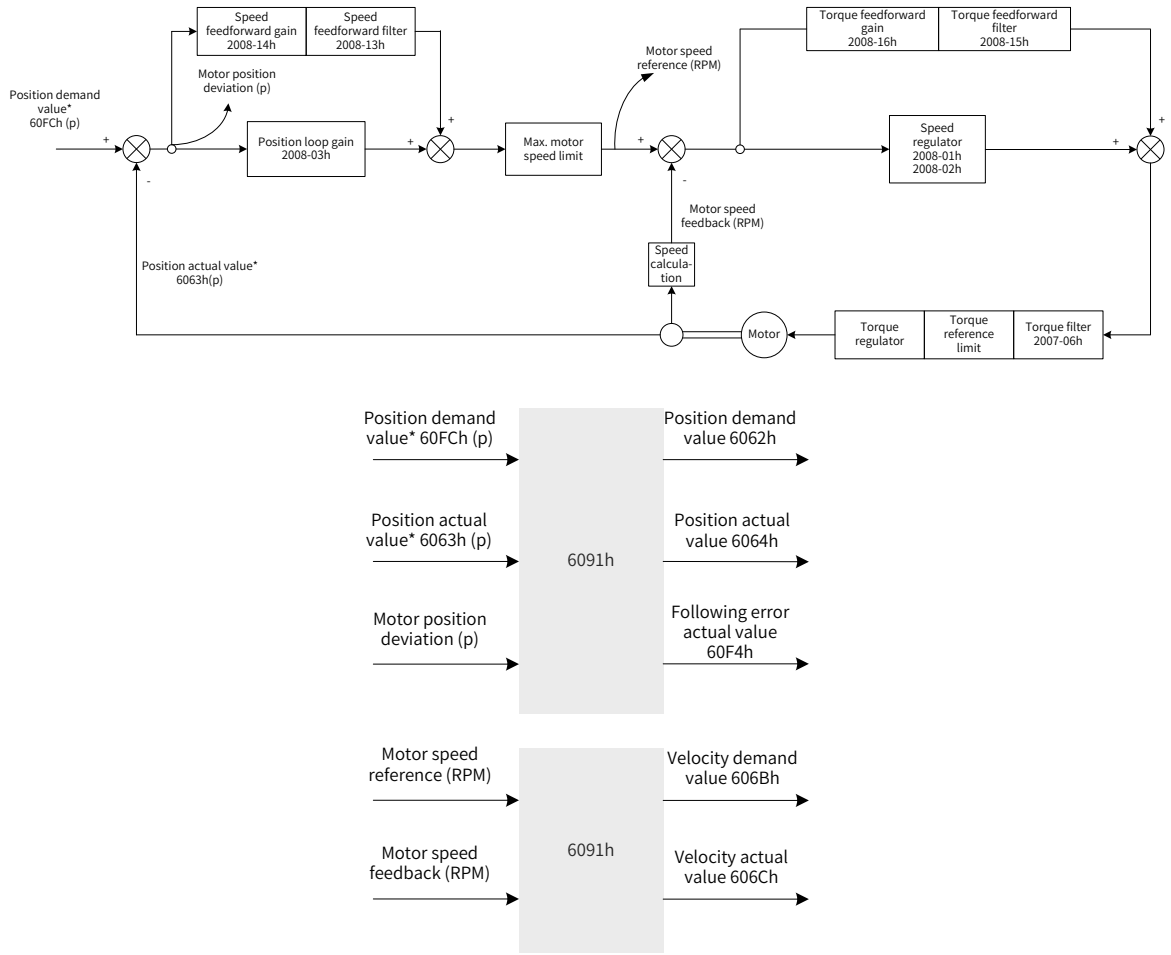


Figure 4-2 Control block diagram of the profile position mode

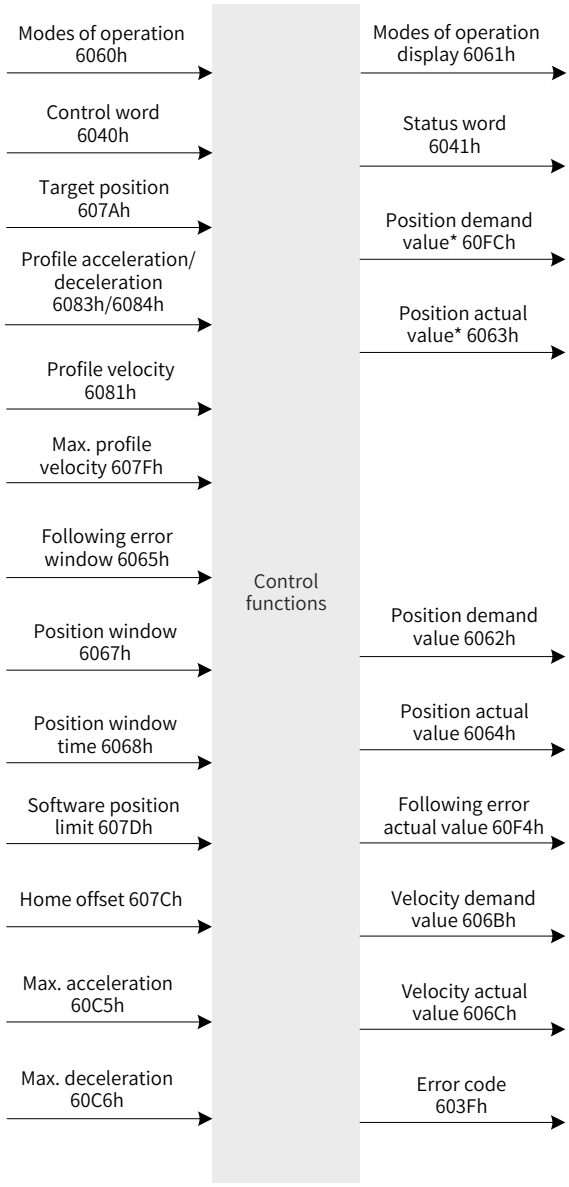


Figure 4-3 Input/output objects in the profile position mode

Displacement curve planning involves the target position 607Ah (in reference unit), profile velocity 6081h (in reference unit), profile acceleration 6083h (in reference unit), and profile deceleration 6084h (in reference unit).

References of the host controller, which are input in reference unit, are in encoder unit after being limited and converted through the conversion factor.

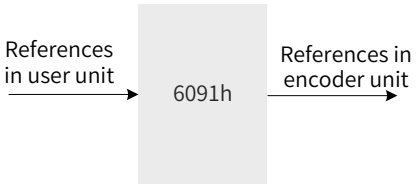


Figure 4-4 Description of the reference unit

The processing of the target position, profile velocity, and profile acceleration/deceleration by the servo drive is shown in Figure 4-5, Figure 4-6, and Figure 4-7.

Software limit: The software limit can be enabled by setting 0x200A-02h to 1 (H0A_01). The software limit is disabled by default (200A-02h = 0). After the software limit is enabled, the motor stops and reports an overtravel warning upon reaching the software limit, and bit11 of 6041h is set to 1. In this case, send a reverse run command to release the servo drive from the limit state and zero out bit11 of 6041h. When the external DI limit switch and internal software position limit are both activated, the limit state is determined by the external DI limit switch.

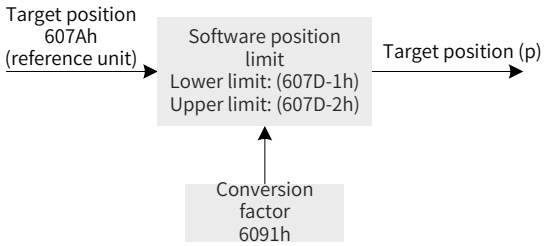


Figure 4-5 Target position 607Ah - internal software position limit

The profile velocity (6081h) is used to set the maximum speed when the displacement reference is running, which cannot exceed the maximum velocity (607Fh) set by the user and the corresponding maximum motor speed after conversion. The processing block diagram is as follows.

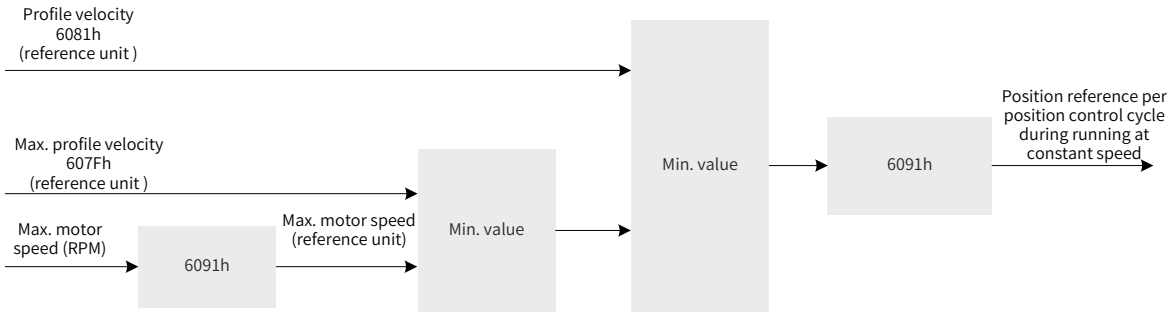


Figure 4-6 Profile velocity 6081h - speed limit

The profile acceleration (6083h) and profile deceleration (6084h) are used to set the acceleration/deceleration rate when the displacement reference is running, which cannot exceed the maximum acceleration (60C5h) and maximum deceleration (60C6h) set by the user. The processing block diagram is as follows.

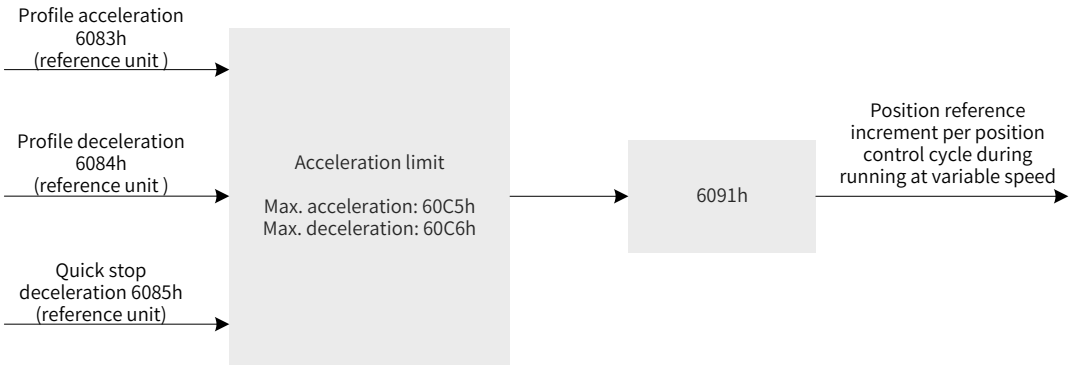


Figure 4-7 Profile acceleration limit

Description of acceleration/deceleration settings:

The following formula applies if a 23-bit motor needs to run at 400 RPM (6081: 400 x 8388608/60) with acceleration rate being 400 RPM/s (6083: 400 x 8388608/60) and deceleration rate being 200 RPM/s (6084: 200 x 8388608/60) under a gear ratio of 1:1:

Acceleration time t_{up} =Δ6081/Δ6083 =1 (s); Deceleration time t_{down} =Δ6081/Δ6084 =2 (s)

4.6.2 Related Object Settings

1) Positioning completed

Index 6067h	Name	Position window					Data Structure	VAR	Data Type	Uint32
	Access	RW	Mapping	YES	Related Mode	pp/hm/ip	Data Range	0 to 4294967295	Default	734p
Subindex: 00 When the position deviation (60F4h) of the reference unit is smaller than 6067h and the time reaches 6068h, bit10 of 6041h is set to 1. When either condition is not met, the position reached is invalid.										

Index 6068h	Name	Position window time					Data Structure	VAR	Data Type	Uint16
	Access	RW	Mapping	YES	Related Mode	pp/hm/ip	Data Range	0 to 65535	Default	0 ms
Subindex: 00 When the position deviation (60F4h) of the reference unit is smaller than 6067h and the time reaches 6068h, bit10 of 6041h is set to 1. When either condition is not met, the position reached is invalid.										

2) Following error window detection

Index 6065h	Name	Following error window					Data Structure	VAR	Data Type	Uint32
	Access	RW	Mapping	YES	Related Mode	pp/hm/ip	Data Range	0 to 4294967295	Default	3435868
Subindex: 00 When the position deviation is larger than this value, Er.B00 occurs.										

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
603Fh	00h	Error code	RO	TPDO	Uint16	-	0 to 65535	-
6040h	00h	Control word	RW	YES	Uint16	-	0 to 65535	0
6041h	00h	Status word	RO	TPDO	Uint16	-	0 to 65535	-
6060h	00h	Modes of operation	RW	YES	Int8	-	0 to 7	0
6061h	00h	Modes of operation display	RO	TPDO	Int8	-	0 to 7	-
6062h	00h	Position demand value	RO	TPDO	Int32	Reference unit	-2^{31} to $+(2^{31}-1)$	-

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
6063h	00h	Position actual value*	RO	TPDO	Int32	Encoder unit	-2^{31} to $+(2^{31}-1)$	-
6064h	00h	Position actual value	RO	TPDO	Int32	Reference unit	-2^{31} to $+(2^{31}-1)$	-
6065h	00h	Following error window	RW	YES	Uint32	Reference unit	0 to $(2^{32}-1)$	3435868
6067h	00h	Position window	RW	YES	Uint32	Reference unit	0 to $(2^{32}-1)$	734
6068h	00h	Position window time	RW	YES	Uint16	ms	0 to 65535	0
606Bh	00h	Velocity demand value	RO	TPDO	Int32	Reference unit	-2^{31} to $+(2^{31}-1)$	-
606Ch	00h	Velocity actual value	RO	TPDO	Int32	Reference unit	-2^{31} to $+(2^{31}-1)$	-
607Ah	00h	Target position	RW	YES	Int32	Reference unit	-2^{31} to $+(2^{31}-1)$	0
607Dh	01h	Min. position limit	RW	YES	Int32	Reference unit	-2^{31} to $+(2^{31}-1)$	-2^{31}
	02h	Max. position limit	RW	YES	Int32	Reference unit	-2^{31} to $+(2^{31}-1)$	$2^{31}-1$
607Ch	00h	Home offset	RW	YES	Int32	Reference unit	-2^{31} to $+(2^{31}-1)$	0
6081h	00h	Profile velocity	RW	YES	Uint32	Reference unit	0 to $(2^{32}-1)$	1747627
6083h	00h	Profile acceleration	RW	YES	Uint32	Reference unit	0 to $(2^{32}-1)$	174762666
6084h	00h	Profile deceleration	RW	YES	Uint32	Reference unit	0 to $(2^{32}-1)$	174762666
60F4h	00h	Following error actual value	RO	TPDO	Int32	p	-2^{31} to $+(2^{31}-1)$	-
60FCh	00h	Position demand value	RO	TPDO	Int32	p	-2^{31} to $+(2^{31}-1)$	-
2005h	05h	First-order low-pass filter time constant	RW	YES	Uint16	ms	0 to 6553.5	0.0
	07h	Moving average filter time constant	RW	YES	Uint16	ms	0 to 128.0	0.0
2007h	06h	Torque reference filter time constant	RW	YES	Uint16	ms	0 to 30.00	0.79

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
2008h	01h	Speed loop gain	RW	YES	Uint16	Hz	0.1 to 2000.0	25.0
	02h	Speed loop integral time constant	RW	YES	Uint16	ms	0.15 to 512.00	31.83
	03h	Position loop gain	RW	YES	Uint16	Hz	0.0 to 2000.0	40.0
	13h	Speed feedback filter time constant	RW	YES	Uint16	ms	0.00 to 64.00	0.50
	14h	Speed feedforward gain	RW	YES	Uint16	%	0.0 to 100.0	0.0
	15h	Torque feedforward filter time constant	RW	YES	Uint16	ms	0.00 to 64.00	0.50
	16h	Torque feedforward gain	RW	YES	Uint16	%	0.0 to 200.0	0.0

4.6.3 Control Commands in the Profile Position Mode

Table 4-3 Relation between status switchover and control commands

CiA402 Status Switchover		Control Word 6040h	bit0 to bit9[1] of 6041h
0	Power-on → Initialization	Natural transition, no control command required	0x0000h
1	Initialization → No fault	Natural transition, no control command required If an error occurs during initialization, the servo drive directly enters status 13.	0x0250h
2	No fault → Servo ready	0x06h	0x0231h
3	Servo ready → Wait for the S-ON signal	0x07h	0x0233h
4	Wait for the S-ON signal → Servo running	0x0Fh	0x0237h
5	Servo running → Wait for the S-ON signal	0x07h	0x0233h
6	Wait for the S-ON signal → Servo ready	0x06h	0x0231h
7	Servo ready → No fault	0x00h	0x0250h
8	Servo running → Servo ready	0x06h	0x0231h
9	Servo running → No fault	0x00h	0x0250h
10	Wait for the S-ON signal → No fault	0x00h	0x0250h
11	Servo running → Quick stop	0x02h	0x0217h
12	Quick stop → No fault	Set 605A to a value between 0 to 3. Natural transition applies after stop and no control command is required.	0x0250h

CiA402 Status Switchover		Control Word 6040h	bit0 to bit9[1] of 6041h
13	→ Stop at fault	Once a fault occurs in any state other than "fault", the servo drive automatically switches to the stop-at-fault state, requiring no control command.	0x021Fh
14	Stop at fault → Fault	Natural transition applies after stop at fault, requiring no control command.	0x0218h
15	Fault → No fault	0x80h The bit7 is rising edge-triggered. The bit7 is kept to 1 and other control commands are invalid.	0x0250h
16	Quick stop → Servo running	Set 605A to a value between 5 to 7. 0x0F will be sent after stop.	0x0237h

[1] bit10 to bit15 (bit14 is meaningless) of 6041h are related to the running state of the servo drive in different modes and are set to 0 in the preceding table. See the operation modes for specific status of the bits.

The control word 6040h in the profile position mode is described as follows.

Index 6040h	Name	Control word					Data Structure	VAR	Data Type	Uint16
	Access	RW	Mapping	YES	Related Mode	All	Data Range	0 to 65535	Default	-

Defines the control commands in the profile position mode.

Control Word 6040h					
Bit	bit7 to bit15	bit6	bit5	bit4	bit0 to bit3
Name	-	Position reference type	Position reference update mode [1]	Enabling new position reference (edge-triggered)	-
Value	See Table 4-2 for details.	-	-	-	See Table 4-2 for details.
Description	For details, see *"6.5.3 Profile-Defined Parameters"*.	0: Target position (607Ah) being absolute 1: Target position (607Ah) being relative	0: Not updated immediately 1: Updated immediately	0 to 1 A new displacement reference is enabled in advance. However, whether the reference can be enabled successfully depends on the servo state. 1 to 0 The bit12 of 6041h is cleared in advance. However, whether bit12 is cleared successfully depends on the servo state.	For details, see *"6.5.3 Profile-Defined Parameters"*.

NOTE

- When the servo drive meets certain conditions and the displacement reference is updated, the two attributes of the reference, namely the update mode and the reference type, are latched and unmodifiable when the displacement reference is running. Other attributes can be modified in the immediate update mode.
- The attributes of a displacement reference includes profile acceleration (6083), profile deceleration (6084), profile velocity (6081), target position (607A), reference update mode (bit5 of 6040), and reference type (bit6 of 6040).

The status word 6041h in the profile position mode is described as follows.

Index 6041h	Name	Status word					Data Structure	VAR	Data Type	Uint16
	Access	RO	Mapping	TPDO	Related Mode	All	Data Range	0 to 65535	Default	-

Indicates the servo drive state in the profile position mode.

Status Word 6041h							
Bit	bit15	bit14	bit13	bit12	bit11	bit10	bit0 to bit9
Name	Homing completed	N/A	Position deviation state	Position reference state	Software internal setting beyond the limit	Target reached	-
Value	-	-	See Table 4-2 for details.	-	-	-	See Table 4-2 for details.
Description	0: Homing not performed or done 1: Homing done and reference point found	Reserved	0: Position deviation within the following error window (6065h) 1: Position deviation beyond the following error window (6065h)	0: New displacement reference can be received by the servo drive 1: New displacement reference cannot be received by the servo drive	0: Position reference not reaching the software position limit (607Dh) 1: Position reference or position feedback reaching software internal position limit[1]	0: Target position not reached. 1: Target position reached[2].	For details, see *"6.5.3 Profile-Defined Parameters"*.

[1] The software internal position limit can be enabled according to the setting of 0x200A-02h. For details, see descriptions of 607Dh in *"6.5.3 Profile-Defined Parameters"*.

[2] When the position deviation is within the position window (6067h) and the time reaches 6068h, the target position is reached. If either condition is not met, the target position is not reached.

1) Control command sequence 1: Immediate update

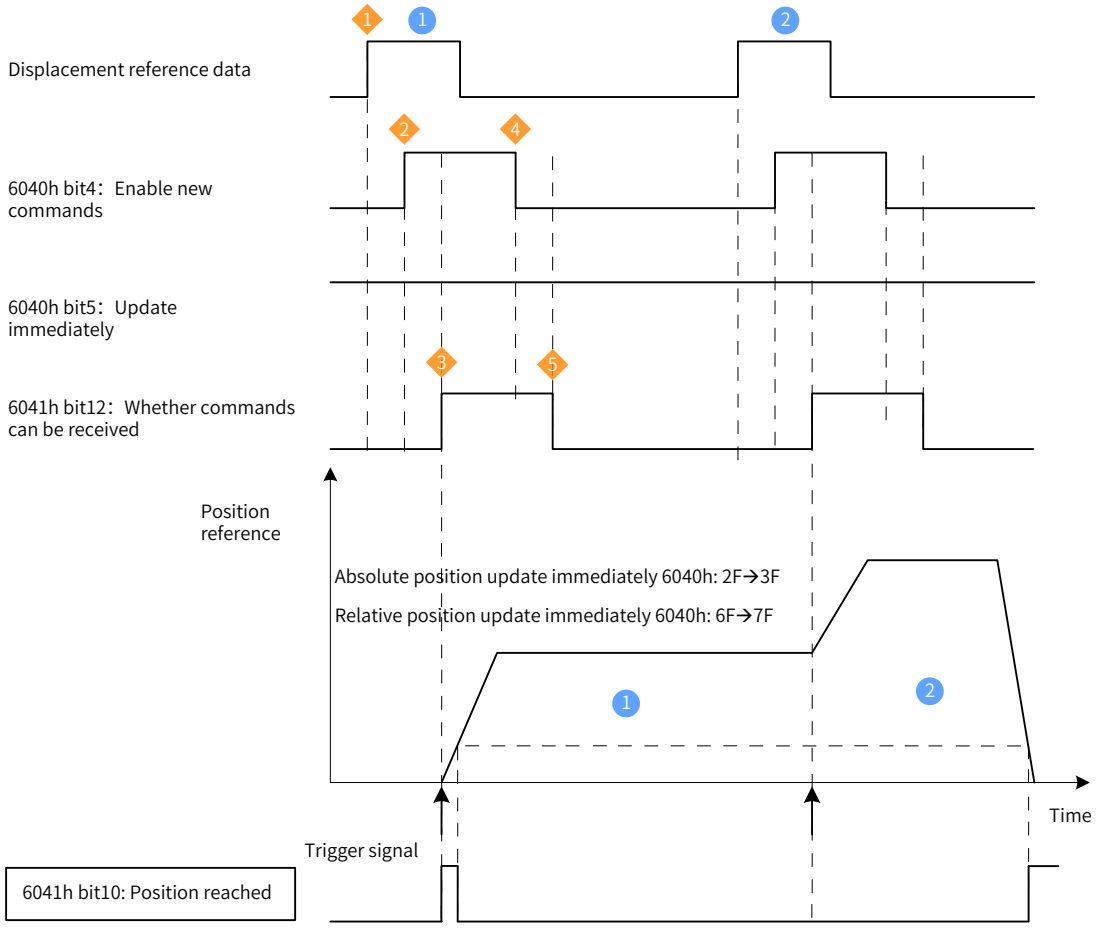


Figure 4-8 Immediate update sequence and motor running curve 1

NOTE

- A trigger signal needs to be transmitted again when any parameter of the displacement reference needs to be modified.
- ◆1 The host controller modifies other attributes (profile acceleration (6083h), profile deceleration (6084h), profile velocity (6081h), and target position (607Ah)) of the displacement reference as needed.
- ◆2 The host controller changes bit4 of 6040h from 0 to 1, informing the slave that a new displacement reference needs to be enabled.
- ◆3 After receiving the rising edge of bit4 of 6040h, the slave determines whether to receive the new displacement reference.

If bit5 of 6040h is 1 initially and bit12 of 6041h is 0, the slave can receive the new displacement reference ①. After receiving the new displacement reference, the slave changes bit12 of 6041h from 0 to 1, indicating the new displacement reference ① is received and no new displacement reference can be received.

In the mode of immediate update, the servo drive immediately executes the new displacement reference received (bit12 of 6041h changed from 0 to 1).

- ◆4 After bit12 of 6041h received by the host controller from the slave changes to 1, the host controller releases the displacement reference data and changes bit4 of 6040h from 1 to 0, indicating there is no new position reference currently.

As bit4 of 6040h is edge-triggered, this operation does not interrupt the displacement reference being executed.

- ◆5 After the slave detects bit4 of 6040h changes from 1 to 0, you can set bit12 of 6041h from 1 to 0, indicating the slave is ready to receive a new displacement reference.

In the immediate update mode, when the slave detects bit4 of 6040h changes from 1 to 0, bit12 of 6041h will be set to 0.

In the immediate update mode, if a new displacement reference ② is received when the current displacement reference ① is being executed, the displacement reference not executed in ① will not be aborted. For a relative position reference, after the second displacement reference is done, the total displacement increment equals to the sum of the target position increment (607Ah) of ① and the target position increment (607Ah) of ② . For an absolute position reference, after the second position reference is done, the user absolute position is the target position (607Ah) of ② .

Example:

Two position references updated immediately upon absolute position reference

Displacement reference ① :

- Target position 607A = 100000000 p
- 6081 = 1000 x 1048576/60 p/s (1000 RPM)

Displacement reference ② :

- Target position 607A = 200000000 p
- 6081 = 2000 x 1048576/60 p/s (2000 RPM)

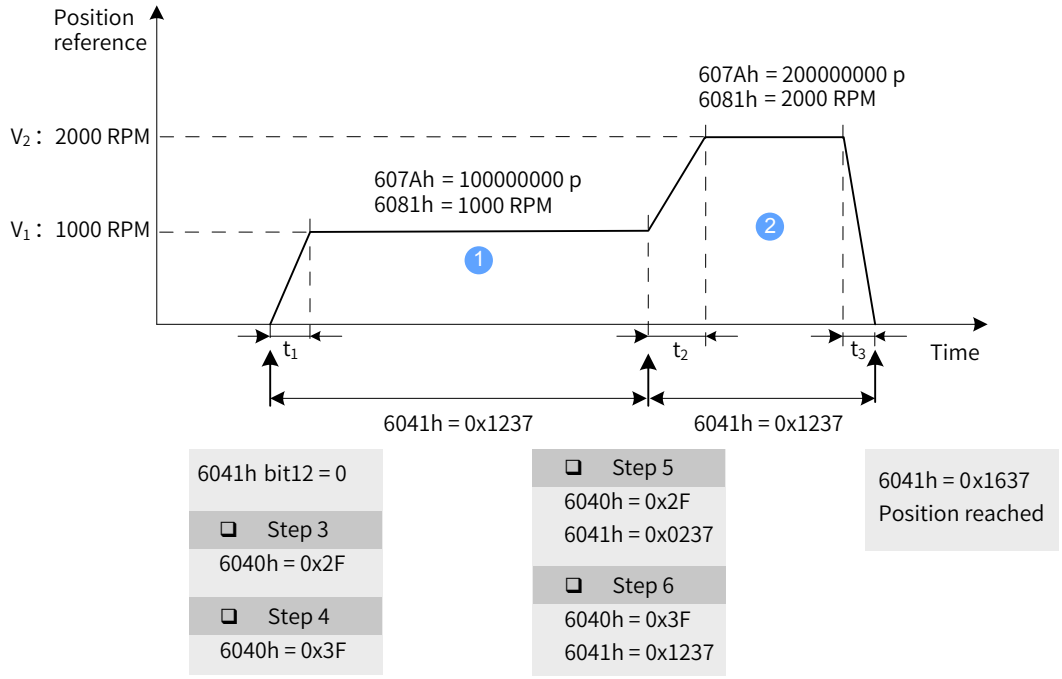


Figure 4-9 Immediate update sequence and motor running curve 2

$$t_1 = \frac{V_1}{6083h}\ s \qquad t_2 = \frac{V_2 - V_1}{6083h}\ s \qquad t_3 = \frac{V_2}{6084h}\ s$$

Operating Steps	Control Command 6040h	Status of 6041h	Description
1	0x06	0x0231	The servo drive is ready to receive a new reference.
2	0x07	0x0233	The servo drive is ready to receive a new reference and the S-ON signal can be enabled.
3	0x2F	0x0637	A new reference can be received and the servo drive is enabled (as no other position references are executed before execution of the displacement reference ① , the target position is considered to be 0 and bit12 of 6041h is 1).

Operating Steps	Control Command 6040h	Status of 6041h	Description
4	0x3F	0x1237	The servo drive already receives a new reference and is executing the reference. The target position is not reached.
If the target position (607Ah) remains unchanged, the profile velocity (6081h) needs to be modified. Perform the following operations when the displacement reference is not done.			
5	0x2F	0x0237	The bit12 of 6041h is released and the servo drive can receive new references again. Current reference is running and the target position is not reached.
6	0x3F	0x1237	The servo drive already receives a reference and is executing the reference. The target position is not reached.
If a new target position (607Ah) is not required and parameters of the current displacement reference need no modification, wait until the current displacement reference is done. After positioning is done, current position actual value* (6063h) is the target position (607Ah) and 6041h is 0x1637. If a new target displacement is required and smooth transition between positions is required, repeat operations 5 and 6 before current position reference is done.			
7	0x3F	0x1637	The target position is not reached.

2) Control command sequence 2: Non-immediate update

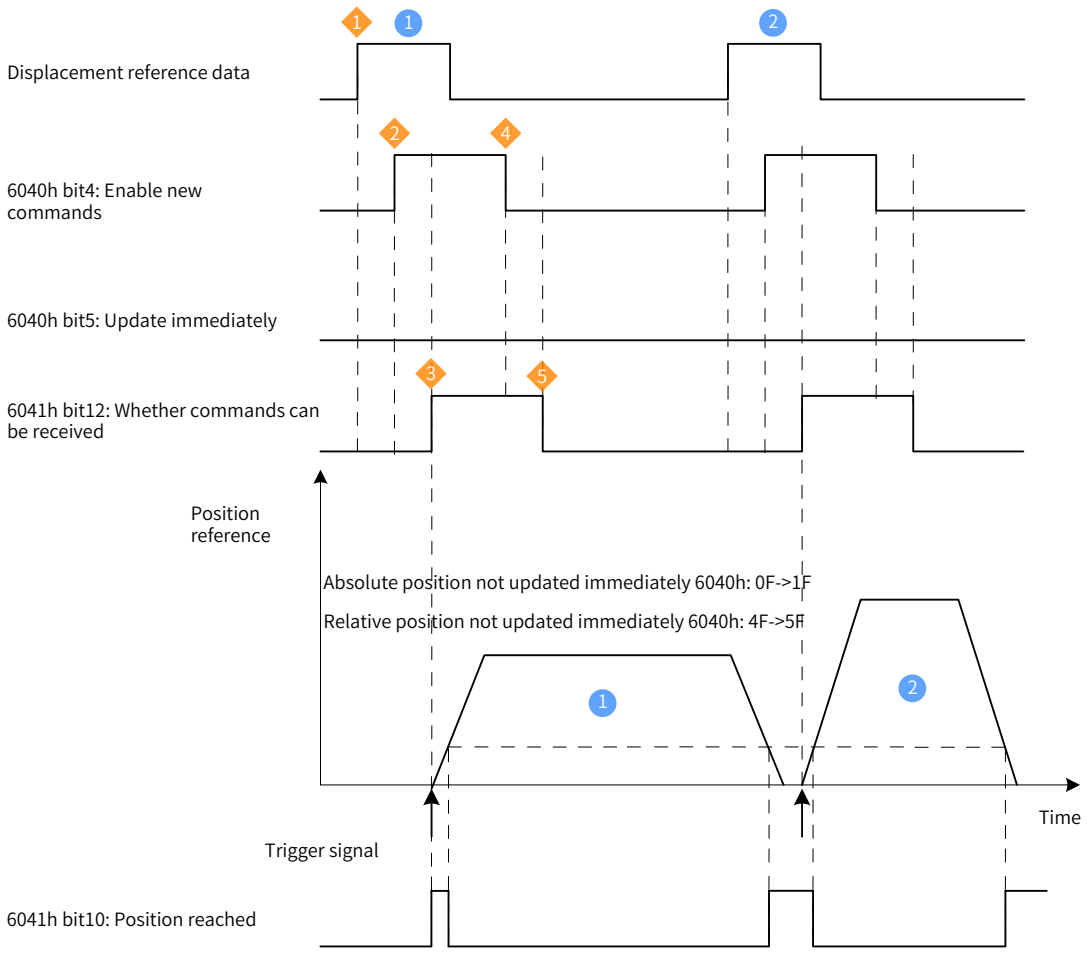


Figure 4-10 Non-immediate update sequence and motor running curve 1

NOTE

- A trigger signal needs to be transmitted again when any parameter of the displacement reference needs to be modified.
- 1 The host controller modifies other attributes (profile acceleration (6083h), profile deceleration (6084h), profile velocity (6081h), and target position (607Ah)) of the displacement reference as needed.
- 2 The host controller changes bit4 of 6040h from 0 to 1, informing the slave that a new displacement reference needs to be enabled.
- 3 After receiving the rising edge of bit4 of 6040h, the servo drive determines whether to receive the new displacement reference.

If bit5 of 6040h is 0 initially and bit12 of 6041h is 0, the slave can receive the new displacement reference 1 . After receiving the new reference, the slave changes bit12 0f 6041h from 0 to 1, indicating the new displacement reference 1 is received and no new displacement reference can be received.

- 4 After bit12 of 6041h received by the host controller changes to 1, the displacement reference data can be released and bit4 of 6040h changes from 1 to 0, indicating there is no new position reference currently.

As bit4 of 6040h is edge-triggered, this operation does not interrupt the displacement reference being executed.

- 5 After the slave detects bit4 of 6040h changes from 1 to 0, bit12 of 6041h will be released after current displacement reference is done, indicating the slave is ready to receive a new displacement reference. In the non-immediate update mode, the servo drive can execute the new reference received (bit12 of 6041h changed from 0 to 1) only after current displacement reference is executed.

Example:

Two position references not updated immediately upon absolute position reference

Displacement reference 1 :

- Target position 607A = 100000000 p
- 6081 = 1000 x 1048576/60 p/s (1000 RPM)

Displacement reference 2 :

- Target position 607A = 200000000 p
- 6081 = 2000 x 1048576/60 p/s (2000 RPM)

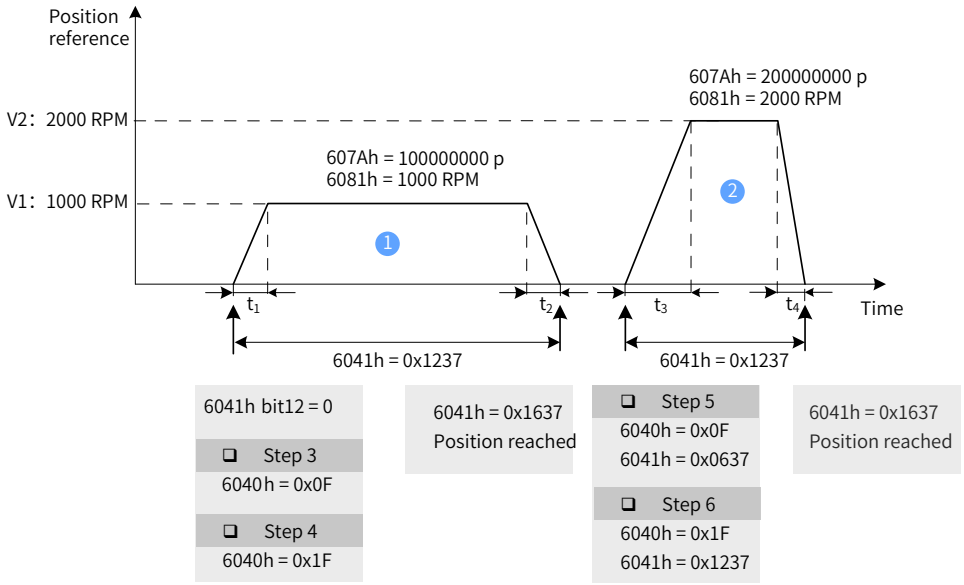


Figure 4-11 Non-immediate update sequence and motor running curve 2

$$t_1=\frac{V_1}{6083h}\ s \quad t_2=\frac{V_1}{6084h}\ s \quad t_3=\frac{V_2}{6083h}\ s \quad t_4=\frac{V_2}{6084h}\ s$$

Operating Sequence	Control Command 6040h	Status of 6041h	Description
1	0x06	0x0231	The servo drive is ready to receive a new reference.
2	0x07	0x0233	The servo drive is ready to receive a new reference and the S-ON signal can be activated.
3	0x0F	0x0637	A new reference can be received and the servo drive is enabled (as no other position references are executed before execution of the displacement reference 1 , the target position is considered to be 0 and bit12 of 6041h is 1).
4	0x1F	0x1237	The servo drive already receives a reference and is executing the reference. The target position is not reached.
Wait for the completion of the present displacement reference (6041h = 0x1637). To continue running, modify related data (607Ah, 6081h, 6083h, and 6084h) of the displacement reference as needed and repeat operations 3 and 4.			

4.6.4 Configuration Example

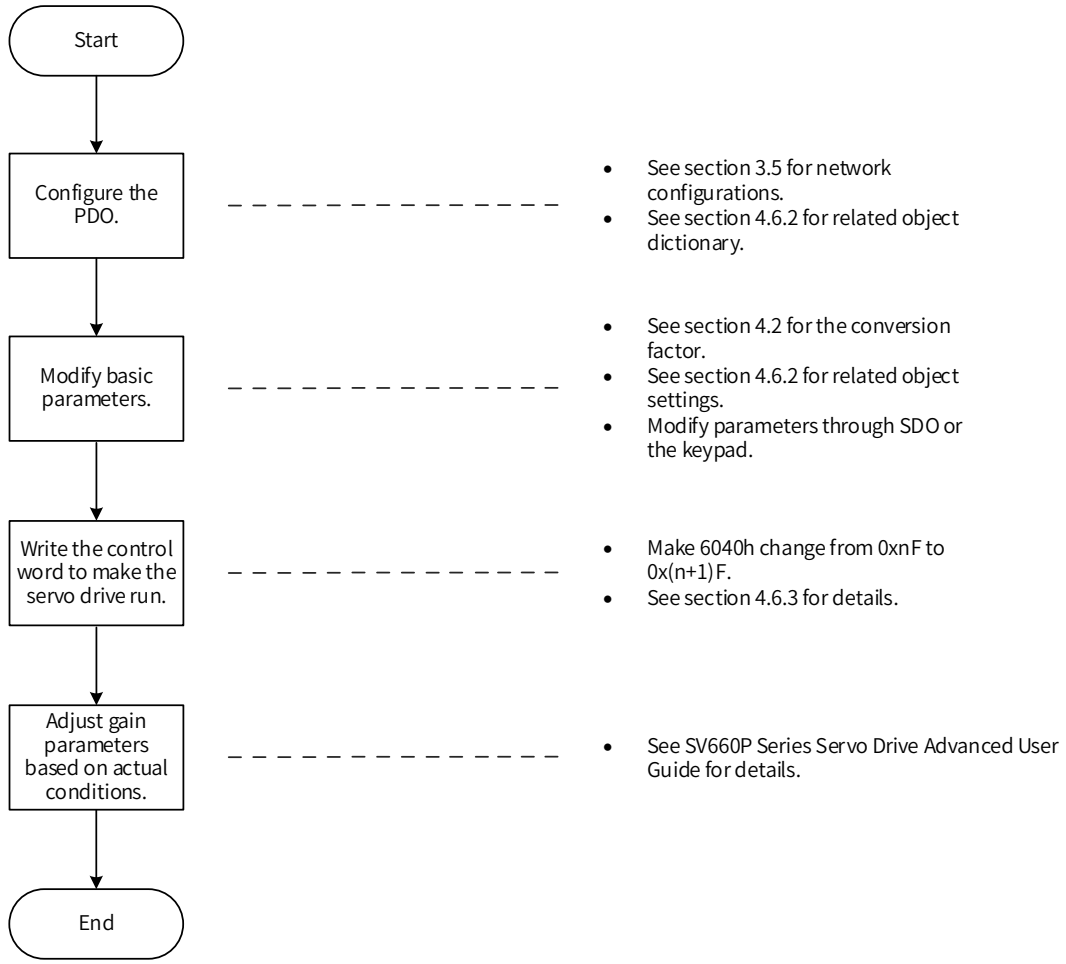


Figure 4-12 Example of setting process of profile position mode

Para. No.	Object	Mapping Object	Input	Description
H2D-32	1600h-00h	Number of RPDO1 mapping objects	2	
H2D-33	1600h-01h	6040h-00h	60400010h	The first mapping parameter of RPDO1 is 6040-00h, with a length of 16 bits.
H2D-35	1600h-02h	6060h-00h	60600008h	The second mapping parameter of RPDO1 is 6060-00h, with a length of 8 bits.
H2D-49	1601h-00h	Number of RPDO2 mapping objects	2	
H2D-50	1601h-01h	607Ah-00h	607A0020h	The first mapping parameter of RPDO2 is 607A-00h, with a length of 32 bits.
H2D-52	1601h-02h	6081h-00h	60810020h	The second mapping parameter of RPDO2 is 6081-00h, with a length of 32 bits.
H2D-66	1602h-00h	Number of RPDO3 mapping objects	2	
H2D-67	1602h-01h	6083h-00h	60830020h	The first mapping parameter of RPDO3 is 6083-00h, with a length of 32 bits.
H2D-69	1602h-02h	6084h-00h	60840020h	The second mapping parameter of RPDO3 is 6084-00h, with a length of 32 bits.
H2E-20	1A00h-00h	Number of TPDO1 mapping objects	2	
H2E-21	1A00h-01h	6041h-00h	60410010h	The first mapping parameter of TPDO1 is 6041-00h, with a length of 16 bits.
H2E-23	1A00h-02h	6061h-00h	60610008h	The second mapping parameter of TPDO1 is 6061-00h, with a length of 8 bits.
H2E-37	1A01h-00h	Number of TPDO2 mapping objects	2	
H2E-38	1A01h-01h	6064h-00h	60640020h	The first mapping parameter of TPDO2 is 6064-00h, with a length of 32 bits.
H2E-40	1A01h-02h	606Ch-00h	606C0020h	The second mapping parameter of TPDO2 is 606C-00h, with a length of 32 bits.

- Write the servo drive operation mode (6060h) to 0x01 to make it run in the profile position mode.
- Write the target position 607Ah (in reference unit, default value: 0 p).
- Set the constant running speed of current displacement reference (6081h) in reference unit.
- Set the profile acceleration (6083h) and profile deceleration (6084h) of each displacement reference in reference unit as needed.
- Write the control word 6040h from 0xnF to 0x(n+1)F and enable the servo drive.

Position Reference Type (bit6 of 6040h)	Reference Update Mode (bit5 of 6040h)	6040h	Description
0	0	0x0F → 0x1F	Absolute position, not updated immediately
0	1	0x2F → 0x3F	Absolute position, updated immediately
1	0	0x4F → 0x5F	Relative position, not updated immediately
1	1	0x6F → 0x7F	Relative position, updated immediately

Monitoring parameters:

- Position demand value 6062h (in reference unit), position demand value* 60FCh (in encoder unit)
- Position actual value* 6063h (in encoder unit), position actual value 6062h (in reference unit)
- Following error actual value 60F4h (in reference unit)
- Status word 6041h

For specific operations on different reference types and update types, see *"4.6.3 Control Commands in the Profile Position Mode"*.

4.7 Homing Mode

This mode is used to find the mechanical home and determine the position relation between the mechanical home and mechanical zero.

- Mechanical home: a fixed location on the machine, which may correspond to a specific home switch or the motor Z signal.
- Mechanical zero: absolute zero point on the machine

After homing is done, the motor stops at the mechanical home. The relation between the mechanical home and mechanical zero can be set in 607Ch.

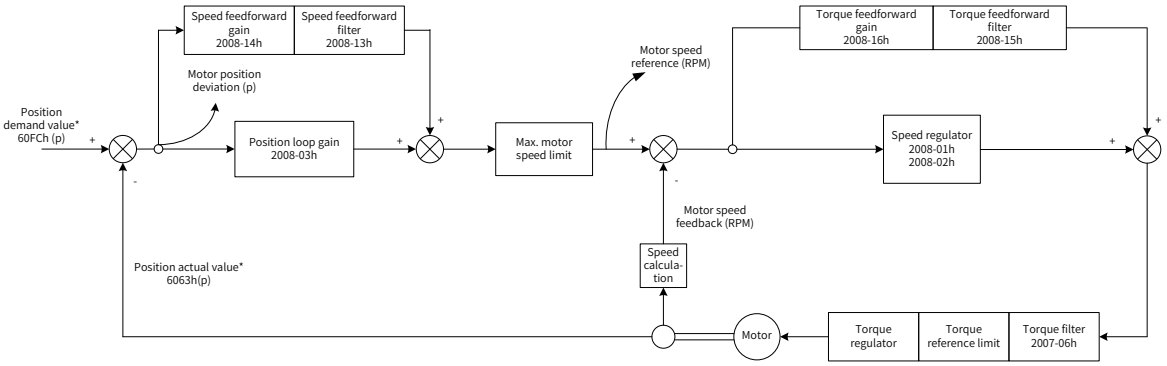
Mechanical home = Mechanical zero + 607C (Home offset)

When 607C is 0, the mechanical home overlaps with the mechanical zero.

In the homing mode, the host controller selects the homing method (6098h) first, set the homing speed (6099-1h and 6099-2h) and homing acceleration (609Ah), and then send the homing trigger signal. Upon receiving the homing trigger signal, the servo drive automatically searches for the mechanical home according to the settings and sets the relative position relation between the mechanical home and mechanical zero. The position control, speed control, and torque control are executed by the servo drive.

Note: The active travel of the home switch signal must be greater than or equal to the travel of acceleration/deceleration and the travel per motor revolution. Otherwise, the home switch may be exceeded, leading to a homing error.

4.7.1 Control Block Diagram



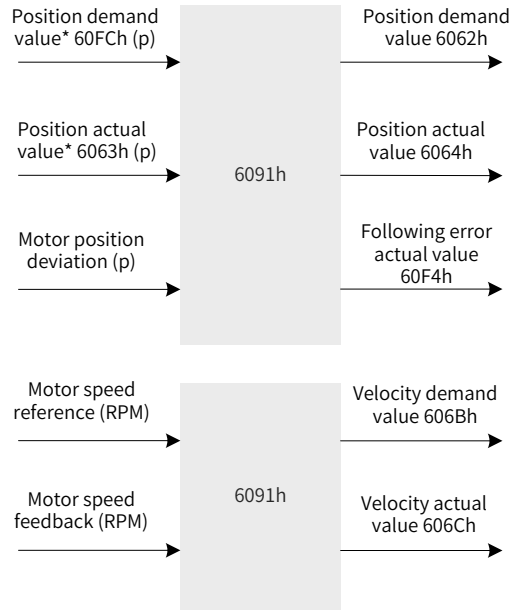


Figure 4-13 Block diagram for the homing mode

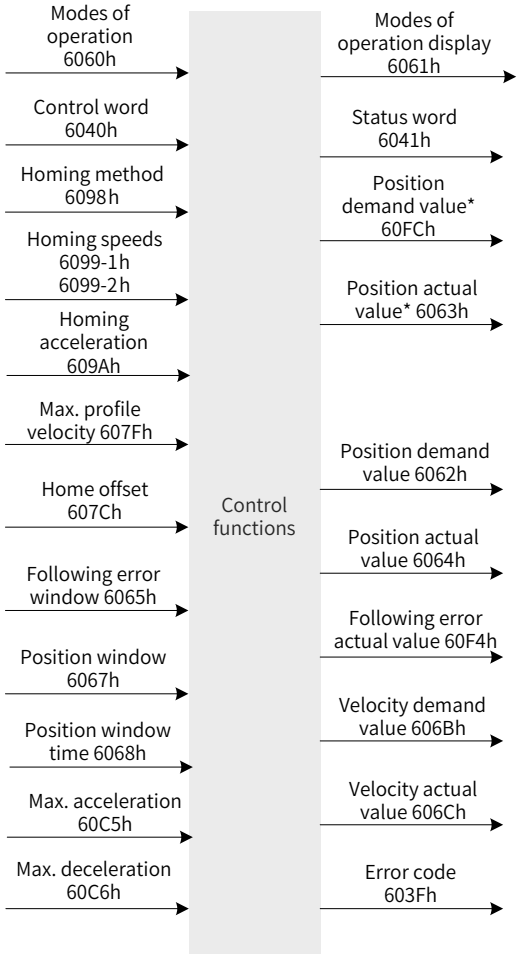


Figure 4-14 Input/output block diagram in the homing mode

Processing of the homing speed and homing acceleration/deceleration are shown in Figure 4-15 and Figure 4-16.

Two speeds are involved during homing. One is the speed during search for switch (6099-1h in reference unit) and the other is the speed during search for zero (6099-2h in reference unit). 6099-1h can be set to a large value to prevent homing timeout due to long homing duration. 6099-2h can be set to a small value to prevent overshoot upon stop at a high speed and large deviation between the stop position and the preset mechanical home.

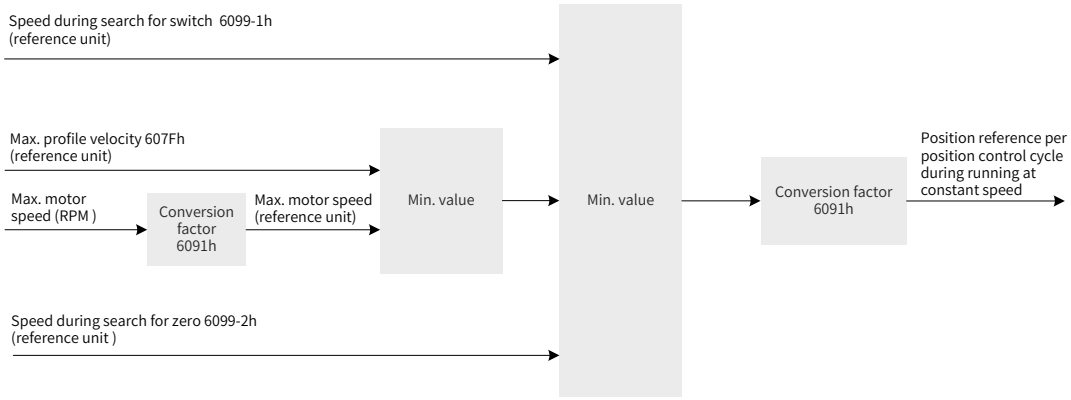


Figure 4-15 Homing speeds 6099h: Speed limit

Homing acceleration (609Ah) is used in the acceleration/deceleration phase.

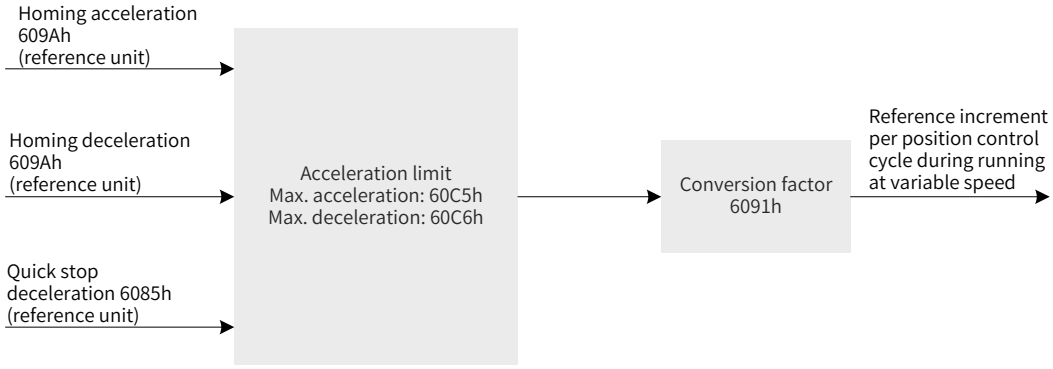


Figure 4-16 Homing acceleration (609Ah) limit

4.7.2 Related Object Settings

1) Homing timeout

Index 2005h	Name	Time of home searching					Data Structure	VAR	Data Type	Uint16
	Access	RW	Mapping	YES	Related Mode	hm	Data Range	0 to 65535	Default	50000
Subindex: 24h Unit: 10 ms If homing is not done within the time defined by 2005h, Er.601 will be reported. The homing timeout fault can be reset.										

2) Positioning completed

Index 6067h	Name	Position window					Data Structure	VAR	Data Type	Uint32
	Access	RW	Mapping	YES	Related Mode	pp/hm/ip	Data Range	0 to 4294967295	Default	734p
Subindex: 00 When the position deviation (60F4h) of the reference unit is smaller than 6067h and the time reaches 6068h, bit10 of 6041h is set to 1. When either condition is not met, the position reached is invalid.										

Index 6068h	Name	Position window time					Data Structure	VAR	Data Type	Uint16
	Access	RW	Mapping	YES	Related Mode	pp/hm/ip	Data Range	0 to 65535	Default	0 ms
Subindex: 00 When the position deviation (60F4h) of the reference unit is smaller than 6067h and the time reaches 6068h, bit10 of 6041h is set to 1. When either condition is not met, the position reached is invalid.										

3) Following error window detection

Index 6065h	Name	Following error window					Data Structure	VAR	Data Type	Uint32
	Access	RW	Mapping	YES	Related Mode	pp/hm/ip	Data Range	0 to 4294967295	Default	3435868
Subindex: 00 When the position deviation is larger than this value, Er.B00 occurs.										

4) Homing speed

Index 6099h	Name	Homing speeds					Data Structure	ARR	Data Type	Uint32
	Access	RW	Mapping	YES	Related Mode	All	Data Range	OD data range	Default	OD default value
Defines the homing speed.										

Sub-index 00h	Name	Number of entries					Data Structure	-	Data Type	Uint8
	Access	RO	Mapping	NO	Related Mode	-	Data Range	2	Default	2

Sub-index 01h	Name	Speed during search for switch					Data Structure	-	Data Type	Uint32
	Access	RW	Mapping	YES	Related Mode	-	Data Range	0 to 4294967295	Default	1747627
Defines the speed (in reference unit) during searching for the deceleration point signal . This speed can be set to a large value to prevent homing timeout due to long homing duration. Note: After finding the deceleration point, the slave decelerates and blocks the change of the home signal during deceleration. To prevent the slave from encountering the home signal during deceleration, set the switch position of the deceleration point signal properly to leave sufficient deceleration distance or increase the homing acceleration rate to shorten the deceleration time.										

Sub-index 02h	Name	Speed during search for zero					Data Structure	-	Data Type	Int32
	Access	RW	Mapping	YES	Related Mode	-	Data Range	0 to 4294967295	Default	174763
Defines the speed (in reference unit) during search for the home signal. Set this speed to a small value to prevent overshoot upon stop at a high speed and large deviation between the stop position and the preset mechanical home.										

5) Homing acceleration

Index 609Ah	Name	Homing acceleration					Data Structure	VAR	Data Type	Uint32
	Access	RW	Mapping	YES	Related Mode	hm	Data Range	0 to 4294967295	Default	174762666
Defines the acceleration during speed change in the homing mode. This parameter is used during acceleration and deceleration. The setpoint is activated after homing is enabled.										

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Value Range	Default
603Fh	00h	Error code	RO	TPDO	Uint16	-	0 to 65535	-
6040h	00h	Control word	RW	YES	Uint16	-	0 to 65535	0
6041h	00h	Status word	RO	TPDO	Uint16	-	0 to 65535	-
6060h	00h	Modes of operation	RW	YES	Int8	-	0 to 7	0
6061h	00h	Modes of operation display	RO	TPDO	Int8	-	0 to 7	-
6062h	00h	Position demand value	RO	TPDO	Int32	Reference unit	-2^{31} to $+(2^{31}-1)$	-
6063h	00h	Position actual value*	RO	TPDO	Int32	Encoder unit	-2^{31} to $+(2^{31}-1)$	-
6064h	00h	Position actual value	RO	TPDO	Int32	Reference unit	-2^{31} to $+(2^{31}-1)$	-
6065h	00h	Following error window	RW	YES	Uint32	Reference unit	0 to $(2^{32}-1)$	3435868
6067h	00h	Position window	RW	YES	Uint32	Reference unit	0 to $(2^{32}-1)$	734
6068h	00h	Position window time	RW	YES	Uint16	ms	0 to 65535	0
606Bh	00h	Velocity demand value	RO	TPDO	Int32	Reference unit	-2^{31} to $+(2^{31}-1)$	-
606Ch	00h	Velocity actual value	RO	TPDO	Int32	RPM	-2^{31} to $+(2^{31}-1)$	-
607Dh	01h	Min. position limit	RW	YES	Int32	Reference unit	-2^{31} to $+(2^{31}-1)$	-2^{31}
	02h	Max. position limit	RW	YES	Int32	Reference unit	-2^{31} to $+(2^{31}-1)$	$2^{31}-1$
607Ch	00h	Home offset	RW	YES	Int32	Reference unit	-2^{31} to $+(2^{31}-1)$	0

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Value Range	Default
6098h	00h	Homing method	RW	YES	Int8	-	0 to 35	1
6099h	01h	Speed during search for switch	RW	YES	Uint32	Reference unit	0 to (2^{32}-1)	1747627
	02h	Speed during search for zero	RW	YES	Int32	Reference unit	0 to (2^{32}-1)	174763
609Ah	00h	Homing acceleration	RW	YES	Uint32	RPM/ms	0 to (2^{32}-1)	174762666
2005h	24h	Homing duration limit	RW	YES	Uint16	10 ms	0 to 65535	50000
60F4h	00h	Following error actual value	RO	TPDO	Int32	Reference unit	-2^{31} to +(2^{31}-1)	-
60FCh	00h	Position demand value*	RO	TPDO	Int32	Reference unit	-2^{31} to +(2^{31}-1)	-
2007h	06h	Torque reference filter time constant	RW	YES	Uint16	ms	0 to 30.00	0.79
2008h	01h	Speed loop gain	RW	YES	Uint16	Hz	0.1 to 2000.0	25.0
	02h	Speed loop integral time constant	RW	YES	Uint16	ms	0.15 to 512.00	31.83
	03h	Position loop gain	RW	YES	Uint16	Hz	0.0 to 2000.0	40.0
	13h	Speed feedback filter time constant	RW	YES	Uint16	ms	0.00 to 64.00	0.50
	14h	Speed feedforward gain	RW	YES	Uint16	%	0.0 to 100.0	0.0
	15h	Torque feedforward filter time constant	RW	YES	Uint16	ms	0.00 to 64.00	0.50
	16h	Torque feedforward gain	RW	YES	Uint16	%	0.0 to 200.0	0.0

4.7.3 Control Commands in the Homing Mode

Table 4-4 Relation between state switchover and control commands

CiA402 Status Switchover		Control Word 6040h	bit0 to bit9 of Status Word 6041h
0	Power-on → Initialization	Natural transition, no control command required	0x0000h
1	Initialization → No fault	Natural transition, no control command required If an error occurs during initialization, the servo drive directly enters status 13.	0x0250h
2	No fault → Servo ready	0x06h	0x0231h
3	Servo ready → Wait for the S-ON signal	0x07h	0x0233h

CiA402 Status Switchover		Control Word 6040h	bit0 to bit9 of Status Word 6041h
4	Wait for the S-ON signal → Servo running	0x0Fh	0x0237h
5	Servo running → Wait for the S-ON signal	0x07h	0x0233h
6	Wait for the S-ON signal → Servo ready	0x06h	0x0231h
7	Servo ready → No fault	0x00h	0x0250h
8	Servo running → Servo ready	0x06h	0x0231h
9	Servo running → No fault	0x00h	0x0250h
10	Wait for the S-ON signal → No fault	0x00h	0x0250h
11	Servo running → Quick stop	0x02h	0x0217h
12	Quick stop → No fault	Set 605A to a value between 0 to 3. Natural transition applies after stop, requiring no control command.	0x0250h
13	→ Stop at fault	Once a fault occurs in any state other than "fault", the servo drive automatically switches to the stop-at-fault state, requiring no control command.	0x021Fh
14	Stop at fault → Fault	Natural transition applies after stop at fault, requiring no control command.	0x0218h
15	Fault → No fault	0x80h The bit7 is rising edge-triggered. The bit7 is kept to 1 and other control commands are invalid.	0x0250h
16	Quick stop → Servo running	Set 605A to a value between 5 to 7. 0x0F will be sent after stop.	0x0237h

The control word 6040h in the homing mode is described as follows.

Index 6040h	Name	Control word					Data Structure	VAR	Data Type	Uint16
	Access	RW	Mapping	YES	Related Mode	All	Data Range	0 to 65535	Default	-

Defines the control commands in the homing mode.

Control Word 6040h				
Bit	bit7 to bit15	bit5 to bit6	bit4	bit0 to bit3
Name	-	N/A	Homing enable	-
Value	See Table 4-2 for details.	-	-	See Table 4-2 for details.
Description	For details, see "6.5.3 Profile-Defined Parameters".	-	0: Homing not activated 0 → 1: Homing enabled 1: Homing in execution 1 → 0: Homing interrupted The bit4 must always be 1 during homing.	For details, see "6.5.3 Profile-Defined Parameters".

The status word 6041h in the homing mode is described as follows.

Index 6041h	Name	Status word					Data Structure	VAR	Data Type	Uint16
	Access	RO	Mapping	TPDO	Related Mode	All	Data Range	0 to 65535	Default	-

Defines the servo drive state in the homing mode.

Status word 6041h							
Bit	bit15	bit14	bit13	bit12	bit11	bit10	bit0 to bit9
Name	Homing completed	N/A	Homing error	Homing completed	Software internal setting beyond the limit	Target reached	-
Setting value	-	-	See Table 4-2 for details.	-	-	-	See Table 4-2 for details.
Description	0: Homing not performed or done 1: Homing done and reference point found This bit is unrelated to the operation mode or the servo state.	Reserved	0: No error 1: Homing error occurred [1]	0: Homing not done 1: Homing done	0: Actual position value not reaching the software position limit 1: Actual position value reaching the software position limit [2]	0: Target position not reached 1: Target position reached [3]	For details, see *"6.5.3 Profile-Defined Parameters"*.

[1] When a homing error occurs, Er.601 (Homing timeout) will be reported. If any error or warning occurs during homing, bit13 of 6041 will be set to 1.

[2] The software internal position limit can be enabled according to the setting of 0x200A-02h. For details, see description of 607Dh in *"6.5.3 Profile-Defined Parameters"*.

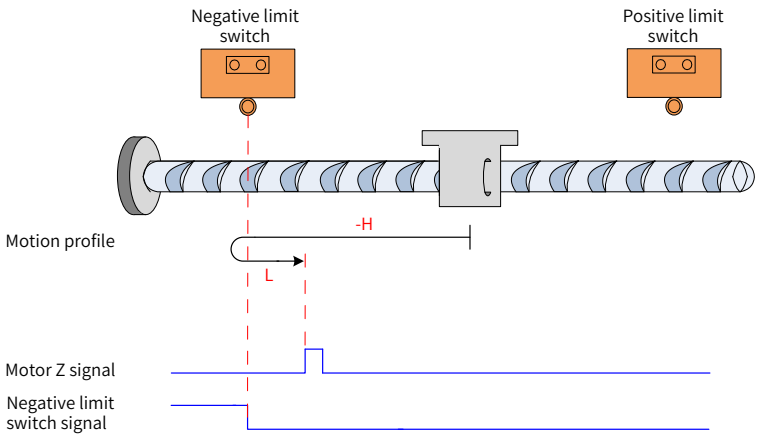
[3] When the position deviation is within the position window (6067h) and the time reaches 6068h, the target position is reached. If either condition is not met, the target position is not reached.

4.7.4 Introduction to the Homing Mode

1 6098h = 1

- Mechanical home: motor Z signal
- Deceleration point: negative limit switch (N-OT)

1) N-OT signal inactive at the start of homing



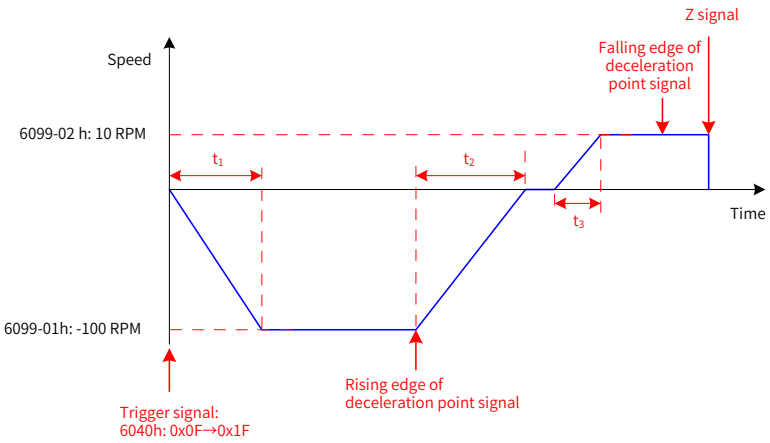


Figure 4-17 Mode ① in which 6098h = 1 and N-OT signal inactive

NOTE

In Figure 4-17, "H" indicates the speed during search for switch (6099-1h) and "L" indicates the speed during search for zero (6099-2h).

6099-1h = 100 x 1048576/60 p/s (100 RPM), 6099-2h = 10 x 1048576/60 p/s (10 RPM), 609Ah = 100 x 1048576/60 p/s^2 (100 RPM/s):

$$t_1 = \frac{6099\text{-}01h}{609Ah}\ s \qquad t_2 = \frac{6099\text{-}01h}{609Ah}\ s \qquad t_3 = \frac{6099\text{-}02h}{609Ah}\ s$$

The N-OT signal is inactive initially and the motor starts homing in the reverse direction at a high speed. After reaching the rising edge of the N-OT signal, the motor decelerates and changes to run in the forward direction. After reaching the falling edge of the N-OT signal, the motor stops at the first motor Z signal.

2) N-OT signal active at the start of homing

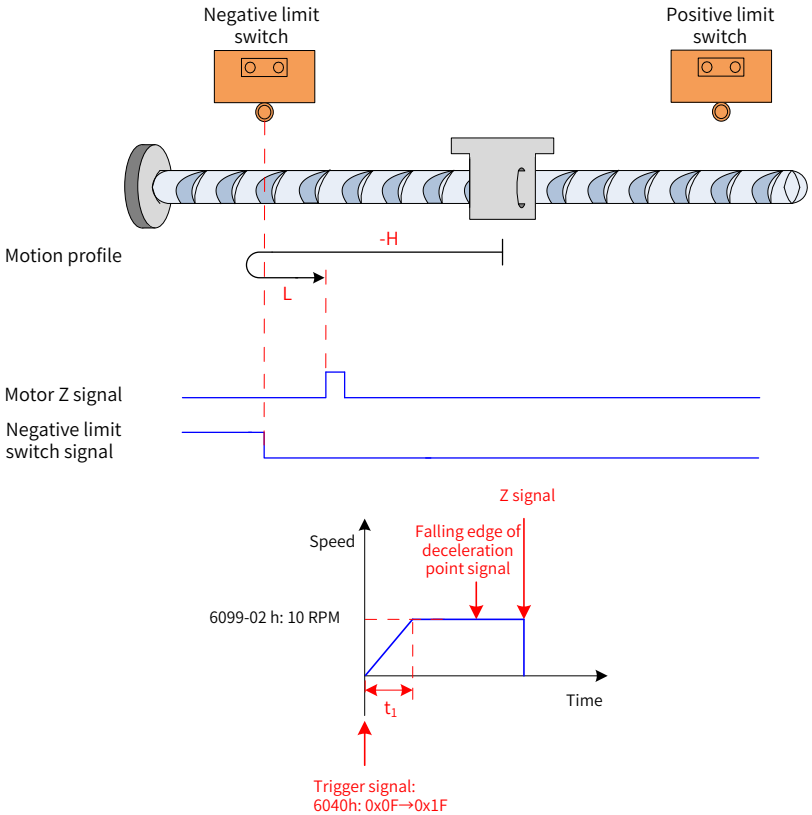


Figure 4-18 Mode ② in which 6098h = 1 and N-OT signal active

$$t_1 = \frac{6099\text{-}02h}{609Ah}\ s$$

The N-OT signal is active initially and the motor starts homing in the forward direction at a low speed. After reaching the falling edge of the N-OT signal, the motor stops at the first motor Z signal.

2 6098h = 2

- Home: Z signal
- Deceleration point: positive limit switch (P-OT)

1) P-OT signal inactive at the start of homing

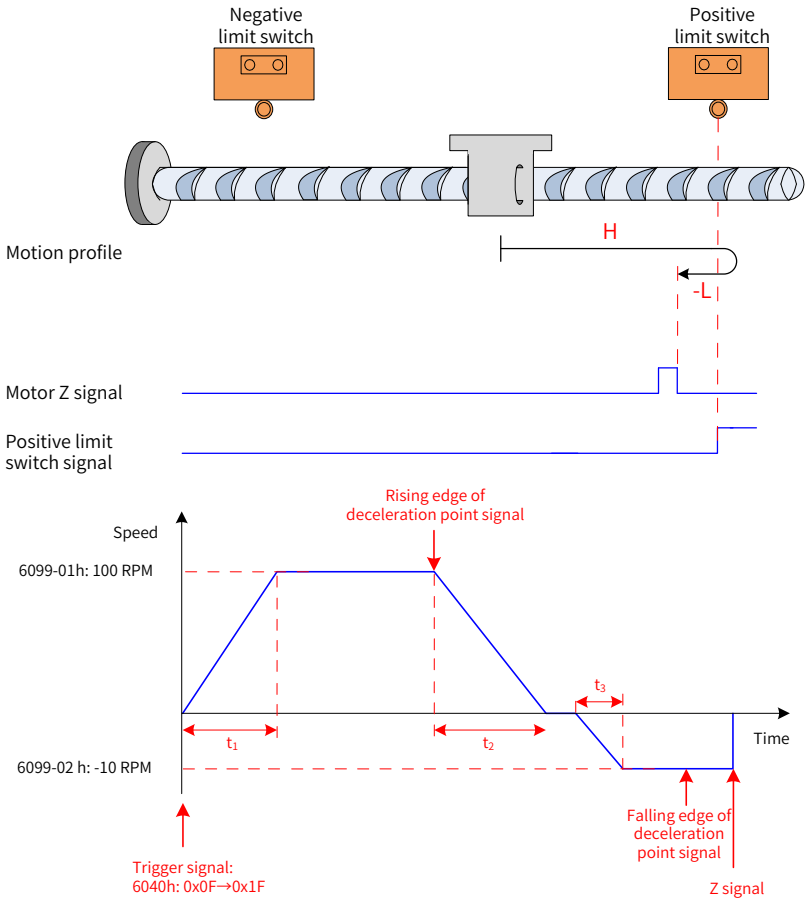


Figure 4-19 Mode ① in which 6098h = 2 and P-OT signal inactive

$$t_1 = \frac{6099\text{-}01h}{609Ah}\ s \qquad t_2 = \frac{6099\text{-}01h}{609Ah}\ s \qquad t_3 = \frac{6099\text{-}02h}{609Ah}\ s$$

The P-OT signal is inactive initially and the motor starts homing in the forward direction at a high speed. After reaching the rising edge of the P-OT signal, the motor decelerates and changes to run in the reverse direction. After reaching the falling edge of the P-OT signal, the motor stops at the first motor Z signal.

2) P-OT signal active at the start of homing

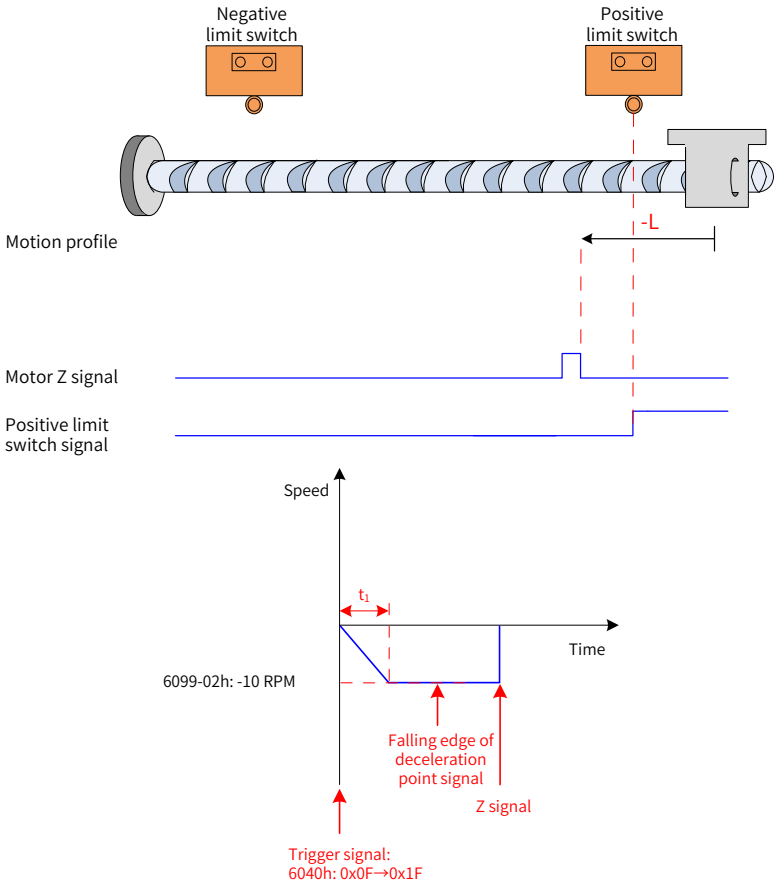


Figure 4-20 Mode ② in which 6098h = 2 and P-OT signal active

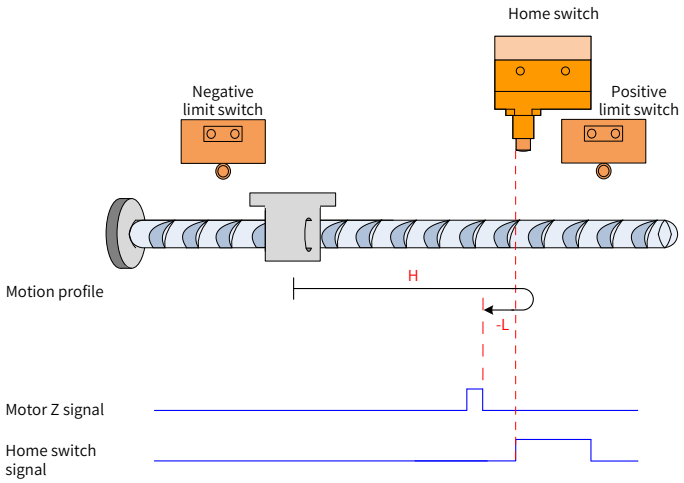
$$t_1 = \frac{6099\text{-}02h}{609Ah} \quad s$$

The P-OT signal is active initially and the motor starts homing in the reverse direction at a low speed. After reaching the falling edge of the P-OT signal, the motor stops at the first motor Z signal.

3 6098h = 3

- Home: Z signal
- Deceleration point: home switch (HW)

1) HW signal inactive at the start of homing



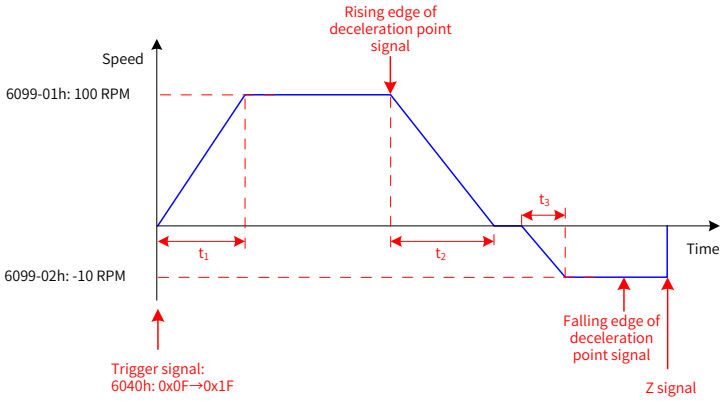


Figure 4-21 Mode ① in which 6098h = 3 and HW signal inactive

$$t_1 = \frac{6099\text{-}01h}{609Ah}\ s \qquad t_2 = \frac{6099\text{-}01h}{609Ah}\ s \qquad t_3 = \frac{6099\text{-}02h}{609Ah}\ s$$

The HW signal is inactive initially and the motor starts homing in the forward direction at a high speed. After reaching the rising edge of the HW signal, the motor decelerates and changes to run in the reverse direction. After reaching the falling edge of the HW signal, the motor stops at the first motor Z signal.

2) HW signal active at the start of homing

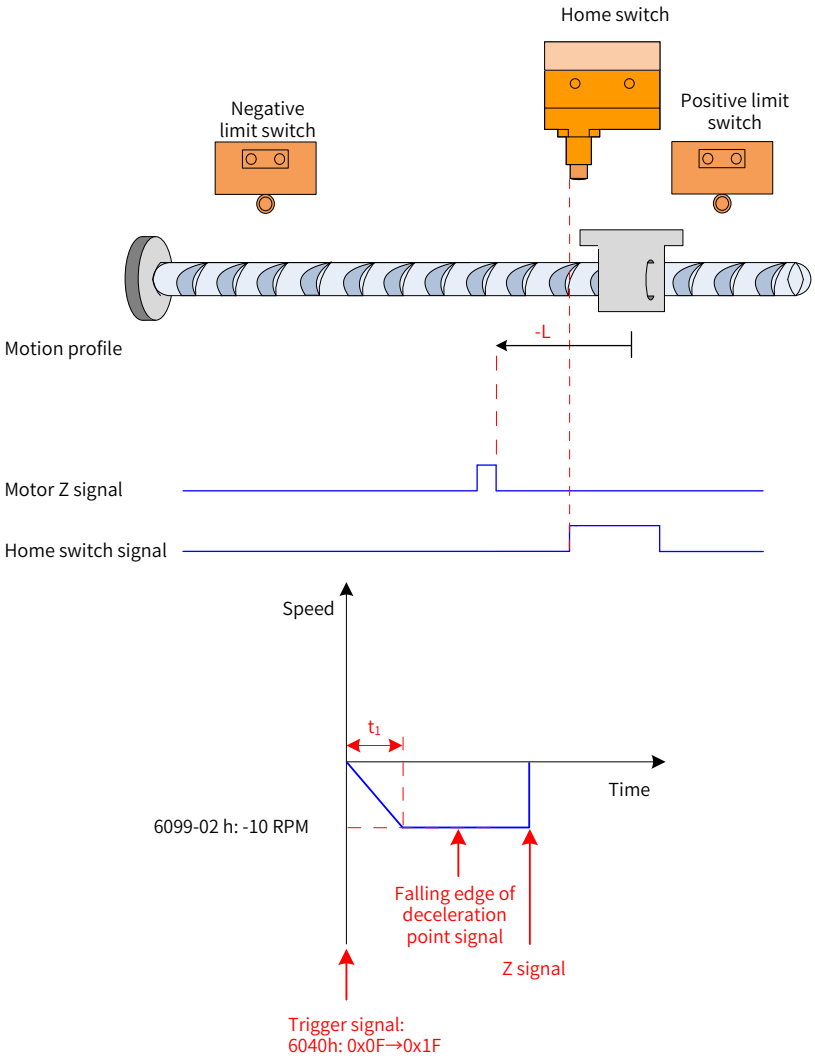


Figure 4-22 Mode ② in which 6098h = 3 and HW signal active

$$t_1 = \frac{6099\text{-}02h}{609Ah} \quad s$$

The HW signal is active initially and the motor starts homing in the reverse direction at a low speed. After reaching the falling edge of the HW signal, the motor stops at the first motor Z signal.

4 6098h = 4

- Home: Z signal
- Deceleration point: home switch (HW)

1) HW signal inactive at the start of homing

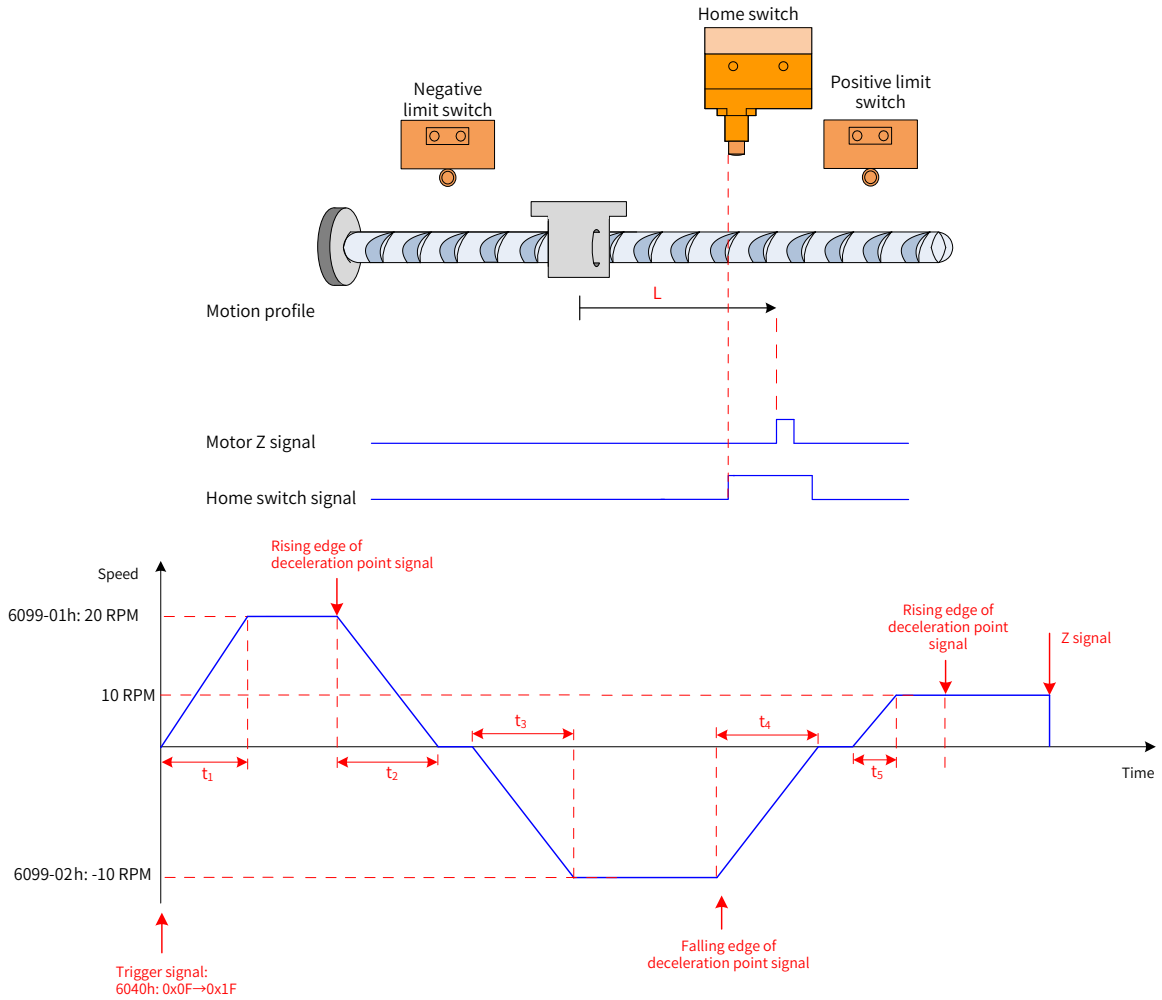


Figure 4-23 Mode ① in which 6098h = 4 and HW signal inactive

The HW signal is inactive at start, and the motor starts homing in the forward direction at a high speed. After reaching the rising edge of the HW signal, the motor decelerates and changes to run in the reverse direction. Then, after reaching the falling edge of the HW signal, the motor decelerates and changes to run in the forward direction until it stops at the first Z signal upon reaching the rising edge of the HW signal.

2) HW signal active at the start of homing

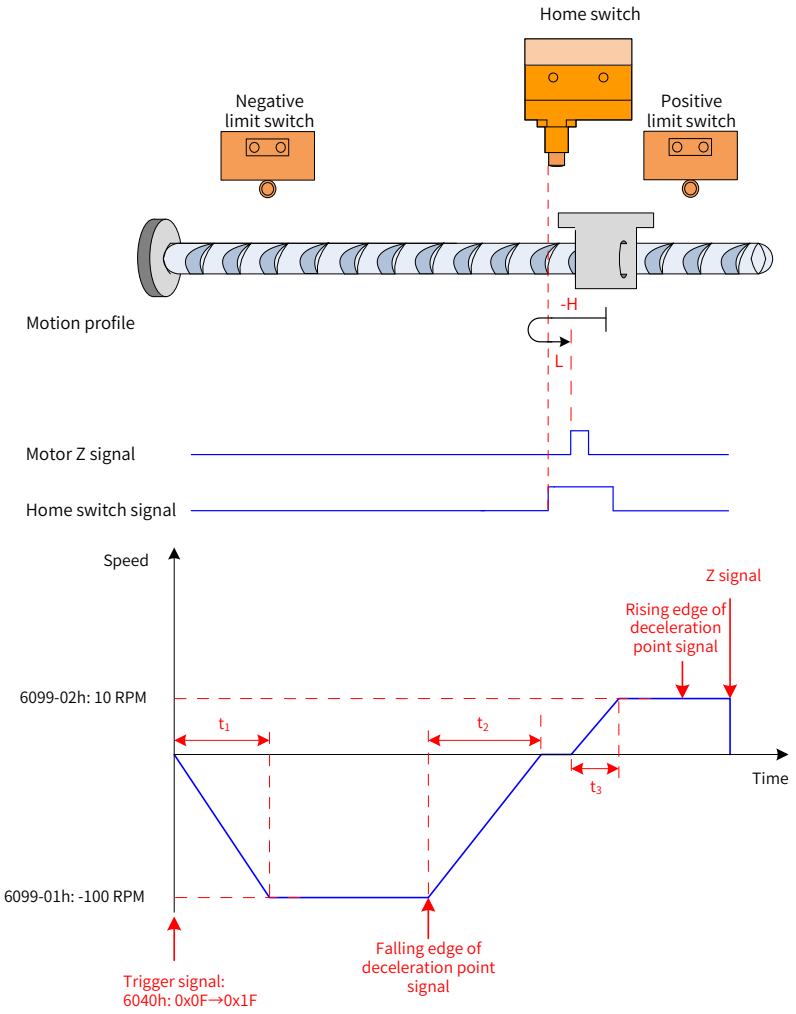


Figure 4-24 Mode ② in which 6098h = 4 and HW signal active

$$t_1 = \frac{\text{6099-01h}}{\text{609Ah}}\ \text{s} \qquad t_2 = \frac{\text{6099-01h}}{\text{609Ah}}\ \text{s} \qquad t_3 = \frac{\text{6099-02h}}{\text{609Ah}}\ \text{s}$$

The HW signal is active initially and the motor starts homing in the reverse direction at a high speed. After reaching the falling edge of the HW signal, the motor decelerates and changes to run in the forward direction at a low speed. After reaching the rising edge of the HW signal, the motor stops at the first motor Z signal.

5 6098h = 5

- Home: Z signal
- Deceleration point: home switch (HW)

1) HW signal inactive at the start of homing

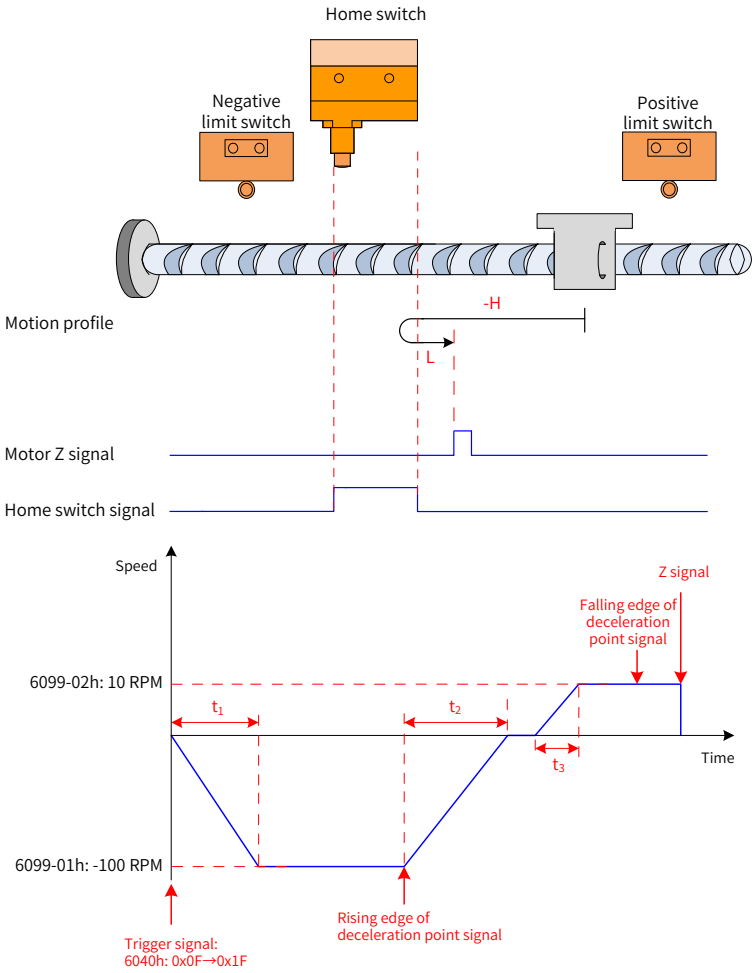


Figure 4-25 Mode ① in which 6098h = 5 and HW signal inactive

$$t_1 = \frac{6099\text{-}01h}{609Ah}\ s \quad t_2 = \frac{6099\text{-}01h}{609Ah}\ s \quad t_3 = \frac{6099\text{-}02h}{609Ah}\ s$$

The HW signal is inactive initially and the motor starts homing in the reverse direction at a high speed. After reaching the rising edge of the HW signal, the motor decelerates and changes to run in the forward direction. After reaching the falling edge of the HW signal, the motor stops at the first motor Z signal.

2) HW signal active at the start of homing

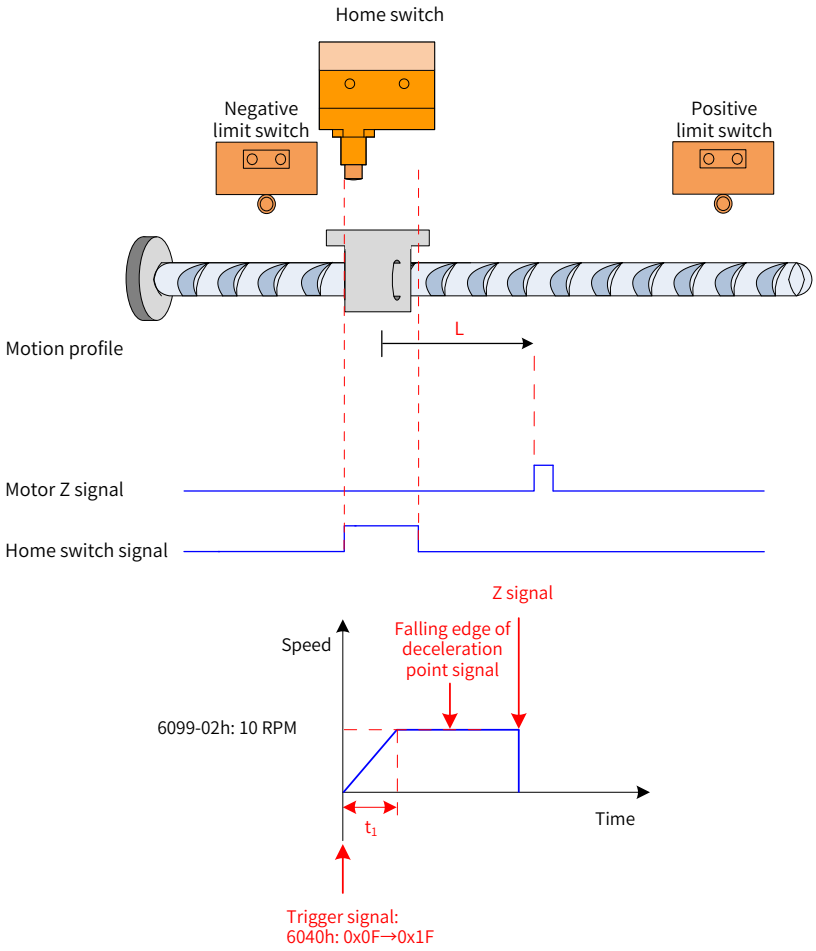


Figure 4-26 Mode ② in which 6098h = 5 and HW signal active

$$t_1 = \frac{6099\text{-}02h}{609Ah} \quad s$$

The HW signal is active initially and the motor starts homing in the forward direction at a low speed. After reaching the falling edge of the HW signal, the motor stops at the first motor Z signal.

6 6098h = 6

- Home: Z signal
- Deceleration point: home switch (HW)

1) HW signal inactive at the start of homing

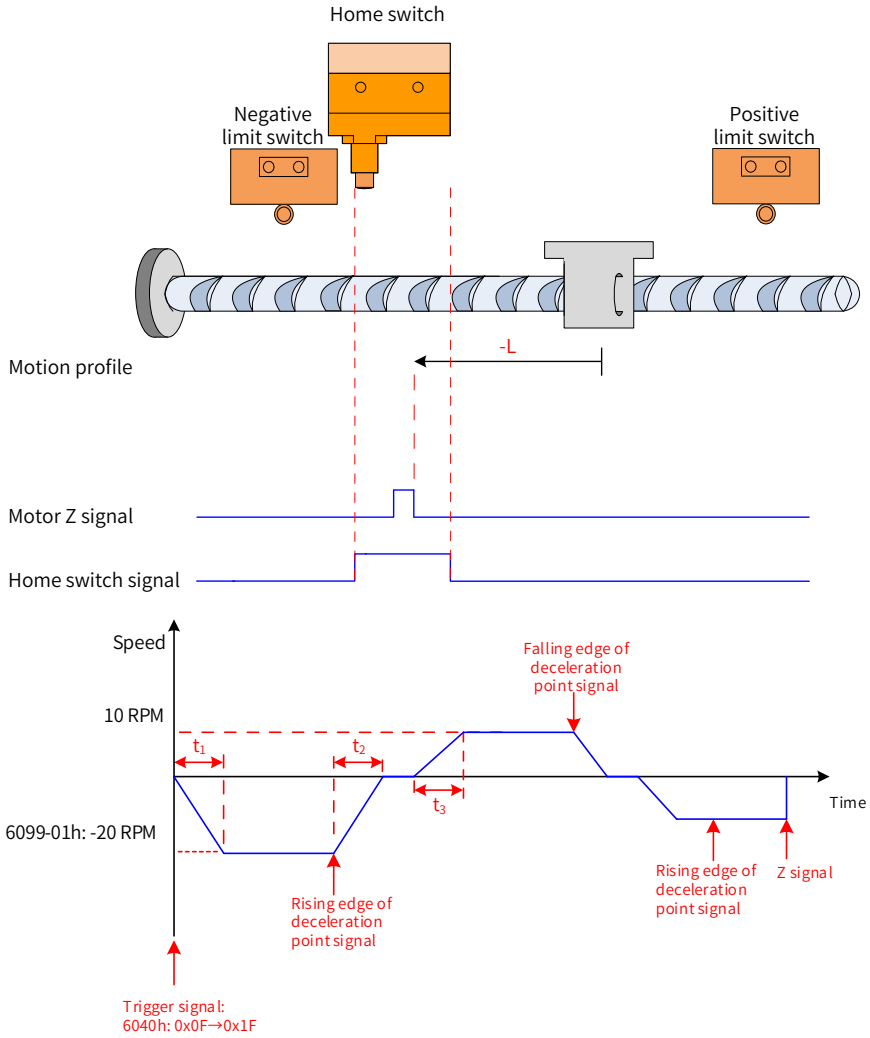


Figure 4-27 Mode ① in which 6098h = 6 and HW signal inactive

The HW signal is inactive initially, and the motor starts homing in reverse direction at a high speed. After reaching the rising edge of the HW signal, the motor decelerates and changes to run in forward direction. After reaching the falling edge of the HW signal, the motor changes to run in the reverse direction at a low speed and stops at the first motor Z signal after reaching the rising edge of the HW signal.

2) HW signal active at the start of homing

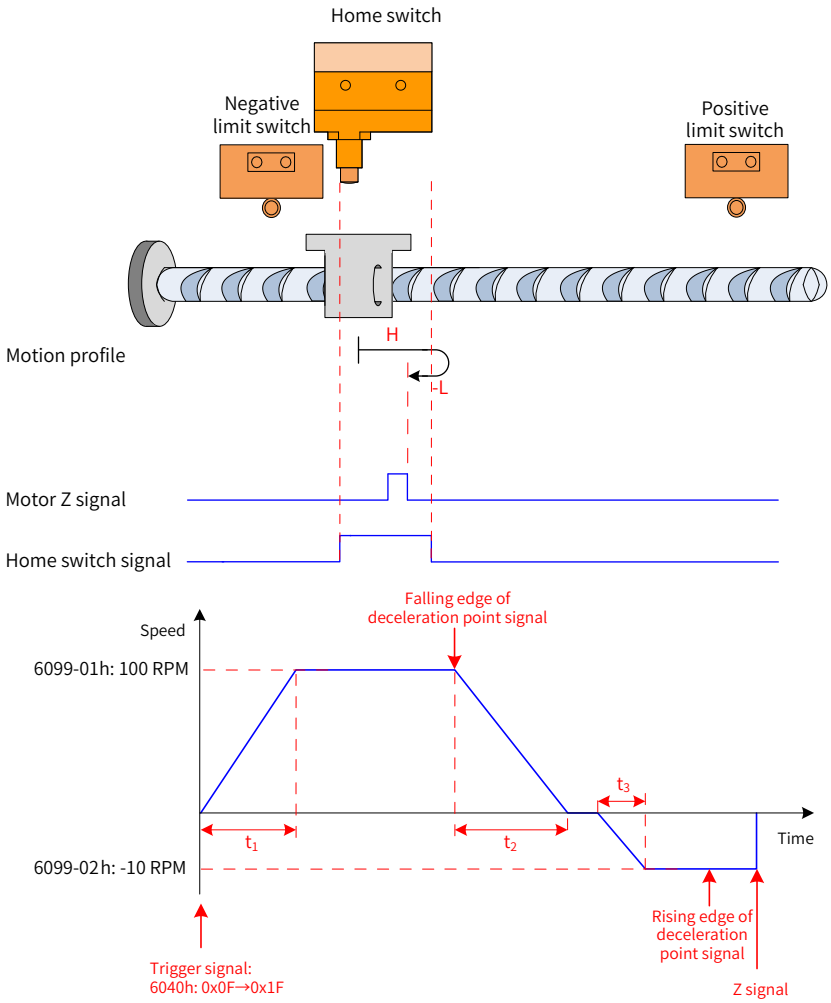


Figure 4-28 Mode ② in which 6098h = 6 and HW signal active

$$t_1 = \frac{\text{6099-01h}}{\text{609Ah}}\ \text{s} \quad t_2 = \frac{\text{6099-01h}}{\text{609Ah}}\ \text{s} \quad t_3 = \frac{\text{6099-02h}}{\text{609Ah}}\ \text{s}$$

The HW signal is active initially and the motor starts homing in the forward direction at a high speed. After reaching the falling edge of the HW signal, the motor decelerates and changes to run in the reverse direction. After reaching the rising edge of the HW signal, the motor stops at the first motor Z signal.

7 6098h = 7

- Home: Z signal
- Deceleration point: home switch (HW)

1) HW signal inactive at the start of homing, not hitting the positive limit switch

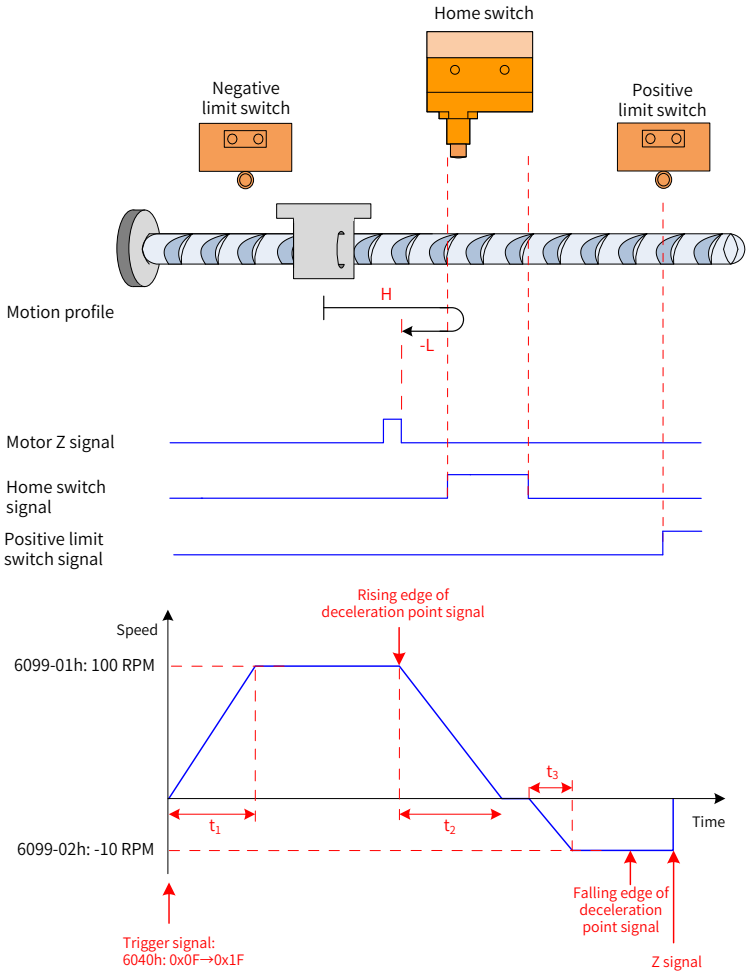


Figure 4-29 Mode ① in which 6098 = 7, HW signal inactive and limit switch not hit

$$t_1 = \frac{6099\text{-}01h}{609Ah}\ s \qquad t_2 = \frac{6099\text{-}01h}{609Ah}\ s \qquad t_3 = \frac{6099\text{-}02h}{609Ah}\ s$$

The HW signal is inactive initially and the motor starts homing in the forward direction at a high speed. If the motor does not hit the limit switch and reaches the rising edge of the HW signal, it decelerates and changes to run in the reverse direction. After reaching the falling edge of the HW signal, the motor stops at the first motor Z signal.

2) HW signal inactive at the start of homing, hitting the positive limit switch

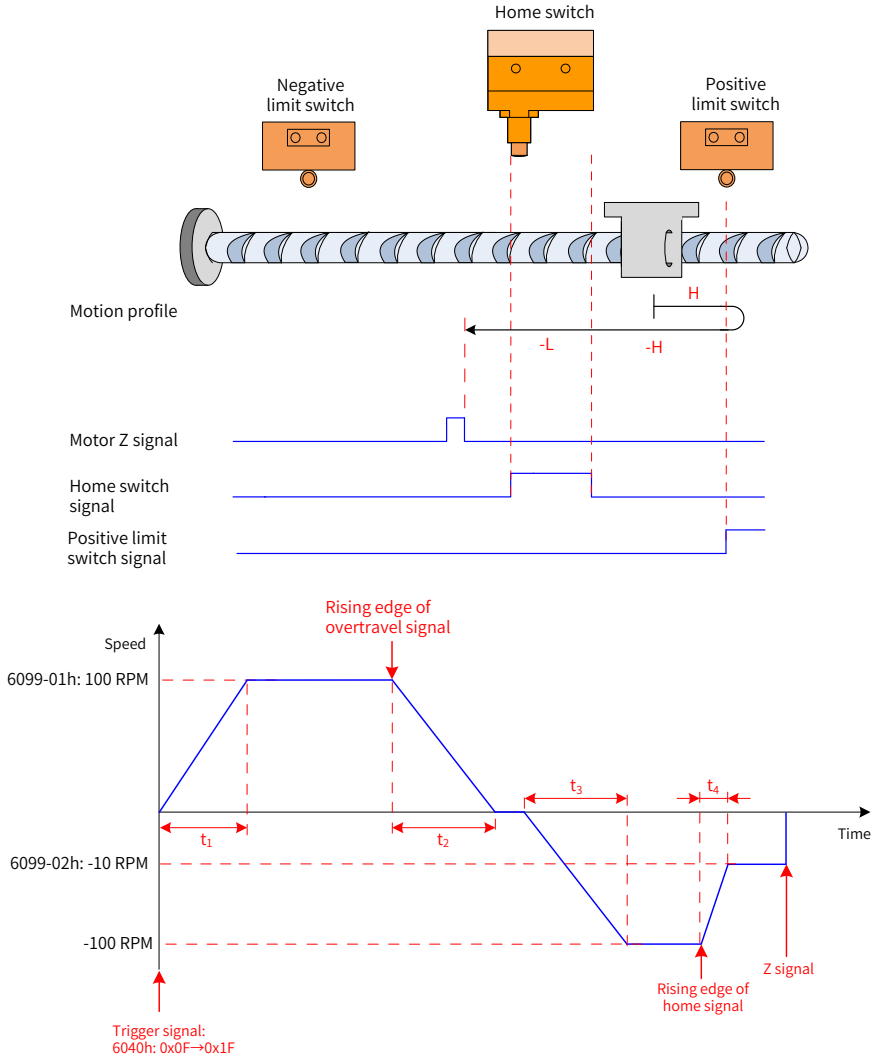


Figure 4-30 Mode ② in which 6098 = 7, HW signal inactive and positive limit switch being hit

$$t_1 = \frac{\text{6099-01h}}{\text{609Ah}}\ \text{s} \quad t_2 = \frac{\text{6099-01h}}{\text{609Ah}}\ \text{s} \quad t_3 = \frac{\text{6099-01h}}{\text{609Ah}}\ \text{s} \quad t_4 = \frac{\text{[6099-01h] - [6099-02h]}}{\text{609Ah}}\ \text{s}$$

The HW signal is inactive at start, and the motor starts homing in the forward direction at a high speed. If the motor hits the limit switch, it changes to run in the reverse direction. After reaching the rising edge of the HW signal, the motor decelerates and continues running in the reverse direction. Then, after reaching the falling edge of the HW signal, the motor changes to run in the forward direction until it decelerates and changes to run in the reverse direction again upon reaching the rising edge of HW signal. Finally, the motor stops at the first Z signal after reaching the falling edge of the HW signal.

3) HW signal active at the start of homing

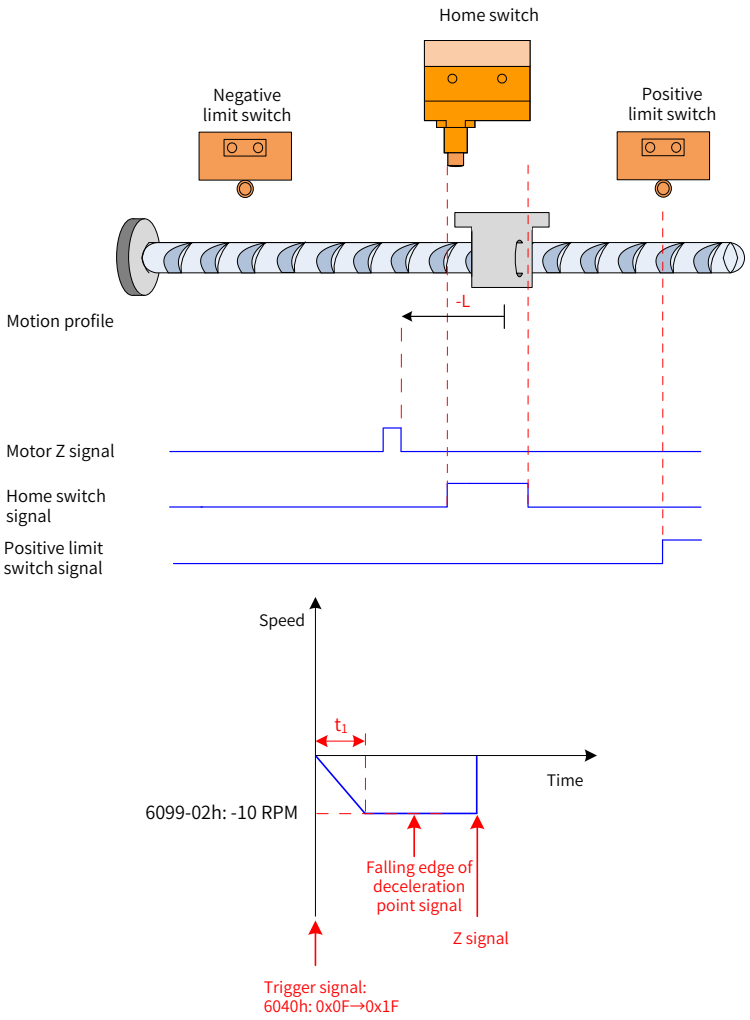


Figure 4-31 Mode ③ in which 6098 = 7 and HW signal active

$$t_1 = \frac{6099\text{-}02h}{609Ah} \quad s$$

The HW signal is active initially, and the motor starts homing in the reverse direction at a low speed. After reaching the falling edge of the HW signal, the motor stops at the first Z signal.

8 6098h = 8

- Home: Z signal
- Deceleration point: home switch (HW)

1) HW signal inactive at the start of homing, not hitting the positive limit switch

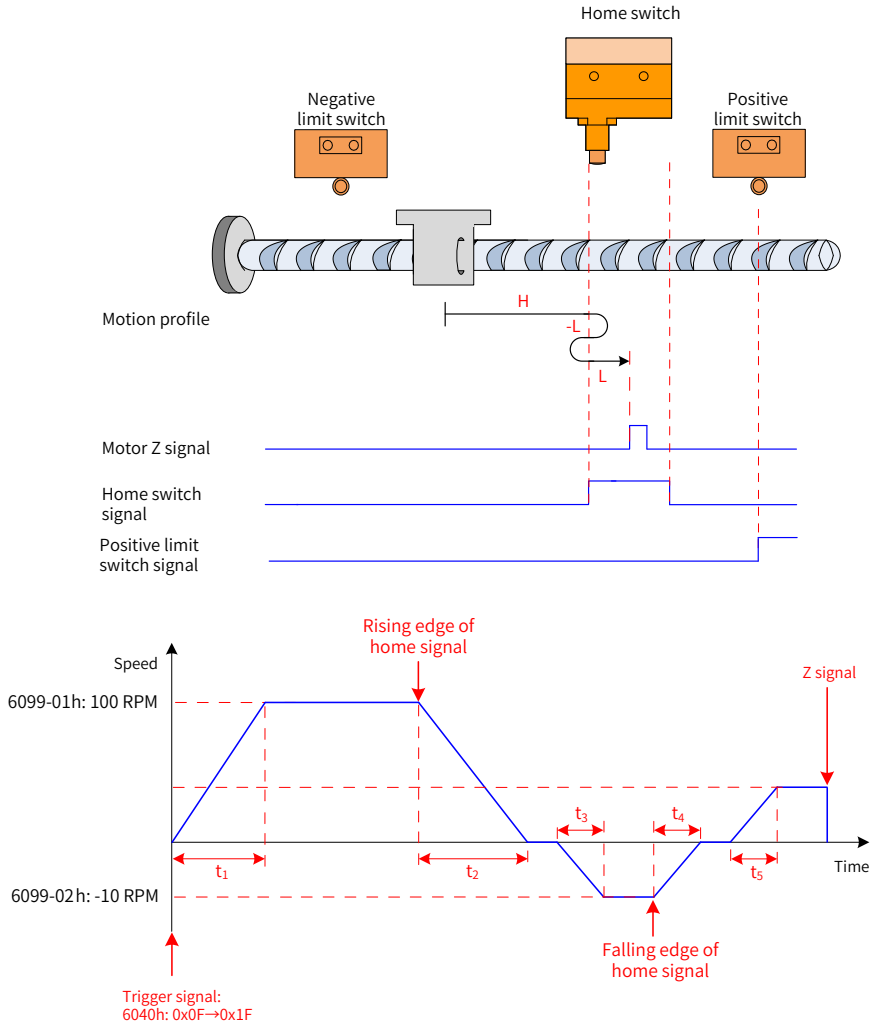


Figure 4-32 Mode ① in which 6098h = 8, HW signal inactive and positive limit switch not hit

$$t_1 = \frac{6099\text{-}01h}{609Ah}\ s \quad t_2 = \frac{6099\text{-}01h}{609Ah}\ s \quad t_3 = \frac{6099\text{-}02h}{609Ah}\ s \quad t_4 = \frac{6099\text{-}02h}{609Ah}\ s \quad t_5 = \frac{6099\text{-}02h}{609Ah}\ s$$

The HW signal is inactive initially and the motor starts homing in the forward direction at a high speed. If the motor does not hit the limit switch, it decelerates and changes to run in the reverse direction after reaching the rising edge of the HW signal. After reaching the falling edge of the HW signal, the motor changes to run in the forward direction at a low speed and stops at the first motor Z signal after reaching the rising edge of the HW signal.

2) HW signal inactive at the start of homing, hitting the positive limit switch

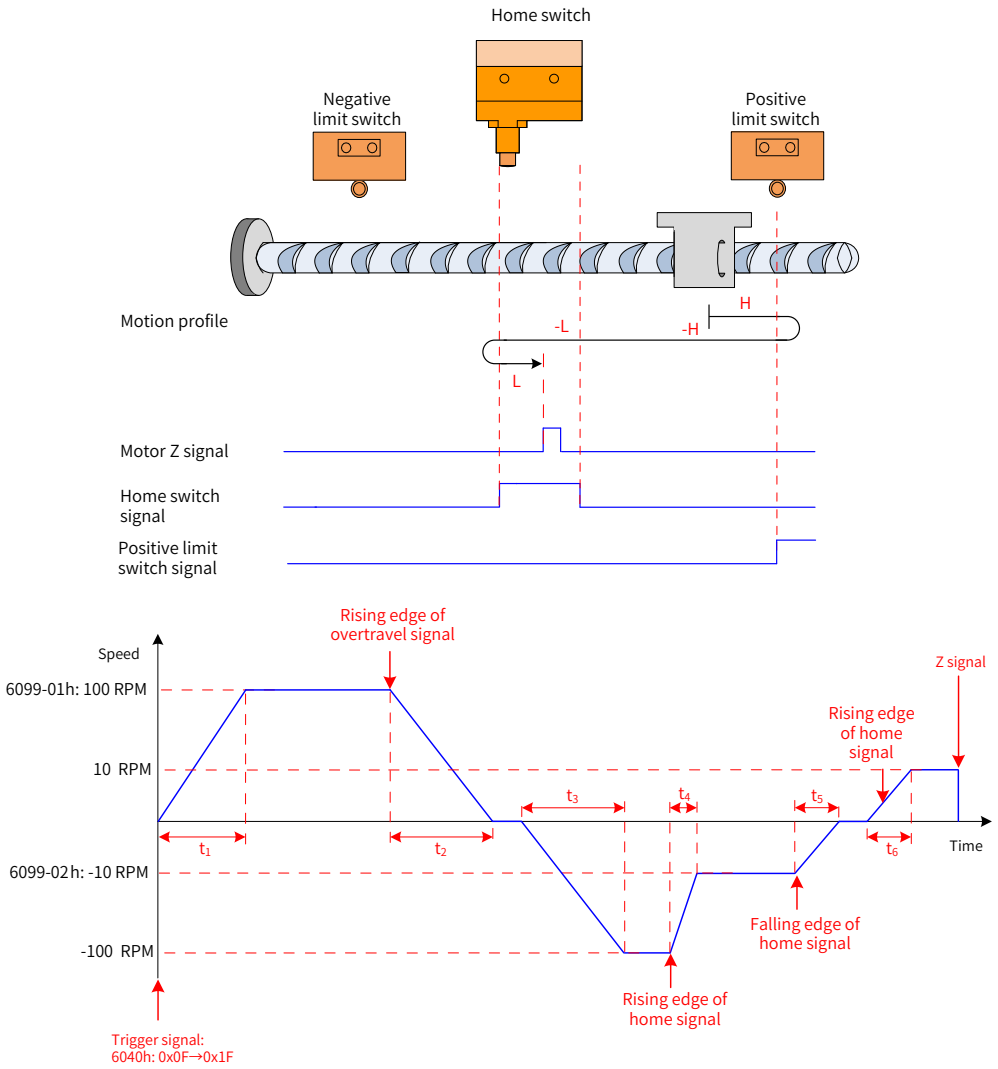


Figure 4-33 Mode ② in which 6098h = 8, HW signal inactive and positive limit switch being hit

$$t_1 = \frac{6099\text{-}01h}{609Ah}\ s \quad t_2 = \frac{6099\text{-}01h}{609Ah}\ s \quad t_3 = \frac{6099\text{-}01h}{609Ah}\ s \quad t_4 = \frac{[6099\text{-}01h] - [6099\text{-}02h]}{609Ah}\ s$$

$$t_5 = \frac{6099\text{-}02h}{609Ah}\ s \quad t_6 = \frac{6099\text{-}02h}{609Ah}\ s$$

The HW signal is inactive initially and the motor starts homing in the forward direction at a high speed. If the motor hits the limit switch, it changes to run in the reverse direction. After reaching the rising edge of the HW signal, the motor decelerates and runs in the reverse direction. After reaching the falling edge of the HW signal, the motor changes to run in the forward direction at a low speed and stops at the first motor Z signal after reaching the rising edge of the HW signal.

3) HW signal active at the start of homing

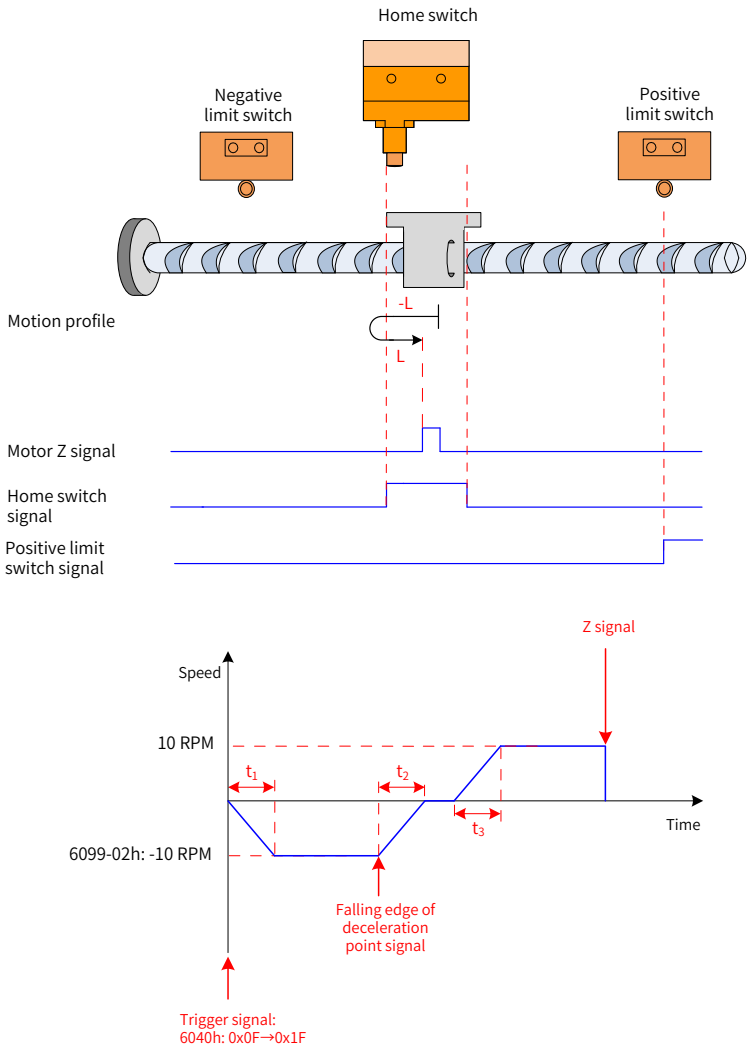


Figure 4-34 Mode ③ in which 6098h = 8 and HW signal active

$$t_1 = \frac{6099\text{-}02h}{609Ah}\ s \qquad t_2 = \frac{6099\text{-}02h}{609Ah}\ s \qquad t_3 = \frac{6099\text{-}02h}{609Ah}\ s$$

The HW signal is active initially and the motor starts homing in the reverse direction at a low speed. After reaching the falling edge of the HW signal, the motor changes to run in the forward direction. After reaching the rising edge of the HW signal, the motor stops at the first motor Z signal.

9 6098h = 9

- Home: Z signal
- Deceleration point: home switch (HW)

1) HW signal inactive at the start of homing, not hitting the positive limit switch

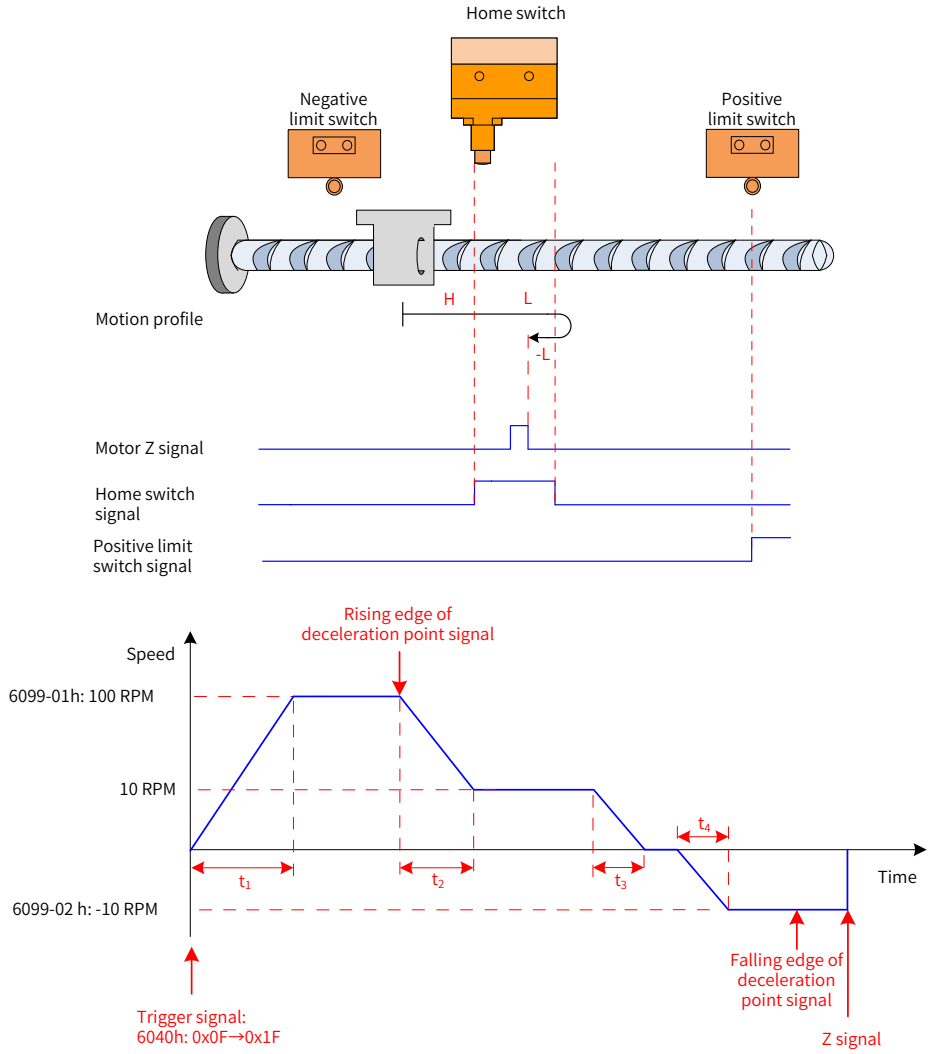


Figure 4-35 Mode ① in which 6098h = 9, HW signal inactive and positive limit switch not hit

$$t_1 = \frac{6099\text{-}01h}{609Ah}\ s \quad t_2 = \frac{[6099\text{-}01h] - [6099\text{-}02h]}{609Ah}\ s \quad t_3 = \frac{6099\text{-}02h}{609Ah}\ s \quad t_4 = \frac{6099\text{-}02h}{609Ah}\ s$$

The HW signal is inactive initially and the motor starts homing in the forward direction at a high speed. If the motor does not hit the limit switch, it decelerates after reaching the rising edge of the HW signal. Then, after reaching the falling edge of the HW signal, the motor changes to run in the reverse direction at a low speed and stops at the first motor Z signal after reaching the rising edge of the HW signal.

2) HW signal inactive at the start of homing, hitting the positive limit switch

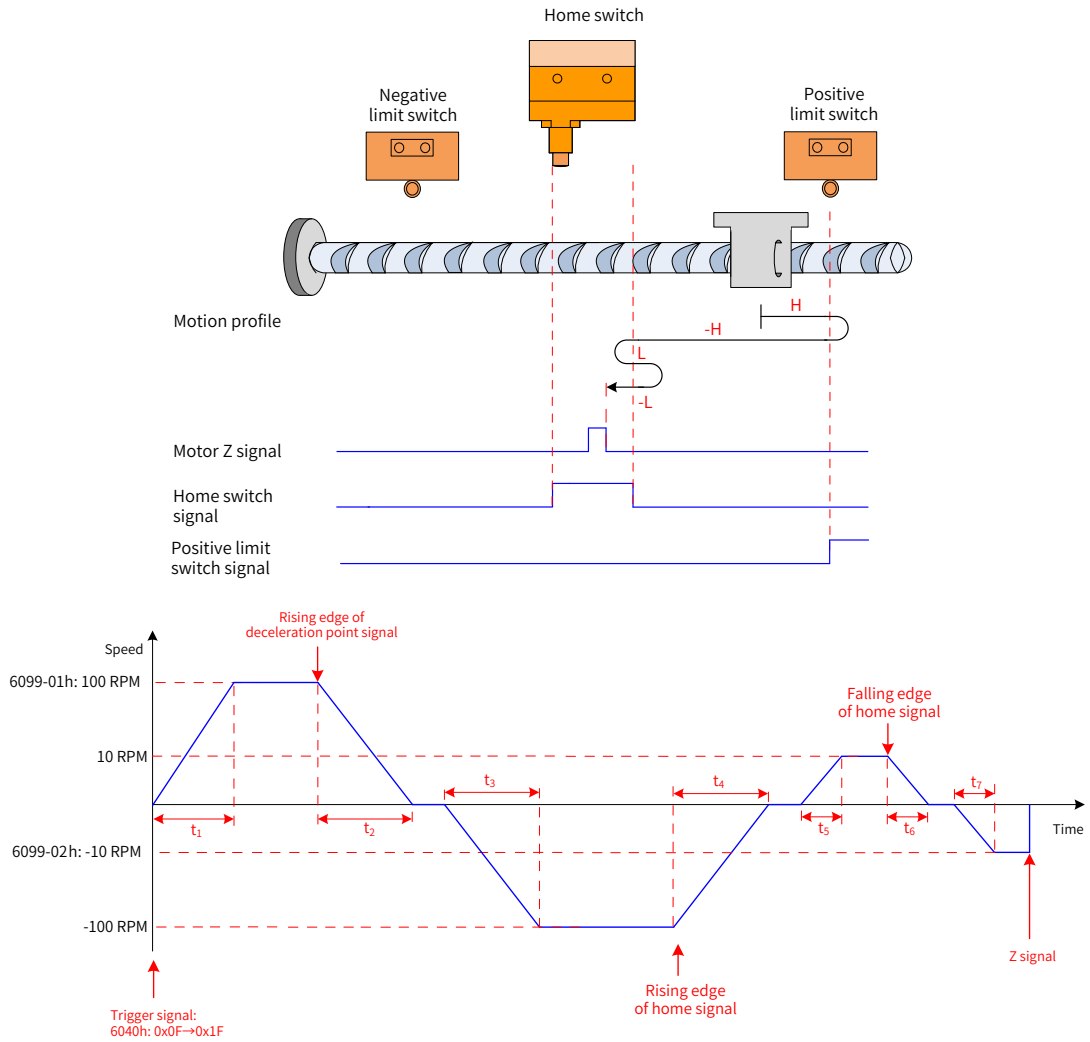


Figure 4-36 Mode ② in which 6098h = 9, HW signal inactive and positive limit switch being hit

$$t_1 = \frac{6099\text{-}01h}{609Ah}\ s \quad t_2 = \frac{6099\text{-}01h}{609Ah}\ s \quad t_3 = \frac{6099\text{-}01h}{609Ah}\ s \quad t_4 = \frac{6099\text{-}01h}{609Ah}\ s \quad t_5 = \frac{6099\text{-}02h}{609Ah}\ s$$

$$t_6 = \frac{6099\text{-}02h}{609Ah}\ s \quad t_7 = \frac{6099\text{-}02h}{609Ah}\ s$$

The HW signal is inactive initially and the motor starts homing in the forward direction at a high speed. If the motor hits the limit switch, it changes to run in the reverse direction. After reaching the rising edge of the HW signal, the motor decelerates and changes to run in the forward direction. After reaching the falling edge of the HW signal, the motor changes to run in the reverse direction at a low speed until it stops at the first motor Z signal upon reaching the rising edge of the HW signal.

3) HW signal active at the start of homing

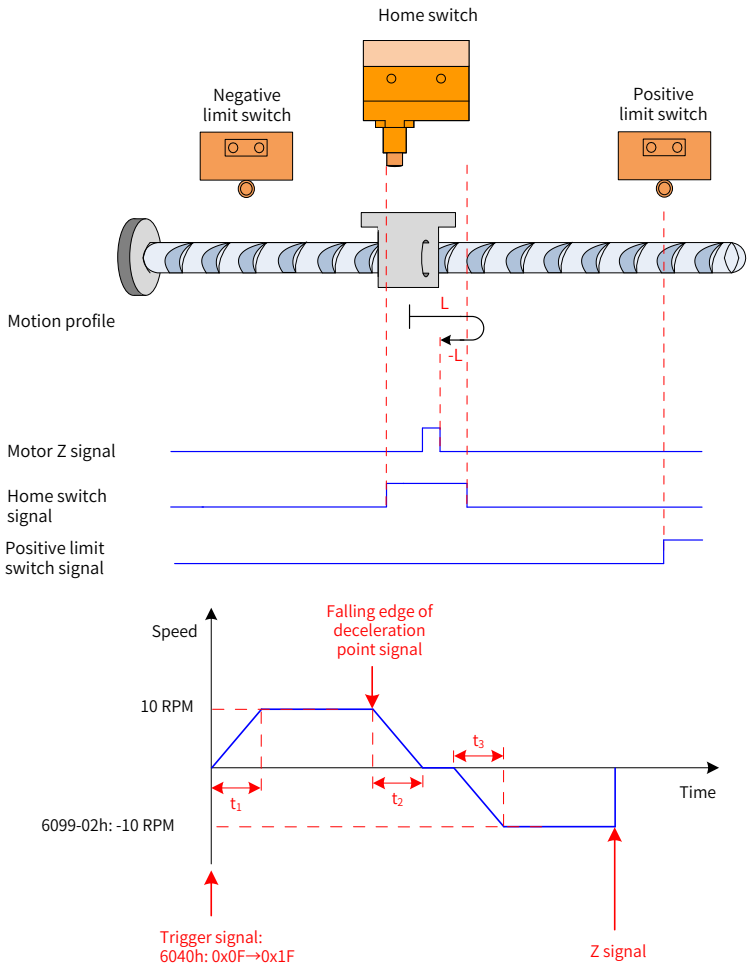


Figure 4-37 Mode ③ in which 6098h = 9 and HW signal active

$$t_1 = \frac{6099\text{-}02h}{609Ah}\ s \quad t_2 = \frac{6099\text{-}02h}{609Ah}\ s \quad t_3 = \frac{6099\text{-}02h}{609Ah}\ s$$

The HW signal is active initially and the motor starts homing in the forward direction at a low speed. After reaching the falling edge of the HW signal, the motor changes to run in the reverse direction and stops at the first motor Z signal upon reaching the rising edge of the HW signal.

10 6098h = 10

- Home: Z signal
- Deceleration point: home switch (HW)

1) HW signal inactive at the start of homing, not hitting the positive limit switch

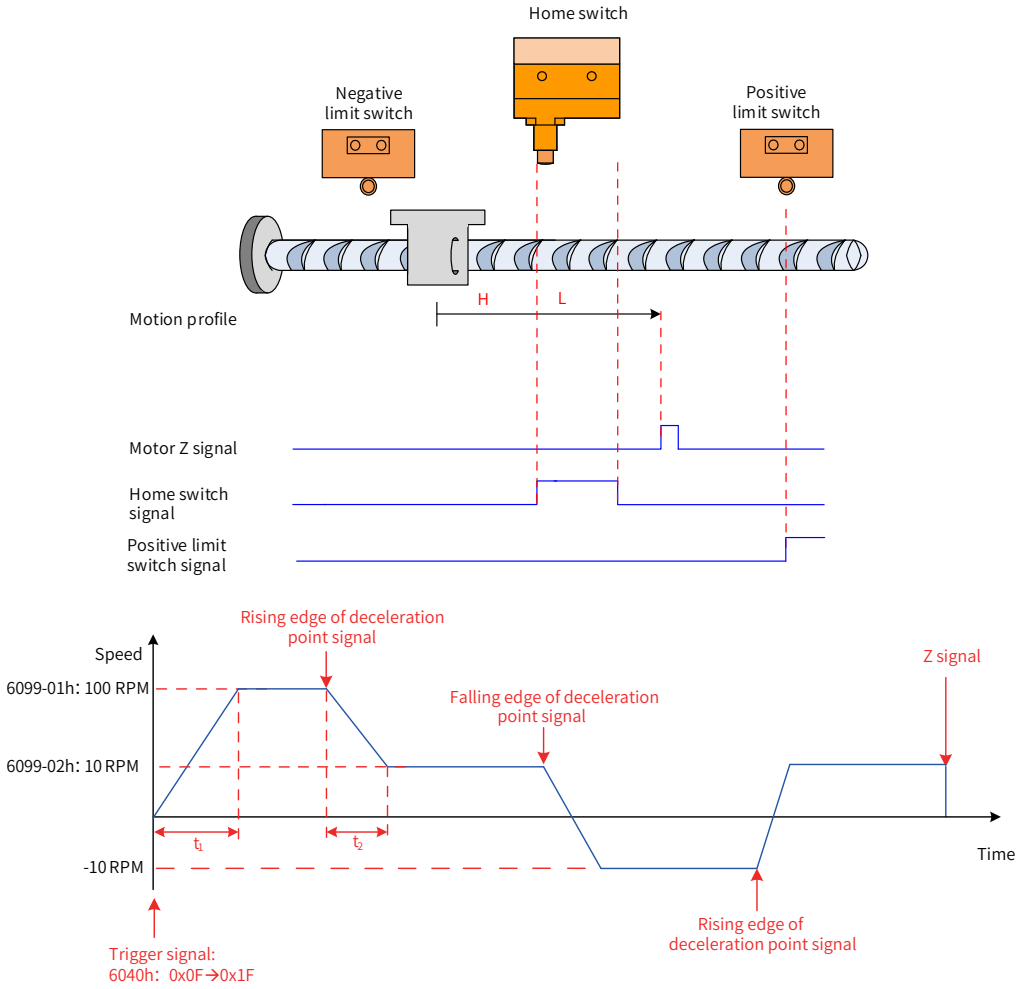


Figure 4-38 Mode ① in which 6098h = 10, HW signal inactive and positive limit switch not hit

$$t_1 = \frac{6099\text{-}01h}{609Ah}\ s \quad t_2 = \frac{[6099\text{-}01h] - [6099\text{-}02h]}{609Ah}\ s$$

The HW signal is inactive at the start of homing and the motor starts homing in the forward direction at a high speed until reaching the rising edge of the HW signal where it decelerates. The motor changes to run in the reverse direction at a low speed upon reaching the falling edge of the HW signal. After reaching the rising edge of the HW signal, the motor decelerates and changes to run in the forward direction until it stops at the first Z signal upon reaching the falling edge of the HW signal.

2) HW signal inactive at the start of homing, hitting the positive limit switch

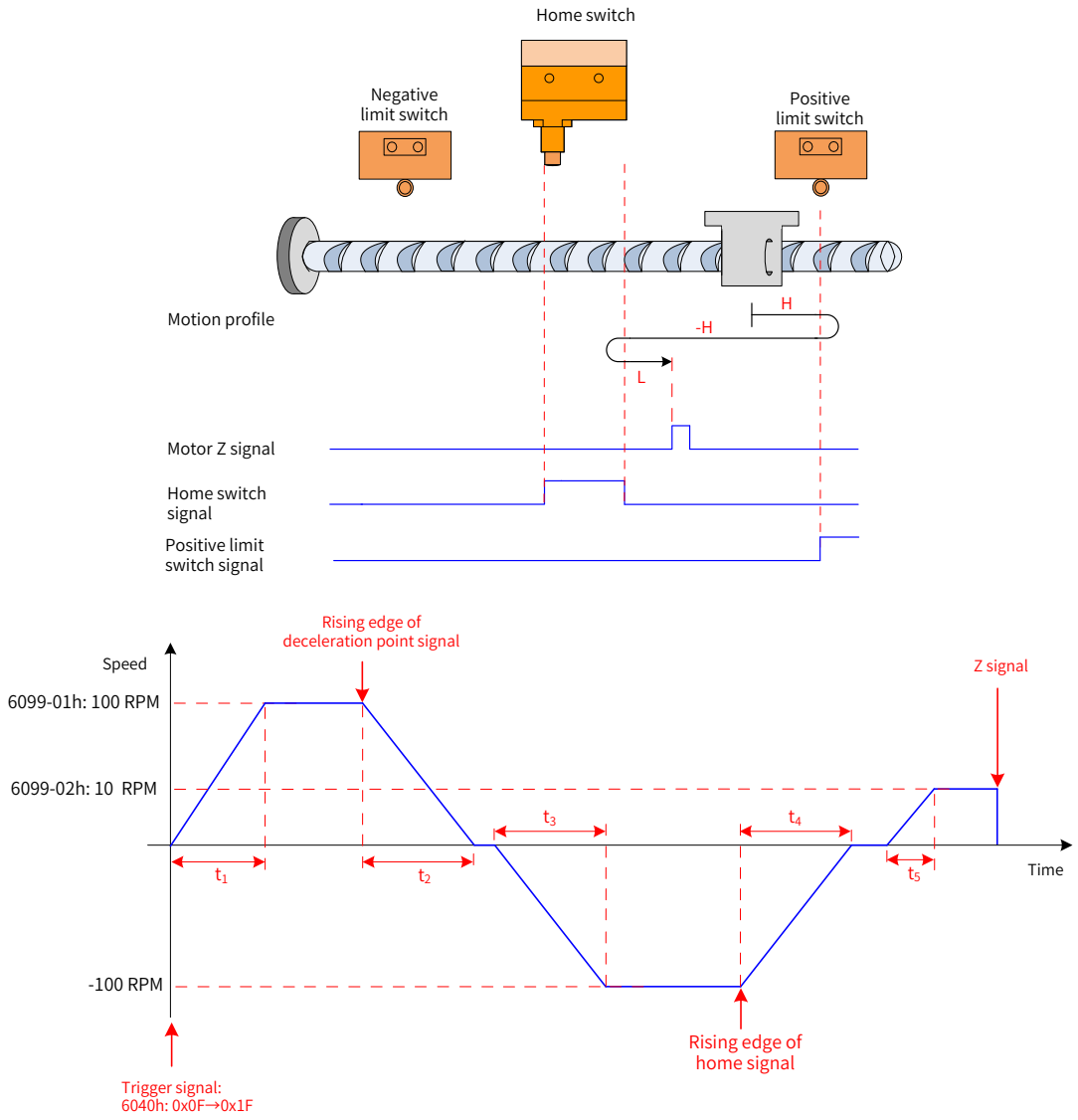


Figure 4-39 Mode ② in which 6098h = 10, HW signal inactive and positive limit switch being hit

$$t_1 = \frac{6099\text{-}01h}{609Ah}\ s \quad t_2 = \frac{6099\text{-}01h}{609Ah}\ s \quad t_3 = \frac{6099\text{-}01h}{609Ah}\ s \quad t_4 = \frac{6099\text{-}01h}{609Ah}\ s \quad t_5 = \frac{6099\text{-}02h}{609Ah}\ s$$

The HW signal is inactive initially and the motor starts homing in the forward direction at a high speed. If the motor hits the limit switch, it changes to run in the reverse direction at a high speed. After reaching the rising edge of the HW signal, the motor decelerates and changes to run in the forward direction again. After reaching the falling edge of the HW signal, the motor stops at the first motor Z signal.

3) HW signal active at the start of homing

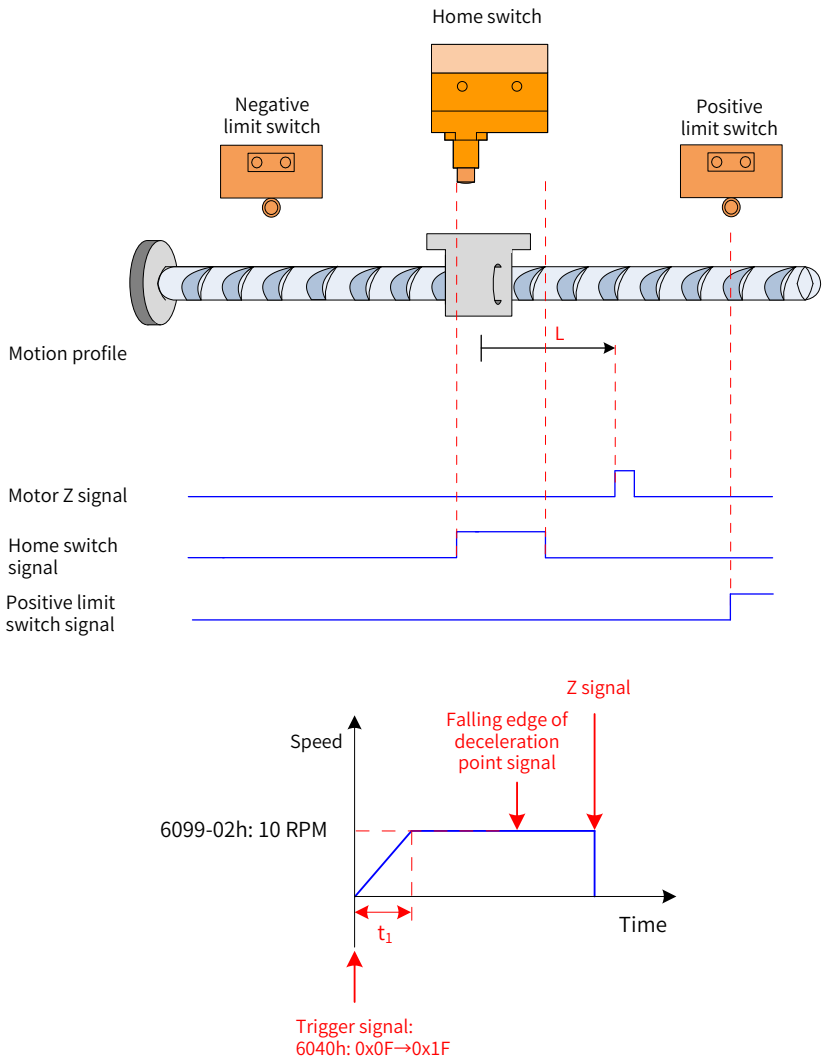


Figure 4-40 Mode ③ in which 6098h = 10 and HW signal active

$$t_1 = \frac{6099\text{-}02h}{609Ah} \quad s$$

The HW signal is active initially and the motor starts homing in the forward direction at a low speed. After reaching the falling edge of the HW signal, the motor stops at the first motor Z signal.

11 6098h = 11, 12, 13, or 14

Similar to the profile when 6098h = 7, 8, 9, or 10, opposite in the initial running direction only

12 6098h = 17 to 30

Same as the profile when 6098 = 1 to 14, without the final step of searching for the motor Z signal

The motor stops immediately upon reaching the following home signals.

Homing Mode 6098	Home Signal
17	N-OT falling edge
18	P-OT falling edge
19	HW falling edge

Homing Mode 6098	Home Signal
20	HW rising edge
21	HW falling edge
22	HW rising edge
23	HW falling edge
24	HW rising edge
25	HW rising edge
26	HW falling edge
27	HW falling edge
28	HW rising edge
29	HW rising edge
30	HW falling edge

13 6098h = 31/32

This mode is not defined in the standard 402 protocol. It can be used for extension purpose.

14 6098h = 33/34

- Home: Z signal
- Deceleration point: None

1) Homing mode 33: The motor runs in the reverse direction at a low speed and stops at the first motor Z signal.
2) Homing mode 34: The motor runs in the forward direction at a low speed and stops at the first motor Z signal.

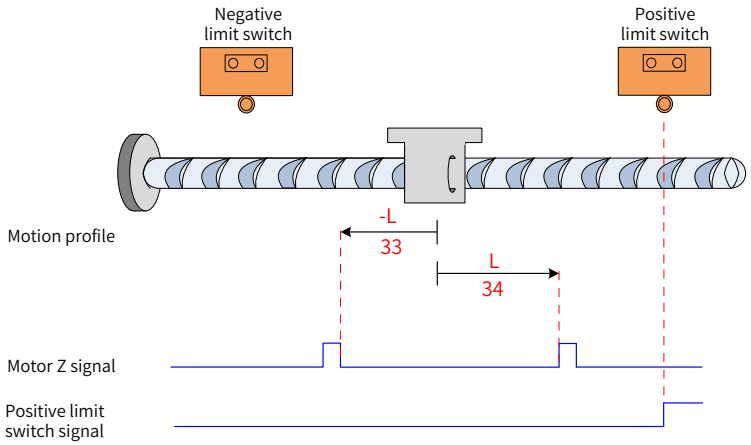


Figure 4-41 6098h = 33 or 34

15 6098h = 35

Homing mode 35: The present position is taken as the mechanical home. The position actual value (6064h) equals to the home offset (60Ch) after homing is triggered (control word 6040h: 0x0F → 0x1F).

Position actual value 6064h = 607C

4.7.5 Configuration Example

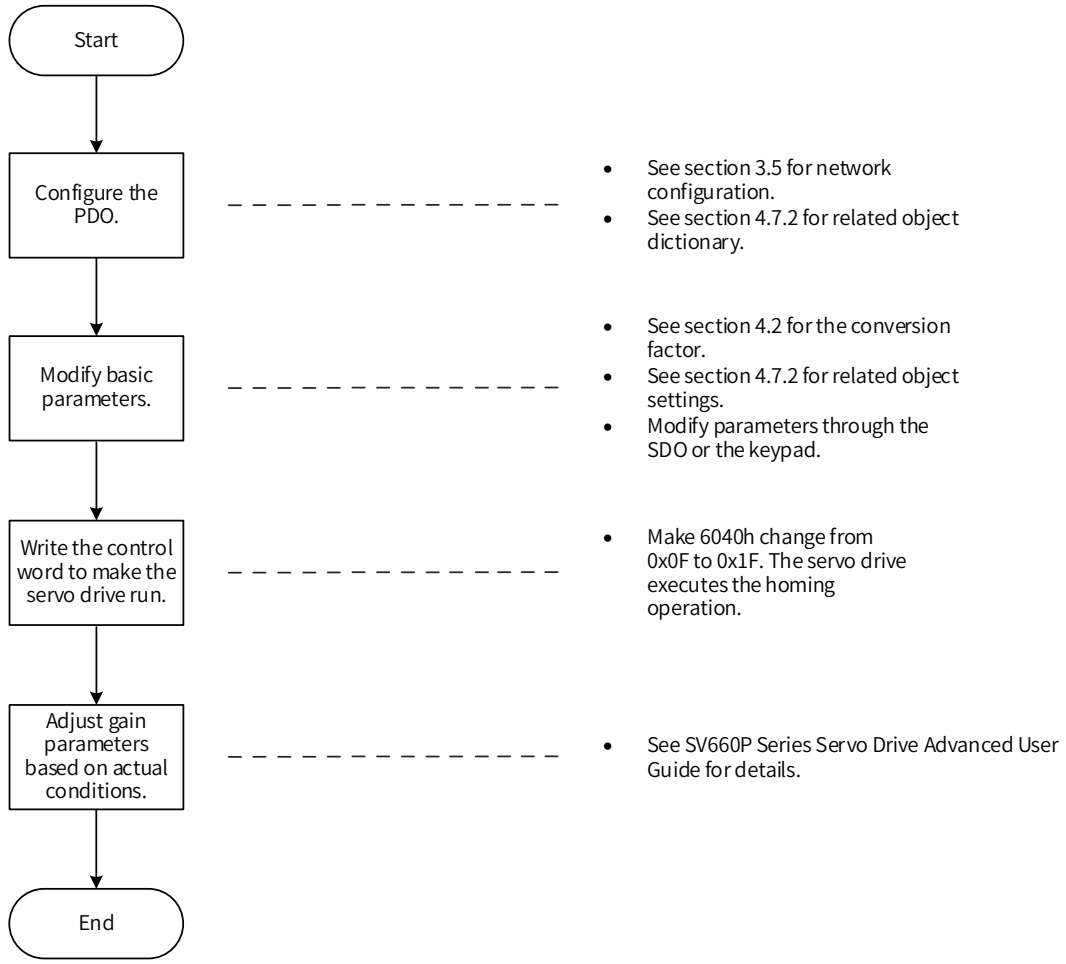


Figure 4-42 Example of homing mode setting process

Para. No.	Object	Mapping Object	Input	Description
H2D-32	1600h-00h	Number of RPDO1 mapping objects	2	
H2D-33	1600h-01h	6040h-00h	60400010h	The first mapping parameter of RPDO1 is 6040-00h, with a length of 16 bits.
H2D-35	1600h-02h	6060h-00h	60600008h	The second mapping parameter of RPDO1 is 6060-00h, with a length of 8 bits.
H2D-49	1601h-00h	Number of RPDO2 mapping objects	2	
H2D-50	1601h-01h	6098h-00h	60980008h	The first mapping parameter of RPDO2 is 6098-00h, with a length of 8 bits.
H2D-52	1601h-02h	609Ah-00h	609A0020h	The second mapping parameter of RPDO2 is 609A-00h, with a length of 32 bits.

Para. No.	Object	Mapping Object	Input	Description
H2D-66	1602h-00h	Number of RPDO3 mapping objects	2	
H2D-67	1602h-01h	6099h-01h	60990120h	The first mapping parameter of RPDO3 is 6099-01h, with a length of 32 bits.
H2D-69	1602h-02h	6099h-02h	60990220h	The second mapping parameter of RPDO3 is 6099-02h, with a length of 32 bits.
H2E-20	1A00h-00h	Number of TPDO1 mapping objects	2	
H2E-21	1A00h-01h	6041h-00h	60410010h	The first mapping parameter of TPDO1 is 6041-00h, with a length of 16 bits.
H2E-23	1A00h-02h	6061h-00h	60610008h	The second mapping parameter of TPDO1 is 6061-00h, with a length of 8 bits.
H2E-37	1A01h-00h	Number of TPDO2 mapping objects	2	
H2E-38	1A01h-01h	6064h-00h	60640020h	The first mapping parameter of TPDO2 is 6064-00h, with a length of 32 bits.
H2E-40	1A01h-02h	606Ch-00h	606C0020h	The second mapping parameter of TPDO2 is 606C-00h, with a length of 32 bits.

- Write the servo drive operation mode (6060h) to 0x06 to make it run in the homing mode.
- Write the homing mode 6098h (in reference unit, default value: 1).
- Write the speed during search for switch 6099-01h (in reference unit, default value: 1747627) and the speed during search for zero 6099-02h (in reference unit, default value: 174763).
- Set the homing acceleration 609Ah (in reference unit, default value: 174762666).
- Write 6040h from 0x0F to 0x1F. The servo drive executes the homing operation.

Monitoring parameters:

- Position demand value 6062h (in reference unit), position demand value* 60FCh (in encoder unit)
- Position actual value* 6063h (in encoder unit), position actual value 6062h (in reference unit)
- Following error actual value 60F4h (in reference unit)
- Status word 6041h

Example:

When 6060h = 0x06, 6098h = 3:

- Speed during search for switch: 6099-1h = 100 x 1048576/60 p/s (100 RPM)
- Speed during search for zero: 6099-2h = 10 x 1048576/60 p/s (10 RPM)
- Homing acceleration: 609Ah = 100 x 1048576/60 p/s^2 (100 RPM/s)

Operating Steps	Control Command 6040h	Status of 6041h	Description
1	0x06	0x0231	The servo drive is ready.
2	0x07	0x0233	The servo drive is ready and can be enabled.
3	0x0F	0x0637	Homing is not started and the target position is reached.
4	0x1F	0x9637	Homing is done and the target position is reached.

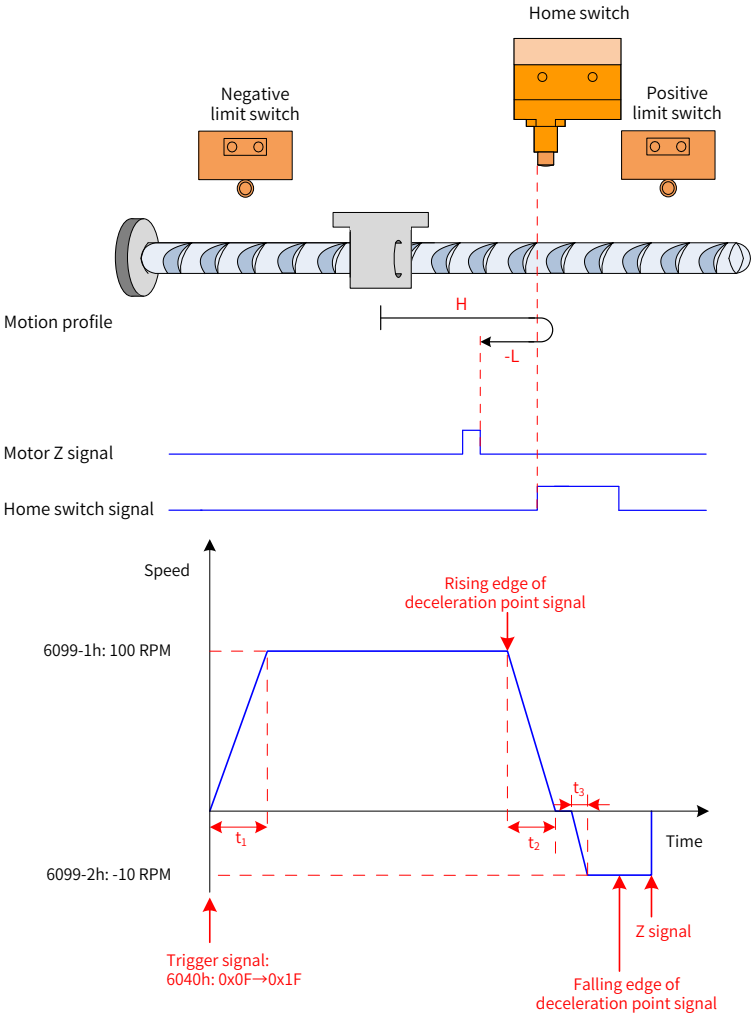


Figure 4-43 6060h = 0x06 and 6098h = 3

$$t_1 = \frac{(6099\text{-}1h)}{609Ah}\ s \qquad t_2 = \frac{(6099\text{-}1h)}{609Ah}\ s \qquad t_3 = \frac{(6099\text{-}2h)}{609Ah}\ s$$

4.8 Interpolation Mode

The interpolation mode can be used to achieve synchronous operations of multi-axis servo drives or single-axis servo drives. The host controller sets the interpolation cycle before the servo drive is enabled and plans the displacement curve in advance based on actual application needs. After the servo drive starts running, the host controller sends different absolute positions in the displacement curve to the slave cyclically. The slave receives the displacement reference synchronously and evenly divides and transmits the displacement reference increment based on the position loop control cycle. The servo drive performs the position control, speed control, and torque control.

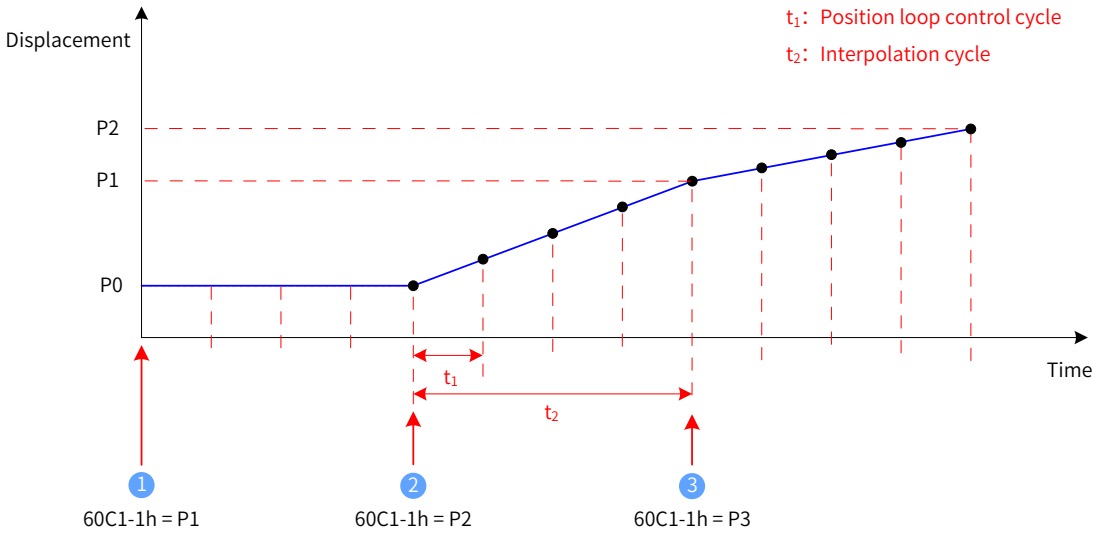
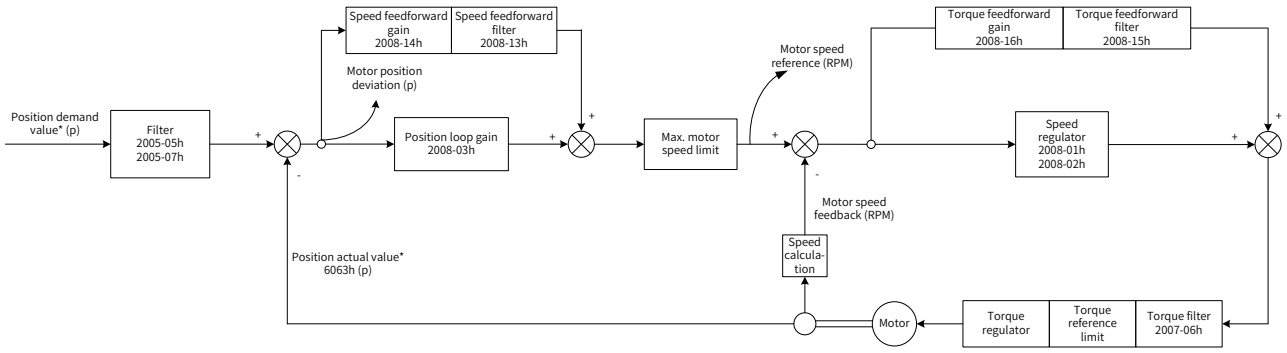


Figure 4-44 Motor displacement curve in single-axis linear interpolation

NOTE

- 1 : The current absolute position of the servo motor is P0, the first absolute position reference is received, and the planning of the first displacement curve starts.
- 2 : The current absolute position of the servo motor is P0. The servo motor starts to move towards the first absolute position P1, meanwhile, the second absolute position reference P2 is received, the planning of the second displacement curve starts.
- 3 : The servo motor reaches the first absolute position P1 and starts to move towards the second absolute position P2, meanwhile, the third absolute position reference P3 is received, the planning of the third displacement curve starts.
- t_1 - position loop control cycle, which is determined by the servo drive
- t_2 - interpolation cycle, which is set in the object dictionary 60C2h. The SV660C servo drive supports a synchronization cycle of 1 ms to 20 ms. Synchronization cycles beyond this range will be limited to this range.
- P0/P1/P2 - absolute position. An absolute position reference is sent through 60C1-1h. The interpolation mode supports absolute position references only.
- When using the interpolation mode, set the position loop cycle to 1k (H01-13).

4.8.1 Control Block Diagram



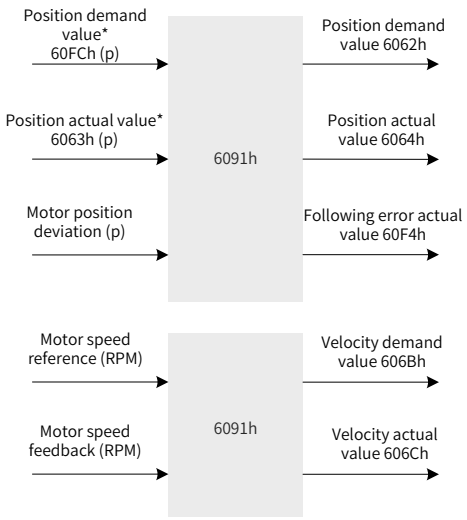


Figure 4-45 Control block diagram of the interpolation mode

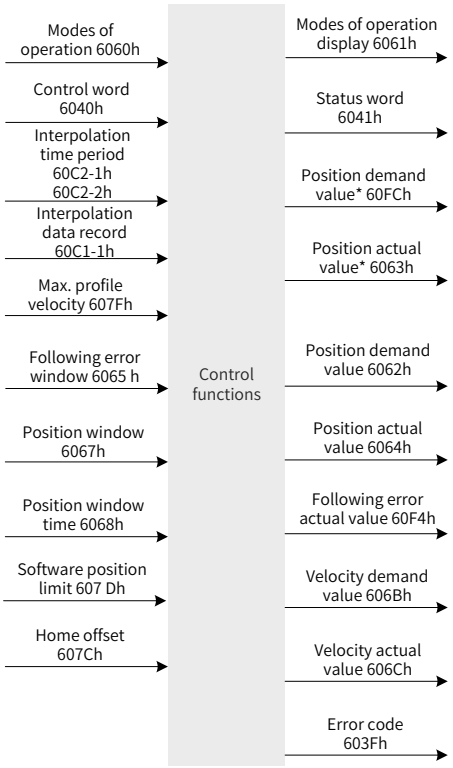


Figure 4-46 Input/output block diagram in the interpolation mode

Setting 0x200A-02A to 1 (H0A_01) enables the software limit. The software limit is disabled by default (200A-02h = 0). After the software limit is enabled, the motor stops and reports an overtravel warning upon reaching the limit, and bit11 of 6041h is set to 1. In this case, send a reverse run command to exit the servo drive from the limit state and zero out bit11 of 6041h. When the external DI limit switch and internal software position limit are both activated, the overtravel status will be determined by the external DI limit switch.

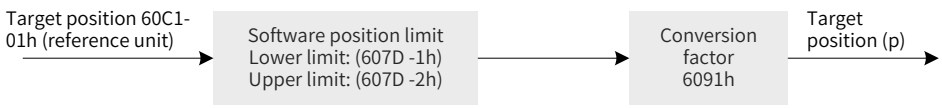


Figure 4-47 Interpolation displacement 60C1h - internal software position limit

4.8.2 Related Object Settings

1) Positioning completed

Index 6067h	Name	Position window					Data Structure	VAR	Data Type	Uint32
	Access	RW	Mapping	YES	Related Mode	pp/hm/ip	Data Range	0 to 4294967295	Default	734p
Subindex: 00 When the position deviation (60F4h) of the reference unit is smaller than 6067h and the time reaches 6068h, bit10 of 6041h is set to 1. When either condition is not met, the position window is invalid.										

Index 6068h	Name	Position window time					Data Structure	VAR	Data Type	Uint16
	Access	RW	Mapping	YES	Related Mode	pp/hm/ip	Data Range	0 to 65535	Default	0 ms
Subindex: 00 When the position deviation (60F4h) of the reference unit is smaller than 6067h and the time reaches 6068h, bit10 of 6041h is set to 1. When either condition is not met, the position window is invalid.										

2) Following error window detection

Index 6065h	Name	Following error window					Data Structure	VAR	Data Type	Uint32
	Access	RW	Mapping	YES	Related Mode	pp/hm/ip	Data Range	0 to 4294967295	Default	3435868
Subindex: 00 When the position deviation is larger than this value, Er.B00 occurs.										

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
603Fh	00h	Error code	RO	TPDO	Uint16	-	0 to 65535	-
6040h	00h	Control word	RW	YES	Uint16	-	0 to 65535	0
6041h	00h	Status word	RO	TPDO	Uint16	-	0 to 65535	-
6060h	00h	Modes of operation	RW	YES	Int8	-	0 to 7	0
6061h	00h	Modes of operation display	RO	TPDO	Int8	-	0 to 7	-
6062h	00h	Position demand value	RO	TPDO	Int32	Reference unit	-2^{31} to $+(2^{31}-1)$	-
6063h	00h	Position actual value*	RO	TPDO	Int32	Encoder unit	-2^{31} to $+(2^{31}-1)$	-
6064h	00h	Position actual value	RO	TPDO	Int32	Reference unit	-2^{31} to $+(2^{31}-1)$	-
6065h	00h	Following error window	RW	YES	Uint32	Reference unit	0 to $(2^{32}-1)$	3435868
6067h	00h	Position window	RW	YES	Uint32	Reference unit	0 to $(2^{32}-1)$	734
6068h	00h	Position window time	RW	YES	Uint16	ms	0 to 65535	0

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
606Bh	00h	Velocity demand value	RO	TPDO	Int32	Reference unit	-2^{31} to $+(2^{31}-1)$	-
606Ch	00h	Velocity actual value	RO	TPDO	Int32	Reference unit	-2^{31} to $+(2^{31}-1)$	-
607Dh	01h	Min. position limit	RW	YES	Int32	Reference unit	-2^{31} to $+(2^{31}-1)$	-2^{31}
	02h	Max. position limit	RW	YES	Int32	Reference unit	-2^{31} to $+(2^{31}-1)$	$2^{31}-1$
607Ch	00h	Home offset	RW	YES	Int32	Reference unit	-2^{31} to $+(2^{31}-1)$	0
6098h	00h	Homing method	RW	YES	Int8	-	0 to 35	1
6099h	01h	Speed during search for switch	RW	YES	Uint32	Reference unit	0 to $(2^{32}-1)$	1747627
	02h	Speed during search for zero	RW	YES	Int32	Reference unit	0 to $(2^{32}-1)$	174763
609Ah	00h	Homing acceleration	RW	YES	Uint32	RPM/ms	0 to $(2^{32}-1)$	174762666
60C1h	01h	Interpolation data record	RW	YES	Int32	-	-2^{31} to $+(2^{31}-1)$	0
60C2h	01h	Interpolation time units	RW	YES	Uint8	-	1 to 20	1
	02h	Interpolation time index	RO	TPDO	Int8	ms	-3	-3
60F4h	00h	Following error actual value	RO	TPDO	Int32	Reference unit	-2^{31} to $+(2^{31}-1)$	-
60FCh	00h	Position demand value	RO	TPDO	Int32	Encoder unit	-2^{31} to $+(2^{31}-1)$	-
2007h	06h	Torque reference filter time constant	RW	YES	Uint16	ms	0 to 30.00	0.79
2008h	01h	Speed loop gain	RW	YES	Uint16	Hz	0.1 to 2000.0	25.0
	02h	Speed loop integral time constant	RW	YES	Uint16	ms	0.15 to 512.00	31.83
	03h	Position loop gain	RW	YES	Uint16	Hz	0.0 to 2000.0	40.0
	13h	Speed feedforward filter time constant	RW	YES	Uint16	ms	0.00 to 64.00	0.50
	14h	Speed feedforward gain	RW	YES	Uint16	%	0.0 to 100.0	0.0
	15h	Torque feedforward filter time constant	RW	YES	Uint16	ms	0.00 to 64.00	0.50
	16h	Torque feedforward gain	RW	YES	Uint16	%	0.0 to 200.0	0.0

4.8.3 Control Commands in the Interpolation Mode

The control word 6040h in the interpolation mode is described as follows:

Index	Name	Control word					Data Structure	VAR	Data Type	Uint16
6040h	Access	RW	Mapping	YES	Related Mode	All	Data Range	0 to 65535	Default	-

Defines the control commands in the interpolation mode.

Control Word 6040h				
Bit	bit7 to bit15	bit5 to bit6	bit4	bit0 to bit3
Name	-	N/A	Enable interpolation mode	-
Setting value	See Table 4-2 for details.	-	-	See Table 4-2 for details.
Description	For details, see *"6.5.3 Profile-Defined Parameters"*.	-	0: Interpolation interrupted 1: Interpolation enabled The bit4 must always be 1 during interpolation. The bit12 of 6041h can be used as a feedback indicating whether the interpolation mode is activated.	For details, see *"6.5.3 Profile-Defined Parameters"*.

The status word 6041h in the interpolation mode is described as follows:

Index	Name	Status word					Data Structure	VAR	Data Type	Uint16
6041h	Access	RO	Mapping	TPDO	Related Mode	All	Data Range	0 to 65535	Default	-

Indicates the status of the servo drive in the interpolation mode.

Status word 6041h							
Bit	bit15	bit14	bit13	bit12	bit11	bit10	bit0 to bit9
Name	Homing completed	N/A	Unused	Interpolation activated	Software internal setting beyond the limit	Target reached	-
Value	-	-	-	-	-	-	See Table 4-2 for details.
Description	0: Homing not performed or done 1: Homing done and reference point found	-	-	0: Interpolation not activated 1: Interpolation activated	0: Actual position value not reaching the software internal position limit 1: Actual position value reaching the software internal position limit [1].	0: Target position not reached 1: Target position reached [2]	For details, see *"6.5.3 Profile-Defined Parameters"*.

[1] The software internal position limit can be enabled according to the setting of 0x200A-02h. For details, see description of 607Dh in *"6.5.3 Profile-Defined Parameters"*.

[2] When the position deviation is within the position window (6067h) and the time reaches 6068h, the target position is reached. If either condition is not met, the target position is not reached.

4.8.4 Configuration Example

Para. No.	Object	Mapping Object	Input	Description
H2D-32	1600h-00h	Number of RPDO1 mapping objects	2	-
H2D-33	1600h-01h	6040h-00h	60400010h	The first mapping parameter of RPDO1 is 6040-00h, with a length of 16 bits.
H2D-35	1600h-02h	6060h-00h	60600008h	The second mapping parameter of RPDO1 is 6060-00h, with a length of 8 bits.
H2D-49	1601h-00h	Number of RPDO2 mapping objects	1	-
H2D-50	1601h-01h	60C1h-01h	60C10020h	The first mapping parameter of RPDO2 is 60C1-00h, with a length of 32 bits.
H2D-50	1601h-01h	-	0	-
H2E-20	1A00h-00h	Number of TPDO1 mapping objects	2	-
H2E-21	1A00h-01h	6041h-00h	60410010h	The first mapping parameter of TPDO1 is 6041-00h, with a length of 16 bits.
H2E-23	1A00h-02h	6061h-00h	60610008h	The first mapping parameter of TPDO1 is 6061-00h, with a length of 8 bits.
H2E-37	1A01h-00h	Number of TPDO2 mapping objects	2	-
H2E-38	1A01h-01h	6064h-00h	60640020h	The first mapping parameter of TPDO2 is 6064-00h, with a length of 32 bits.
H2E-40	1A01h-02h	606Ch-00h	606C0020h	The second mapping parameter of TPDO2 is 606C-00h, with a length of 32 bits.

Example:

6060h = 0x07:

When the servo drive stops, if 60C2-1h is set to 10 through the SDO, the interpolation cycle is 10 ms.

The interpolation displacement record 60C1-01h needs to be configured to the synchronous PDO type.

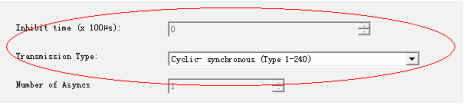


Figure 4-48 Configuration example of 60C1-01h

- Set the servo drive operation mode (6060h) to 0x07 to make it run in the interpolation mode.
- Write the interpolation position 60C1-1h (absolute position references supported only), interpolation time constant 60C2-1h, and interpolation time index 60C2-2h (the default value is -3 ms and can be modified to -2 (10 ms)). The SYNC cycle must be set to a value between 1 ms to 20 ms.
- Write the control word 6040h from 0x0F to 0x1F so that the servo drive can run. Here is an example of the configuration:

Operating Step	Control Command 6040h	Status of 6041h	Description
1	0x06	0x0231	No fault → Servo ready
2	0x07	0x0233	Servo running → Wait for the S-ON signal
3	0x0F	0x0637	Target position reached
4	0x0F	0x0A37	Target position not reached and position reference exceeding the limit
5	0x0F	0x0E37	Target position reached and position reference exceeding the limit
6	0x1F	0x1237	Interpolation mode activated and target position not reached
7	0x1F	0x1637	Interpolation mode activated and target position reached
8	0x1F	0x1A37	Interpolation mode activated, target position not reached, and position reference exceeding the limit
9	0x1F	0x1E37	Interpolation mode activated, target position reached, and position reference exceeding the limit

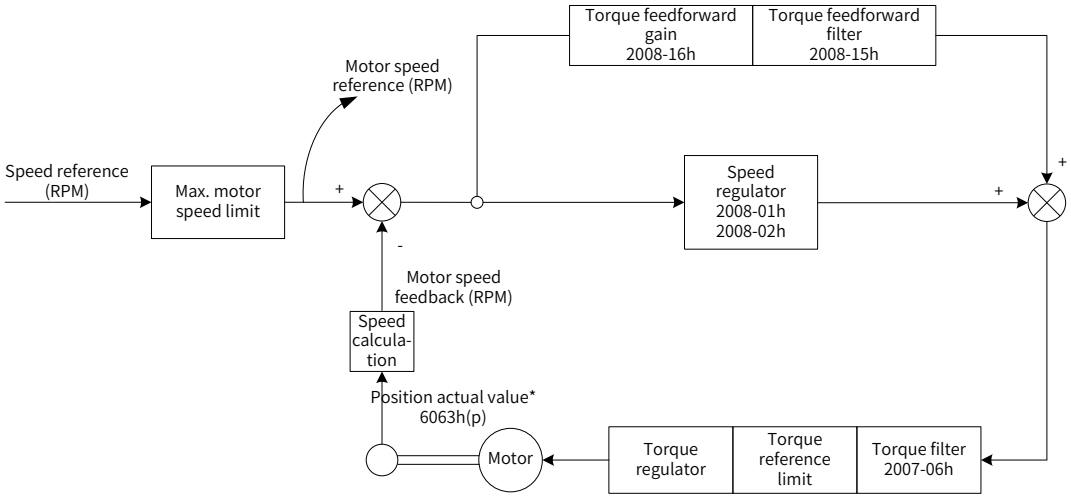
Monitoring parameters:

- Position demand value 6062h (in reference unit), position demand value* 60FCh (in encoder unit)
- Position actual value* 6063h (in encoder unit), position actual value 6062h (in reference unit)
- Following error actual value 60F4h (in reference unit)
- Status word 6041h

4.9 Profile Velocity Mode

In the profile velocity mode, after the user sets the speed, acceleration, and deceleration, the servo drive plans the motor velocity curve based on user settings and switches between different velocity references smoothly.

4.9.1 Control Block Diagram



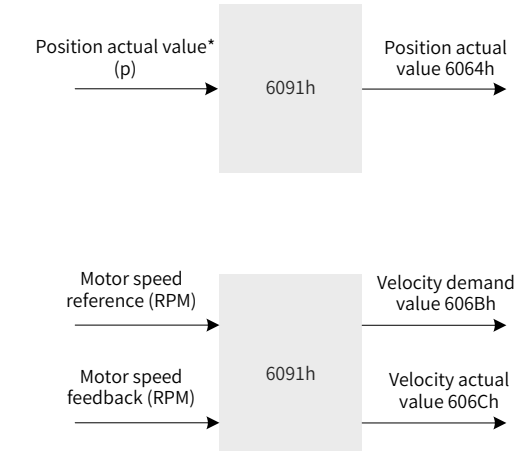


Figure 4-49 Control block diagram of the profile velocity mode

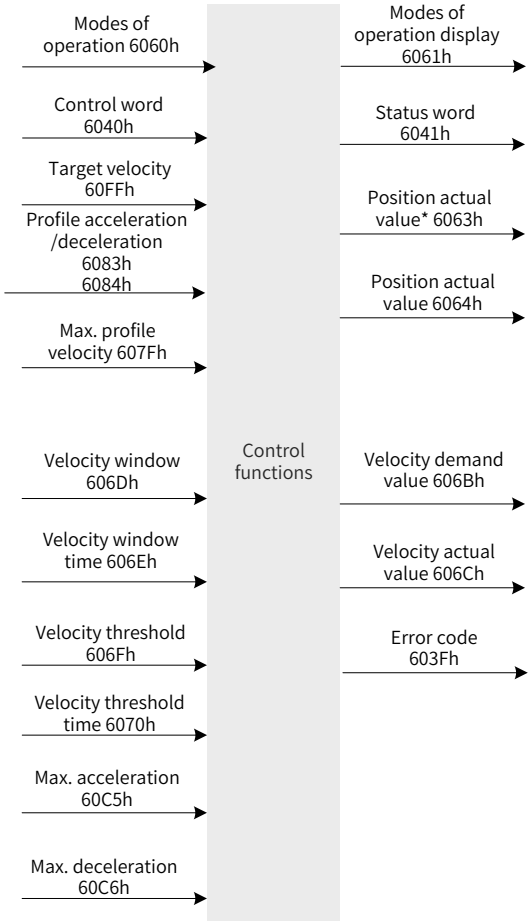


Figure 4-50 Input/output block diagram in the profile velocity mode

Velocity curve planning involves the target velocity 60FFh (in reference unit), profile acceleration 6083h (in reference unit), and profile deceleration 6084h (in reference unit). References of the host controller, which are input in reference units, are in the encoder unit after being limited and converted through the conversion factor. The processing of the target velocity, profile acceleration, and profile deceleration by the servo drive are shown in Figure 4-51, Figure 4-52, and Figure 4-53.

Setting 0x200A-02A to 1 (H0A_01) enables the software limit. The software limit is disabled by default (200A-02h = 0). After the software limit is enabled, the motor stops and reports an overtravel warning upon reaching the limit, and bit11 of 6041h is set to 1. In this case, send a reverse run command to exit the servo drive from the limit state and zero out bit11 of 6041h. When the external DI limit switch and internal software position limit are both activated, the overtravel status is determined by the external DI limit switch.

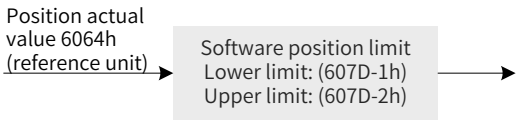


Figure 4-51 Position actual value 6064h - internal software position detection

The target velocity (60FFh) is used to set the maximum speed when the velocity reference is running. It cannot exceed the maximum velocity (607Fh) set by the user and the maximum motor speed after conversion. Figure 4-52 shows the block diagram.

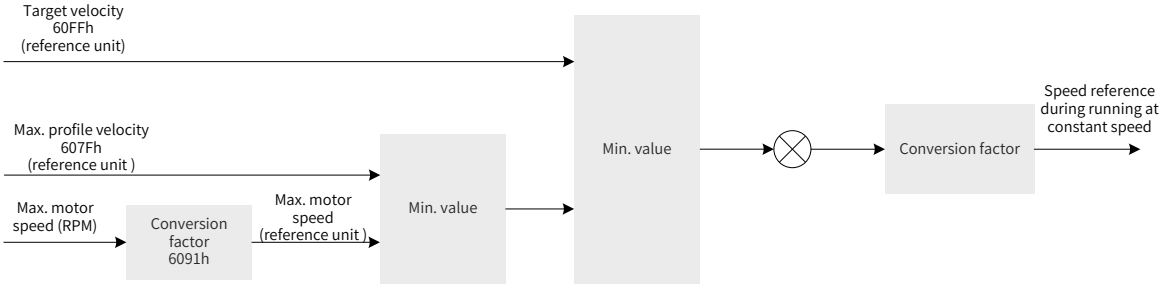


Figure 4-52 Target velocity 60FFh - velocity limit

The profile acceleration (6083h) and profile deceleration (6084h) are used to set the acceleration/deceleration when the velocity reference is running. The values of 6083h and 6084h cannot exceed the maximum acceleration/deceleration (60C5h/60C6h) set by the user. Figure 4-53 shows the block diagram.

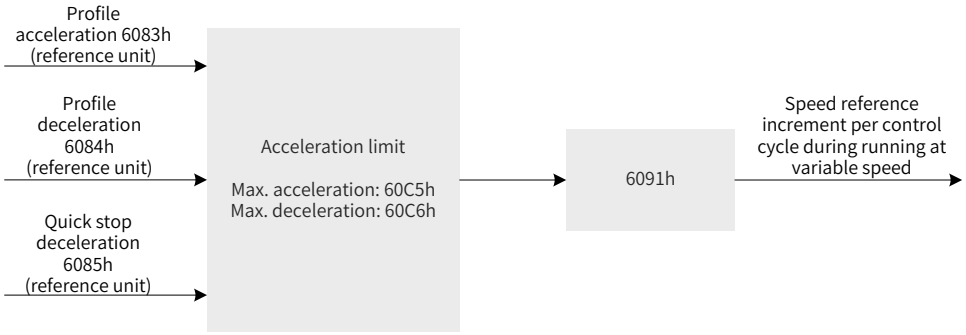


Figure 4-53 Profile acceleration limit

4.9.2 Related Object Settings

1) Zero clamp

Index 2006h	Name	Speed threshold for zero clamp					Data Structure	VAR	Data Type	Uint16
	Access	RW	Mapping	YES	Related Mode	pv	Data Range	0 to 6000	Default	10 RPM
Subindex: 10h When the actual velocity is smaller than this value and the corresponding DI function 12 is enabled, the motor enters the position lock state.										

2) Velocity threshold

Index 606Fh	Name	Velocity threshold					Data Structure	VAR	Data Type	Uint16
	Access	RW	Mapping	YES	Related Mode	pv	Data Range	0 to 65535	Default	10 RPM
Subindex: 00h When the velocity actual value (606Ch) of the reference unit is smaller than 606Fh and the time reaches 6070h, bit12 of 6041h is set to 1. When either condition is not met, the speed reached is invalid.										

Index 6070h	Name	Velocity threshold time					Data Structure	VAR	Data Type	Uint16
	Access	RW	Mapping	YES	Related Mode	pv	Data Range	0 to 65535	Default	0 ms
Subindex: 00h When the velocity actual value (606Ch) of the reference unit is smaller than 606Fh and the time reaches 6070h, bit12 of 6041h is set to 1. When either condition is not met, the speed reached is invalid.										

3) Speed reached

Index 606Dh	Name	Velocity window					Data Structure	VAR	Data Type	Uint16
	Access	RW	Mapping	YES	Related Mode	pv	Data Range	0 to 65535	Default	10 RPM
Subindex: 00h When the deviation between the velocity actual value (606Ch) of the reference unit and the target velocity (60FFh) is smaller than 606Dh and the time reaches 606Eh, bit10 of 6041h is set to 1. When either condition is not met, the speed reached is invalid.										

Index 606Eh	Name	Velocity window time					Data Structure	VAR	Data Type	Uint16
	Access	RW	Mapping	YES	Related Mode	pv	Data Range	0 to 65535	Default	0 ms
Subindex: 00h When the deviation between the velocity actual value (606Ch) of the reference unit and the target velocity (60FFh) is smaller than the value of 606Dh and the time reaches 606Eh, bit10 of 6041h is set to 1. When either condition is not met, the speed reached is invalid.										

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
603Fh	00h	Error code	RO	TPDO	Uint16	-	0 to 65535	-
6040h	00h	Control word	RW	YES	Uint16	-	0 to 65535	0
6041h	00h	Status word	RO	TPDO	Uint16	-	0 to 65535	-
6060h	00h	Modes of operation	RW	YES	Int8	-	0 to 7	0
6061h	00h	Modes of operation display	RO	TPDO	Int8	-	0 to 7	-

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
6063h	00h	Position actual value*	RO	TPDO	Int32	Encoder unit	-2^{31} to $+(2^{31}-1)$	-
6064h	00h	Position actual value	RO	TPDO	Int32	Reference unit	-2^{31} to $+(2^{31}-1)$	-
606Bh	00h	Velocity demand value	RO	TPDO	Int32	Reference unit	-2^{31} to $+(2^{31}-1)$	-
606Ch	00h	Velocity actual value	RO	TPDO	Int32	Reference unit	-2^{31} to $+(2^{31}-1)$	-
606Dh	00h	Velocity window	RW	YES	Uint16	RPM	0 to 65535	10
606Eh	00h	Velocity window time	RW	YES	Uint16	ms	0 to 65535	0
606Fh	00h	Velocity threshold	RW	YES	Uint16	RPM	0 to 65535	10
6070h	00h	Velocity threshold time	RW	YES	Uint16	ms	0 to 65535	0
607Ch	00h	Home offset	RW	YES	Int32	Reference unit	-2^{31} to $+(2^{31}-1)$	0
607Dh	01h	Min. position limit	RW	YES	Int32	Reference unit	-2^{31} to $+(2^{31}-1)$	-2^{31}
	02h	Max. position limit	RW	YES	Int32	Reference unit	-2^{31} to $+(2^{31}-1)$	$2^{31}-1$
6083h	00h	Profile acceleration	RW	YES	Uint32	RPM/ms	0 to $(2^{32}-1)$	174762666
6084h	00h	Profile deceleration	RW	YES	Uint32	RPM/ms	0 to $(2^{32}-1)$	174762666
6091h	00h	Highest subindex supported	RO	NO	Uint8	-	-	2
	01h	Motor revolutions	RW	PRDO	Uint32	-	1 to $(2^{32}-1)$	1
	02h	Shaft revolutions	RW	PRDO	Uint32	-	1 to $(2^{32}-1)$	1
60C5h	00h	Max. acceleration	RW	YES	Uint32	p/ms	0 to $(2^{32}-1)$	2147483647
60C6h	00h	Max. deceleration	RW	YES	Uint32	p/ms	0 to $(2^{32}-1)$	2147483647
2007h	06h	Torque reference filter time constant	RW	YES	Uint16	ms	0 to 30.00	0.79
2008h	01h	Speed loop gain	RW	YES	Uint16	Hz	0.1 to 2000.0	25.0
	02h	Speed loop integral time constant	RW	YES	Uint16	ms	0.15 to 512.00	31.83
	15h	Torque feedforward filter time constant	RW	YES	Uint16	ms	0.00 to 64.00	0.50
	16h	Torque feedforward gain	RW	YES	Uint16	%	0.0 to 200.0	0.0

4.9.3 Control Commands in the Profile Velocity Mode

The control word 6040h in the profile velocity mode is described as follows:

Index 6040h	Name	Control word					Data Structure	VAR	Data Type	Uint16
	Access	RW	Mapping	YES	Related Mode	All	Data Range	0 to 65535	Default	-

Used to set control commands in the profile velocity mode to make them be the same as those in the state machine.

6040	Description
0x06	The servo drive is ready.
0x07	The servo drive is ready and can be enabled.
0x0F	The servo is enabled and runs according to the reference velocity curve.

The status word 6041h in the profile velocity mode is described as follows.

Index 6041h	Name	Status word					Data Structure	VAR	Data Type	Uint16
	Access	RO	Mapping	TPDO	Related Mode	All	Data Range	0 to 65535	Default	-

Indicates the status of the servo drive in the profile velocity mode.

Status word 6041h							
Bit	bit15	bit14	bit13	bit12	bit11	bit10	bit0 to bit9
Name	Homing completed	N/A	Unused	Zero speed signal	Software internal setting overlimit	Target reached	-
Value	-	-	-	-	-	-	See Table 4-2 for details.
Description	0: Homing not performed or done 1: Homing done and reference point found	-	-	0: The user velocity is not 0. 1: The user velocity is 0.[1]	0: Position feedback not reaching the software internal position limit 1: Position feedback reaching the software internal position limit [2]	0: Target velocity not reached 1: Target velocity reached[3]	For details, see *"6.5.3 Profile-Defined Parameters"*.

[1] When the user velocity is within the velocity threshold (606Fh) and the time reaches 6070h, the user velocity is 0. When either condition is not met, the user velocity is not 0. This flag bit, which is meaningful only in the profile velocity mode, is unrelated to the enable/disable state of the servo drive.

[2] The software internal position limit can be enabled according to the setting of 0x200A-02h. For details, see descriptions of 607Dh in *"6.5.3 Profile-Defined Parameters"*.

[3] When the target velocity is within the velocity window (606Dh) and the time reaches 606Eh, the target velocity is reached. If either condition is not met, the target velocity is not reached. This flag bit is meaningful only when the servo drive is enabled in the profile velocity mode.

4.9.4 Configuration Example

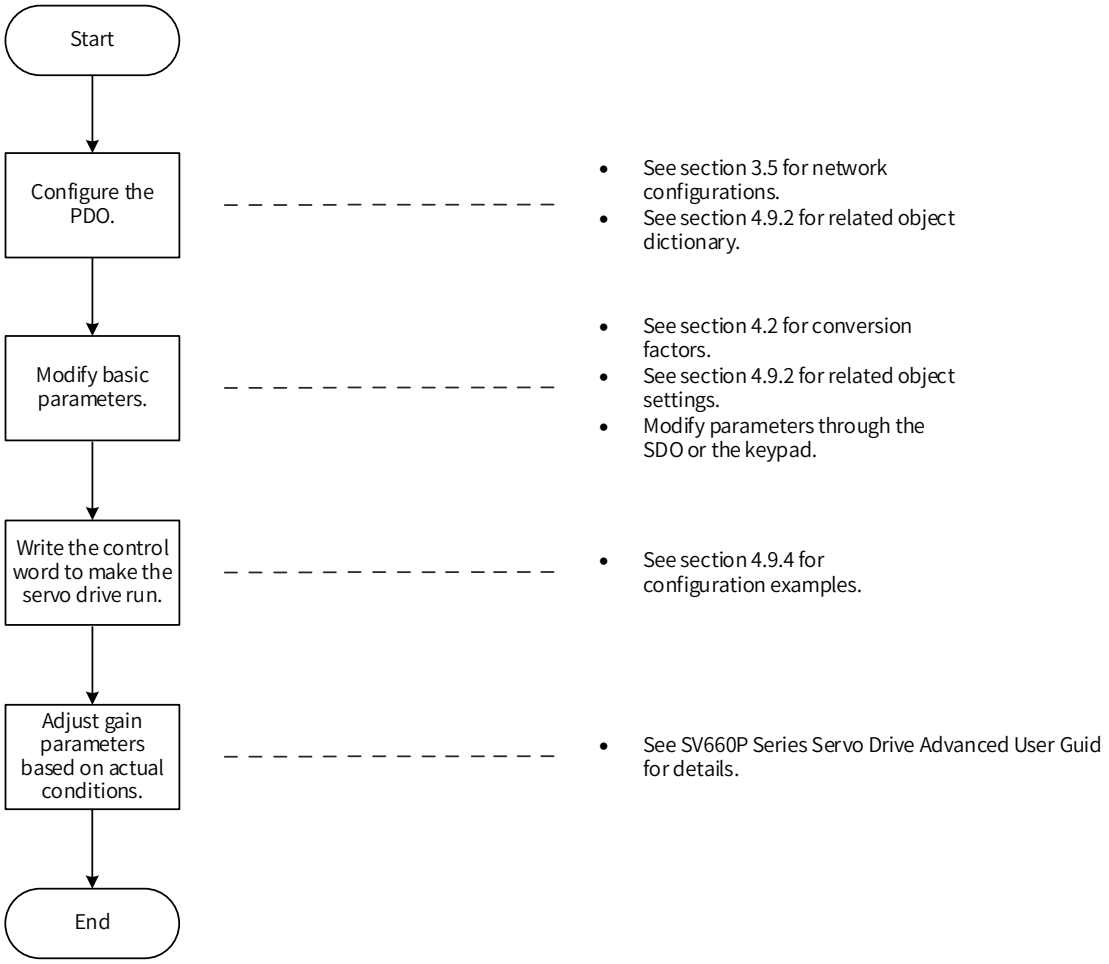


Figure 4-54 Example of the setting process of profile velocity mode

Para. No.	Object	Mapping Object	Input	Description
H2D-32	1600h-00h	Number of RPDO1 mapping objects	2	-
H2D-33	1600h-01h	6040h-00h	60400010h	The first mapping parameter of RPDO1 is 6040-00h, with a length of 16 bits.
H2D-35	1600h-02h	6060h-00h	60600008h	The second mapping parameter of RPDO1 is 6060-00h, with a length of 8 bits.
H2D-49	1601h-00h	Number of RPDO2 mapping objects	1	-
H2D-50	1601h-01h	60FFh-00h	60FF0020h	The first mapping parameter of RPDO2 is 60FF-00h, with a length of 32 bits.
H2D-50	1601h-01h	-	0	-
H2D-66	1602h-00h	Number of RPDO3 mapping objects	2	-
H2D-67	1602h-01h	6083h-00h	60830020h	The first mapping parameter of RPDO3 is 6083-00h, with a length of 32 bits.
H2D-69	1602h-02h	6084h-00h	60840020h	The second mapping parameter of RPDO3 is 6084-00h, with a length of 32 bits.
H2E-20	1A00h-00h	Number of TPDO1 mapping objects	2	
H2E-21	1A00h-01h	6041h-00h	60410010h	The first mapping parameter of TPDO1 is 6041-00h, with a length of 16 bits.

Para. No.	Object	Mapping Object	Input	Description
H2E-23	1A00h-02h	6061h-00h	60610008h	The second mapping parameter of TPDO1 is 6061-00h, with a length of 8 bits.
H2E-37	1A01h-00h	Number of TPDO2 mapping objects	2	
H2E-38	1A01h-01h	6064h-00h	60640020h	The first mapping parameter of TPDO2 is 6064-00h, with a length of 32 bits.
H2E-40	1A01h-02h	606Ch-00h	606C0020h	The second mapping parameter of TPDO2 is 606C-00h, with a length of 32 bits.

- Write the servo operation mode (6060h) to 0x03 to make it operate in the profile velocity mode.
- Write the target velocity ① : V_1 = 60FFh = 1000 x 1048576/60 p/s (1000 RPM)
- Write the profile acceleration ① : 6083h = 100 x 1048576/60 p/s^2 (100 RPM/s)
- Write the profile deceleration ① : 6084h = 100 x 1048576/60 p/s^2 (100 RPM/s)
- Write the target velocity ② : V_2 = 60FFh = 2000 x 1048576/60 p/s (2000 RPM)
- Write the profile acceleration ② : 6083h = 10 x 1048576/60 p/s^2 (10 RPM/s)
- Write the profile deceleration ② : 6084h = 10 x 1048576/60 p/s^2 (10 RPM/s)
- Write the control word 6040h and enable the servo drive. Here is an example of the configuration:

Operating Step	Control Command 6040h	Status of 6041h	Description
1	0x06	0x1231	The servo is ready and the velocity threshold is reached.
2	0x07	0x1233	The servo is ready and can be enabled. The velocity threshold is reached.
3	0x0F	0x0637	Homing is not started and the target position is reached.
4	0x06/0x07	0x1231	The profile velocity mode is interrupted and the velocity threshold is reached.

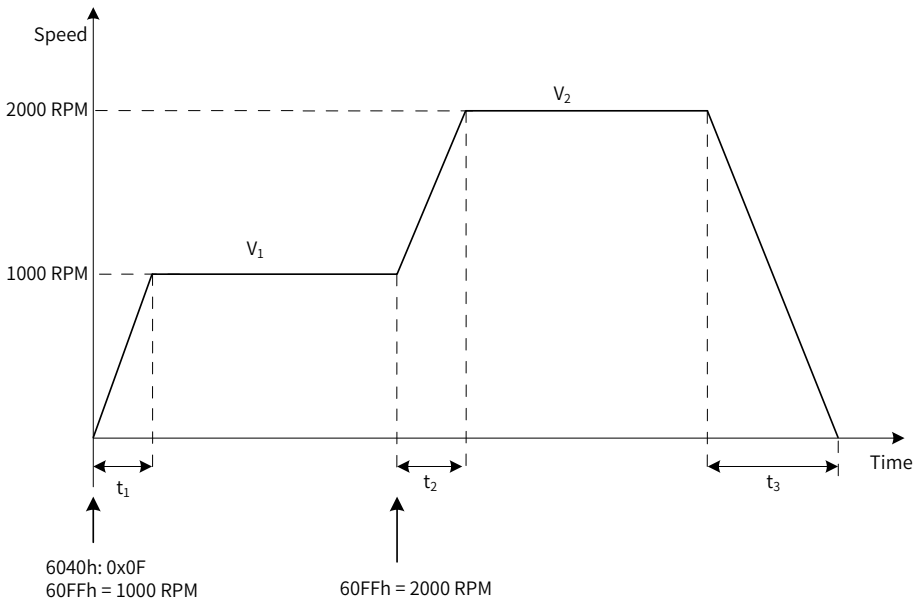


Figure 4-55 Profile velocity running curve

$$t_1 = \frac{V_1}{6083h}\ s \qquad t_2 = \frac{V_2 - V_1}{6083h}\ s \qquad t_3 = \frac{V_2}{6084h}\ s$$

4.10 Profile Torque Mode

In this mode, the host controller sends the target torque (6071h) and the torque slope (6087h) to the servo drive, and the servo drive executes torque adjustment. The speed adjustment applies when the speed limit is reached. The maximum output of adjustment cannot exceed the torque reference limit.

4.10.1 Control Block Diagram

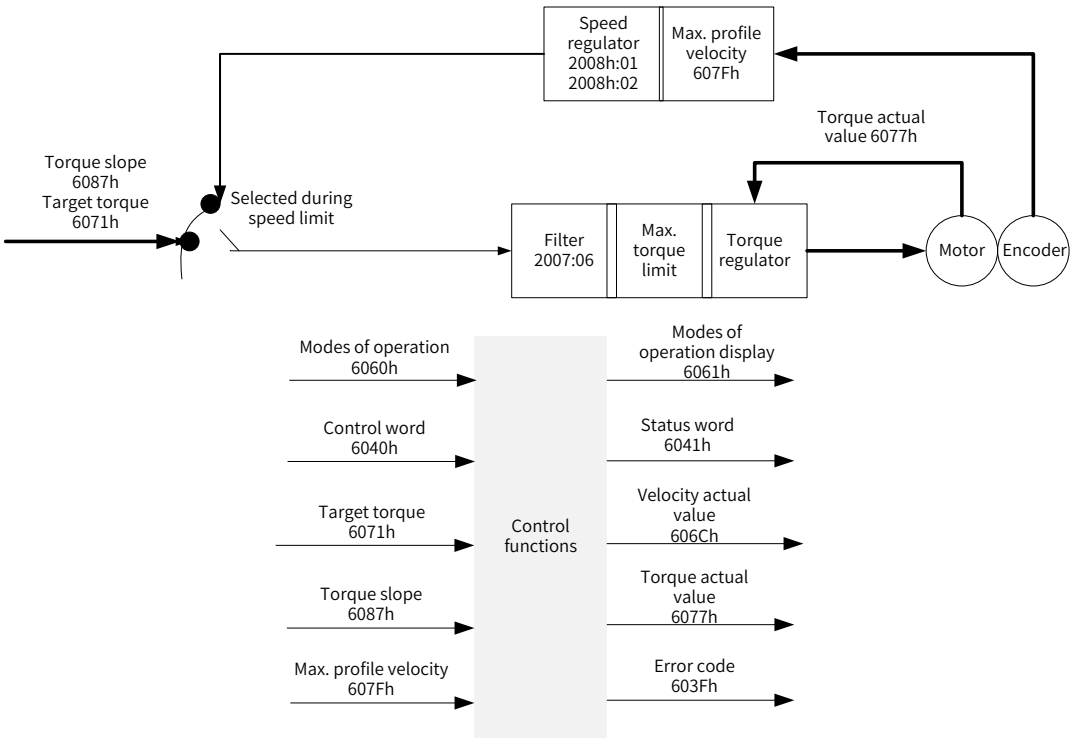


Figure 4-56 Control block diagram of the profile torque mode

4.10.2 Related Object Settings

Control Word 6040h		
Bit	Name	Description
0	Servo ready	If bit0 to bit3 are 1, the servo drive is started.
1	Main circuit switch-on	
2	Quick stop	
3	Servo running	

Status Word 6041h		
Bit	Name	Description
10	Target reached	0: Target torque reached 1: Target torque not reached
11	Software internal position limit violation	0: Position feedback within the limit 1: Position feedback beyond the limit
15	Homing completed	0: Homing not completed 1: Homing completed

Index (hex)	Sub-index (hex)	Name	Access	Size	Unit	Value Range	Default
603F	00	Error code	RO	UINT16	-	0 to 65535	0
6040	00	Control word	RW	UINT16	-	0 to 65535	0
6041	00	Status word	RO	UINT16	-	0 to 65535	0
6060	00	Modes of operation	RW	INT8	-	0 to 10	0
6061	00	Modes of operation display	RO	INT8	-	0 to 10	0
606C	00	Velocity actual value	RO	INT32	Reference unit/s	-2^{31} to $+(2^{31}-1)$	0
6071	00	Target torque	RW	INT16	0.1%	-5000 to +5000	0
6072	00	Max. torque	RW	UINT16	0.1%	0 to 0x0FA0	0x0BB8
6074	00	Torque demand value	RO	INT16	0.1%	-	0
6077	00	Torque actual value	RO	INT16	0.1%	-5000 to +5000	0
607F	00	Max. profile velocity	RW	UINT32	Reference unit/s	0 to $(2^{32}-1)$	1048576000
6087	00	Torque slope	RW	UINT32	0.1%/s	0 to $(2^{32}-1)$	4294967295
2007	06	Torque filter time	RW	UINT16	0.01 ms	0 to 65535	79
2008	01	Speed loop gain	RW	UINT16	0.1 Hz	1 to 20000	250
	02	Speed loop integral time	RW	UINT16	0.01 ms	15 to 51200	3183

■ Torque reached signal setting

When the difference between the torque and reference value is larger than the value of 2007-17h, the signal TOQREACH is output and bit10 of the status word 6041h is set to 1. When the difference is smaller than the value of 2007-18h, the output is invalid and bit10 of the status word 6041h is set to 0.

Index (hex)	Subindex (hex)	Name	Attribute	Size	Unit	Value Range	Default
2007	16	Reference value for torque reached	RW	UINT16	0.1%	0 to 8000	0
2007	17	Valid value for torque reached	RW	UINT16	0.1%	0 to 8000	200
2007	18	Invalid value for torque reached	RW	UINT16	0.1%	0 to 8000	100

4.10.3 Speed Limit in the Profile Torque Mode

The speed limit is determined by 607Fh (Max. profile velocity) and H00_15 (Max. speed).

Forward/Reverse speed: V = min{607Fh, H00_15}

5 Troubleshooting

When a communication error or servo drive error occurs, the SV660C series servo drive sends an emergency message to the network as a producer or sends an abort response when SDO transmission is abnormal. Error codes and auxiliary information related to nodes and emergency messages are listed as follows.

5.1 CANopen Communication Fault Codes

Display	Fault Name	Resettable or Not	Error Code (603Fh)	Auxiliary Code (203Fh)
Er.101	Parameter abnormal	No	0x6320	0x01010101
Er.102	Programmable logic configuration fault	No	0x7500	0x01020102
Er.104	Programmable logic interrupt	No	0x7500	0x01040104 0x01000104 0x0E940104
Er.105	Internal program error	No	0x6320	0x01050105
Er.108	Parameter storage error	No	0x5530	0x01080108
Er.110	Frequency-division pulse output setting error	Yes	0x6320	0x01100110
Er.111	Parameters in group H00/H01 being abnormal	No	0x6320	0x01110111
Er.120	Product mismatch	No	0x7122	0x01200120
Er.121	S-ON command invalid	Yes	0x5441	0x01210121
Er.130	Same function allocated to different DIs	Yes	0x6320	0x01300130
Er.131	DO allocation beyond the limit	Yes	0x6320	0x01310131
Er.136	Data check error or no parameter stored in the motor ROM	No	0x7305	0x01360136
Er.200	Overcurrent 1	No	0x2311	0x02000200
Er.201	Overcurrent 2	No	0x2312	0x02010201
Er.207	D/Q axis current overflow	Yes	0xFF00	0x02070207
Er.208	FPGA system sampling operation timeout	No	0xFF00	0x02080208
Er.210	Output short-circuited to ground	No	0x2330	0x02100210
Er.220	Wrong UVW phase sequence	No	0xFF00	0x02200220
Er.234	Runaway	No	0xFF00	0x02340234
Er.400	Main circuit overvoltage	Yes	0x3210	0x04000400
Er.410	Main circuit undervoltage	Yes	0x3220	0x04100410
Er.420	Main circuit power cable phase loss	Yes	0x3130	0x04200420
Er.430	Control circuit undervoltage	No	0x3120	0x04300430
Er.500	Motor overspeed	Yes	0x8400	0x05000500

Display	Fault Name	Resettable or Not	Error Code (603Fh)	Auxiliary Code (203Fh)
Er.510	Pulse output overspeed	Yes	0xFF00	0x05100510
Er.601	Homing timeout	Yes	0xFF00	0x06010601
Er.602	Angle auto-tuning failure	Yes	0xFF00	0x06020602
Er.610	Servo drive overload	Yes	0x3230	0x06100610
Er.620	Motor overload	Yes	0x3230	0x06200620
Er.630	Motor rotor locked	Yes	0x7121	0x06300630
Er.650	Heatsink over-temperature	Yes	0x4210	0x06500650
Er.740	Encoder interference	No	0x7305	0x07400740
Er.834	AD sampling overvoltage	No	0x5210	0x08340834
Er.900	DI emergency braking	Yes	0x5442	0x09000900
Er.909	Motor overload warning	Yes	0x3230	0x09090909
Er.920	Regenerative resistor overload	Yes	0x3210	0x09200920
Er.922	Resistance of external regenerative resistor too small	Yes	0x6320	0x09220922
Er.939	Motor power cable disconnected	Yes	0x3331	0x09390939
Er.941	Parameter modifications activated at next power-on	Yes	0x6320	0x09410941
Er.942	Parameters saved too frequently	Yes	0x7600	0x09420942
Er.950	Forward overtravel warning	Yes	0x5443	0x09500950
Er.952	Reverse overtravel warning	Yes	0x5444	0x09520952
Er.980	Encoder fault	Yes	0x7305	0x09800980
Er.990	Input phase loss	Yes	0x3130	0x09900990
Er.995	CANopen network disconnection recovery	Yes	0x8140	0x09950995
Er.996	CANopen network passive error	Yes	0x8120	0x09960996
Er.b03	Electronic gear ratio limit violated	Yes	0x6320	0x0b030b03
Er.A33	Encoder data error	No	0x7305	0x0A330A33
Er.A34	Encoder echo check error	No	0x7305	0x0A340A34
Er.A35	Z signal lost	No	0x7305	0x0A350A35
Er.A40	Parameter auto-tuning failure	Yes	0xFF00	0x0A400A40
Er.b00	Position deviation too large	Yes	0x8611	0x0b000b00
Er.b01	Pulse input error	Yes	0xFF00	0x0b010b01
Er.b02	Position deviation in full closed loop too large	Yes	0x8611	0x0b020b02
Er.d04	CANopen node guarding or heartbeat timeout	Yes	0x8130	0x0d040d04
Er.d05	NMT turned to initialization state when motor enabled	No	0x8160	0x0d050d05

Display	Fault Name	Resettable or Not	Error Code (603Fh)	Auxiliary Code (203Fh)
Er.d06	NMT turned to stop state when motor enabled	No	0x8170	0x0d060d06
Er.d07	CANopen network disconnected	Yes	0x8141	0x0d070d07
Er.d08	CANopen PDO transmission length error	Yes	0x8210	0x0d080d08
Er.d09	Software position upper/lower limit setting error	Yes	0x6320	0x0d090d09
Er.d10	Home offset setting error	Yes	0x6320	0x0d100d10
Er.d11	Synchronization cycle error too large	Yes	0x6320	0x0d110d11

5.2 Solutions to Communication Faults

This section describes the solutions to communication faults only. For solutions to the servo drive faults, see *SV660P Series Servo Drive Advanced User Guide* or *SV660P Series Servo Drive User Guide*.

Fault Code	Name	Cause	Solution
Er.d04	CANopen node guarding or heartbeat timeout	The slave reaches the time configured by the consumer or the node guarding time.	◆ Check whether all CAN nodes are online, or check the CANopen configuration, reset the node or communication.
Er.d05	NMT turned to initialization state when motor enabled	After the motor is enabled, the communication state machine turns to the initialization state (slave offline, heartbeat abnormal, load ratio too high, data frame loss, and false master reset).	◆ Check: 1. Whether the cable is shielded against interference. 2. Whether proper grounding is performed. 3. Whether the load ratio is too high due to excessive number of axes involved. 4. Whether asynchronous transmission is configured and whether the suppression time is too short. 5. Whether a false reset frame is triggered by the host controller. 6. Whether the termination resistor is installed.
Er.d06	NMT turned to stop state when motor enabled	A NMT stop command is received when the motor is enabled.	◆ Reset the NMT node. When the NMT is modified, disable the output stage.
Er.d07	CANopen network disconnected	There are too many errors.	◆ Check the CANopen network and connect the network again.
Er.d08	CANopen PDO transmission length error	The length of the content transmitted through the PDO is inconsistent with the mapping length during configuration.	◆ Re-configure the PDO and reset the node or communication.
Er.d09	Software position upper/lower limit setting error	The lower limit of the software position is larger than the upper limit.	◆ Set 0x607D correctly and ensure the value of 607D-1h is smaller than that of 607D-2h.
Er.d10	Home offset setting error	The home offset is outside the software position lower/upper limit.	◆ Set 607D and 607C correctly and ensure: 607C > (607D-1h) 607C < (607D-2h)

Fault Code	Name	Cause	Solution
Er.d11	Synchronization cycle error too large	The error of the synchronization cycle exceeds 1/4 of the set value.	◆ Check the settings of 60C2-1h and 60C2-2h and ensure the synchronization cycle parameters are set properly. ◆ Ensure the synchronization cycle of the host controller is set correctly and is consistent with the setting of 60C2h. ◆ Check the cable connections between the slave and the master.

5.3 SDO Transmission Abort Code

Abort Code	Description
0503 0000	Trigger bits not alternated
0504 0000	SDO protocol timeout
0504 0001	Client/Server command word illegal or unknown
0504 0005	Memory overflow
0601 0000	Access to objects not supported
0601 0001	Attempt to read a write-only object
0601 0002	Attempt to write a read-only object
0602 0000	Object not existed in the object dictionary
0604 0041	Object cannot be mapped to PDO
0604 to 0042	Number and length of mapped objects exceeding the PDO length
0604 0043	General parameters incompatible
0604 0047	General device incompatible internally
0606 0000	Access to object fails due to a hardware error
0607 0010	Data type and service parameter length not match
0607 0012	Data type not match, service parameter length too long
0607 0013	Data type not match, service parameter length too short
0609 0011	Subindex not existed
0609 0030	Invalid parameter value
0609 0031	Parameter value written too large
0609 0032	Parameter value written too small
0609 0036	Maximum value smaller than the minimum value
0800 0000	General error
0800 to 0020	Data cannot be transmitted or stored to the application
0800 to 0021	Data cannot be transmitted or stored to the application due to local control
0800 to 0022	Data cannot be transmitted or stored to the application due to current device status
0800 to 0023	Object dictionary error occurred or object dictionary not existed
0800 to 0024	Value not existed

6 Object Dictionary

6.1 Object Types

★ Definitions of terms

"Index": This field (in hexadecimal) specifies the position of each object in the object dictionary.

"Data type": See Table 6-1 for details.

Table 6-1 Data type

Data Type	Value Range	Data Length	DS301 Value
Int8	-128 to +127	1 byte	0002
Int16	-32768 to +32767	2 bytes	0003
Int32	-2147483648 to +2147483647	4 bytes	0004
Uint8	0 to 255	1 byte	0005
Uint16	0 to 65535	2 bytes	0006
Uint32	0 to 4294967295	4 bytes	0007
String	ASCII	-	0009

"Read/write type": See Table 6-2 for details.

Table 6-2 Read/write type

Read/Write Type	Description
RW	Read/Write
WO	Write-only
RO	Read-only
CONST	Constant, read-only

"Object type": See Table 6-3 for details.

Table 6-3 Object type

Type	Meaning	DS301 Value
VAR	Single simple value, including data types Int8, Uint16, and String	7
ARR	Data block of the same type	8
REC	Data block of different types	9

6.2 Object Group 1000h

The object group 1000h includes parameters required in CANopen communication. The parameters cannot be mapped to PDOs.

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
1000h	-	Device type	RO	NO	Uint32	VAR	Uint 32	0x20192
1001h	-	Error register	RO	NO	Uint8	VAR	Uint 8	0x0
1003h	-	Pre-defined error field	RO	NO	Uint32	ARR	-	-
	1h to 4h	Error field	RW	NO	Uint32	-	Uint 32	0
1005h	-	Synchronization message COB-ID	RW	NO	Uint32	VAR	Uint 32	0x80
1006h	-	Synchronization cycle	RW	NO	Uint32	VAR	Uint 32	0
1008h	-	Device manufacturer name	CONST	NO	String	VAR	String	SV660C servo drive
1009h	-	Hardware version	CONST	NO	String	VAR	String	V0.0
100Ah	-	Software version	CONST	NO	String	VAR	String	402.XX
100Ch	-	Node guarding time	RW	NO	Uint16	VAR	Uint 16	0
100Dh	-	Life factor	RW	NO	Uint8	VAR	Uint 8	0
1010h	-	Save parameters	RW	NO	Uint32	ARR	Uint 8	0
	1h	Save parameters of all objects	RW	NO	Uint32	-	-	1
	2h	Save parameters of communication objects	RW	NO	Uint32	-	-	1
	3h	Save parameters of objects in the profile area	RW	NO	Uint32	-	-	1
1011h	0h	Restore default parameters	RW	NO	Uint32	ARR	-	-
	1h	Restore default parameters of all objects	RW	NO	Uint32	-	-	1
	2h	Restore default parameters of communication objects	RW	NO	Uint32	-	-	1
	3h	Restore default parameters of objects in the profile area	RW	NO	Uint32	-	-	1
1014h	-	Emergency message COB-ID	RW	NO	Uint32	VAR	Uint 32	0x80_Node_ID
1016h	-	Consumer heartbeat time	RW	NO	Uint32	ARR	-	-
	1h to 5h	Consumer heartbeat time	RW	NO	Uint32	-	Uint 32	0
1017h	-	Producer heartbeat time	RW	NO	Uint16	VAR	Uint 16	0

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
1018h	-	Device object description	RO	NO	Individually related	REC	-	-
	1h	Manufacturer ID	RO	NO	Uint32	-	Uint 32	0x3B9
	2h	Device code	RO	NO	Uint32	-	Uint 32	0xD0107
	3h	Device revision version	RO	NO	Uint32	-	Uint 32	0x00020003
1029h	-	Wrong behavior object	RW	NO	Uint8	ARR	-	-
	1h	Communication error	RW	NO	Uint8	-	Uint 8	0
1200h	-	SDO server parameter	RO	NO	SDO parameter	REC	-	-
	1h	Client to server COB-ID	RO	NO	Uint32	-	Uint 32	0x600+ Node_ID
	2h	Server to client COB-ID	RO	NO	Uint32	-	Uint 32	0x580+ Node_ID
1400h	-	RPDO1 parameter	RW	NO	PDO parameter	REC	-	-
	1h	COB-ID of RPDO1	RW	NO	Uint32	-	Uint 32	0x00000200 +Node_ID
	2h	Transmission type of RPDO1	RW	NO	Uint8	-	Uint 8	255
1401h		RPDO2 parameter	RW	NO	PDO parameter	REC	-	-
	1	COB-ID of RPDO2	RW	NO	Uint32	-	Uint 32	0x00000300 +Node_ID
	2	Transmission type of RPDO2	RW	NO	Uint8	-	Uint 8	255
1402h		RPDO3 parameter	RW	NO	PDO parameter	REC	-	-
	1h	COB-ID of RPDO3	RW	NO	Uint32	-	Uint 32	0x00000400 +Node_ID
	2h	Transmission type of RPDO3	RW	NO	Uint8	-	Uint 8	255
1403h		RPDO4 parameter	RW	NO	PDO parameter	REC	-	-
	1h	COB-ID of RPDO4	RW	-	Uint32	-	Uint 32	0x00000500 +Node_ID
	2h	Transmission type of RPDO4	RW	NO	Uint8	-	Uint 8	255
1600h		Mapping parameter of RPDO1	RW	NO	RPDO mapping parameter	REC	-	-
	1h to 8h	Mapping object of RPDO1	RW	NO	Uint32	-	Uint 32	-
1601h		Mapping parameter of RPDO2	RW	NO	RPDO mapping parameter	REC	-	-
	1h to 8h	Mapping object of RPDO2	RW	NO	Uint32	-	Uint 32	-

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
1602h		Mapping parameter of RPDO3	RW	NO	RPDO mapping parameter	REC	-	-
	1h to 8h	Mapping object of RPDO3	RW	NO	Uint32	-	Uint 32	-
1603h		Mapping parameter of RPDO4	RW	NO	RPDO mapping parameter	REC	-	-
	1h to 8h	Mapping object of RPDO4	RW	NO	Uint32	-	Uint 32	-
1800h		Communication parameter of TPDO1	RW	NO	PDO communication parameter	REC	-	-
	1h	COB-ID of TPDO1	RW	NO	Uint32	-	Uint 32	0x40000180 +Node_ID
	2h	Transmission type of TPDO1	RW	NO	Uint8	-	Uint 8	255
	3h	Inhibit time	RW	NO	Uint16	100 us	Uint 16	0
	5h	Event timer	RW	NO	Uint16	-	Uint 16	0
1801h		Communication parameter of TPDO2	RW	NO	PDO communication parameter	REC	-	-
	1h	COB-ID of TPDO2	RW	NO	Uint32	-	Uint 32	0xC0000280 +Node_ID
	2h	Transmission type of TPDO2	RW	NO	Uint8	-	Uint 8	255
	3h	Inhibit time	RW	NO	Uint16	-	Uint 16	0
	5h	Event timer	RW	NO	Uint16	-	Uint 16	0
1802h		Communication parameter of TPDO3	RW	NO	PDO communication parameter	REC	-	-
	1h	COB-ID of TPDO3	RW	NO	Uint32	-	Uint 32	0xC0000380 +Node_ID
	2h	Transmission type of TPDO3	RW	NO	Uint8	-	Uint 8	255
	3h	Inhibit time	RW	NO	Uint16	-	Uint 16	0
	5h	Event timer	RW	NO	Uint16	-	Uint 16	0
1803h		Communication parameter of TPDO4	RW	NO	PDO communication parameter	REC	-	-
	1h	COB-ID of TPDO4	RW	NO	Uint32	-	Uint 32	0xC0000480 +Node_ID
	2h	Transmission type of TPDO4	RW	NO	Uint8	-	Uint 8	255
	3h	Inhibit time	RW	NO	Uint16	-	Uint 16	0
	5h	Event timer	RW	NO	Uint16	-	Uint 16	0

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
1A00h		Mapping parameter of TPDO1	RW	NO	PDO mapping parameter	REC	-	-
	1h to 8h	Mapping object of TPDO1	RW	NO	Uint32	-	Uint 32	-
1A01h		Mapping parameter of TPDO2	RW	NO	PDO mapping parameter	REC	-	-
	1h to 8h	Mapping object of TPDO2	RW	NO	Uint32	-	Uint 32	-
1A02h		Mapping parameter of TPDO3	RW	NO	PDO mapping parameter	REC	-	-
	1h	Mapping object of TPDO3	RW	NO	Uint32	-	Uint 32	-
1A03h		Mapping parameter of TPDO4	RW	NO	PDO mapping parameter	REC	-	-
	1h to 8h	Mapping object of TPDO4	RW	NO	Uint32	-	Uint 32	-

6.3 Object Group 2000h

The object group 2000h, as an object table defined by Inovance, is associated with the corresponding device parameters. All objects in this area support PDO mapping.

2000h Servo Motor Parameters

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
2000h	1h	Motor code	RW	YES	Uint16	-	0 to 65535	-
	3h	Customized No.	RO	TPDO	Uint32	-	-	-
	5h	Encoder version	RO	TPDO	Uint16	-	-	-
	6h	Serial encoder motor model	RO	TPDO	Uint16	-	-	-

NOTE

The modification on 2000-01h is activated at next power-on. Default values of some parameters can be updated only after factory settings are restored (H02-31 = 1).

2001h Servo Drive Parameters

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
2001h	1h	MCU software version	RO	TPDO	Uint16	-	0 to 65535	-
	2h	FPGA software version	RO	TPDO	Uint16	-	0 to 65535	-
	3h	Servo drive SN	RW	YES	Uint16	-	0 to 65535	-

2002h Basic Control Parameters

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
2002h	1h	Control mode selection	RW	YES	Uint16	-	0 to 8	8
	2h	Absolute system selection	RW	YES	Uint16	-	0 to 2	0
	3h	Rotation direction	RW	YES	Uint16	-	0 to 1	0
	4h	Output pulse phase	RW	YES	Uint16	-	0 to 1	0
	6h	Stop mode at S-ON OFF	RW	YES	Uint16	-	0 to 1	0
	7h	Stop mode at No. 2 fault	RW	YES	Uint16	-	0 to 1	0
	8h	Stop mode at overtravel	RW	YES	Uint16	-	0 to 2	1
	9h	Stop mode at No.1 fault	RW	YES	Uint16	-	0	0
	0Ah	Delay from brake output ON to command received	RW	YES	Uint16	ms	0 to 500	250
	0Bh	Delay from brake output OFF to motor de-energized in static state	RW	YES	Uint16	ms	1 to 1000	150
	0Ch	Motor speed threshold at brake output OFF in rotational state	RW	YES	Uint16	RPM	0 to 3000	30
	0Dh	Delay from motor de-energized to brake output OFF in rotational state	RW	YES	Uint16	ms	1 to 1000	500
	0Fh	Warning display on the keypad	RW	YES	Uint16	-	0 to 1	0
	13h	S-ON filter time constant	RW	YES	Uint16	ms	0 to 64	0
	16h	Permissible minimum resistance of regenerative resistor	RO	TPDO	Uint16	Ω	-	-
	17h	Power of built-in regenerative resistor	RO	TPDO	Uint16	W	-	-
	18h	Resistance of built-in regenerative resistor	RO	TPDO	Uint16	Ω	-	-
	19h	Resistor heat dissipation coefficient	RW	YES	Uint16	%	10 to 100	30
	1Ah	Regenerative resistor type	RW	YES	Uint16	-	0 to 3	0
	1Bh	Power of external regenerative resistor	RW	YES	Uint16	W	1 to 65535	-
	1Ch	Resistance of external regenerative resistor	RW	YES	Uint16	Ω	1 to 1000	-
	1Fh	User password	WO	RPDO	Uint16	-	0 to 65535	0
	20h	System parameter initialization	WO	RPDO	Uint16	-	0 to 2	0
	21h	Default keypad display	RW	YES	Uint16	-	0 to 99	50
	2Ah	Factory password	WO	NO	Uint16	-	-	-

2003h Terminal Input Parameters

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
2003h	1h	DI function allocation 1 (activated upon power-on)	RW	YES	Uint16	-	0 to 0xFFFF	0
	2h	DI function allocation 2 (activated upon power-on)	RW	YES	Uint16	-	0 to 0xFFFF	0
	3h	DI1 function selection	RW	YES	Uint16	-	0 to 37	14
	4h	DI1 logic selection	RW	YES	Uint16	-	0 to 4	0
	5h	DI2 function selection	RW	YES	Uint16	-	0 to 37	15
	6h	DI2 logic selection	RW	YES	Uint16	-	0 to 4	0
	7h	DI3 function selection	RW	YES	Uint16	-	0 to 37	13
	8h	DI3 logic selection	RW	YES	Uint16	-	0 to 4	0
	9h	DI4 function selection	RW	YES	Uint16	-	0 to 37	2
	0Ah	DI4 logic selection	RW	YES	Uint16	-	0 to 4	0
	0Bh	DI5 function selection	RW	YES	Uint16	-	0 to 37	1
	0Ch	DI5 logic selection	RW	YES	Uint16	-	0 to 4	0
	11h	DI8 function selection	RW	YES	Uint16	-	0 to 37	31
	12h	DI8 logic selection	RW	YES	Uint16	-	0 to 4	0
	13h	DI9 function selection	RW	YES	Uint16	-	0 to 37	0
	14h	DI9 logic selection	RW	YES	Uint16	-	0 to 4	0
	23h	DI function allocation 3 (activated upon power-on)	RW	YES	Uint16	-	0 to 0xFFFF	0
	24h	DI function allocation 4 (activated upon power-on)	RW	YES	Uint16	-	0 to 0xFFFF	0

2004h Terminal Output Parameters

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
2004h	1h	DO1 function selection	RW	YES	Uint16	-	0 to 19	1
	2h	DO1 logic selection	RW	YES	Uint16	-	0 to 1	0
	3h	DO2 function selection	RW	YES	Uint16	-	0 to 19	5
	4h	DO2 logic selection	RW	YES	Uint16	-	0 to 1	0
	5h	DO3 function selection	RW	YES	Uint16	-	0 to 19	3
	6h	DO3 logic selection	RW	YES	Uint16	-	0 to 1	0

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
2004h	7h	DO4 function selection	RW	YES	Uint16	-	0 to 19	11
	8h	DO4 logic selection	RW	YES	Uint16	-	0 to 1	0
	9h	DO5 function selection	RW	YES	Uint16	-	0 to 19	16
	0Ah	DO5 logic selection	RW	YES	Uint16	-	0 to 1	0
	17h	DO source	RW	YES	Uint16	-	0 to 31	0

2005h Position Control Parameters

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
2005h	1h	Position reference source	RW	YES	Uint16	-	0 to 2	0
	2h	Pulse reference input terminal selection	RW	YES	Uint16	-	0 to 1	0
	3h	Position references per motor revolution	RW	YES	Uint32	p/r	0 to 1048576	0
	5h	First-order low-pass filter time constant	RW	YES	Uint16	ms	0 to 6553.5	0.0
	6h	Step amount	RW	YES	Int16	Reference unit	-9999 to +9999	50
	7h	Moving average filter time constant	RW	YES	Uint16	ms	0 to 128.0	0.0
	8h	Electronic gear ratio 1 (numerator)	RW	YES	Uint32	-	1 to 1073741824	1048576
	0Ah	Electronic gear ratio 1 (denominator)	RW	YES	Uint32	-	1 to 1073741824	1000
	0Ch	Electronic gear ratio 2 (numerator)	RW	YES	Uint32	-	1 to 1073741824	1048576
	0Eh	Electronic gear ratio 2 (denominator)	RW	YES	Uint32	-	1 to 1073741824	10000
	10h	Pulse reference form	RW	YES	Uint16	-	0 to 3	0
	11h	Clear action selection	RW	YES	Uint16	-	0 to 2	0
	12h	Number of encoder frequency-division pulses	RW	YES	Uint16	p/r	35 to 327567	2500
	14h	Speed feedforward control	RW	YES	Uint16	-	0 to 3	1
	15h	Condition for positioning completed signal output	RW	YES	Uint16	-	0 to 2	0
	16h	Positioning completed threshold	RW	YES	Uint16	Encoder unit	1 to 65535	734
	17h	Positioning proximity threshold	RW	YES	Uint16	Encoder unit	1 to 65535	65535
	18h	Interrupt positioning selection	RW	YES	Uint16	-	0 to 1	0
	19h	Displacement of interrupt positioning	RW	YES	Uint32	Reference unit	0 to 1073741824	10000
	1Bh	Constant running speed at interrupt positioning	RW	YES	Uint16	RPM	0 to 6000	200

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
2005h	1Ch	Acceleration/Deceleration time of interrupt positioning	RW	YES	Uint16	ms	0 to 1000	10
	1Eh	Interrupt positioning cancel signal	RW	YES	Uint16	-	0 to 1	1
	1Fh	Homing selection	RW	YES	Uint16	-	0 to 6	0
	20h	Homing mode	RW	YES	Uint16	-	0 to 13	0
	21h	Speed in high-speed searching for the home switch signal	RW	YES	Uint16	RPM	0 to 3000	100
	22h	Speed in low-speed searching for the home switch signal	RW	YES	Uint16	RPM	0 to 1000	10
	23h	Acceleration/Deceleration time during homing	RW	YES	Uint16	ms	0 to 1000	1000
	24h	Homing duration limit	RW	YES	Uint16	ms	0 to 65535	10000
	25h	Mechanical home offset	RW	YES	Uint32	Reference unit	-1073741824 to +1073741824	0
	27h	Servo pulse output source	RW	YES	Uint16	-	0 to 2	0
	28h	Electronic gear ratio switchover condition	RW	YES	Uint16	-	0 to 1	0
	29h	Mechanical home offset and action upon hitting the limit switch	RW	YES	Uint16	-	0 to 3	0
	2Ah	Z pulse output polarity	RW	YES	Uint16	-	0 to 1	1
	2Ch	Position pulse edge selection	RW	YES	Uint16	1	0 to 1	0
	2Fh	Position offset in absolute position linear mode (low 32 bits)	RW	YES	int32	Encoder unit	-2147483648 to +2147483647	0
	31h	Position offset in absolute position linear mode (high 32 bits)	RW	YES	int32	Encoder unit	-2147483648 to +2147483647	0
	33h	Mechanical gear ratio in absolute position rotation mode (numerator)	RW	YES	Uint16	1	1 to 65535	65535
	34h	Mechanical gear ratio in absolute position rotation mode (denominator)	RW	YES		1	1 to 65535	1
	35h	Pulses per revolution in absolute position rotation mode (low 32 bits)	RW	YES	Uint32	Encoder unit	0 to 4294967295	0
	37h	Pulses per revolution in absolute position rotation mode (high 32 bits)	RW	YES	Uint16	Encoder unit	0 to 127	0
	39h	Speed threshold of homing upon hit-and-stop	RW	YES	Uint16	RPM	0 to 1000	2
	3Bh	Torque limit of homing upon hit-and-stop	RW	YES	Uint16	%	0 to 300.0	100.0%
	3Ch	Positioning window time	RW	YES	Uint16	ms	0 to 30000	1
	3Dh	Positioning completed holding time	RW	YES	Uint16	ms	0 to 30000	1
	3Eh	Number of encoder frequency-division pulses (32 bits)	RW	YES	Uint32	P/r	0 to 262143	0

2006h Speed Control Parameters

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
2006h	1h	Source of main speed reference A	RW	YES	Uint16	-	0 to 2	0
	2h	Source of auxiliary speed reference B	RW	YES	Uint16	-	0 to 5	1
	3h	Speed reference selection	RW	YES	Uint16	-	0 to 4	0
	4h	Speed reference set through keypad	RW	YES	Int16	RPM	-6000 to +6000	200
	5h	Jog speed setpoint	RW	YES	Uint16	RPM	0 to 6000	100
	6h	Acceleration ramp time constant of speed reference	RW	YES	Uint16	ms	0 to 65535	0
	7h	Deceleration ramp time constant of speed reference	RW	YES	Uint16	ms	0 to 65535	0
	8h	Max. speed threshold	RW	YES	Uint16	RPM	0 to 6000	6000
	9h	Forward speed threshold	RW	YES	Uint16	RPM	0 to 6000	6000
	0Ah	Reverse speed threshold	RW	YES	Uint16	RPM	0 to 6000	6000
	0Ch	Torque feedforward control	RW	YES	Uint16	-	0 to 1	1
	10h	Speed threshold for zero clamp	RW	YES	Uint16	RPM	0 to 6000	10
	11h	Motor speed threshold	RW	YES	Uint16	RPM	0 to 1000	20
	12h	Threshold of speed matching signal	RW	YES	Uint16	RPM	0 to 100	10
	13h	Threshold of speed reached signal	RW	YES	Uint16	RPM	10 to 6000	1000
	14h	Threshold of zero speed signal output	RW	YES	Uint16	RPM	1 to 6000	10

2007h Torque Control Parameters

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
2007h	1h	Source of main torque reference A	RW	YES	Uint16	-	0 to 2	0
	2h	Source of auxiliary torque reference B	RW	YES	Uint16	-	0 to 2	1
	3h	Torque reference selection	RW	YES	Uint16	-	0 to 3	0
	4h	Torque reference set through keypad	RW	YES	Int16	%	-300.0 to +300.0	0
	6h	Torque reference filter time constant	RW	YES	Uint16	ms	0 to 30.00	0.79

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
2007h	7h	2nd torque reference filter time constant	RW	YES	Uint16	ms	0 to 30.00	0.79
	8h	Torque limit source	RW	YES	Uint16	-	0 to 3	0
	9h	T-LMT selection	RW	YES	Uint16	-	1 to 2	2
	0Ah	Forward internal torque limit	RW	YES	Uint16	%	0.0 to 300.0	300.0
	0Bh	Reverse internal torque limit	RW	YES	Uint16	%	0.0 to 300.0	300.0
	0Ch	Forward external torque limit	RW	YES	Uint16	%	0.0 to 300.0	300.0
	0Dh	Reverse external torque limit	RW	YES	Uint16	%	0.0 to 300.0	300.0
	10h	Emergency stop torque	RW	YES	Uint16	%	0.0 to 300.0	100.0
	12h	Speed limit source	RW	YES	Uint16	-	0 to 2	0
	13h	V-LMT selection	RW	YES	Uint16	-	1 to 2	1
	14h	Torque control forward speed limit/Torque control speed limit 1	RW	YES	Uint16	RPM	0 to 6000	3000
	15h	Torque control reverse speed limit/Torque control speed limit 2	RW	YES	Uint16	RPM	0 to 6000	3000
	16h	Reference value for torque reached	RW	YES	Uint16	%	0.0 to 300.0	0.0
	17h	Valid value for torque reached	RW	YES	Uint16	%	0.0 to 300.0	20.0
	18h	Invalid value for torque reached	RW	YES	Uint16	%	0.0 to 300.0	10.0
	29h	Speed limit window in torque control mode	RW	YES	Uint16	ms	0.5 to 30.0	1.0

2008h Gain Control Parameters

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
2008h	1h	Speed loop gain	RW	YES	Uint16	Hz	0.1 to 2000.0	25.0
	2h	Speed loop integral time constant	RW	YES	Uint16	ms	0.15 to 512.00	31.83
	3h	Position loop gain	RW	YES	Uint16	Hz	0.0 to 2000.0	40.0
	4h	2nd speed loop gain	RW	YES	Uint16	Hz	0.1 to 2000.0	40.0
	5h	Second speed loop integral time constant	RW	YES	Uint16	ms	0.15 to 512.00	40.00
	6h	2nd position loop gain	RW	YES	Uint16	Hz	0.0 to 2000.0	64.0

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
2008h	9h	2nd gain mode	RW	YES	Uint16	-	0 to 1	1
	0Ah	Gain switchover condition	RW	YES	Uint16	-	0 to 10	0
	0Bh	Gain switchover delay	RW	YES	Uint16	ms	0.0 to 1000.0	5.0
	0Ch	Gain switchover level	RW	YES	Uint16	Based on switchover conditions	0 to 20000	50
	0Dh	Gain switchover hysteresis	RW	YES	Uint16	Based on switchover conditions	0 to 20000	30
	0Eh	Position gain switchover time	RW	YES	Uint16	ms	0.0 to 1000.0	3.0
	10h	Load moment of inertia ratio	RW	YES	Uint16	Times	0.00 to 120.00	1.00
	13h	Speed feedback filter time constant	RW	YES	Uint16	ms	0.00 to 64.00	0.50
	14h	Speed feedforward gain	RW	YES	Uint16	%	0.0 to 100.0	0.0
	15h	Torque feedforward filter time constant	RW	YES	Uint16	ms	0.00 to 64.00	0.50
	16h	Torque feedforward gain	RW	YES	Uint16	%	0.0 to 200.0	0.0
	17h	Speed feedback filter selection	RW	YES	Uint16	-	0 to 4	0
	18h	Cutoff frequency of speed feedback low-pass filter	RW	YES	Uint16	Hz	100 to 4000	4000
	19h	PDFF control coefficient	RW	YES	Uint16	-	0.0 to 100.0	100.0

2009h Gain Auto-tuning Parameters

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
2009h	1h	Gain auto-tuning mode	RW	YES	Uint16	-	0 to 2	0
	2h	Stiffness level	RW	YES	Uint16	-	0 to 31	12
	3h	Adaptive notch mode	RW	YES	Uint16	-	0 to 4	0
	4h	Online inertia auto-tuning mode	RW	YES	Uint16	-	0 to 3	0
	5h	Low-frequency resonance suppression mode	RW	YES	Uint16	-	0 to 1	0
	6h	Offline inertia auto-tuning mode	RW	YES	Uint16	-	0 to 1	0
	7h	Maximum speed of inertia auto-tuning	RW	YES	Uint16	RPM	100 to 1000	500

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
2009h	8h	Time constant for accelerating to the maximum speed during inertia auto-tuning	RW	YES	Uint16	ms	20 to 800	125
	9h	Inertia auto-tuning interval	RW	YES	Uint16	ms	50–10000	800
	0Ah	Number of motor revolutions per inertia auto-tuning	RO	TPDO	Uint16	r	0.00 to 2.00	-
	0Dh	Frequency of the 1st notch	RW	YES	Uint16	Hz	50 to 4000	4000
	0Eh	Width level of the 1st notch	RW	YES	Uint16	-	0 to 20	2
	0Fh	Depth level of the 1st notch	RW	YES	Uint16	-	0 to 99	0
	10h	Frequency of the 2nd notch	RW	YES	Uint16	Hz	50 to 4000	4000
	11h	Width level of the 2nd notch	RW	YES	Uint16	-	0 to 20	2
	12h	Depth level of the 2nd notch	RW	YES	Uint16	-	0 to 99	0
	13h	Frequency of the 3rd notch	RW	YES	Uint16	Hz	50 to 4000	4000
	14h	Width level of the 3rd notch	RW	YES	Uint16	-	0 to 20	2
	15h	Depth level of the 3rd notch	RW	YES	Uint16	-	0 to 99	0
	16h	Frequency of the 4th notch	RW	YES	Uint16	Hz	50 to 4000	0
	17h	Width level of the 4th notch	RW	YES	Uint16	-	0 to 20	0
	18h	Depth level of the 4th notch	RW	YES	Uint16	-	0 to 99	0
	19h	Auto-tuned resonance frequency	RO	-	Uint16	Hz	0 to 2	0
	1Fh	Torque disturbance compensation gain	RW	YES	Int16	%	0.0 to 100.0	0.0
	20h	Time constant of torque disturbance observer filter	RW	YES	Uint16	ms	0.00 to 25.00	0.50
	27h	Frequency of low-frequency resonance	RW	YES	Uint16	Hz	1.0 to 100.0	100.0
	28h	Low-frequency resonance frequency filter setting	RW	YES	Uint16	-	0 to 10	2

200Ah Fault and Protection Parameters

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
200Ah	1h	Power input phase loss protection	RW	YES	Uint16	-	0 to 2	0
	2h	Software limit	RW	YES	Uint16	-	0 to 2	0
	4h	Power-off memory	RW	YES	Uint16	-	0 to 1	0
	5h	Motor overload protection gain	RW	YES	Uint16	%	50 to 300	100
	9h	Overspeed threshold	RW	YES	Uint16	RPM	0 to 10000	0
	0Ah	Max. position pulse frequency	RW	YES	Uint16	kHz	100 to 4000	4000
	0Bh	Excessive position deviation threshold	RW	YES	Uint32	Encoder unit	1 to 1073741824	3145728
	0Dh	Runaway protection	RW	YES	Uint16	-	0 to 1	1
	11h	Threshold of low-frequency resonance position deviation	RW	YES	Uint16	Encoder unit	1 to 1000	5
	12h	Position setting unit	RW	YES	Uint16	-	0 to 1	0
	14h	DI8 filter time constant	RW	YES	Uint16	25 ns	0 to 255	80
	15h	DI9 filter time constant	RW	YES	Uint16	25 ns	0 to 255	80
	19h	Filter time constant of low-speed pulse input pin	RW	YES	Uint6	25 ns	0 to 255	30
	1Ah	Filter time constant of displayed speed feedback	RW	YES	Uint16	ms	0 to 5000	50
	1Bh	Motor overload warning/ fault selection	RW	YES	Uint16	-	0 to 1	0
	1Ch	Speed DO filter time constant	RW	YES	Uint16	ms	0 to 5000	10
	1Dh	Quadrature encoder filter time constant	RW	YES	Uint16	25 ns	0 to 255	5
	1Eh	Linear encoder filter time	RW	YES	Uint16	25 ns	0 to 255	15
	1Fh	High-speed pulse input pin filter time constant	RW	YES	Uint16	25 ns	0 to 255	3
	21h	Time window of locked rotor over-temperature protection	RW	YES	Uint16	ms	10 to 65535	200
	22h	Locked rotor over-temperature protection	RW	YES	Uint16	-	0 to 1	1
	25h	Encoder multi-turn overflow fault	RW	YES	Uint16	-	0 to 1	0
	30h	Brake protection detection	RW	YES	Uint16	-	0 to 1	1
	31h	Gravity load detection value	RW	YES	Uint16	%	0 to 300.0	30.0

200Bh Display Parameters

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
200Bh	1h	Actual motor speed	RO	TPDO	Int16	RPM	-	-
	2h	Speed reference	RO	TPDO	Int16	RPM	-	-
	3h	Internal torque reference (relative to rated torque)	RO	TPDO	Int16	%	-	-
	4h	Monitored DI status	RO	TPDO	Uint16	-	-	-
	6h	Monitored DO status	RO	TPDO	Uint16	-	-	-
	8h	Absolute position counter	RO	TPDO	Int32	Reference unit	-	-
	0Ah	Mechanical angle (starting from pulses of the home)	RO	TPDO	Uint16	Encoder unit	-	-
	0Bh	Electrical angle	RO	TPDO	Uint16	°	-	-
	0Ch	Speed corresponding to the input position reference	RO	TPDO	Int16	RPM	-	-
	0Dh	Average load ratio	RO	TPDO	Uint16	%	-	-
	0Eh	Input position reference counter	RO	TPDO	Int32	Reference unit	-	-
	10h	Encoder position deviation counter	RO	TPDO	Int32	Encoder unit	-	-
	12h	Feedback pulse counter	RO	TPDO	Int32	Encoder unit	-	-
	14h	Total power-on time	RO	TPDO	Uint32	s	-	-
	19h	RMS value of phase current	RO	TPDO	Uint16	A	-	-
	1Bh	Bus voltage	RO	TPDO	Uint16	V	-	-
	1Ch	Inverter module heatsink temperature	RO	TPDO	Uint16	℃	-	-
	22h	Fault log	RW	YES	Uint16	-	0 to 9	0
	23h	Fault code of the selected fault	RO	TPDO	Uint16	-	-	-
	24h	Time stamp upon occurrence of the selected fault	RO	TPDO	Uint32	s	-	-
	26h	Motor speed upon occurrence of the selected fault	RO	TPDO	Int16	RPM	-	-
	27h	Motor phase U current upon occurrence of the selected fault	RO	TPDO	Int16	A	-	-
	28h	Motor phase V current upon occurrence of the selected fault	RO	TPDO	Int16	A	-	-
	29h	Bus voltage upon occurrence of the selected fault	RO	TPDO	Uint16	V	-	-
	2Ah	Input terminal status upon occurrence of the selected fault	RO	TPDO	Uint16	-	-	-
	2Bh	Output terminal status upon occurrence of the selected fault	RO	TPDO	Uint16	-	-	-
	36h	Position deviation counter	RO	TPDO	Int32	Reference unit	-	-
	38h	Actual motor speed	RO	TPDO	Int32	RPM	-	-

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
200Bh	3Bh	Mechanical absolute position (low 32 bits)	RO	TPDO	int32	Encoder unit	-	0
	3Dh	Mechanical absolute position (high 32 bits)	RO	TPDO	int32	Encoder unit	-	0
	41h	Real-time position reference counter	RO	TPDO	int32	Reference unit	-	-
	47h	Number of absolute encoder revolutions	RO	TPDO	Uint16	r	-	0
	48h	Position of the absolute encoder within one turn	RO	TPDO	Uint32	Encoder unit	-	0
	4Eh	Absolute position of absolute encoder (low 32 bits)	RO	TPDO	int32	Encoder unit	-	0
	50h	Absolute position of absolute encoder (high 32 bits)	RO	TPDO	int32	Encoder unit	-	0
	52h	Single-turn position of the rotating load (low 32 bits)	RO	TPDO	Uint32	Encoder unit	-	0
	54h	Single-turn position of the rotating load (high 32 bits)	RO	TPDO	Uint32	Encoder unit	-	0
	56h	Single-turn position of the rotating load	RO	TPDO	Uint32	Reference unit	-	0

200Ch Communication Parameters

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
200Ch	1h	Servo drive axis address	RW	YES	Uint16	-	1 to 247	1
	3h	Serial baud rate	RW	YES	Uint16	-	0 to 5	5
	4h	Modbus data format	RW	YES	Uint16	-	0 to 3	0
	5h	Excessive CANopen synchronization error threshold	RW	YES	Uint16	-	0 to 5	0
	9h	CAN communication rate	RW	YES	Uint16	-	0 to 7	5
	0Ah	Communication VDI	RW	YES	Uint16	-	0 to 1	0
	0Bh	VDI default value upon power-on	RW	YES	Uint16	-	0 to 65535	0
	0Ch	Communication VDO	RW	YES	Uint16	-	0 to 1	0
	0Dh	Default level when VDO function set to 0	RW	YES	Uint16	-	0 to 65535	0
	0Eh	Update parameters written through Modbus communication to EEPROM	RW	YES	Uint16	-	0 to 1	1
	0Fh	Modbus error code	RO	TPDO	Uint16	1	0 to 65535	-

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
200Ch	1Ah	Modbus response delay	RW	YES	Uint16	1	0 to 1	1
	1Bh	Modbus communication data sequence	RW	YES	Uint16	1	0 to 1	1
	1Fh	Modbus error frame format	RW	YES	Uint16	1	0 to 1	1

200Dh Auxiliary Parameters

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
200Dh	1h	Software reset	RW	YES	Uint16	-	0 to 1	0
	2h	Fault reset	RW	YES	Uint16	-	0 to 1	0
	6h	Emergency stop	RW	YES	Uint16	-	0 to 1	0
	12h	Forced DI/DO selection	RW	YES	Uint16	-	0 to 3	0
	13h	Forced DI setting	RW	YES	Uint16	-	0 to 0x01FF	0x01FF
	14h	Forced DO setting	RW	YES	Uint16	-	0 to 0x001F	0

200Fh Fully Closed-Loop Parameters

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
200Fh	1h	Encoder feedback mode	RW	YES	Uint16	-	0 to 2	0
	2h	Usage of external encoder	RW	YES	Uint16	-	0 to 1	0
	5h	Number of external encoder pulses per motor revolution	RW	YES	Uint32	External encoder unit	0 to 1073741824	10000
	9h	Excessive position deviation threshold in fully closed-loop mode	RW	YES	Uint32	External encoder unit	0 to 1073741824	10000
	0Bh	Clear position deviation in fully closed-loop mode	RW	YES	Uint16	r	0 to 100	0
	0Eh	Filter time constant of hybrid vibration suppression	RW	YES	Uint16	ms	0 to 6553.5	0
	11h	Position deviation counter in fully closed-loop mode	RO	TPDO	Uint32	External encoder unit	-1073741824 to +1073741824	0

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
200Fh	13h	Internal encoder feedback value	RO	TPDO	Uint32	Internal encoder unit	-1073741824 to +1073741824	0
	15h	External encoder feedback value	RO	TPDO	Uint32	External encoder unit	-1073741824 to +1073741824	0

2011h Multi-Position Parameters

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
2011h	1h	Multi-position running mode	RW	YES	Uint16	1	0 to 3	1
	2h	Number of displacements	RW	YES	Uint16	1	1 to 16	1
	3h	Start position upon restart after pause	RW	YES	Uint16	1	0 to 1	0
	4h	Waiting time unit	RW	YES	Uint16	1	0 to 1	0
	5h	Displacement reference type	RW	YES	Uint16	1	0 to 1	0
	6h	Start position in cyclic running	RW	YES	Uint16	1	0 to 16	0
	0Dh	1st displacement	RW	YES	Uint32	1 reference unit	-1073741824 to +1073741824	10000
	0Fh	Constant running speed of 1st displacement	RW	YES	Uint16	1 RPM	1 to 9000	200
	10h	Acceleration/Deceleration time of 1st displacement	RW	YES	Uint16	1 ms (1s)	0 to 65535	10
	11h	Interval upon completion of the 1st displacement	RW	YES	Uint16	1 ms (1s)	0 to 10000	10
	12h	2nd displacement	RW	YES	Uint32	1 reference unit	-1073741824 to +1073741824	10000
	14h	Constant running speed of 2nd displacement	RW	YES	Uint16	1 RPM	1 to 9000	200
	15h	Acceleration/Deceleration time of 2nd displacement	RW	YES	Uint16	1 ms (1s)	0 to 65535	10
	16h	Interval upon completion of the 2nd displacement	RW	YES	Uint16	1 ms (1s)	0 to 10000	10
	17h	3rd displacement	RW	YES	Uint32	1 reference unit	-1073741824 to +1073741824	10000
	19h	Constant running speed of 3rd displacement	RW	YES	Uint16	1 RPM	1 to 9000	200
	1Ah	Acceleration/Deceleration time of 3rd displacement	RW	YES	Uint16	1 ms (1s)	0 to 65535	10

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
2011h	1Bh	Interval upon completion of the 3rd displacement	RW	YES	Uint16	1 ms (1s)	0 to 10000	10
	1Ch	4th displacement	RW	YES	Uint32	1 reference unit	-1073741824 to +1073741824	10000
	1Eh	Constant running speed of 4th displacement	RW	YES	Uint16	1 RPM	1 to 9000	200
	1Fh	Acceleration/Deceleration time of 4th displacement	RW	YES	Uint16	1 ms (1s)	0 to 65535	10
	20h	Interval upon completion of the 4th displacement	RW	YES	Uint16	1 ms (1s)	0 to 10000	10
	21h	5th position	RW	YES	Uint32	1 reference unit	-1073741824 to +1073741824	10000
	23h	Constant running speed of 5th displacement	RW	YES	Uint16	1 RPM	1 to 9000	200
	24h	Acceleration/Deceleration time of 5th displacement	RW	YES	Uint16	1 ms (1s)	0 to 65535	10
	25h	Interval upon completion of the 5th displacement	RW	YES	Uint16	1 ms (1s)	0 to 10000	10
	26h	6th displacement	RW	YES	Uint32	1 reference unit	-1073741824 to +1073741824	10000
	28h	Constant running speed of 6th displacement	RW	YES	Uint16	1 RPM	1 to 9000	200
	29h	Acceleration/Deceleration time of 6th displacement	RW	YES	Uint16	1 ms (1s)	0 to 65535	10
	2Ah	Interval upon completion of the 6th displacement	RW	YES	Uint16	1 ms (1s)	0 to 10000	10
	2Bh	7th displacement	RW	YES	Uint32	1 reference unit	-1073741824 to +1073741824	10000
	2Dh	Constant running speed of 7th displacement	RW	YES	Uint16	1 RPM	1 to 9000	200
	2Eh	Acceleration/Deceleration time of 7th displacement	RW	YES	Uint16	1 ms (1s)	0 to 65535	10
	2Fh	Interval upon completion of the 7th displacement	RW	YES	Uint16	1 ms (1s)	0 to 10000	10
	30h	8th displacement	RW	YES	Uint32	1 reference unit	-1073741824 to +1073741824	10000
	32h	Constant running speed of 8th displacement	RW	YES	Uint16	1 RPM	1 to 9000	200
	33h	Acceleration/Deceleration time of 8th displacement	RW	YES	Uint16	1 ms (1s)	0 to 65535	10
	34h	Interval upon completion of the 8th displacement	RW	YES	Uint16	1 ms (1s)	0 to 10000	10

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
2011h	35h	9th displacement	RW	YES	Uint32	1 reference unit	-1073741824 to +1073741824	10000
	37h	Constant running speed of 9th displacement	RW	YES	Uint16	1 RPM	1 to 9000	200
	38h	Acceleration/Deceleration time of 9th displacement	RW	YES	Uint16	1 ms (1s)	0 to 65535	10
	39h	Interval upon completion of the 9th displacement	RW	YES	Uint16	1 ms (1s)	0 to 10000	10
	3Ah	10th displacement	RW	YES	Uint32	1 reference unit	-1073741824 to +1073741824	10000
	3Ch	Constant running speed of 10th displacement	RW	YES	Uint16	1 RPM	1 to 9000	200
	3Dh	Acceleration/Deceleration time of 10th displacement	RW	YES	Uint16	1 ms (1s)	0 to 65535	10
	3Eh	Interval upon completion of the 10th displacement	RW	YES	Uint16	1 ms (1s)	0 to 10000	10
	3Fh	11th displacement	RW	YES	Uint32	1 reference unit	-1073741824 to +1073741824	10000
	41h	Constant running speed of 11th displacement	RW	YES	Uint16	1 RPM	1 to 9000	200
	42h	Acceleration/Deceleration time of 11th displacement	RW	YES	Uint16	1 ms (1s)	0 to 65535	10
	43h	Interval upon completion of the 11th displacement	RW	YES	Uint16	1 ms (1s)	0 to 10000	10
	44h	12th displacement	RW	YES	Uint32	1 reference unit	-1073741824 to +1073741824	10000
	46h	Constant running speed of 12th displacement	RW	YES	Uint16	1 RPM	1 to 9000	200
	47h	Acceleration/Deceleration time of 12th displacement	RW	YES	Uint16	1 ms (1s)	0 to 65535	10
	48h	Interval upon completion of the 12th displacement	RW	YES	Uint16	1 ms (1s)	0 to 10000	10
	49h	13th displacement	RW	YES	Uint32	1 reference unit	-1073741824 to +1073741824	10000
	4Bh	Constant running speed of 13th displacement	RW	YES	Uint16	1 RPM	1 to 9000	200
	4Ch	Acceleration/Deceleration time of 13th displacement	RW	YES	Uint16	1 ms (1s)	0 to 65535	10
	4Dh	Interval upon completion of the 13th displacement	RW	YES	Uint16	1 ms (1s)	0 to 10000	10
	4Eh	14th displacement	RW	YES	Uint32	1 reference unit	-1073741824 to +1073741824	10000

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
2011h	50h	Constant running speed of 14th displacement	RW	YES	Uint16	1 RPM	1 to 9000	200
	51h	Acceleration/Deceleration time of 14th displacement	RW	YES	Uint16	1 ms (1s)	0 to 65535	10
	52h	Interval upon completion of the 14th displacement	RW	YES	Uint16	1 ms (1s)	0 to 10000	10
	53h	15th displacement	RW	YES	Uint32	1 reference unit	-1073741824 to +1073741824	10000
	55h	Constant running speed of 15th displacement	RW	YES	Uint16	1 RPM	1 to 9000	200
	56h	Acceleration/Deceleration time of 15th displacement	RW	YES	Uint16	1 ms (1s)	0 to 65535	10
	57h	Interval upon completion of the 15th displacement	RW	YES	Uint16	1 ms (1s)	0 to 10000	10
	58h	16th displacement	RW	YES	Uint32	1 reference unit	-1073741824 to +1073741824	10000
	5Ah	Constant running speed of 16th displacement	RW	YES	Uint16	1 RPM	1 to 9000	200
	5Bh	Acceleration/Deceleration time of 16th displacement	RW	YES	Uint16	1 ms (1s)	0 to 65535	10
	5Ch	Interval upon completion of the 16th displacement	RW	YES	Uint16	1 ms (1s)	0 to 10000	10

2012h Multi-Speed Parameters

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
2012h	1h	Multi-speed reference running mode	RW	YES	Uint16	1	0 to 2	1
	2h	Number of speeds	RW	YES	Uint16	1	1 to 16	16
	3h	Running time unit	RW	YES	Uint16	1	0 to 1	0
	4h	Acceleration time 1	RW	YES	Uint16	1 ms	0 to 65535	10
	5h	Deceleration time 1	RW	YES	Uint16	1 ms	0 to 65535	10
	6h	Acceleration time 2	RW	YES	Uint16	1 ms	0 to 65535	50
	7h	Deceleration time 2	RW	YES	Uint16	1 ms	0 to 65535	50
	8h	Acceleration time 3	RW	YES	Uint16	1 ms	0 to 65535	100
	9h	Deceleration time 3	RW	YES	Uint16	1 ms	0 to 65535	100
	Ah	Acceleration time 4	RW	YES	Uint16	1 ms	0 to 65535	150
	Bh	Deceleration time 4	RW	YES	Uint16	1 ms	0 to 65535	150
	15h	1st speed	RW	YES	Uint16	1 RPM	-9000 to +9000	0

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
2012h	16h	Running time of the 1st speed	RW	YES	Uint16	0.1s (m)	0 to 65535	50
	17h	Acceleration/ Deceleration time of 1st speed	RW	YES	Uint16	1	0 to 4	0
	18h	2nd speed	RW	YES	Uint16	1 RPM	-9000 to +9000	100
	19h	Running time of the 2nd speed	RW	YES	Uint16	0.1s (m)	0 to 65535	50
	1Ah	Acceleration/ Deceleration time of the 2nd speed	RW	YES	Uint16	1	0 to 4	0
	1Bh	3rd speed	RW	YES	Uint16	1 RPM	-9000 to +9000	300
	1Ch	Running time of the 3rd speed	RW	YES	Uint16	0.1s (m)	0 to 65535	50
	1Dh	Acceleration/ Deceleration time of the 3rd speed	RW	YES	Uint16	1	0 to 4	0
	1Eh	4th speed	RW	YES	Uint16	1 RPM	-9000 to +9000	500
	1Fh	Running time of the 4th speed	RW	YES	Uint16	0.1s (m)	0 to 65535	50
	20h	Acceleration/ Deceleration time of the 4th speed	RW	YES	Uint16	1	0 to 4	0
	21h	5th speed	RW	YES	Uint16	1 RPM	-9000 to +9000	700
	22h	Running time of the 5th speed	RW	YES	Uint16	0.1s (m)	0 to 65535	50
	23h	Acceleration/ Deceleration time of the 5th speed	RW	YES	Uint16	1	0 to 4	0
	24h	6th speed	RW	YES	Uint16	1 RPM	-9000 to +9000	900
	25h	Running time of the 6th speed	RW	YES	Uint16	0.1s (m)	0 to 65535	50
	26h	Acceleration/ Deceleration time of the 6th speed	RW	YES	Uint16	1	0 to 4	0
	27h	7th speed	RW	YES	Uint16	1 RPM	-9000 to +9000	600
	28h	Running time of the 7th speed	RW	YES	Uint16	0.1s (m)	0 to 65535	50
	29h	Acceleration/ Deceleration time of the 7th speed	RW	YES	Uint16	1	0 to 4	0
	2Ah	8th speed	RW	YES	Uint16	1 RPM	-9000 to +9000	300
	2Bh	Running time of the 8th speed	RW	YES	Uint16	0.1s (m)	0 to 65535	50

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
2012h	2Ch	Acceleration/ Deceleration time of the 8th speed	RW	YES	Uint16	1	0 to 4	0
	2Dh	9th speed	RW	YES	Uint16	1 RPM	-9000 to +9000	100
	2Eh	Running time of the 9th speed	RW	YES	Uint16	0.1s (m)	0 to 65535	50
	2Fh	Acceleration/ Deceleration time of the 9th speed	RW	YES	Uint16	1	0 to 4	0
	30h	10th speed	RW	YES	Uint16	1 RPM	-9000 to +9000	-100
	31h	Running time of the 10th speed	RW	YES	Uint16	0.1s (m)	0 to 65535	50
	32h	Acceleration/ Deceleration time of the 10th speed	RW	YES	Uint16	1	0 to 4	0
	33h	11th speed	RW	YES	Uint16	1 RPM	-9000 to +9000	-300
	34h	Running time of the 11th speed	RW	YES	Uint16	0.1s (m)	0 to 65535	50
	35h	Acceleration/ Deceleration time of the 11th speed	RW	YES	Uint16	1	0 to 4	0
	36h	12th speed	RW	YES	Uint16	1 RPM	-9000 to +9000	-500
	37h	Running time of the 12th speed	RW	YES	Uint16	0.1s (m)	0 to 65535	50
	38h	Acceleration/ Deceleration time of the 12th speed	RW	YES	Uint16	1	0 to 4	0
	39h	13th speed	RW	YES	Uint16	1 RPM	-9000 to +9000	-700
	3Ah	Running time of the 13th speed	RW	YES	Uint16	0.1s (m)	0 to 65535	50
	3Bh	Acceleration/ Deceleration time of the 13th speed	RW	YES	Uint16	1	0 to 4	0
	3Ch	14th speed	RW	YES	Uint16	1 RPM	-9000 to +9000	-900
	3Dh	Running time of the 14th speed	RW	YES	Uint16	0.1s (m)	0 to 65535	50
	3Eh	Acceleration/ Deceleration time of the 14th speed	RW	YES	Uint16	1	0 to 4	0
	3Fh	15th speed	RW	YES	Uint16	1 RPM	-9000 to +9000	-600
	40h	Running time of the 15th speed	RW	YES	Uint16	0.1s (m)	0 to 65535	50

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
2012h	41h	Acceleration/ Deceleration time of the 15th speed	RW	YES	Uint16	1	0 to 4	0
	42h	16th speed	RW	YES	Uint16	1 RPM	-9000 to +9000	-300
	43h	Running time of the 16th speed	RW	YES	Uint16	0.1s (m)	0 to 65535	50

2017h VDI/VDO function

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
2017h	1h	VDI1 function selection	RW	YES	Uint16	-	0 to 37	0
	2h	VDI1 logic selection	RW	YES	Uint16	-	0 to 1	0
	3h	VDI2 function selection	RW	YES	Uint16	-	0 to 37	0
	4h	VDI2 logic selection	RW	YES	Uint16	-	0 to 1	0
	5h	VDI3 function selection	RW	YES	Uint16	-	0 to 37	0
	6h	VDI3 logic selection	RW	YES	Uint16	-	0 to 1	0
	7h	VDI4 function selection	RW	YES	Uint16	-	0 to 37	0
	8h	VDI4 logic selection	RW	YES	Uint16	-	0 to 1	0
	9h	VDI5 function selection	RW	YES	Uint16	-	0 to 37	0
	0Ah	VDI5 logic selection	RW	YES	Uint16	-	0 to 1	0
	0Bh	VDI6 function selection	RW	YES	Uint16	-	0 to 37	0
	0Ch	VDI6 logic selection	RW	YES	Uint16	-	0 to 1	0
	0Dh	VDI7 function selection	RW	YES	Uint16	-	0 to 37	0
	0Eh	VDI7 logic selection	RW	YES	Uint16	-	0 to 1	0
	0Fh	VDI8 function selection	RW	YES	Uint16	-	0 to 37	0
	10h	VDI8 logic selection	RW	YES	Uint16	-	0 to 1	0
	11h	VDI9 function selection	RW	YES	Uint16	-	0 to 37	0
	12h	VDI9 logic selection	RW	YES	Uint16	-	0 to 1	0
	13h	VDI10 function selection	RW	YES	Uint16	-	0 to 37	0
	14h	VDI10 logic selection	RW	YES	Uint16	-	0 to 1	0
	15h	VDI11 function selection	RW	YES	Uint16	-	0 to 37	0
	16h	VDI11 logic selection	RW	YES	Uint16	-	0 to 1	0
	17h	VDI12 function selection	RW	YES	Uint16	-	0 to 37	0
	18h	VDI12 logic selection	RW	YES	Uint16	-	0 to 1	0
	19h	VDI13 function selection	RW	YES	Uint16	-	0 to 37	0
	1Ah	VDI13 logic selection	RW	YES	Uint16	-	0 to 1	0

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
2017h	1Bh	VDI14 function selection	RW	YES	Uint16	-	0 to 37	0
	1Ch	VDI14 logic selection	RW	YES	Uint16	-	0 to 1	0
	1Dh	VDI15 function selection	RW	YES	Uint16	-	0 to 37	0
	1Eh	VDI15 logic selection	RW	YES	Uint16	-	0 to 1	0
	1Fh	VDI16 function selection	RW	YES	Uint16	-	0 to 37	0
	20h	VDI16 logic selection	RW	YES	Uint16	-	0 to 1	0
	21h	VDO virtual level	RO	TPDO	Uint16	-	-	-
	22h	VDO1 function selection	RW	YES	Uint16	-	0 to 19	0
	23h	VDO1 logic selection	RW	YES	Uint16	-	0 to 1	0
	24h	VDO2 function selection	RW	YES	Uint16	-	0 to 19	0
	25h	VDO2 logic selection	RW	YES	Uint16	-	0 to 1	0
	26h	VDO3 function selection	RW	YES	Uint16	-	0 to 19	0
	27h	VDO3 logic selection	RW	YES	Uint16	-	0 to 1	0
	28h	VDO4 function selection	RW	YES	Uint16	-	0 to 19	0
	29h	VDO4 logic selection	RW	YES	Uint16	-	0 to 1	0
	2Ah	VDO5 function selection	RW	YES	Uint16	-	0 to 19	0
	2Bh	VDO5 logic selection	RW	YES	Uint16	-	0 to 1	0
	2Ch	VDO6 function selection	RW	YES	Uint16	-	0 to 19	0
	2Dh	VDO6 logic selection	RW	YES	Uint16	-	0 to 1	0
	2Eh	VDO7 function selection	RW	YES	Uint16	-	0 to 19	0
	2Fh	VDO7 logic selection	RW	YES	Uint16	-	0 to 1	0
	30h	VDO8 function selection	RW	YES	Uint16	-	0 to 19	0
	31h	VDO8 terminal logic selection	RW	YES	Uint16	-	0 to 1	0
	32h	VDO9 function selection	RW	YES	Uint16	-	0 to 19	0
	33h	VDO9 logic selection	RW	YES	Uint16	-	0 to 1	0
	34h	VDO10 function selection	RW	YES	Uint16	-	0 to 19	0
	35h	VDO10 logic selection	RW	YES	Uint16	-	0 to 1	0
	36h	VDO11 function selection	RW	YES	Uint16	-	0 to 19	0
	37h	VDO11 logic selection	RW	YES	Uint16	-	0 to 1	0
	38h	VDO12 function selection	RW	YES	Uint16	-	0 to 19	0
	39h	VDO12 logic selection	RW	YES	Uint16	-	0 to 1	0
	3Ah	VDO13 function selection	RW	YES	Uint16	-	0 to 19	0

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
2017h	3Bh	VDO13 logic selection	RW	YES	Uint16	-	0 to 1	0
	3Ch	VDO14 function selection	RW	YES	Uint16	-	0 to 19	0
	3Dh	VDO14 logic selection	RW	YES	Uint16	-	0 to 1	0
	3Eh	VDO15 function selection	RW	YES	Uint16	-	0 to 19	0
	3Fh	VDO15 logic selection	RW	YES	Uint16	-	0 to 1	0
	40h	VDO16 function selection	RW	YES	Uint16	-	0 to 19	0
	41h	VDO16 logic selection	RW	YES	Uint16	-	0 to 1	0

202Dh CANopen Communication Parameters 1

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
202Dh	1h	Synchronization message COB-ID (0x1005h)	No	No	Uint32	-	128 to 1073741824	128 (0x80)
	3h	Synchronization cycle (0x1006h)	No	No	Uint32	-	0 to 2147483647	0
	5h	Node guarding time (0x100Ch)	No	No	Uint16	-	0 to 65535	0
	6h	Life factor (0x100Dh)	No	No	Uint8	-	0 to 255	0
	7h	Emergency message COB-ID (0x1014h)	No	No	Uint32	-	0 to 2147483647	128 (0x80)
	9h	Consumer heartbeat time 1 (0x1016-01h)	No	No	Uint32	-	0 to 2147483647	0
	0Bh	Consumer heartbeat time 2 (0x1016-02h)	No	No	Uint32	-	0 to 2147483647	0
	0Dh	Consumer heartbeat time 3 (0x1016-03h)	No	No	Uint32	-	0 to 2147483647	0
	0Fh	Consumer heartbeat time 4 (0x1016-04h)	No	No	Uint32	-	0 to 2147483647	0
	11h	Consumer heartbeat time 5 (0x1016-05h)	No	No	Uint32	-	0 to 2147483647	0
	13h	Producer heartbeat time (0x1017h)	No	No	Uint16	-	0 to 65535	0
	14h	Error behavior object - Communication error (0x1029-01h)	No	No	Uint8	-	0 to 255	0
	15h	COB-ID (0x1400-01h) of RPDO1	No	No	Uint32	-	-2147483647 to +2147483647	512 (0x200)
	17h	Transmission type of RPDO1 (0x1400-02h)	No	No	Uint8	-	0 to 255	255

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
202Dh	18h	COB-ID (0x1401-01h) of RPDO2	No	No	Uint32	-	-2147483647 to +2147483647	0
	1Ah	Transmission type of RPDO2 (0x1401-02h)	No	No	Uint8	-	0 to 255	255
	1Bh	COB-ID (0x1402-01h) of RPDO3	No	No	Uint32	-	-2147483647 to +2147483647	0
	1Dh	Transmission type of RPDO3 (0x1402-02h)	No	No	Uint8	-	0 to 255	255
	1Eh	COB-ID (0x1403-01h) of RPDO4	No	No	Uint32	-	-2147483647 to +2147483647	0
	20h	Transmission type of RPDO4 (0x1403-02h)	No	No	Uint8	-	0 to 255	255
	21h	Number of valid mapping objects of RPDO1 (0x1600-00h)	No	No	Uint8	-	0 to 8	1
	22h	RPDO1 mapping object 1 (0x1600-01h)	No	No	Uint32	-	0 to 2147483647	1614807056 (0x60400010)
	24h	RPDO1 mapping object 2 (0x1600-02h)	No	No	Uint32	-	0 to 2147483647	0
	26h	RPDO1 mapping object 3 (0x1600-03h)	No	No	Uint32	-	0 to 2147483647	0
	28h	RPDO1 mapping object 4 (0x1600-04h)	No	No	Uint32	-	0 to 2147483647	0
	2Ah	RPDO1 mapping object 5 (0x1600-05h)	No	No	Uint32	-	0 to 2147483647	0
	2Ch	RPDO1 mapping object 6 (0x1600-06h)	No	No	Uint32	-	0 to 2147483647	0
	2Eh	RPDO1 mapping object 7 (0x1600-07h)	No	No	Uint32	-	0 to 2147483647	0
	30h	RPDO1 mapping object 8 (0x1600-08h)	No	No	Uint32	-	0 to 2147483647	0
	32h	Number of valid mapping objects of RPDO2 (0x1601-00h)	No	No	Uint8	-	0 to 8	2
	33h	RPDO2 mapping object 1 (0x1601-01h)	No	No	Uint32	-	0 to 2147483647	1614807056 (0x60400010)
	35h	RPDO2 mapping object 2 (0x1601-02h)	No	No	Uint32	-	0 to 2147483647	1616904200 (0x60600008)
	37h	RPDO2 mapping object 3 (0x1601-03h)	No	No	Uint32	-	0 to 2147483647	0
	39h	RPDO2 mapping object 4 (0x1601-04h)	No	No	Uint32	-	0 to 2147483647	0

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
202Dh	3Bh	RPDO2 mapping object 5 (0x1601-05h)	No	No	Uint32	-	0 to 2147483647	0
	3Dh	RPDO2 mapping object 6 (0x1601-06h)	No	No	Uint32	-	0 to 2147483647	0
	3Fh	RPDO2 mapping object 7 (0x1601-07h)	No	No	Uint32	-	0 to 2147483647	0
	41h	RPDO2 mapping object 8 (0x1601-08h)	No	No	Uint32	-	0 to 2147483647	0
	43h	Number of valid mapping objects of RPDO3 (0x1602-00h)	No	No	Uint8	-	0 to 8	2
	44h	RPDO3 mapping object 1 (0x1602-01h)	No	No	Uint32	-	0 to 2147483647	1614807056 (0x60400010)
	46h	RPDO3 mapping object 2 (0x1602-02h)	No	No	Uint32	-	0 to 2147483647	1618608160 (0x607A0020)
	48h	RPDO3 mapping object 3 (0x1602-03h)	No	No	Uint32	-	0 to 2147483647	0
	4Ah	RPDO3 mapping object 4 (0x1602-04h)	No	No	Uint32	-	0 to 2147483647	0
	4Ch	RPDO3 mapping object 5 (0x1602-05h)	No	No	Uint32	-	0 to 2147483647	0
	4Eh	RPDO3 mapping object 6 (0x1602-06h)	No	No	Uint32	-	0 to 2147483647	0
	50h	RPDO3 mapping object 7 (0x1602-07h)	No	No	Uint32	-	0 to 2147483647	0
	52h	RPDO3 mapping object 8 (0x1602-08h)	No	No	Uint32	-	0 to 2147483647	0
	54h	Number of valid mapping objects of RPDO4 (0x1603-00h)	No	No	Uint8	-	0 to 8	2
	55h	RPDO4 mapping object 1 (0x1603-01h)	No	No	Uint32	-	0 to 2147483647	1614807056 (0x60400010)
	57h	RPDO4 mapping object 2 (0x1603-02h)	No	No	Uint32	-	0 to 2147483647	1627324448 (0x60FF0020)
	59h	RPDO4 mapping object 3 (0x1603-03h)	No	No	Uint32	-	0 to 2147483647	0
	5Bh	RPDO4 mapping object 4 (0x1603-04h)	No	No	Uint32	-	0 to 2147483647	0
	5Dh	RPDO4 mapping object 5 (0x1603-05h)	No	No	Uint32	-	0 to 2147483647	0
	5Fh	RPDO4 mapping object 6 (0x1603-06h)	No	No	Uint32	-	0 to 2147483647	0
	61h	RPDO4 mapping object 7 (0x1603-07h)	No	No	Uint32	-	0 to 2147483647	0

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
202Dh	63h	RPDO4 mapping object 8 (0x1603-08h)	No	No	Uint32	-	0 to 2147483647	0

202Eh CANopen Communication Parameters 2

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
202Eh	1h	COB-ID (0x1800-01h) of TPDO1	No	No	Uint32	-	-2147483647 to +2147483647	1073742208 (0x40000180)
	3h	Transmission type of TPDO1 (0x1800-02h)	No	No	Uint8	-	0 to 255	255
	4h	Inhibit time of TPDO1 (0x1800-03h)	No	No	Uint16	-	0 to 65535	0
	5h	Event timer of TPDO1 (0x1800-05h)	No	No	Uint16	-	0 to 65535	0
	6h	COB-ID (0x1801-01h) of TPDO2	No	No	Uint32	-	-2147483647 to +2147483647	0
	8h	Transmission type of TPDO2 (0x1801-02h)	No	No	Uint8	-	0 to 255	255
	9h	Inhibit time of TPDO2 (0x1801-03h)	No	No	Uint16	-	0 to 65535	0
	0Ah	Event timer of TPDO2 (0x1801-05h)	No	No	Uint16	-	0 to 65535	0
	Bh	COB-ID(0x1802-01h) of TPDO3	No	No	Uint32	-	-2147483647 to +2147483647	0
	Dh	Transmission type of TPDO3 (0x1802-02h)	No	No	Uint8	-	0 to 255	255
	Eh	Inhibit time of TPDO3 (0x1802-03h)	No	No	Uint16	-	0 to 65535	0
	Fh	Event timer of TPDO3 (0x1802-05h)	No	No	Uint16	-	0 to 65535	0
	10h	COB-ID (0x1803-01h) of TPDO4	No	No	Uint32	-	-2147483647 to +2147483647	0
	12h	Transmission type of TPDO4 (0x1803-02h)	No	No	Uint8	-	0 to 255	255
	13h	Inhibit time of TPDO4 (0x1803-03h)	No	No	Uint16	-	0 to 65535	0
	14h	Event timer of TPDO4 (0x1803-05h)	No	No	Uint16	-	0 to 65535	0
	15h	Number of valid mapping objects of TPDO1	No	No	Uint8	-	0 to 8	1
	16h	TPDO1 mapping object 1 (0x1A00-01h)	No	No	Uint32	-	0 to 2147483647	1614872592 (0x60410010)

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
202Eh	18h	TPDO1 mapping object 2 (0x1A00-02h)	No	No	Uint32	-	0 to 2147483647	0
	1Ah	TPDO1 mapping object 3 (0x1A00-03h)	No	No	Uint32	-	0 to 2147483647	0
	1Ch	TPDO1 mapping object 4 (0x1A00-04h)	No	No	Uint32	-	0 to 2147483647	0
	1Eh	TPDO1 mapping object 5 (0x1A00-05h)	No	No	Uint32	-	0 to 2147483647	0
	20h	TPDO1 mapping object 6 (0x1A00-06h)	No	No	Uint32	-	0 to 2147483647	0
	22h	TPDO1 mapping object 7 (0x1A00-07h)	No	No	Uint32	-	0 to 2147483647	0
	24h	TPDO1 mapping object 8 (0x1A00-08h)	No	No	Uint32	-	0 to 2147483647	0
	26h	Number of valid mapping objects of TPDO2	No	No	Uint8	-	0 to 8	2
	27h	TPDO2 mapping object 1 (0x1A01-01h)	No	No	Uint32	-	0 to 2147483647	1614872592 (0x60410010)
	29h	TPDO2 mapping object 2 (0x1A01-02h)	No	No	Uint32	-	0 to 2147483647	1616969736 (0x60610008)
	2Bh	TPDO2 mapping object 3 (0x1A01-03h)	No	No	Uint32	-	0 to 2147483647	0
	2Dh	TPDO2 mapping object 4 (0x1A01-04h)	No	No	Uint32	-	0 to 2147483647	0
	2Fh	TPDO2 mapping object 5(0x1A01-05h)	No	No	Uint32	-	0 to 2147483647	0
	31h	TPDO2 mapping object 6 (0x1A01-06h)	No	No	Uint32	-	0 to 2147483647	0
	33h	TPDO2 mapping object 7 (0x1A01-07h)	No	No	Uint32	-	0 to 2147483647	0
	35h	TPDO2 mapping object 8 (0x1A01-08h)	No	No	Uint32	-	0 to 2147483647	0
	37h	Number of valid mapping objects of TPDO3	No	No	Uint8	-	0 to 8	2
	38h	TPDO3 mapping object 1 (0x1A02-01h)	No	No	Uint32	-	0 to 2147483647	1614872592 (0x60410010)
	3Ah	TPDO3 mapping object 2 (0x1A02-02h)	No	No	Uint32	-	0 to 2147483647	1617166368 (0x60640020)
	3Ch	TPDO3 mapping object 3 (0x1A02-03h)	No	No	Uint32	-	0 to 2147483647	0
	3Eh	TPDO3 mapping object 4 (0x1A02-04h)	No	No	Uint32	-	0 to 2147483647	0
	40h	TPDO3 mapping object 5 (0x1A02-05h)	No	No	Uint32	-	0 to 2147483647	0

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
202Eh	42h	TPDO3 mapping object 6 (0x1A02-06h)	No	No	Uint32	-	0 to 2147483647	0
	44h	TPDO3 mapping object 7 (0x1A02-07h)	No	No	Uint32	-	0 to 2147483647	0
	46h	TPDO3 mapping object 8 (0x1A02-08h)	No	No	Uint32	-	0 to 2147483647	0
	48h	Number of valid mapping objects of TPDO4	No	No	Uint8	-	0 to 8	2
	49h	TPDO4 mapping object 1 (0x1A03-01h)	No	No	Uint32	-	0 to 2147483647	1614872592 (0x60410010)
	4Bh	TPDO4 mapping object 2 (0x1A03-02h)	No	No	Uint32	-	0 to 2147483647	1617690656 (0x606C0020)
	4Dh	TPDO4 mapping object 3 (0x1A03-03h)	No	No	Uint32	-	0 to 2147483647	0
	4Fh	TPDO4 mapping object 4 (0x1A03-04h)	No	No	Uint32	-	0 to 2147483647	0
	51h	TPDO4 mapping object 5 (0x1A03-05h)	No	No	Uint32	-	0 to 2147483647	0
	53h	TPDO4 mapping object 6 (0x1A03-06h)	No	No	Uint32	-	0 to 2147483647	0
	55h	TPDO4 mapping object 7 (0x1A03-07h)	No	No	Uint32	-	0 to 2147483647	0
	57h	TPDO4 mapping object 8 (0x1A03-08h)	No	No	Uint32	-	0 to 2147483647	0

2030h Servo Status Variables Read Through Communication

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
2030	01h	Servo status read through communication	RO	TPDO	Uint16	-	-	0
	02h	DO function state 1 read through communication	RO	TPDO	Uint16	-	0 to 65535	0
	03h	DO function state 2 read through communication	RO	TPDO	Uint16	-	0 to 65535	0

2031h Variables Related to Communication Settings

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
2031	1h	VDI virtual level set through communication	RW	RPDO	Uint16	-	0 to 65535	0
	5h	DO state set through communication	RW	RPDO	Uint16	-	0 to 7	0

203Fh Inovance Servo Fault Codes

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
203F	0h	Inovance servo fault code	RO	TPDO	Uint32	-	-	-

6.4 Object Group 6000h

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
603Fh	-	Error code	RO	TPDO	Uint16	-	0 to 65535	0
6040h	-	Control word	RW	YES	Uint16	-	0 to 65535	0
6041h	-	Status word	RO	TPDO	Uint16	-	0 to 65535	-
605Ah	-	Quick stop option code	RW	YES	Int16	-	0 to 7	2
605Dh	-	Stop option code	RW	YES	Int16	-	0 to 7	1
6060h	-	Modes of operation	RW	YES	Int8	-	0 to 7	0
6061h	-	Modes of operation display	RO	TPDO	Int8	-	0 to 7	-
6062h	-	Position demand value	RO	TPDO	Int32	Reference unit	-2^{31} to $+(2^{31}-1)$	-
6063h	-	Position actual value*	RO	TPDO	Int32	Encoder unit	-2^{31} to $+(2^{31}-1)$	-
6064h	-	Position actual value	RO	TPDO	Int32	Reference unit	-2^{31} to $+(2^{31}-1)$	-
6065h	-	Following error window	RW	YES	Uint32	Reference unit	0 to $(2^{32}-1)$	3435868
6067h	-	Position window	RW	YES	Uint32	Reference unit	0 to $(2^{32}-1)$	734
6068h	-	Position window time	RW	YES	Uint16	ms	0 to 65535	0
606Bh	-	Velocity demand value	RO	TPDO	Int32	Reference unit	-2^{31} to $+(2^{31}-1)$	-
606Ch	-	Velocity actual value	RO	TPDO	Int32	Reference unit	-2^{31} to $+(2^{31}-1)$	-
606Dh	-	Velocity window	RW	YES	Uint16	RPM	0 to 65535	10
606Eh	-	Velocity window time	RW	YES	Uint16	ms	0 to 65535	0
606Fh	-	Velocity threshold	RW	YES	Uint16	RPM	0 to 65535	10
6070h	-	Velocity threshold time	RW	YES	Uint16	ms	0 to 65535	0
6071h	-	Target torque	RW	RPDO	INT16	0.1%	-5000 to +5000	0
6072h	-	Max. torque	RW	RPDO	INT16	0.1%	-5000 to +5000	3000
6074h	-	Torque demand value	RO	TPDO	INT16	0.1%	-5000 to +5000	0
6077h	-	Torque actual value	RO	TPDO	INT16	0.1%	-5000 to +5000	0
607Ah	-	Target position	RW	YES	Int32	Reference unit	-2^{31} to $+2^{31}-1$)	0
607Ch	-	Home offset	RW	YES	Int32	Reference unit	-2^{31} to $+(2^{31}-1)$	0

Index	Sub-index	Name	Access	Mapping	Data Type	Unit	Data Range	Default
607Dh	-	Software position limit	-	-	-	-	-	-
	1h	Min. position limit	RW	YES	Int32	Reference unit	-2^{31} to $+(2^{31}-1)$	-2^{31}
	2h	Max. position limit	RW	YES	Int32	Reference unit	-2^{31} to $+(2^{31}-1)$	$(2^{31}-1)$
607Eh	-	Polarity	RW	Y	Uint8	-	0 to 255	0
607Fh	-	Max. profile velocity	RW	YES	Uint32	Reference unit	0 to $(2^{32}-1)$	1048576000
6081h	-	Profile velocity	RW	YES	Uint32	Reference unit	0 to $(2^{32}-1)$	1747627
6083h	-	Profile acceleration	RW	YES	Uint32	Reference unit	0 to $(2^{32}-1)$	174762666
6084h	-	Profile deceleration	RW	YES	Uint32	Reference unit	0 to $(2^{32}-1)$	174762666
6085h	-	Quick stop deceleration	RW	YES	Uint32	Reference unit	0 to $(2^{32}-1)$	2147483647
6086h	-	Motion profile type	RW	YES	Int16	-	0	0
6087h	-	Torque slope	RW	RPDO	UINT32	0.1%/s	0 to $(2^{32}-1)$	4294967295
6091h	0h	Highest subindex supported	RO	NO	UNIT8	-	-	2
	1h	Motor revolutions	RW	PRDO	Uint32	-	1 to $(2^{32}-1)$	1
	2h	Shaft revolutions	RW	PRDO	Uint32	-	1 to $(2^{32}-1)$	1
6098h	-	Homing mode	RW	YES	Int8	-	0 to 35	1
6099h	-	Homing speeds	-	-	-	-	-	-
	1h	Speed during search for switch	RW	YES	Uint32	Reference unit	0 to $(2^{32}-1)$	1747627
	2h	Speed during search for zero	RW	YES	Uint32	Reference unit	0 to $(2^{32}-1)$	174763
609Ah	-	Homing acceleration	RW	YES	Uint32	Reference unit	0 to $(2^{32}-1)$	174762666
60C1h	-	Interpolation data record	-	-	-	-	-	-
	1h	Interpolation displacement	RW	YES	Int32	-	-2^{31} to $+(2^{31}-1)$	0
60C2h	-	Interpolation time	-	-	-	-	-	-
	1h	Interpolation time unit	RW	YES	Uint8	$10^{\text{ip time index}}$s	1 to 20	1
	2h	Interpolation time index	RW	YES	Int8	-	-3	-3
60C5h	-	Max. acceleration	RW	YES	Uint32	p/ms	0 to $(2^{32}-1)$	2147483647
60C6h	-	Max. deceleration	RW	YES	Uint32	p/ms	0 to $(2^{32}-1)$	2147483647
60F4h	-	Following error actual value	RO	TPDO	Int32	Reference unit	-2^{31} to $+(2^{31}-1)$	-
60FCh	-	Position demand value	RO	TPDO	Int32	Encoder unit	-2^{31} to $+(2^{31}-1)$	-
60FDh	-	DI state	RO	TPDO	Uint32	-	0 to $(2^{32}-1)$	-
60FEh	-	Digital outputs	-	-	-	-	-	-
	1h	DO state	RW	TPDO	Uint32	-	0 to $(2^{32}-1)$	0
60FFh	-	Target velocity	RW	YES	Int32	Reference unit	-2^{31} to $+(2^{31}-1)$	0

6.5 Object Dictionary

6.5.1 Communication Parameters

Index 1000h	Name	Device type					Data Structure	VAR	Data Type	Uint32
	Access	RO	Mapping	NO	Related Mode	-	Data Range	Uint 32	Default	0x20192
Used to describe the device profile or application specification used.										

Index 1001h	Name	Error register					Data Structure	VAR	Data Type	Uint8
	Access	RO	Mapping	NO	Related Mode	-	Data Range	Uint 8	Default	0x0

Indicates information about error types in bits, as listed in the following table:

Bit	Meaning	Bit	Meaning
0	Normal	4	Communication
1	Current	5	Sub-protocol
2	Voltage	6	Reserved
3	Temperature	7	Manufacturer-defined

When an error occurs, the bit corresponding to the error is "1", and once an error occurs, bit0 must be set to 1.

Index 1003h	Name	Pre-defined error field					Data Structure	ARR	Data Type	Uint32
	Access	RO	Mapping	NO	Related Mode	-	Data Range	-	Default	-

Sub-index 00h	Name	Number of errors					Data Structure	-	Data Type	Uint8
	Access	RW	Mapping	NO	Related Mode	-	Data Range	0 to 4	Default	0
Only the value 0 can be written. In this case, all error records are cleared.										

Sub-indexes 1h to 4h	Name	Standard error field					Data Structure	-	Data Type	Uint32
	Access	RW	Mapping	NO	Related Mode	-	Data Range	Uint 32	Default	0

When the subindex is 0, the data is unreadable.

When an error occurs, the error is saved in the following format:

31 16	15 0
Manufacturer error code	Standard error code

MSB LSB

Index	Name	COB-ID (COB-ID SYNC message)					Data Structure	VAR	Data Type	Uint32
1005h	Access	RW	Mapping	NO	Related Mode	-	Data Range	Uint 32	Default	0x80
Only 0x80h and 0x40000080h can be written. When 0x80h is written, the SYNC generator is deactivated. When 0x40000080h is written, the SYNC generator is activated. Before activating the SYNC generator, set the SYNC cycle (1006h) to a non-zero value.										

Index	Name	Communication cycle					Data Structure	VAR	Data Type	Uint32
1006h	Access	RW	Mapping	NO	Related Mode	-	Data Range	Uint 32	Default	0
Applicable to the SYNC generator only (unit: us).										

Index	Name	Manufacturer device name					Data Structure	VAR	Data Type	String
1008h	Access	CONST	Mapping	NO	Related Mode	-	Data Range	String	Default	SV660C servo drive

Index	Name	Manufacturer hardware version					Data Structure	VAR	Data Type	String
1009h	Access	CONST	Mapping	NO	Related Mode	-	Data Range	String	Default	V0.0

Index	Name	Manufacturer software version					Data Structure	VAR	Data Type	String
100Ah	Access	CONST	Mapping	NO	Related Mode	-	Data Range	String	Default	402.XX
In 402.**: **: The upgrade record No. of the CANopen software.										

Index	Name	Guard time					Data Structure	VAR	Data Type	Uint16
100Ch	Access	RW	Mapping	NO	Related Mode	-	Data Range	Uint 16	Default	0
Unit: ms										

Index	Name	Life time factor					Data Structure	VAR	Data Type	Uint8
100Dh	Access	RW	Mapping	NO	Related Mode	-	Data Range	Uint 8	Default	0
The life time factor used must be greater than 1.										

Index 1010h	Name	Save parameters					Data Structure	ARR	Data Type	Uint32
	Access	RW	Mapping	NO	Related Mode	-	Data Range	Uint 8	Default	0

Saving parameters means to save current parameter values to the EEPROM. When the EEPROM is loaded next time (next power-on, node reset, or communication reset) , the saved values will be loaded.

To save parameters, specify the subindex of the saving area and write "save" based on the ASCII code.

The mapping between ASCII codes and hexadecimal data is as follows:

	MSB			LSB
ASCII	e	v	a	s
Hexadecimal	65h	76h	61h	73h

Value	Meaning
0	Parameters can neither be saved automatically nor saved based on commands.
1	Parameters can be saved based on commands only.
2	Parameters can be saved automatically only.
3	Parameters can either be saved automatically or saved based on commands.

The value returned after a subindex is read indicates the mode in which the subindex saves parameters.

The SV660C servo drive saves parameters based on commands only. Parameters will not be saved automatically. The value 1 is returned after a subindex is read and saved.

Sub-index 00h	Name	Highest subindex supported					Data Structure	-	Data Type	Uint8
	Access	RO	Mapping	NO	Related Mode	-	Data Range	4	Default	4

Sub-index 01h	Name	Save all parameters					Data Structure	-	Data Type	Uint32
	Access	RW	Mapping	NO	Related Mode	-	Data Range	-	Default	1

Save all parameters in the object dictionary list.

Sub-index 02h	Name	Save communication parameters					Data Structure	-	Data Type	Uint32
	Access	RW	Mapping	NO	Related Mode	-	Data Range	-	Default	1

Save parameters of the object group 1000h.

Sub-index 03h	Name	Save application parameters					Data Structure	-	Data Type	Uint32
	Access	RW	Mapping	NO	Related Mode	-	Data Range	-	Default	1

Save parameters of the object group 6000h.

Sub-index 04h	Name	Save manufacturer-defined parameters					Data Structure	-	Data Type	Uint32
	Access	RW	Mapping	NO	Related Mode	-	Data Range	-	Default	1
Save parameters of the object group 2000h.										

Index 1011h	Name	Restore default parameters					Data Structure	ARR	Data Type	Uint32
	Access	RW	Mapping	NO	Related Mode	-	Data Range	-	Default	-

Restoring default parameters means to restore default parameters to the EEPROM. However, this operation does not take effect immediately. The default values will be loaded at the next EEPROM loading (next power-on, node reset, or communication reset).

To restore default parameters, specify the subindex of the restoring area and write "load" based on the ASCII code. The mapping between ASCII codes and hexadecimal data is as follows:

	MSB			LSB
ASCII	d	a	o	l
Hexadecimal	64h	61h	6Fh	6Ch

Value	Meaning
0	The device cannot restore default parameters.
1	The device can restore default parameters.

The value returned after a subindex is read indicates the mode in which the subindex restores default parameters.

The SV660C servo drive can restore default parameters. The value 1 will be returned after a non-zero subindex is read.

Sub-index 00h	Name	Highest subindex supported					Data Structure	-	Data Type	Uint8
	Access	RO	Mapping	NO	Related Mode	-	Data Range	4	Default	4

Sub-index 01h	Name	Restore all default parameters					Data Structure	-	Data Type	Uint32
	Access	RW	Mapping	NO	Related Mode	-	Data Range	-	Default	1
Restore all default parameters in the object dictionary list.										

Sub-index 02h	Name	Restore communication default parameters					Data Structure	-	Data Type	Uint32
	Access	RW	Mapping	NO	Related Mode	-	Data Range	-	Default	1
Restore default parameters of the object group 1000h.										

Sub-index 03h	Name	Restore application default parameters					Data Structure	-	Data Type	Uint32
	Access	RW	Mapping	NO	Related Mode	-	Data Range	-	Default	1
Restore default parameters of the object group 6000h.										

Sub-index 04h	Name	Restore manufacturer-defined default parameters					Data Structure	-	Data Type	Uint32
	Access	RW	Mapping	NO	Related Mode	-	Data Range	-	Default	1
Restore default parameters of the object group 2000h.										

Index 1014h	Name	COB-ID (COB-ID emergency message)					Data Structure	VAR	Data Type	Uint32
	Access	RW	Mapping	NO	Related Mode	-	Data Range	Uint 32	Default	0x80+ Node_ID
The most significant bit indicates whether to disable the emergency message of the device. Only the data "0x80+Node_ID" can be written for the bit to enable the emergency message of the device. If the data "0x80000080+Node_ID" is written, the emergency message is disabled. When the emergency message takes effect, its COB-ID must be consistent with this object.										

Index 1016h	Name	Consumer heartbeat time					Data Structure	ARR	Data Type	Uint32
	Access	RW	Mapping	NO	Related Mode	-	Data Range	-	Default	-

Includes the address of the monitored node and the actual consumer time. The consumer heartbeat time must be longer than the heartbeat producer time (in ms) of the corresponding node. Two different consumer time cannot be set for one node.

The parameter content is as follows.

31 24	23 16	15 0
Reserved (0)	Monitored address	Monitoring time

MSB LSB

The value returned after a subindex is read indicates the mode in which the subindex restores default parameters.

The SV660C servo drive can restore default parameters. The value 1 will be returned after a non-zero subindex is read.

Sub-index 00h	Name	Highest subindex supported					Data Structure	-	Data Type	Uint8
	Access	RO	Mapping	NO	Related Mode	-	Data Range	5	Default	5

Sub-indexes 1h to 5h	Name	Consumer heartbeat time					Data Structure	-	Data Type	Uint32
	Access	RW	Mapping	NO	Related Mode	-	Data Range	Uint 32	Default	0

Index 1017h	Name	Producer heartbeat time					Data Structure	VAR	Data Type	Uint16
	Access	RW	Mapping	NO	Related Mode	-	Data Range	Uint 16	Default	0
The unit is ms.										

Index 1018h	Name	Identity object					Data Structure	REC	Data Type	Individually related
	Access	RO	Mapping	NO	Related Mode	-	Data Range	-	Default	-

Sub-index 00h	Name	Highest subindex supported					Data Structure	-	Data Type	Uint8
	Access	RO	Mapping	NO	Related Mode	-	Data Range	3	Default	3

Sub-index 01h	Name	Vendor-ID					Data Structure	-	Data Type	Uint32
	Access	RO	Mapping	NO	Related Mode	-	Data Range	Uint 32	Default	0x3B9

Indicates the unique ID allocated by the CiA organization.

Sub-index 02h	Name	Product code					Data Structure	-	Data Type	Uint32
	Access	RO	Mapping	NO	Related Mode	-	Data Range	Uint 32	Default	0xD0107

The relation between Inovance device codes and product series and product models shown on the electronic labels is as follows.

31 16	15 0
Product series	Product model

MLB LSB

Sub-index 03h	Name	Revision number					Data Structure	-	Data Type	Uint32
	Access	RO	Mapping	NO	Related Mode	-	Data Range	Uint 32	Default	0x00020003

The relation between the revision number and the software version (100Ah) is shown in the following table.

31 16	15 0
Primary revision	Secondary revision

MLB LSB

The main revision number based on the customized number is 0x192 . Each time the code is updated, the secondary revision number accumulates accordingly.

Index 1029h	Name	Error behavior					Data Structure	ARR	Data Type	Uint8
	Access	RW	Mapping	NO	Related Mode	-	Data Range	-	Default	-

Indicates the state that NMT turns to upon occurrence of different errors. NMT turns to different states based on different values.

Value	Meaning
0	Operational state applied currently, turning to the pre-operational state
1	Staying in the present state
2	Turning to the stop state
Others	Reserved

The SV660C servo drive only supports automatic NMT state conversion in the case of communication fault.

Sub-index 00h	Name	Highest subindex supported					Data Structure	-	Data Type	Uint8
	Access	RO	Mapping	NO	Related Mode	-	Data Range	1	Default	1

Sub-index 01h	Name	Communication error					Data Structure	-	Data Type	Uint8
	Access	RW	Mapping	NO	Related Mode	-	Data Range	Uint 8	Default	0
Communication errors include NMT error control timeout, PDO length error, and bus disconnection.										

Index 1200h	Name	SDO server parameter					Data Structure	REC	Data Type	SDO parameter
	Access	RO	Mapping	NO	Related Mode	-	Data Range	-	Default	-
The default SDO always exists and is a read-only constant.										

Sub-index 00h	Name	Highest subindex supported					Data Structure	-	Data Type	Uint8
	Access	RO	Mapping	NO	Related Mode	-	Data Range	2	Default	2

Sub-index 01h	Name	Client to server COB-ID (COB-ID client → Server(rx))					Data Structure	-	Data Type	Uint32
	Access	RO	Mapping	NO	Related Mode	-	Data Range	Uint 32	Default	0x600 + Node_ID

Sub-index 02h	Name	Server to client COB-ID (COB-ID server → Client(tx))					Data Structure	-	Data Type	Uint32
	Access	RO	Mapping	NO	Related Mode	-	Data Range	Uint 32	Default	0x580 + Node_ID

Indexes 1400h to 1403h	Name	RPDO communication parameter					Data Structure	REC	Data Type	PDO parameter
	Access	RW	Mapping	NO	Related Mode	-	Data Range	-	Default	-

Sub-index 00h	Name	Highest subindex supported					Data Structure	-	Data Type	Uint8
	Access	RO	Mapping	NO	Related Mode	-	Data Range	2 to 6	Default	2

Sub-index 01h	Name	COB-ID used by RPDO					Data Structure	-	Data Type	Uint32
	Access	RW	Mapping	NO	Related Mode	-	Data Range	Uint 32	Default	See below for details.

Only the most significant bit (MSB) can be changed. When the MSB is 0, the PDO is valid; when the MSB is 1, the PDO is invalid.

The factory settings are as follows:

1400h: 0x00000200 + Node_ID

1401h: 0x80000300 + Node_ID

1402h: 0x80000400 + Node_ID

1403h: 0x80000500 + Node_ID

Sub-index 02h	Name	RPDO transmission type					Data Structure	-	Data Type	Uint8
	Access	RW	Mapping	NO	Related Mode	-	Data Range	Uint 8	Default	255

This value can be modified only when the PDO is invalid.

Different values indicate different RPDO transmission types, as shown below:

Value	Meaning
0	Synchronous, acyclic
1 to 240	Synchronous, cyclic
254, 255	Asynchronous, acyclic

Indexes 1600h to 1603h	Name	RPDO mapping parameter					Data Structure	REC	Data Type	Mapping parameter of RPDO
	Access	RW	Mapping	NO	Related Mode	-	Data Range	-	Default	-

This value can be modified only when the PDO is invalid. The total length of a mapping object cannot exceed 64 bits. Mapping based on bytes is supported only.

Sub-index 00h	Name	Number of mapped application objects in PDO					Data Structure	-	Data Type	Uint8
	Access	RW	Mapping	NO	Related Mode	-	Data Range	0 to 8	Default	-

When the value 0 is written, the mapping objects of other subindexes are cleared.

Sub-indexes 1h to 8h	Name	Application object					Data Structure	-	Data Type	Uint32
	Access	RW	Mapping	NO	Related Mode	-	Data Range	Uint 32	Default	-

The indexes and subindexes of mapping objects must be present in the object dictionary list, which are writable and mappable.

Subindexes are written in the following format:

31 16	15 8	7 0
Index	Subindex	Object length

MLB LSB

Default mapping content of RPDO

1) RPDO1

Subindex	Value	Meaning
0	1	One object mapped
1	0x60400010	Command word

2) RPDO2:

Subindex	Value	Meaning
0	2	Two objects mapped
1	0x60400010	Command word
2	0x60600008	Operation mode

3) RPDO3:

Subindex	Value	Meaning
0	2	Two objects mapped
1	0x60400010	Command word
2	0x607A0020	Target position

4) RPDO4:

Subindex	Value	Meaning
0	2	Two objects mapped
1	0x60400010	Command word
2	0x60FF0020	Target velocity

Indexes 1800h to 1803h	Name	TPDO communication parameter					Data Structure	REC	Data Type	PDO communication parameter
	Access	RW	Mapping	NO	Related Mode	-	Data Range	-	Default	-

Sub-index 00h	Name	Highest subindex supported					Data Structure	-	Data Type	Uint8
	Access	RO	Mapping	NO	Related Mode	-	Data Range	2 to 6	Default	5

Sub-index 01h	Name	COB-ID used by TPDO					Data Structure	-	Data Type	Uint32
	Access	RW	Mapping	NO	Related Mode	-	Data Range	Uint 32	Default	See below for details.

Only the most significant bit and the second most significant bit can be modified.

When the most significant bit is 0, this PDO is valid. When the most significant bit is 1, this PDO is invalid.

The second most significant bit indicates whether this PDO can be triggered by a remote frame. As the SV660C servo drive does not support this function, this bit is meaningless. It is recommended to set this bit to 1, indicating the PDO cannot be triggered by a remote frame.

The factory settings are as follows:

1800h: 0x40000180 + Node_ID

1801h: 0xC0000280 + Node_ID

1802h: 0xC0000380 + Node_ID

1803h: 0xC0000480 + Node_ID

Sub-index 02h	Name	TPDO transmission type					Data Structure	-	Data Type	Uint8
	Access	RW	Mapping	NO	Related Mode	-	Data Range	Uint 8	Default	255

This value can be modified only when the PDO is invalid. Different values indicate different TPDO transmission types, as shown below.

Value	Meaning
0	Synchronous, acyclic
1 to 240	Synchronous, cyclic
254, 255	Asynchronous, acyclic
Other	Reserved

Sub-index 03h	Name	Inhibit time					Data Structure	-	Data Type	Uint16
	Access	RW	Mapping	NO	Related Mode	-	Data Range	Uint 16	Default	0

This value can be modified only when the PDO is invalid.

The unit is 100 us. The setpoint 0 indicates the inhibit time is invalid.

Sub-index 05h	Name	Event timer					Data Structure	-	Data Type	Uint16
	Access	RW	Mapping	NO	Related Mode	-	Data Range	Uint 16	Default	0

This value can be modified only when the PDO is invalid.

The unit is 1 ms. The setpoint 0 indicates the event timer is invalid.

Indexes 1A00h to 1A03h	Name	TPDO mapping					Data Structure	REC	Data Type	PDO mapping parameters
	Access	RW	Mapping	NO	Related Mode	-	Data Range	-	Default	-

This object can be modified only when the PDO is invalid. The total length of a mapping object cannot exceed 64 bits. Mapping based on bytes is supported only.

Sub-index 00h	Name	Number of mapped application objects in TPDO					Data Structure	-	Data Type	Uint8
	Access	RW	Mapping	NO	Related Mode	-	Data Range	0 to 8	Default	-

When 0 is written, the mapping objects of other subindexes are cleared.

Sub-indexes 1h to 8h	Name	Application object					Data Structure	-	Data Type	Uint32
	Access	RW	Mapping	NO	Related Mode	-	Data Range	Uint 32	Default	-

The indexes and subindexes of mapping objects must be present in the object dictionary list, which are writable and mappable.

Subindexes are written in the following format:

31 16	15 8	7 0
Index	Subindex	Object length

MLB LSB

Default mapping content of the TPDO:

1) TPDO1:

Subindex	Value	Meaning
0	1	One object mapped
1	0x60410010	Status word

2) TPDO2:

Subindex	Value	Meaning
0	2	Two objects mapped
1	0x60410010	Status word
2	0x60610008	Current running mode

3) TPDO3:

Subindex	Value	Meaning
0	2	Two objects mapped
1	0x60410010	Status word
2	0x60640020	Current position

4) TPDO4:

Subindex	Value	Meaning
0	2	Two objects mapped
1	0x60410010	Status word
2	0x606C0020	Current speed

6.5.2 Manufacturer-Defined Parameters

For parameters with the same functions as those of SV660P servo drives, see *SV660P Series Servo Drive Advanced User Guide* for details. This section only describes parameters with functions different from those of SV660P servo drives.

Index 2005h	Name	Position control parameters					Data Structure	ARR	Data Type	Uint16
	Access	-	Mapping	-	Related Mode	-	Data Range	OD data range	Default	OD default value
Defines position control parameters.										

Sub-index 3Eh	Name	Position window unit set					Data Structure	-	Data Type	Uint16
	Access	RW	Mapping	YES	Related Mode	All	Data Range	0 to 1	Default	0

Defines the unit of 6067h (Position window).

Value	Unit
0	Encoder unit
1	Reference unit

Index 2007h	Name	Torque control parameters					Data Structure	ARR	Data Type	Uint16
	Access	-	Mapping	-	Related Mode	-	Data Range	OD data range	Default	OD default value
Defines torque control parameters.										

Sub-index 12h	Name	Speed limit source					Data Structure	-	Data Type	Uint16
	Access	RW	Mapping	YES	Related Mode	All	Data Range	0 to 3	Default	0

Defines the unit of 6067h (Position window).

Value	Speed Limit Source
0	Internal speed limit
1	V-Lmt used as external speed limit input
2	V-SEL used to select speed limit 1 or speed limit 2
3	607F used as the speed limit

Index 200Ah	Name	Fault and protection parameters					Data Structure	ARR	Data Type	Uint16
	Access	-	Mapping	-	Related Mode	-	Data Range	OD data range	Default	OD default value
Defines the fault and protection parameters.										

Sub-index 02h	Name	Absolute position limit					Data Structure	-	Data Type	Uint16
	Access	RW	Mapping	YES	Related Mode	All	Data Range	0 to 2	Default	0

Defines the conditions for enabling software absolute position limit.

Value	Communication rate
0	Disable
1	Enable
2	Software absolute position limit enabled after homing

Index 200Ch	Name	Communication parameters					Data Structure	ARR	Data Type	Uint16
	Access	-	Mapping	-	Related Mode	-	Data Range	OD data range	Default	OD default value
Defines communication parameters.										

Sub-index 01h	Name	Axis address					Data Structure	-	Data Type	Uint16
	Access	RW	Mapping	NO	Related Mode	All	Data Range	1 to 127	Default	1

Defines the servo drive axis address.

When multiple servo drives are connected for networking, each servo drive can only have one unique address. Otherwise, communication error or communication failure may occur.

Sub-index 09h	Name	CAN communication baud rate					Data Structure	-	Data Type	Uint16
	Access	RW	Mapping	NO	Related Mode	All	Data Range	0 to 7	Default	5

Defines the communication rate between the servo drive and the host controller during CAN communication.

The communication rate set in the servo drive must be the same as that in the host controller. Otherwise, communication cannot be executed.

Value	Communication rate
0	20 k
1	50 k
2	100 k
3	125 k
4	250 k
5	500 k
6	1 M
7	1 M

Set a proper communication rate based on actual conditions (such as communication distance and communication data volume).

Sub-index 0Eh	Name	Save parameters modified through communication to EEPROM					Data Structure	-	Data Type	Uint16
	Access	RW	Mapping	YES	Related Mode	All	Data Range	0 to 3	Default	0

Defines whether to save parameters modified through communication to EEPROM.

If parameters need to be saved in the EEPROM, 200C-0Eh must be set to the corresponding value before parameter modification.

Value	Name	Description
0	Not saving	-
1	Saving 2000h series parameters	2000h series parameters refer to parameters of the SV660P servo drive. When 200C-0Eh is set to 1, parameters modified in RS232/RS485 communication can also be saved in the EEPROM.

6.5.3 Profile-Defined Parameters

Index 603Fh	Name	Error code					Data Structure	VAR	Data Type	Uint16
	Access	RO	Mapping	TPDO	Related Mode	All	Data Range	0 to 65535	Default	-

When an error described in the DSP402 profile occurs on the servo drive, 603Fh is the same as the description in DSP402. For details, see *"5.1 CANopen Communication Fault Codes"*.

When an error specified by the user occurs on the servo drive, 603Fh is 0xFF00. The value of 603Fh is in hexadecimal.

In addition, the object dictionary 203Fh displays auxiliary bytes of fault codes in hexadecimal.

203Fh is an Uint32 value, in which the high 16 bits indicate the internal fault code of the manufacturer, and the low 16 bits indicate the external fault code of the manufacturer.

Index 6040h	Name	Control word					Data Structure	VAR	Data Type	Uint16
	Access	RW	Mapping	YES	Related Mode	All	Data Range	0 to 65535	Default	0

Defines the control command.

bit	Name	Description
0	Servo ready	0: Inactive 1: Active
1	Main circuit switch-on	0: Inactive 1: Active
2	Quick stop	0: Active 1: Inactive
3	Servo running	0: Inactive 1: Active
4 to 6		Related to the operation mode.
7	Fault reset	Fault reset is implemented for faults and warnings that can be reset. The bit7 is rising edge-triggered. The bit7 is kept to 0 and other control commands are invalid.
8	Pause	Supported by MC056 program
9 to 10	N/A	Reserved
11 to 15	Manufacturer-defined	Reserved

Note:

- All bits in the control word constitute a control command. One bit is meaningless if it is assigned separately.
- The meanings of bit0 to bit3 and bit7 are the same in each operation mode. The servo drive can switch to the preset state according to the CiA402 state machine switchover process only when commands are sent in sequence. Each command corresponds to a certain state.
- The meanings of bit4 to bit6 vary with the operation mode. For details, see control commands in different modes.

Index 6041h	Name	Status word					Data Structure	VAR	Data Type	Uint16
	Access	RO	Mapping	TPDO	Related Mode	All	Data Range	0 to 65535	Default	-

Indicates the servo drive status.

bit	Name	Description
0	No fault	1: Active, 0: Inactive
1	Wait for the S-ON signal	1: Active, 0: Inactive
2	Servo running	1: Active, 0: Inactive
3	Fault	1: Active, 0: Inactive
4	Main circuit switch-on	1: Active, 0: Inactive
5	Quick stop	0: Active, 1: Inactive
6	Servo ready	1: Active, 0: Inactive
7	Warning	1: Active, 0: Inactive
8	Manufacturer-defined	Reserved
9	Remote control	0: Non-CANopen mode, some functions of SV660P standard software can be used. 1: CANopen remote control mode
10	Target reached	0: Target position not reached 1: Target position reached
11	Software internal position limit violation	0: The position reference or feedback does not reach the software internal position limit. 1: The position reference or feedback reaches the software internal position limit. In position control modes, when the position reference or feedback reaches the internal position limit, the servo drive takes the position limit as the target position and stops after reaching the limit. If a reverse displacement reference is received, the motor exits from the position limit state and this bit is set to 0.
12/13	-	Related to the operation mode.
14	N/A	Reserved
15	Homing completed	0: Homing is not performed or complete. 1: Homing is done and the reference point is found.

Note:

- All bits in the control word constitute a control command to reflect present servo state. One bit is meaningless if it is read separately.
- The meanings of bit0 to bit9 are the same in each operation mode. After commands in 6040h are sent in sequence, the servo drive returns a feedback of the servo state.
- bit12 and bit13 are related to each operation mode (see control commands in different modes for details).
- The meanings of bit10, bit11, and bit15 are the same in each operation mode. These three bits indicate the servo state after a certain operation mode is executed.

Index 605Ah	Name	Quick stop option code					Data Structure	VAR	Data Type	Int16
	Access	RW	Mapping	NO	Related Mode	All	Data Range	0 to 7	Default	2

Defines the quick stop mode.

Value	Stop Mode
0	Coast to stop, keeping de-energized state
1	Ramp to stop as defined by 6084h (hm: 609Ah), keeping de-energized state after stop
2	Ramp to stop as defined by 6085h, keeping de-energized state after stop
3	Stop at the emergency stop torque as defined by 2007-10h, keeping de-energized state after stop
4	N/A
5	Ramp to stop as defined by 6084h (hm: 609Ah), keeping position lock state after stop
6	Ramp to stop as defined by 6085h, keeping position lock state after stop
7	Stop at the emergency stop torque as defined by 2007-10h, keeping position lock state after stop

Index 6060h	Name	Modes of operation					Data Structure	VAR	Data Type	Int8
	Access	RW	Mapping	YES	Related Mode	All	Data Range	0 to 7	Default	0

Defines the servo drive operation mode.

bit	Description	Description
0	N/A	Reserved
1	Profile position mode	For parameter settings, see *"4.6 Profile Position Mode"*.
2	N/A	Reserved
3	Profile velocity mode	For parameter settings, see *"4.9 Profile Velocity Mode"*.
4	Profile torque mode	For parameter settings, see *"4.10 Profile Torque Mode"*.
5	N/A	Reserved
6	Homing mode	For parameter settings, see *"4.7 Homing Mode"*.
7	Interpolation mode	For parameter settings, see *"4.8 Interpolation Mode"*.

- If an unsupported operation mode is selected through the SDO, an SDO error will be returned.
- If an unsupported operation mode is selected through the PDO, the modification on the operation mode will be invalid.

Index 6061h	Name	Modes of operation display					Data Structure	VAR	Data Type	Int8
	Access	RO	Mapping	TPDO	Related Mode	All	Data Range	0 to 7	Default	-

Indicates the actual operation mode.

bit	Description	Description
0	N/A	Reserved
1	Profile position mode	For parameter settings, see *"4.6 Profile Position Mode"*.
2	N/A	Reserved
3	Profile velocity mode	For parameter settings, see *"4.9 Profile Velocity Mode"*.
4	Profile torque mode	For parameter settings, see *"4.10 Profile Torque Mode"*.
5	N/A	Reserved
6	Homing mode	For parameter settings, see *"4.7 Homing Mode"*.
7	Interpolation mode	For parameter settings, see *"4.8 Interpolation Mode"*.

Index 6062h	Name	Position demand value					Data Structure	VAR	Data Type	Int32
	Access	RO	Mapping	TPDO	Related Mode	pp/hm/ip	Data Range	-2^{31} to $+(2^{31}-1)$	Default	-
Indicates the real-time position reference (in reference unit).										

Index 6063h	Name	Position actual value*					Data Structure	VAR	Data Type	Int32
	Access	RO	Mapping	TPDO	Related Mode	All	Data Range	-2^{31} to $+(2^{31}-1)$	Default	-
Indicates the absolute motor position feedback in real time.										

Index 6064h	Name	Position actual value					Data Structure	VAR	Data Type	Int32
	Access	RO	Mapping	TPDO	Related Mode	All	Data Range	-2^{31} to $+(2^{31}-1)$	Default	-
Indicates user absolute position feedback in real time. Position actual value (6064h) x Gear ratio (6091h) = Position actual value* (6063h)										

Index 6065h	Name	Following error window					Data Structure	VAR	Data Type	Uint32
	Access	RW	Mapping	YES	Related Mode	pp/hm/ip	Data Range	0 to $(2^{32}-1)$	Default	3435868
Defines the threshold of excessive position deviation (in reference unit). ◆ When the difference between position demand value (6062h) and position actual value (6064h) exceeds ±6065h, Er.B00 (Position deviation too large) occurs. ◆ When 6065h is set to 0xFFFFFFFF, the position deviation will not be monitored. Use this function with caution.										

Index 6067h	Name	Position window					Data Structure	VAR	Data Type	Uint32
	Access	RW	Mapping	YES	Related Mode	pp/hm/ip	Data Range	0 to $(2^{32}-1)$	Default	734p
Defines the threshold for position reached. If the difference between 6062h and 6064h is within ±6067h and the time reaches 6068h, the position is considered to be reached and bit10 of 6041h is set to 1 in the profile position mode. This flag bit is meaningful only when the S-ON signal is active in the profile position mode.										

Index 6068h	Name	Position window time					Data Structure	VAR	Data Type	Uint16
	Access	RW	Mapping	YES	Related Mode	pp/hm/ip	Data Range	0 to 65535	Default	0 ms
Defines the window time for position reached. If the difference between 6062h and 6064h is within ±6067h and the time reaches 6068h, the position is considered to be reached and bit10 of 6041h is set to 1 in the profile position mode. This flag bit is meaningful only when the S-ON signal is active in the profile position mode.										

Index 606Bh	Name	Velocity demand value					Data Structure	VAR	Data Type	Int32
	Access	RO	Mapping	TPDO	Related Mode	All	Data Range	-2^{31} to $+(2^{31}-1)$	Default	-
Indicates the actual velocity reference. In position control modes, 606Bh indicates the speed reference corresponding to the position regulator. In velocity control modes, 606Bh indicates the input reference of the speed regulator.										

Index 606Ch	Name	Velocity actual value					Data Structure	VAR	Data Type	Int32
	Access	RO	Mapping	TPDO	Related Mode	All	Data Range	-2^{31} to $+(2^{31}-1)$	Default	-
Indicates the actual velocity feedback.										

Index 606Dh	Name	Velocity window					Data Structure	VAR	Data Type	Uint16
	Access	RW	Mapping	YES	Related Mode	pv	Data Range	0 to 65535	Default	10 RPM
Defines the threshold for speed reached. If the difference between 60FFh and 606Ch is within ±606Dh and the time reaches 606Eh, the speed is considered to be reached and bit10 of 6041h is set to 1 in the profile velocity mode. This flag bit is meaningful only when the S-ON signal is active in the profile velocity mode.										

Index 606Eh	Name	Velocity window time					Data Structure	VAR	Data Type	Uint16
	Access	RW	Mapping	YES	Related Mode	pv	Data Range	0 to 65535	Default	0 ms
Defines the window time for speed reached. If the difference between 60FFh and 606Ch is within ±606Dh and the time reaches 606Eh, the speed is considered to be reached and bit10 of 6041h is set to 1 in the profile velocity mode. This flag bit is meaningful only when the S-ON signal is active in the profile velocity mode.										

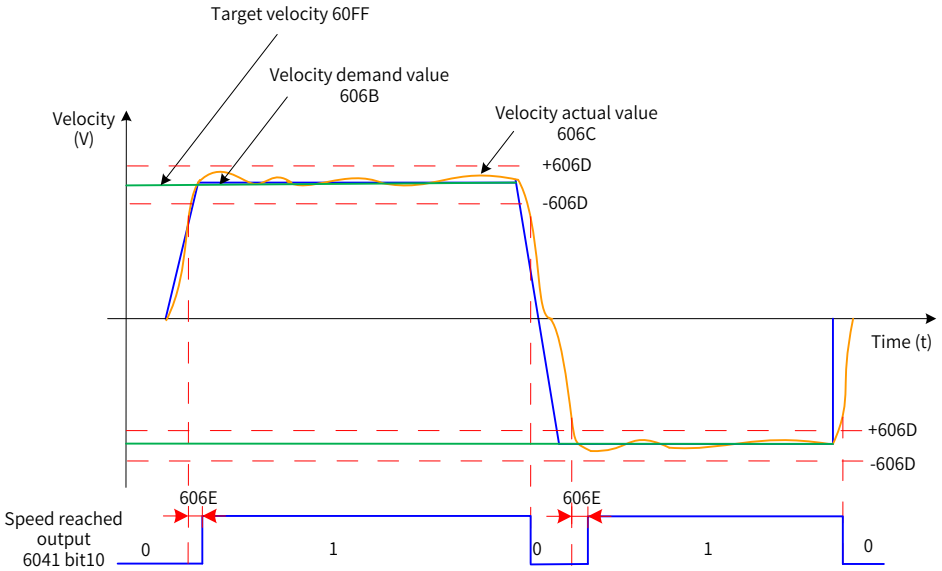


Figure 6-1 Speed reached

Index 606Fh	Name	Velocity threshold					Data Structure	VAR	Data Type	Uint16
	Access	RW	Mapping	YES	Related Mode	pv	Data Range	0 to 65535	Default	10 RPM
Defines the threshold for determining whether the user velocity is 0. When the value of 606Ch (Velocity actual value) is within ±606Fh and the time reaches 6070h, the user velocity is 0. When either condition is not met, the user velocity is considered not to be 0. This flag bit is meaningful only in the profile velocity mode. This flag bit is unrelated to the enable/disable state of the servo drive.										

Index 6070h	Name	Velocity threshold time					Data Structure	VAR	Data Type	Uint16
	Access	RW	Mapping	YES	Related Mode	pv	Data Range	0 to 65535	Default	0 ms
Defines the window time for determining whether the user velocity is 0. When the value of 606Ch (Velocity actual value) is within ±606Fh and the time reaches 6070h, the user velocity is 0. When either condition is not met, the user velocity is considered not to be 0. This flag bit is meaningful only in the profile velocity mode. This flag bit is unrelated to the enable/disable state of the servo drive.										

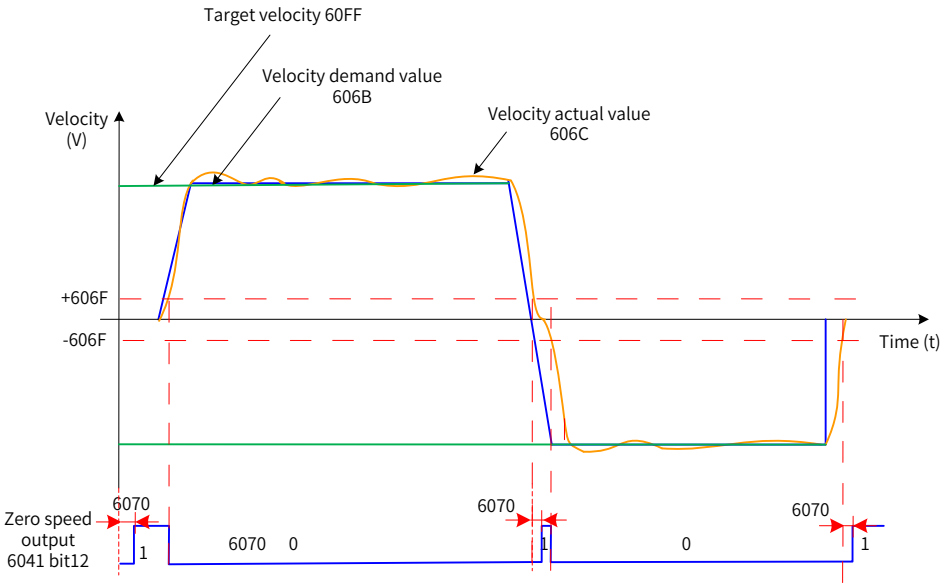


Figure 6-2 Velocity threshold

Index 6071h	Name	Target torque					Data Structure	VAR	Data Type	INTER16
	Access	RW	Mapping	YES	Related Mode	PT CST	Data Range	0xEC78 to 0x1388 (Unit: 0.1%)	Default	0x0000
Defines the target torque of the servo drive in the profile torque mode and cyclic synchronous torque mode. The value 100% corresponds to the rated motor torque.										

Index 6077h	Name	Torque actual value					Data Structure	VAR	Data Type	INTER16
	Access	RO	Mapping	TPDO	Related Mode	All	Data Range	- (Unit: 0.1%)	Default	-
Indicates the internal torque feedback of the servo drive. The value 100% corresponds to the rated motor torque.										

Index 607Ah	Name	Target position					Data Structure	VAR	Data Type	Int32
	Access	RW	Mapping	YES	Related Mode	pp	Data Range	-2^{31} to $+(2^{31}-1)$	Default	0

Defines the target position of the servo drive in the profile position mode.

Bit6 of 6040h	Description
0	607Ah indicates the absolute target position of the current phase. After positioning of the current phase is complete, 6064h equals to 607Ah.
1	607Ah indicates the target increment displacement of the current phase. After positioning of the current phase is done, the user displacement increment equals to 607Ah.

Index 607Ch	Name	Home offset					Data Structure	VAR	Data Type	Int32
	Access	RW	Mapping	YES	Related Mode	All	Data Range	-2^{31} to $+(2^{31}-1)$	Default	0

Defines the physical location of mechanical zero that deviates from the motor home in position control modes (profile position mode, interpolation mode, and homing mode).

◆ The home offset takes effect in the following conditions: The device is powered on, the homing operation is done, and bit15 of 6041h is set to 1.

◆ The home offset has the following effect:

After homing is done: 6064h (position actual value) = 607Ch

◆ If 607Ch is beyond the value of 607Dh (Software position limit), Er.D10 (Home offset setting error) occurs.

Index 607Dh	Name	Software position limit					Data Structure	ARR	Data Type	Int32
	Access	RW	Mapping	YES	Related Mode	All	Data Range	OD data range	Default	OD default value

Defines the minimum and maximum software position limits.
Min. position limit = (607D-01h)
Max. position limit = (607D-02h)

◆ This parameter is used to judge the absolute position. When the homing operation is not performed, 607Dh is meaningless.

◆ The condition for activating the software position limit is set in H0A-01 (object dictionary 0x200A-02h).

0: No absolute software position limit
1: Absolute software position limit activated
2: Absolute software position limit activated after homing

The absolute software position takes effect in the following conditions: The homing operation is done and bit15 of 6041h is set to 1.

◆ If the minimum software position limit is larger than the maximum software position limit, Er.D09 (Software position limit setting error) will occur.

◆ In position control modes, when the position reference or position feedback reaches the software internal position limit, the servo drive takes the position limit as the target position and stops after the motor reaches the limit, with an overtravel fault being reported. In this case, if a reverse displacement command is input, the motor exits from the position limit state and this bit is set to 0.

◆ When the external DI limit switch and internal software position limit are both activated, the limit state is determined by the external DI limit switch.

Sub-index 00h	Name	Number of entries					Data Structure	-	Data Type	Uint8
	Access	RO	Mapping	NO	Related Mode	-	Data Range	2	Default	2

Sub-index 01h	Name	Min. position limit					Data Structure	-	Data Type	Int32
	Access	RW	Mapping	YES	Related Mode	-	Data Range	-2^{31} to $+(2^{31}-1)$	Default	-2^{31}p

Defines the minimum software position limit relative to the mechanical zero.

Min. position limit = (607D-01h)

Sub-index 02h	Name	Max. position limit					Data Structure	-	Data Type	Int32
	Access	RW	Mapping	YES	Related Mode	-	Data Range	-2^{31} to $+(2^{31}-1)$	Default	$(2^{31}-1)$ p

Defines the maximum software position limit relative to the mechanical zero.

Max. software position limit = (607D-02h)

Index 607Eh	Name	Polarity			Data Structure	VAR	Data Type	Uint8
	Access	RW	Mapping	YES	Data Range	OD data range	Default	0

Defines the polarity of position or velocity references.

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Position reference polarity	Velocity reference polarity	Torque reference characteristics	N/A	N/A	N/A	N/A	N/A

Bit7 = 1: Indicates the position reference is multiplied by "-1" and the motor direction is reversed in the standard position mode or interpolation mode.

Bit = 1: Indicates the target velocity (60FFh) is multiplied by "-1" and the motor direction is reversed in the velocity mode.

Bit5 = 1: Indicates the target torque (6071h) is multiplied by "-1" and the motor direction is reversed in the torque mode.

N/A: No meaning

Index 607Fh	Name	Max. profile velocity					Data Structure	VAR	Data Type	Uint32
	Access	RW	Mapping	YES	Related Mode	All	Data Range	0 to $(2^{32}-1)$	Default	1048576000

Defines the maximum user running speed.

The setpoint takes effect when the speed reference of the slave changes.

Index 6081h	Name	Profile velocity					Data Structure	VAR	Data Type	Uint32
	Access	RW	Mapping	YES	Related Mode	pp	Data Range	0 to $(2^{32}-1)$	Default	1747627

Defines the constant running speed of the displacement reference in the profile position mode.

The setpoint takes effect after the slave receives the displacement reference.

Index 6083h	Name	Profile acceleration					Data Structure	VAR	Data Type	Uint32
	Access	RW	Mapping	YES	Related Mode	pp/pv	Data Range	0 to (2^{32}-1)	Default	174762666

Defines the acceleration rate of the displacement reference in the profile position mode.

The setpoint takes effect after the slave receives the displacement reference.

- The following formula applies if a 23-bit motor needs to run at 400 RPM (6081: 400 x 8388608/60) with acceleration rate being 400 RPM/s (6083: 400 x 8388608/60) and deceleration rate being 200 RPM/s (6084: 200 x 8388608/60) under a gear ratio of 1:1:

  Acceleration time t_{up} = Δ6081/Δ6083 = 1 (s); Deceleration time t_{down} = Δ6081/Δ6084 = 2 (s)

- For 6083h, the setpoint 0 will be forcibly changed to 1.

Index 6084h	Name	Profile deceleration					Data Structure	VAR	Data Type	Uint32
	Access	RW	Mapping	YES	Related Mode	pp/pv	Data Range	0 to (2^{32}-1)	Default	174762666

Defines the acceleration rate in the deceleration phase of the displacement reference in the profile position mode.

The setpoint takes effect after the slave receives the displacement reference.

- The following formula applies if a 23-bit motor needs to run at 400 RPM (6081: 400 x 8388608/60) with acceleration rate being 400 RPM/s (6083: 400 x 8388608/60) and deceleration rate being 200 RPM/s (6084: 200 x 8388608/60) under a gear ratio of 1:1:

  Acceleration time t_{up} = Δ6081/Δ6083 =1 (s); Deceleration time t_{down} = Δ6081/Δ6084 = 2 (s)

- For 6084h, the setpoint 0 will be forcibly changed to 1.

Index 6085h	Name	Quick stop deceleration					Data Structure	VAR	Data Type	Uint32
	Access	RW	Mapping	YES	Related Mode	All	Data Range	0 to (2^{32}-1)	Default	2147483647

Defines the deceleration rate when the quick stop command (6040h = 0x0002) is active and 605Ah (Quick stop option code) is set to 2 or 5.

- The following formula applies if a 23-bit motor needs to run at 400 RPM (6081: 400 x 8388608/60) with acceleration rate being 400 RPM/s (6083: 400 x 8388608/60) and deceleration rate being 200 RPM/s (6084: 200 x 8388608/60) under a gear ratio of 1:1:

  Acceleration time t_{up} = Δ6081/Δ6083 = 1 (s); Deceleration time t_{down} = Δ6081/Δ6084 = 2 (s)

- For 6085h, the setpoint 0 will be forcibly changed to 1.

Index 6086h	Name	Motion profile type					Data Structure	VAR	Data Type	Int16
	Access	RW	Mapping	YES	Related Mode	pp/pv	Data Range	0	Default	0

Defines the curve type of motor position references or speed references.

0: Linear

Index 6087h	Name	Torque slope					Data Structure	VAR	Data Type	UNSIGNED32
	Access	RW	Mapping	YES	Related Mode	PT CST	Data Range	0x00000000 to 0xFFFFFFFF (Unit: 0.1%/s)	Default	4294967295

Defines the acceleration (torque reference increment per second) of the torque reference in the profile torque mode.

In the profile torque mode or cyclic synchronous torque mode, if 605A (Quick stop option code) is set to 1, 2, 5 or 6, or 605D (Stop option code) is set to 1 or 2, the servo drive decelerates and stops as defined by 6087h.

If the value of 6087h exceeds the torque reference limit, the torque reference limit will be used.

For 6087h, the setpoint 0 will be forcibly changed to 1.

Index 6098h	Name	Homing method					Data Structure	VAR	Data Type	int8
	Access	RW	Mapping	RPDO	Related Mode	HM	Value Range	0 to 35	Default	1

Defines the homing method.

Mode	Description
1	Reverse homing: Home: Z signal Deceleration point: negative limit switch (N-OT) The falling edge of the N-OT signal must be reached before reaching the Z signal.
2	Forward homing: Home: Z signal Deceleration point: positive limit switch (P-OT) The falling edge of the P-OT signal must be reached before reaching the Z signal.
3	Forward homing: Home: Z signal Deceleration point: home switch (HW) The falling edge of the HW signal on the same side must be reached before reaching the Z signal.
4	Reverse homing: Home: Z signal Deceleration point: home switch (HW) The rising edge of the HW signal on the same side must be reached before reaching the Z signal.
5	Reverse homing: Home: Z signal Deceleration point: home switch (HW) The falling edge of the HW signal on the same side must be reached before reaching the Z signal.
6	Forward homing: Home: Z signal Deceleration point: home switch (HW) The rising edge of the HW signal on the same side must be reached before reaching the Z signal.
7	Forward homing: Home: Z signal Deceleration point: home switch (HW) The falling edge of the HW signal on the same side must be reached before reaching the Z signal.
8	Forward homing: Home: Z signal Deceleration point: home switch (HW) The rising edge of the HW signal on the same side must be reached before reaching the Z signal.
9	Forward homing: Home: Z signal Deceleration point: home switch (HW) The rising edge of the HW signal on the other side must be reached before reaching the Z signal.
10	Forward homing: Home: Z signal Deceleration point: home switch (HW) The falling edge of the HW signal on the other side must be reached before reaching the Z signal.

Index 6098h	Name	Homing method					Data Structure	VAR	Data Type	int8
	Access	RW	Mapping	RPDO	Related Mode	HM	Value Range	0 to 35	Default	1

Mode	Description
11	Reverse homing: Home: Z signal Deceleration point: home switch (HW) The falling edge of the HW signal on the same side must be reached before reaching the Z signal.
12	Reverse homing: Home: Z signal Deceleration point: home switch (HW) The rising edge of the HW signal on the same side must be reached before reaching the Z signal.
13	Reverse homing: Home: Z signal on the other side of the HW signal Deceleration point: home switch (HW) The rising edge of the HW signal on the other side must be reached before reaching the Z signal.
14	Reverse homing: Home: Z signal on the other side of the HW signal Deceleration point: home switch (HW) The falling edge of the HW signal on the other side of the home switch must be reached before reaching the Z signal.
17 to 32	Similar to values 1 to 14, except that the deceleration point overlaps with the home.
33	Reverse homing. The home is the Z signal.
34	Forward homing. The home is the Z signal.
35	The present position is used as the home.

6098h = 15/16/31/32: No meaning, the servo drive does not execute the homing operation.

Index 6099h	Name	Homing speeds					Data Structure	ARR	Data Type	Uint32
	Access	RW	Mapping	YES	Related Mode	All	Data Range	OD data range	Default	OD default value

Defines two speed values used in the homing mode.

- ◆ Speed during search for switch
- ◆ Speed during search for zero

Sub-index 00h	Name	Number of entries					Data Structure	-	Data Type	Uint8
	Access	RO	Mapping	NO	Related Mode	-	Data Range	2	Default	2

Sub-index 01h	Name	Speed during search for switch					Data Structure	-	Data Type	Uint32
	Access	RW	Mapping	YES	Related Mode	-	Data Range	0 to (2^{32}-1)	Default	1747627

Defines the speed (in reference unit) during searching for the deceleration point signal . This speed can be set to a large value to prevent homing timeout due to prolonged homing process.

Note: After finding the deceleration point, the slave decelerates and blocks the change of the home signal during deceleration. To prevent the slave from encountering the home signal during deceleration, set the switch position of the deceleration point signal properly to leave sufficient deceleration distance or increase the homing acceleration rate to shorten the deceleration time.

Sub-index 02h	Name	Speed during search for zero					Data Structure	-	Data Type	Int32
	Access	RW	Mapping	YES	Related Mode	-	Data Range	0 to (2^{32}-1)	Default	174763
This subindex defines the speed (in reference unit) during search for the home signal. Set this subindex to a small value to prevent overshoot during stop at a high speed and large deviation between the stop position and the preset mechanical home.										

Index 609Ah	Name	Homing acceleration					Data Structure	VAR	Data Type	Uint32
	Access	RW	Mapping	YES	Related Mode	hm	Data Range	0 to (2^{32}-1)	Default	174762666
Defines the acceleration rate in the homing mode. The setpoint takes effect after the homing operation is started. ◆ The following formula applies if a 23-bit motor needs to run at 400 RPM (6081: 400 x 8388608/60) with acceleration rate being 400 RPM/s (6083: 400 x 8388608/60) and deceleration rate being 200 RPM/s (6084: 200 x 8388608/60) under a gear ratio of 1:1: Acceleration time t_{up} = Δ6081/Δ6083 = 1 (s); Deceleration time t_{down} = Δ6081/Δ6084 = 2 (s)										

Index 60C1h	Name	Interpolation data record					Data Structure	ARR	Data Type	Int32
	Access	RW	Mapping	YES	Related Mode	ip	Data Range	OD data range	Default	OD default value
Defines the displacement reference in the interpolation mode.										

Sub-index 00h	Name	Number of entries					Data Structure	-	Data Type	Uint8
	Access	RO	Mapping	NO	Related Mode	-	Data Range	1	Default	1

Sub-index 01h	Name	First interpolation point					Data Structure	-	Data Type	Int32
	Access	RW	Mapping	YES	Related Mode	-	Data Range	-2^{31} to +(2^{31}-1)	Default	0
The interpolation displacement is an absolute displacement reference. 60C1-1h must be set to the synchronous PDO and the transmission type 1 must be used in the interpolation mode. The host controller sends a displacement reference to the slave per synchronization cycle.										

Index 60C2h	Name	Interpolation time period					Data Structure	ARR	Data Type	Uint8
	Access	RW	Mapping	YES	Related Mode	ip	Data Range	OD data range	Default	OD default value
Defines the interpolation cycle in the interpolation mode. The SV660C servo drive supports a synchronization cycle of 1 ms to 20 ms. Synchronization cycles are limited to this range. The synchronization cycle must be set after the servo drive stops.										

Sub-index 00h	Name	Number of entries					Data Structure	-	Data Type	Uint8
	Access	RO	Mapping	NO	Related Mode	-	Data Range	2	Default	2

Sub-index 01h	Name	Interpolation time units					Data Structure	-	Data Type	Uint8
	Access	RW	Mapping	YES	Related Mode	-	Data Range	1 to 20	Default	1
Defines the interpolation time units.										

Sub-index 02h	Name	Interpolation time index					Data Structure	-	Data Type	Int8
	Access	RO	Mapping	TPDO	Related Mode	-	Data Range	-3	Default	-3
Defines the interpolation time index. The value "-3" indicates the time unit is ms. The actual interpolation cycle (ms) therefore is 60C2-01h.										

Index 60C5h	Name	Max. profile acceleration					Data Structure	VAR	Data Type	Uint32
	Access	RW	Mapping	YES	Related Mode	All	Data Range	0 to (2^{32}-1)	Default	2147483647 p/ms

Defines the maximum permissible acceleration in the profile position mode, profile velocity mode, and homing mode.
The setpoint takes effect when the motor runs in the acceleration phase next time.

- The following formula applies when a 23-bit motor needs to run at 400 RPM (6081: 400 x 8388608/60) with acceleration rate being 400 RPM/s (6083: 400 x 8388608/60) and deceleration rate being 200 RPM/s (6084: 200 x 8388608/60) under a gear ratio of 1:1:

  Acceleration time t_{up} = Δ6081/Δ6083 = 1 (s); Deceleration time t_{down} = Δ6081/Δ6084 = 2 (s)

- For 60C5h, the setpoint 0 will be forcibly changed to 1.

Index 60C6h	Name	Max. profile deceleration					Data Structure	VAR	Data Type	Uint32
	Access	RW	Mapping	YES	Related Mode	All	Data Range	0 to (2^{32}-1)	Default	2147483647 p/ms

Defines the maximum allowable acceleration in the deceleration phase of profile position mode, profile velocity mode, and homing mode.
The setpoint takes effect when the motor runs in the deceleration phase next time.

- The following formula applies when a 23-bit motor needs to run at 400 RPM (6081: 400 x 8388608/60) with acceleration rate being 400 RPM/s (6083: 400 x 8388608/60) and deceleration rate being 200 RPM/s (6084: 200 x 8388608/60) under a gear ratio of 1:1:

  Acceleration time t_{up} = Δ6081/Δ6083 = 1 (s); Deceleration time t_{down} = Δ6081/Δ6084 = 2 (s)

- For 60C6h, the setpoint 0 will be forcibly changed to 1.

Index 60F4h	Name	Following error actual value					Data Structure	VAR	Data Type	Int32
	Access	RO	Mapping	TPDO	Related Mode	pp/hm/ip	Data Range	-2^{31} to $+(2^{31}-1)$	Default	-
Indicates the real-time position deviation (in user position unit).										

Index 60FCh	Name	Position demand value					Data Structure	VAR	Data Type	Int32
	Access	RO	Mapping	TPDO	Related Mode	pp/hm/ip	Data Range	-2^{31} to $+(2^{31}-1)$	Default	-

Indicates the real-time motor position reference.

Position demand value (6062h) x Gear ratio (6091h) = Position demand value* (60FCh)

Index 60FDh	Name	Digital input					Data Structure	VAR	Data Type	Uint32
	Access	RO	Mapping	TPDO	Related Mode	All	Data Range	0 to $(2^{32}-1)$	Default	-

Indicates whether current DI logic of the servo drive is valid.

0: Invalid

1: Valid

The DI signal indicated by each bit is described as follows:

31 to 16	15 to 4	3	2	1	0
Manufacturer-defined (undefined)	Reserved	Undefined	Home switch	Positive limit switch	Negative limit switch

Index 60FEh	Name	Digital output					Data Structure	ARR	Data Type	Uint32
	Access	RO	Mapping	TPDO	Related Mode	All	Data Range	OD data range	Default	OD default value

Indicates whether current DO logic of the servo drive is valid.

Sub-index 00h	Name	Number of entries					Data Structure	-	Data Type	Uint8
	Access	RO	Mapping	NO	Related Mode	-	Data Range	1	Default	1

Sub-index 01h	Name	Physical outputs					Data Structure	-	Data Type	Uint32
	Access	RO	Mapping	TPDO	Related Mode	-	Data Range	0 to $(2^{32}-1)$	Default	0

Indicates whether current DO logic of the servo drive is valid.

0: Invalid

1: Valid

The DO signal indicated by each bit is described as follows:

31 to 16	15 to 1	0
Manufacturer-defined (undefined)	Reserved	Brake output

Index 60FFh	Name	Target velocity					Data Structure	VAR	Data Type	Int32
	Access	RW	Mapping	YES	Related Mode	pv	Data Range	-2^{31} to $+(2^{31}-1)$	Default	0 RPM

Defines the user velocity reference in the profile velocity mode.

7 Application Cases

Application cases are presented by taking the position mode as an example. For details on the position mode, see *"4.9 Profile Velocity Mode"*

In the position mode, objects used as PDOs are allocated as follows:

Table 7-1 PDO mapping allocation

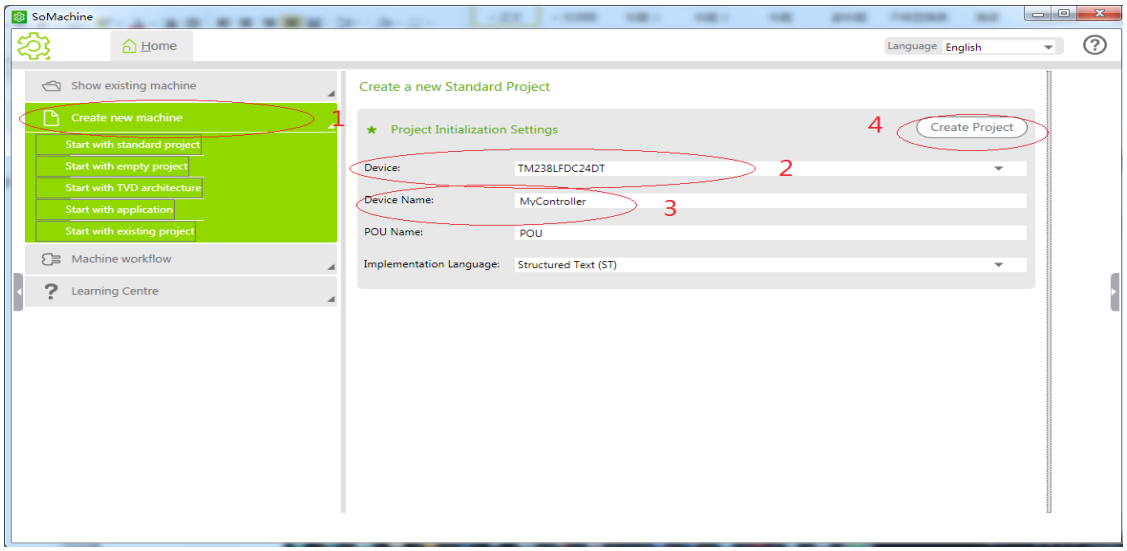
PDO	Object	Meaning	Bit Length
RPDO1	6040h-00h	Command word	Uint16
	6060h-00h	Modes of operation	Int8
RPDO2	6081h-00h	Profile velocity	Uint32
	607Ah-00h	Target position	Int32
TPDO1	6041h-00h	Status word	Uint16
	6061h-00h	Modes of operation display	Int8
TPDO2	606Ch-00h	Velocity actual value	Int32
	6064h-00h	Position actual value	Int32
TPDO3	200Bh-19h	Phase current actual value	Uint16

Write 6083h (Profile acceleration), 6084h (Profile deceleration), and 605Ah (Quick stop option code) for the SDO.

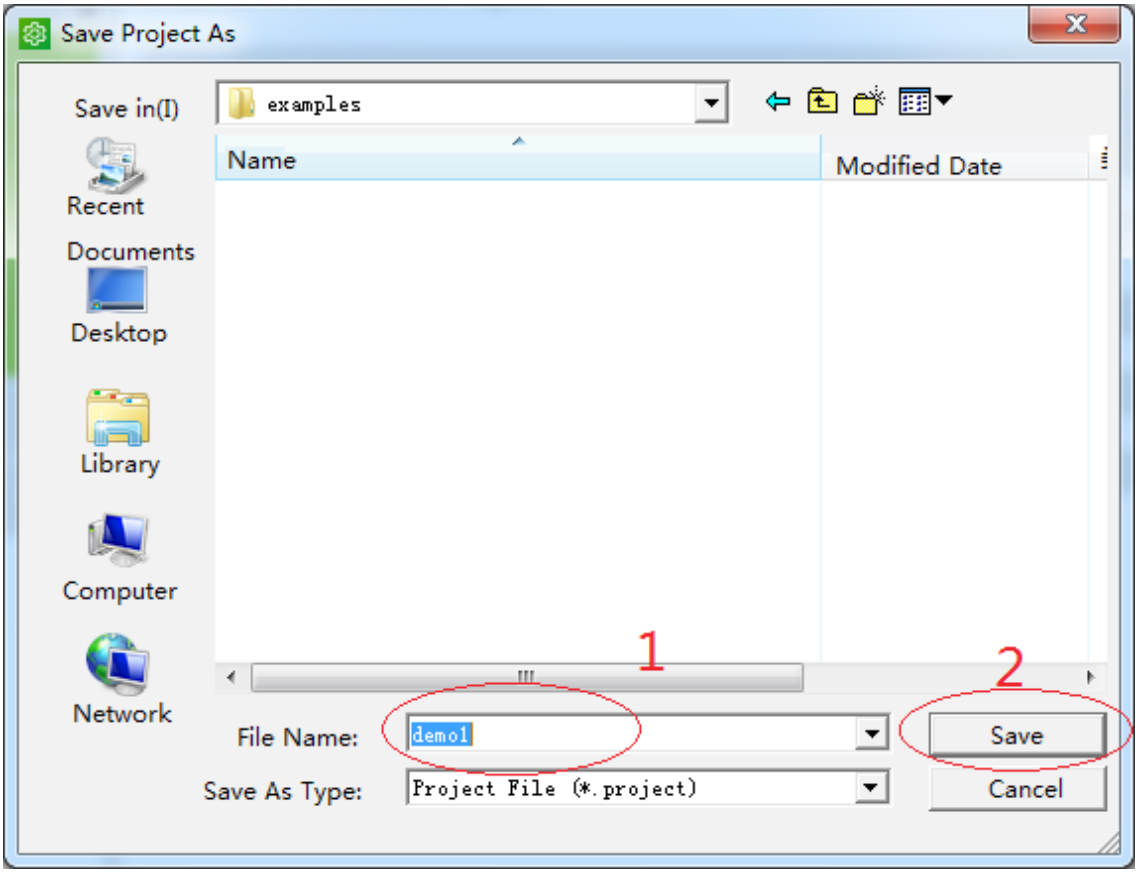
7.1 Connecting SV660C Servo Drive to Schneider 3S Master

SoMachine is the software tool of Schneider 3S series master. This section describes how to connect the SV660C servo drive to Schneider M238 master.

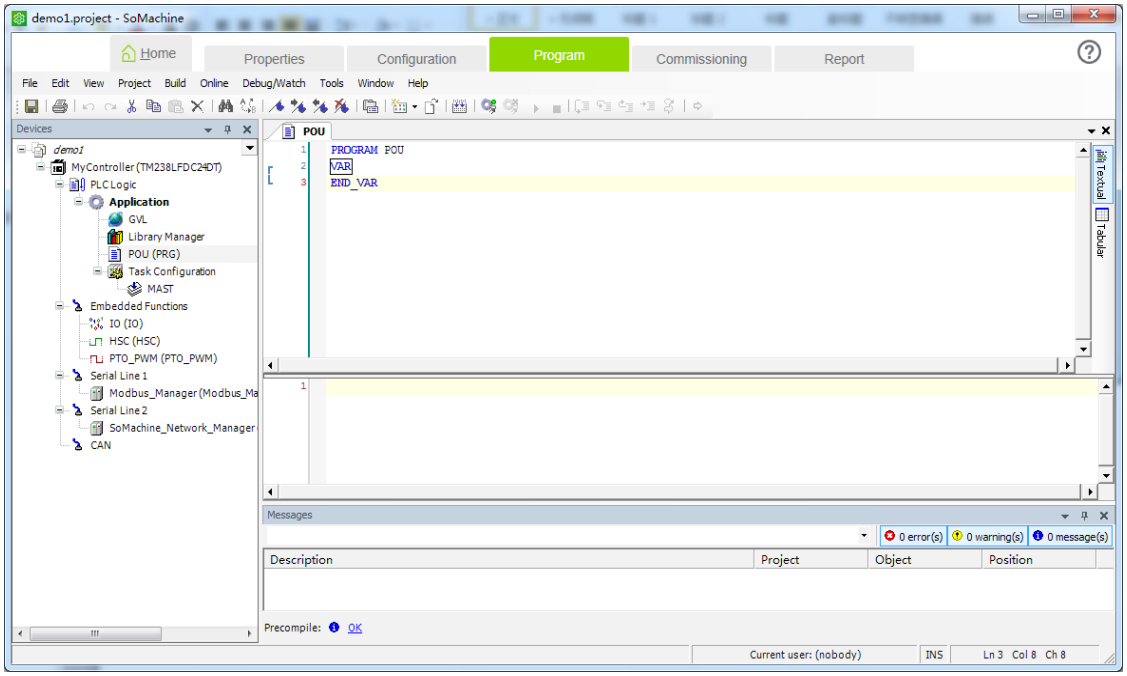
1) Start SoMachine and click **Create new machine** based on a standard project. Select a master device, for example, TM238LFDC24DT, modify the device name, and click **Create Project**, as shown below.



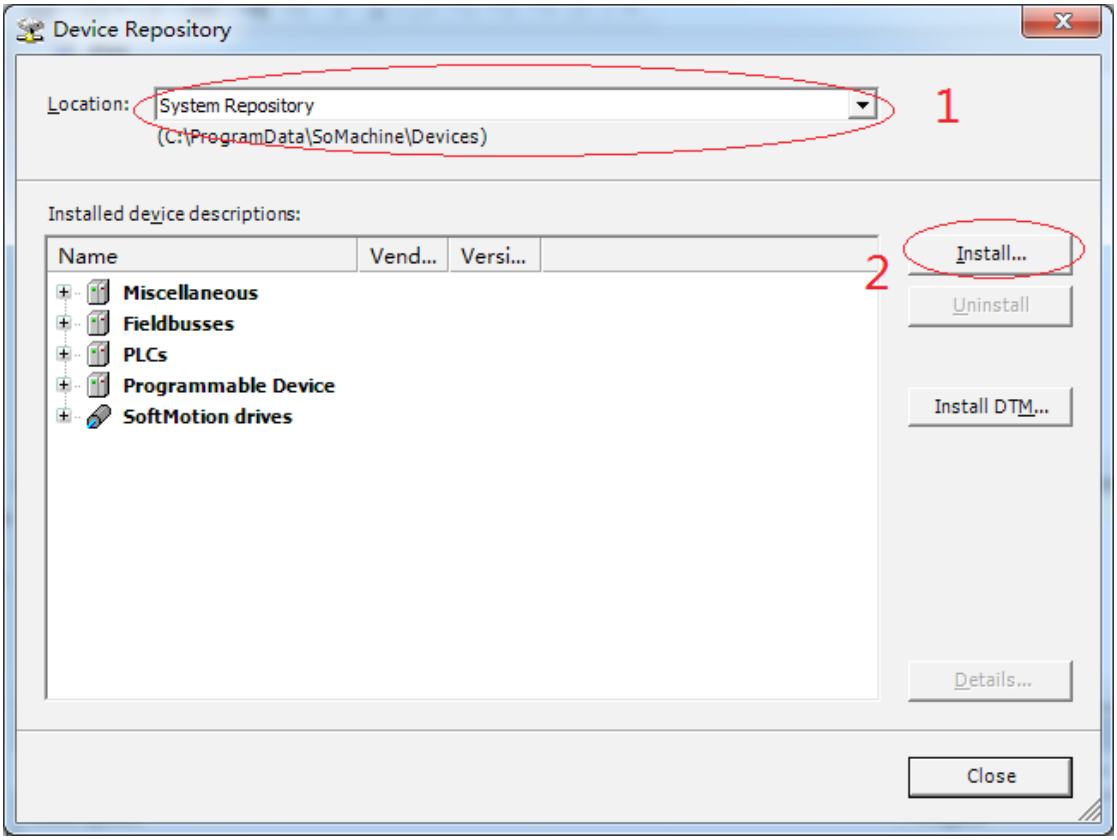
2) Input the file name and click **Save** in the dialog box popped out.



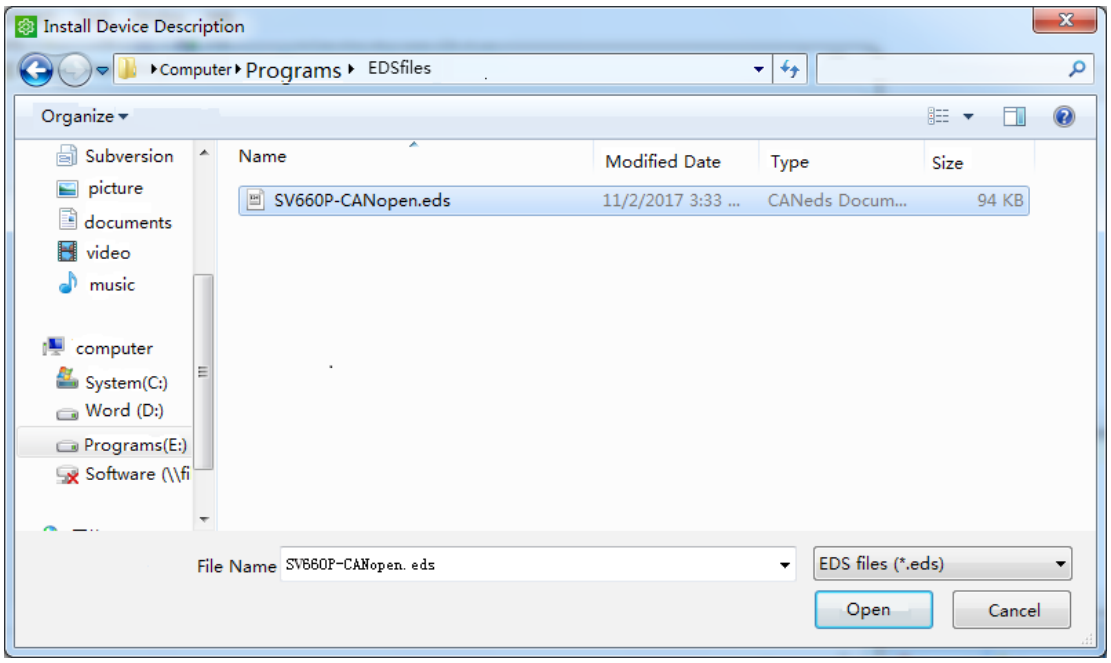
3) The following interface appears.



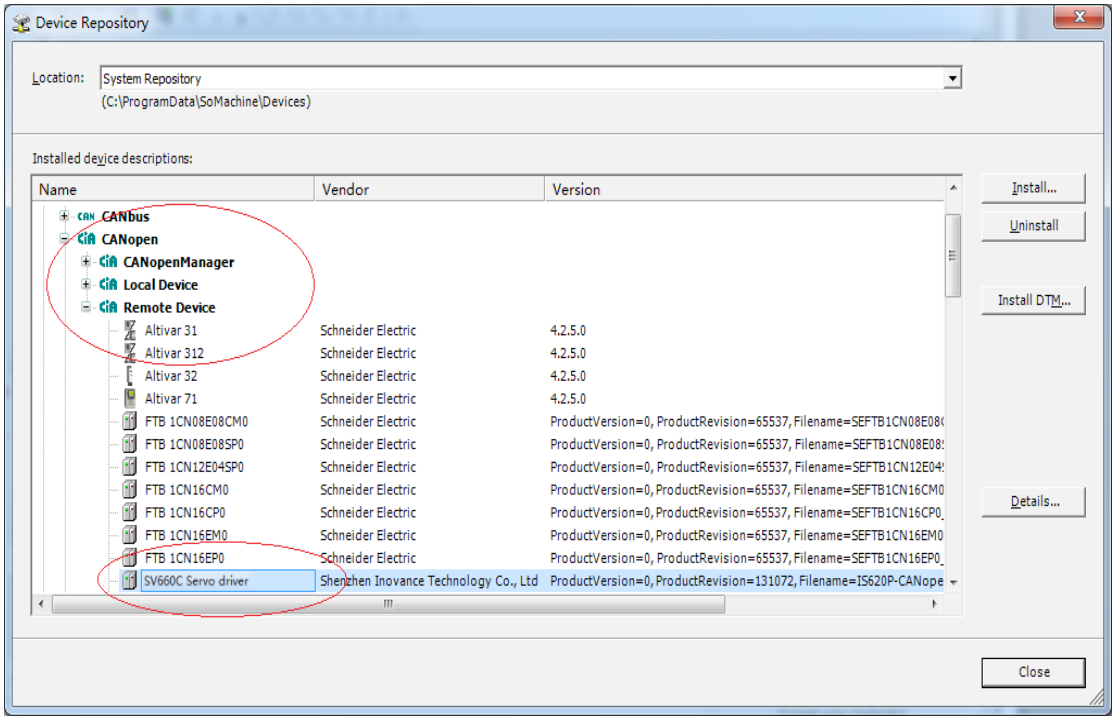
4) Select **Tool** > **Device Repository** to display the following dialog box. (If the EDS file is imported, skip steps 4 to 6.)



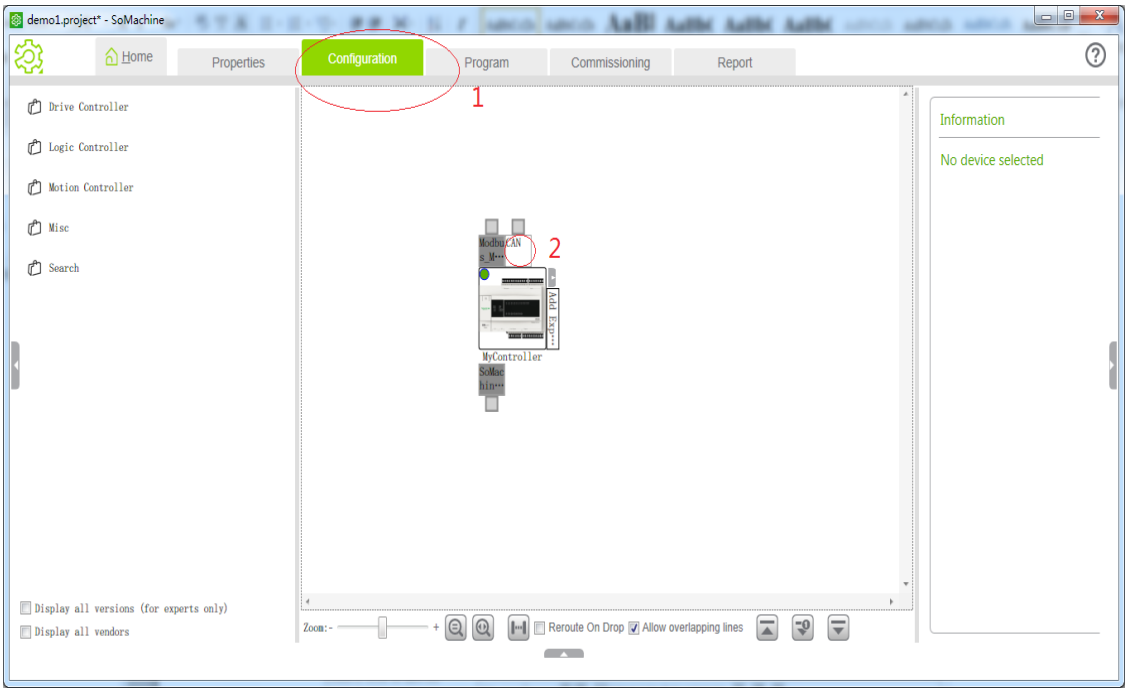
5) As shown in the preceding interface, select **System Repository** and click **Install**. Select a directory for saving the EDS file, as shown below.



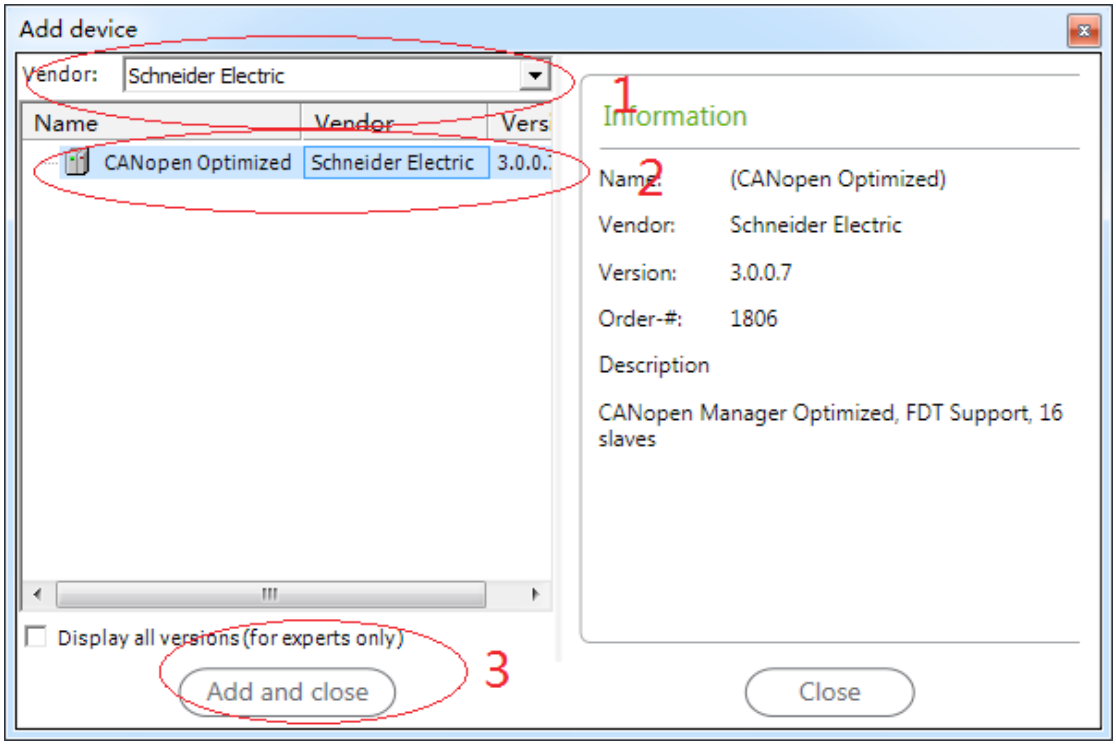
6) As shown in the preceding interface, click **Open** and the EDS file is imported into SoMachine. In the **Device Repository** dialog box, you can select **CANopen** > **Remote Device** to view devices, as shown below.



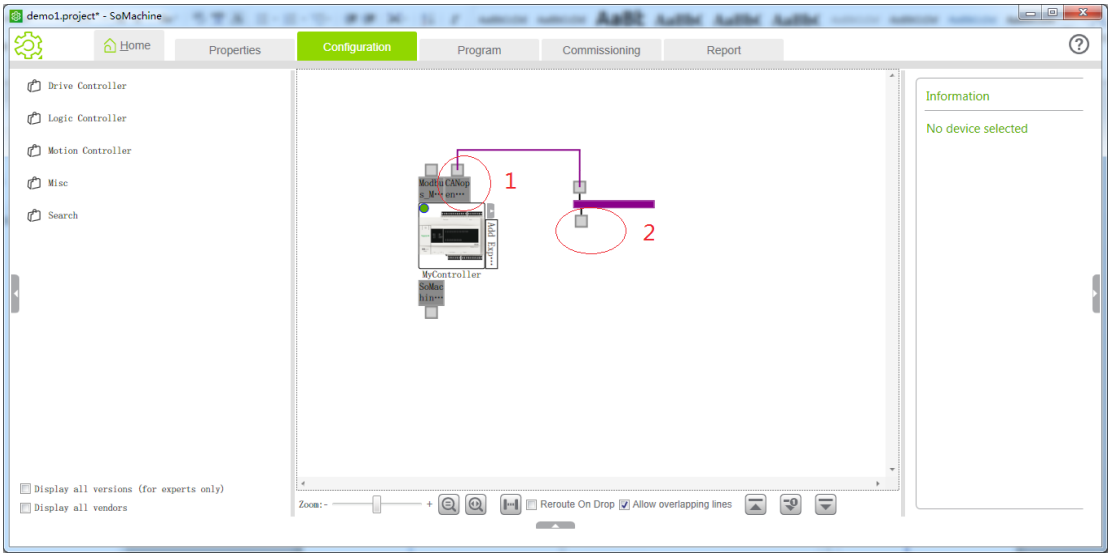
7) Close the preceding dialog box and click **Configuration**. In the interface displayed, only M238 master is available. Click **CAN** in the master.



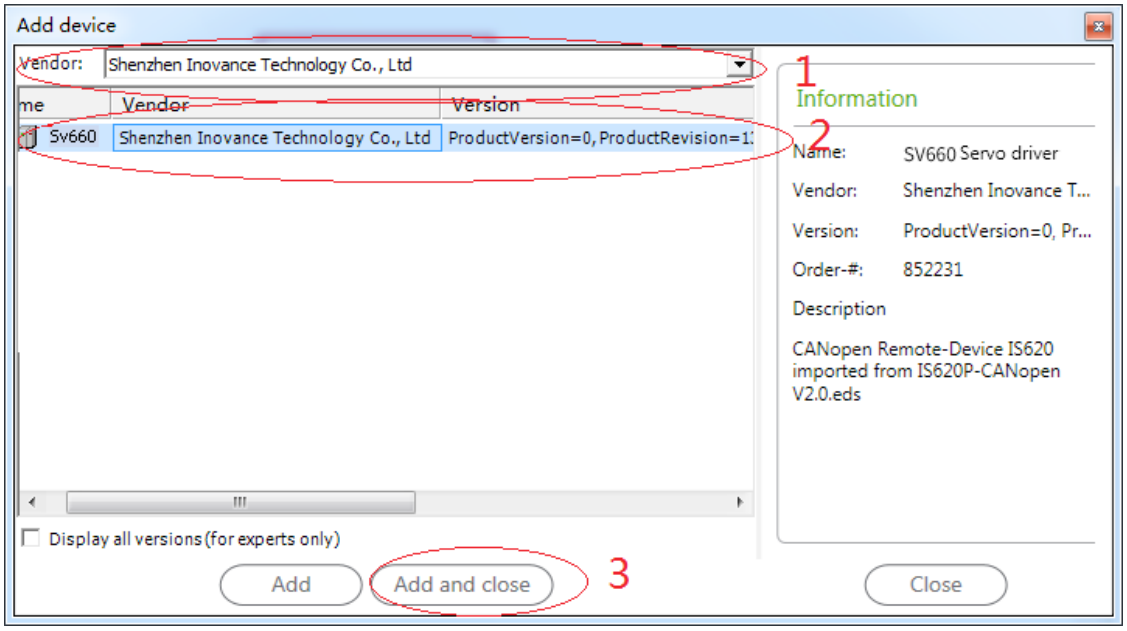
8) The **Add device** dialog box pops out. Add the CANopen gateway and select **Schneider Electric** as the vendor. Then, select **CANopen Optimized** and click **Add and close**.



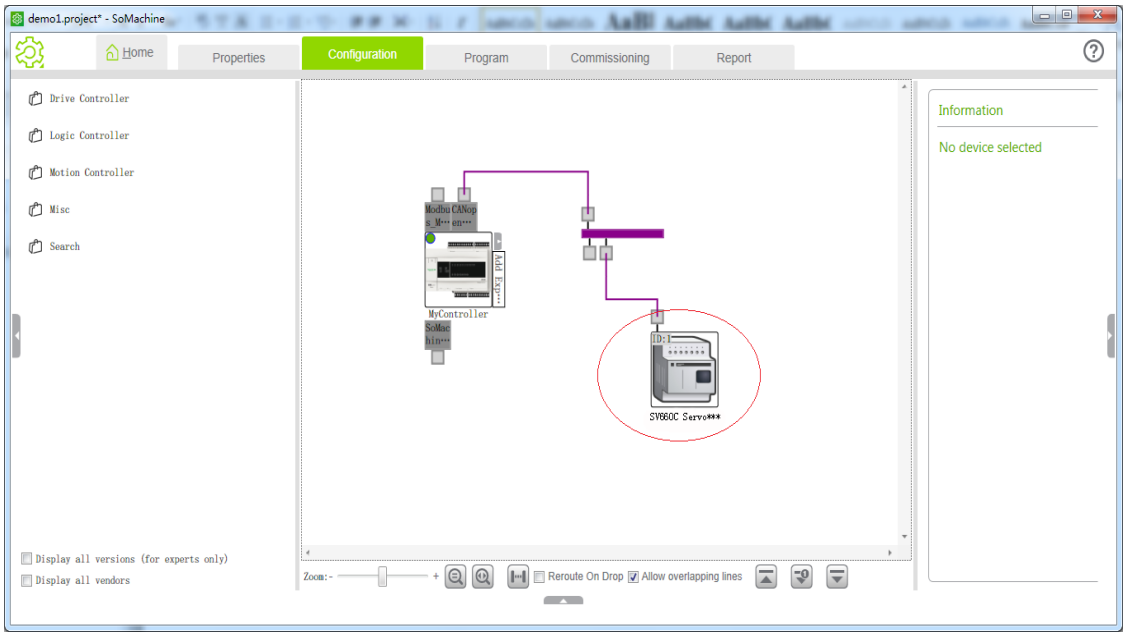
9) Now, the CANopen gateway appears in the interface. Click the position indicated by **2**.



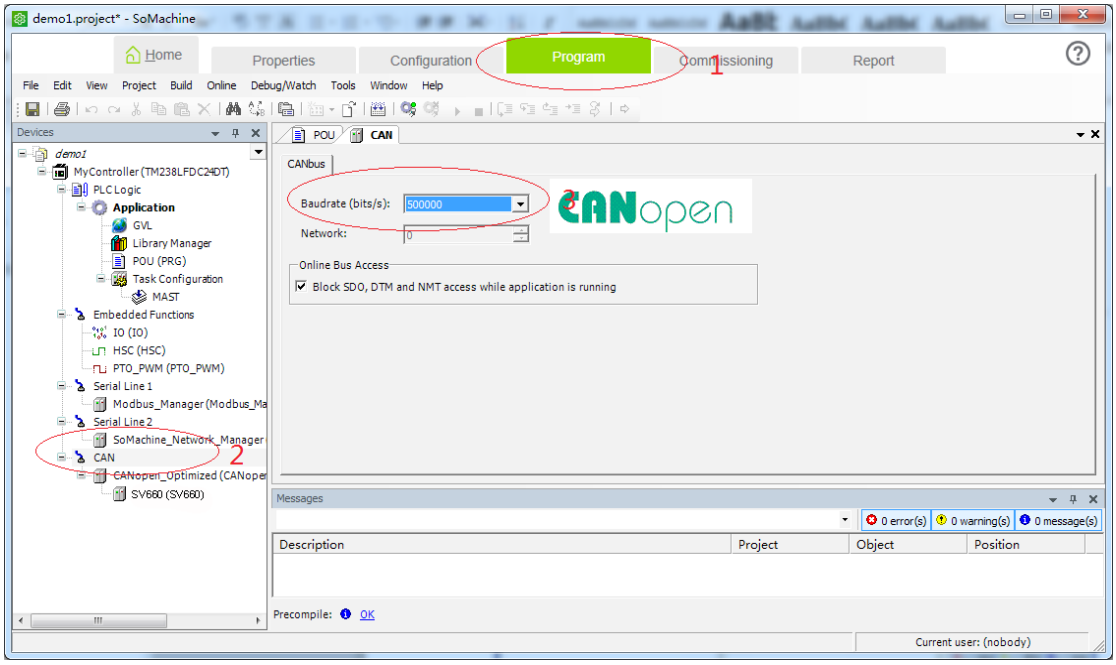
10) The **Add device** dialog box pops out again. Select **Shenzhen Inovance Technology Co., Ltd** as the vendor and **SV660** as the device, and then click **Add and close**.



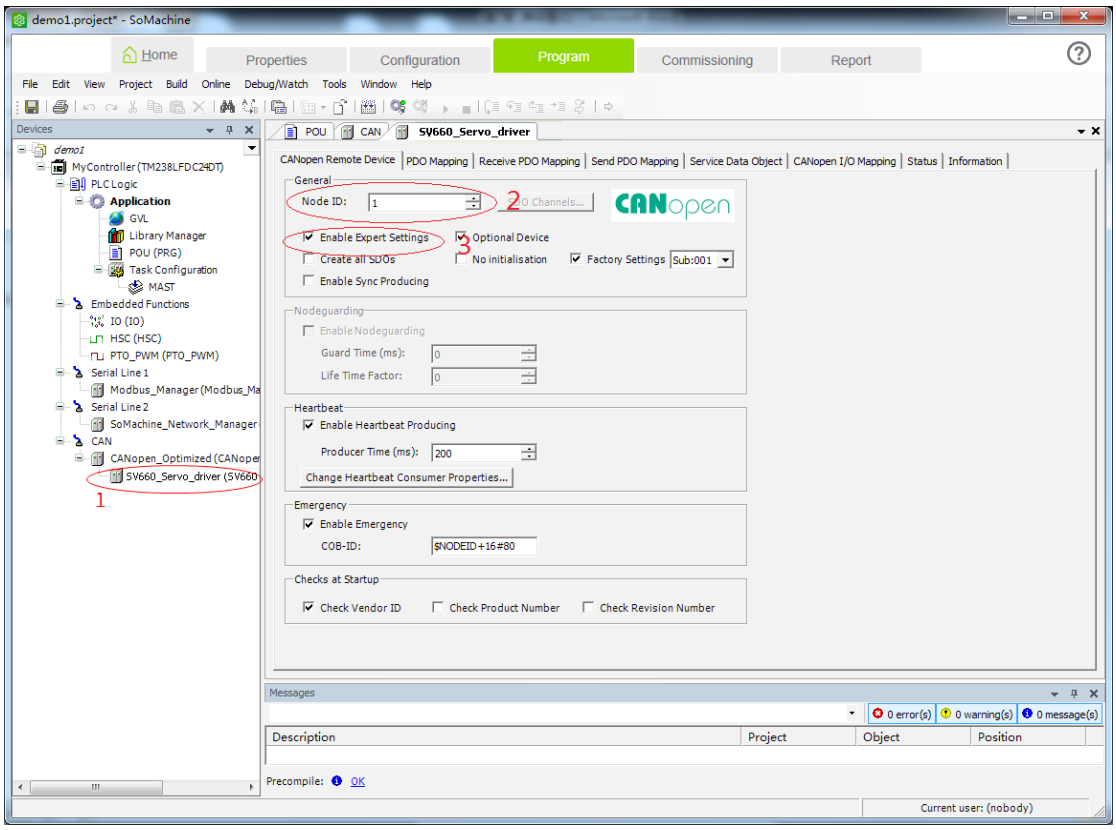
11) Now, the SV660C servo drive appears in the interface.



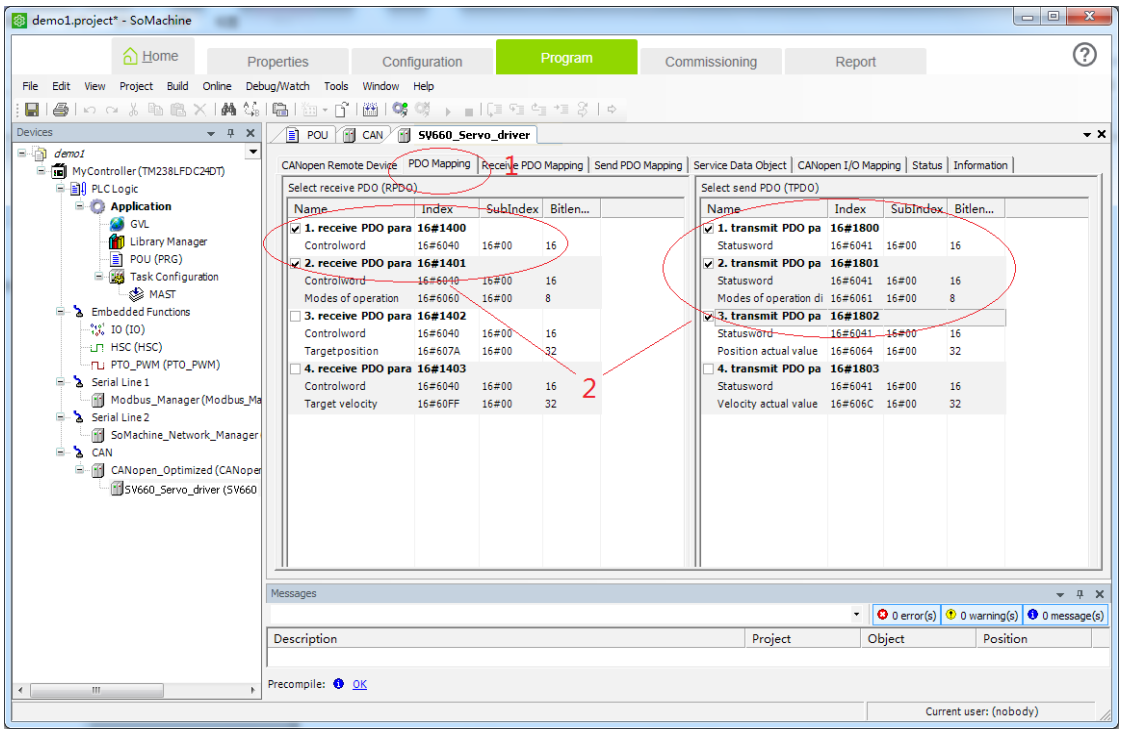
12) Click **Program** and double-click **CAN** on the left to select a proper baud rate. 500 Kbps is selected here.



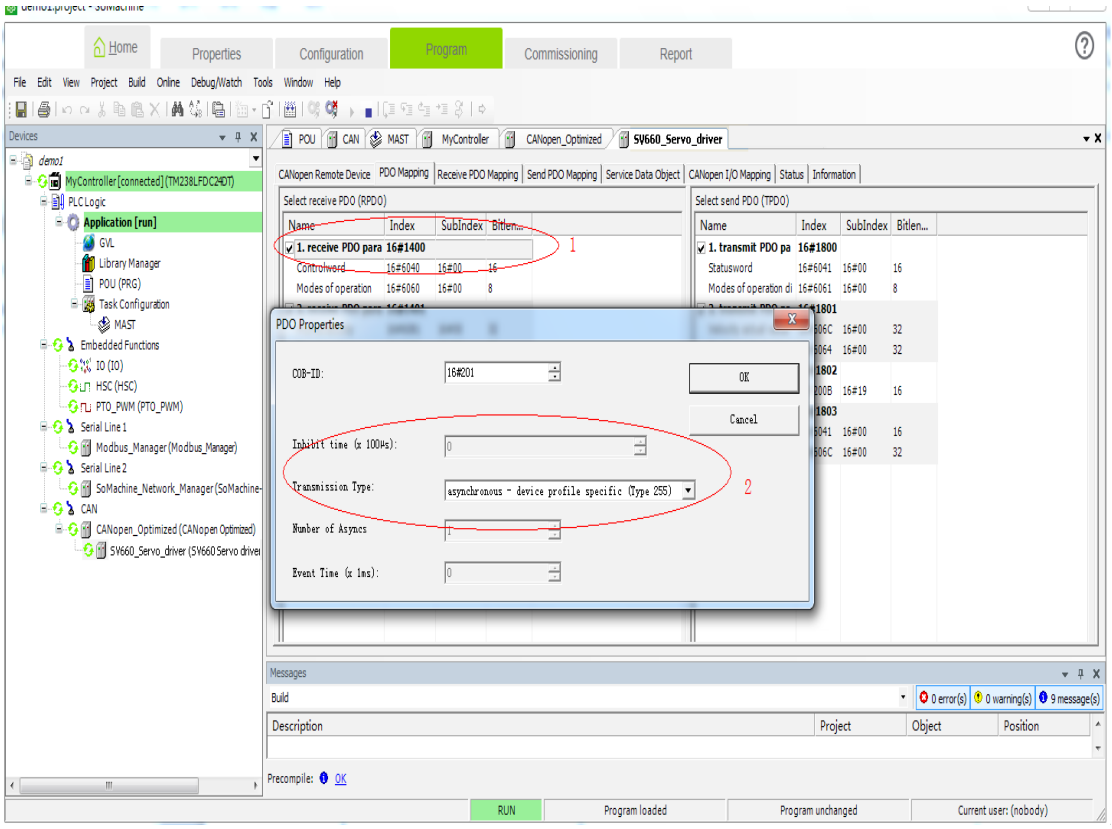
13) To modify the node station No., double-click **SV660_Servo_Driver** on the left. Select **Enable Expert Settings**.



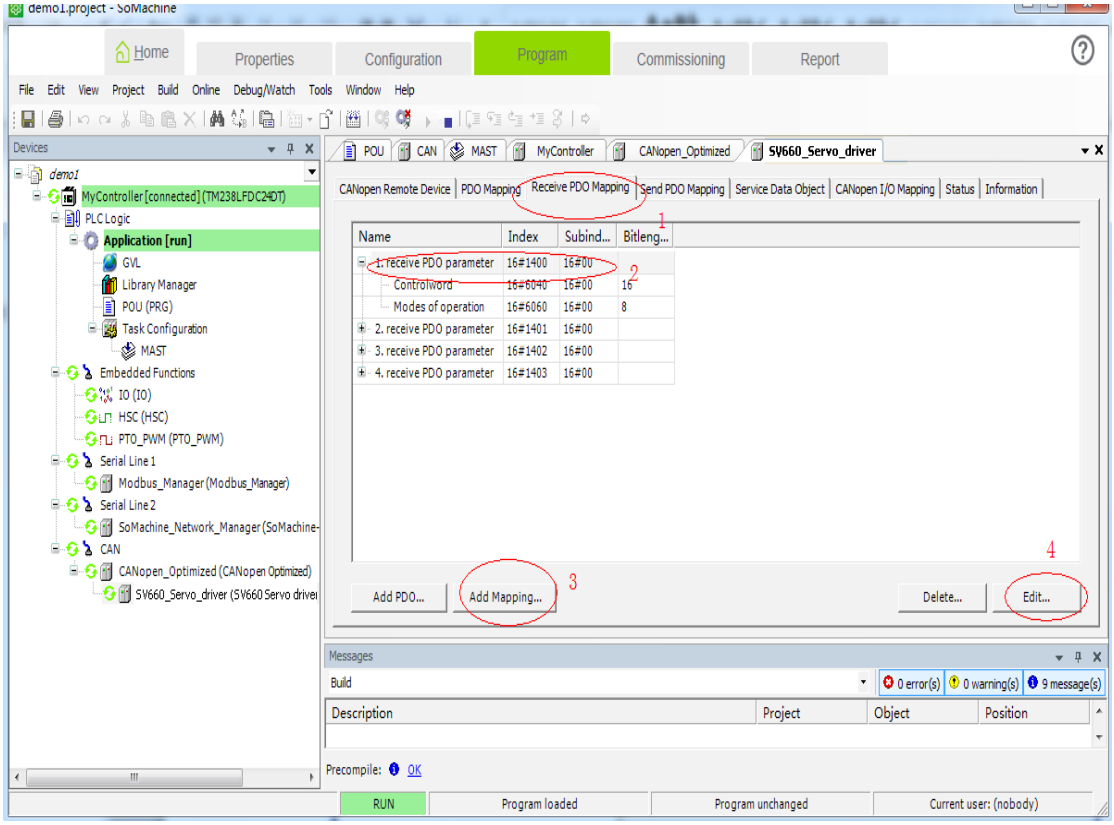
14) Click **PDO Mapping** and check two RPDOs and three TPDOs.



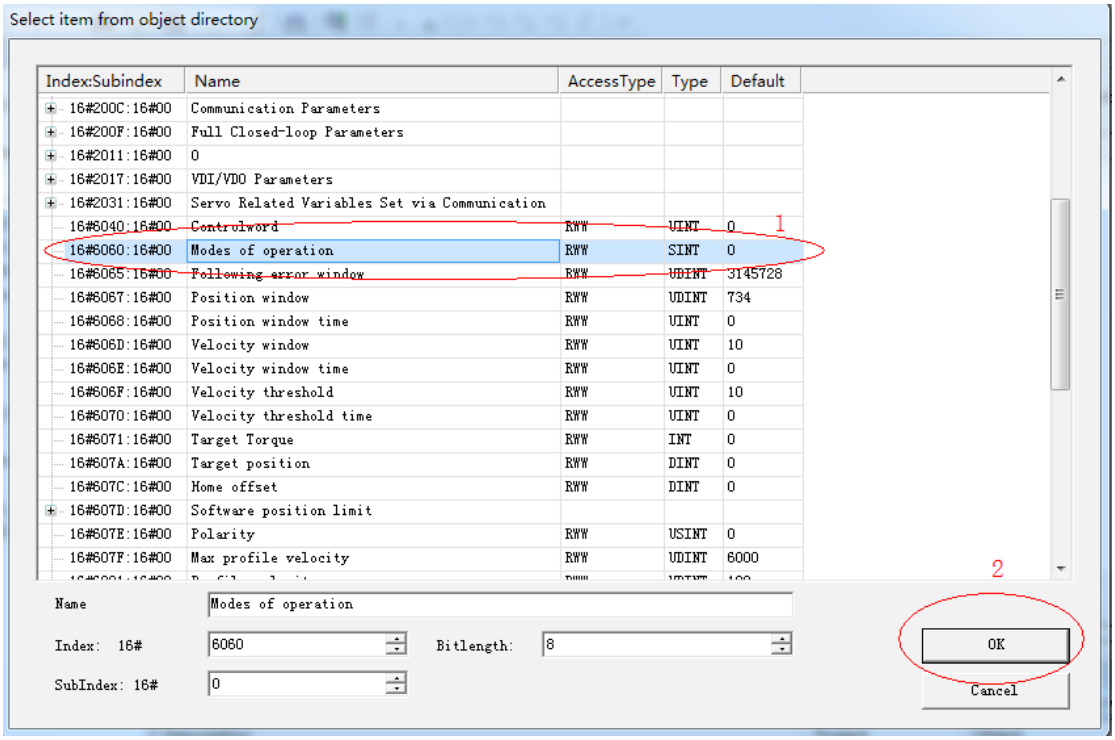
15) Double-click **1. receive PDO para**. In the **PDO Properties** dialog box popped out, modify the transmission type to Type 255. The similar procedure applies to other PDOs.



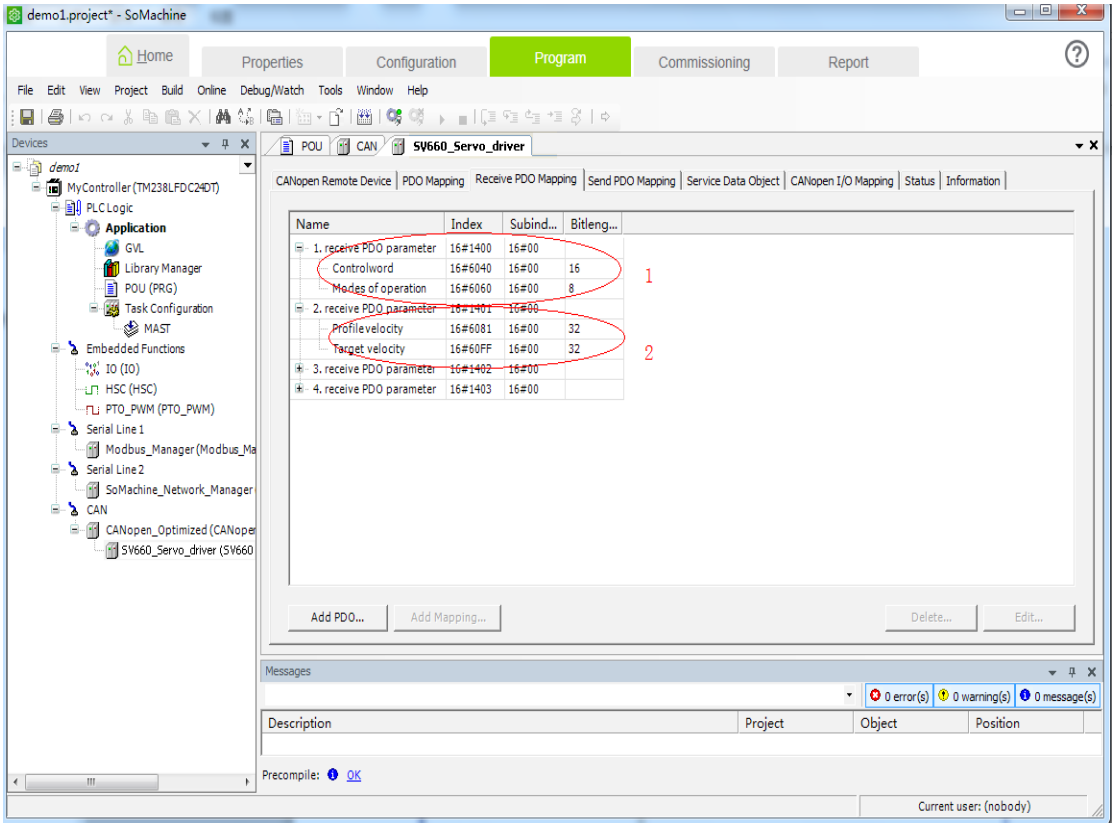
16) Select **Receive PDO Mapping** and click **1. receive PDO parameter**. Click **Add Mapping** or select a mapping and click **Edit**.



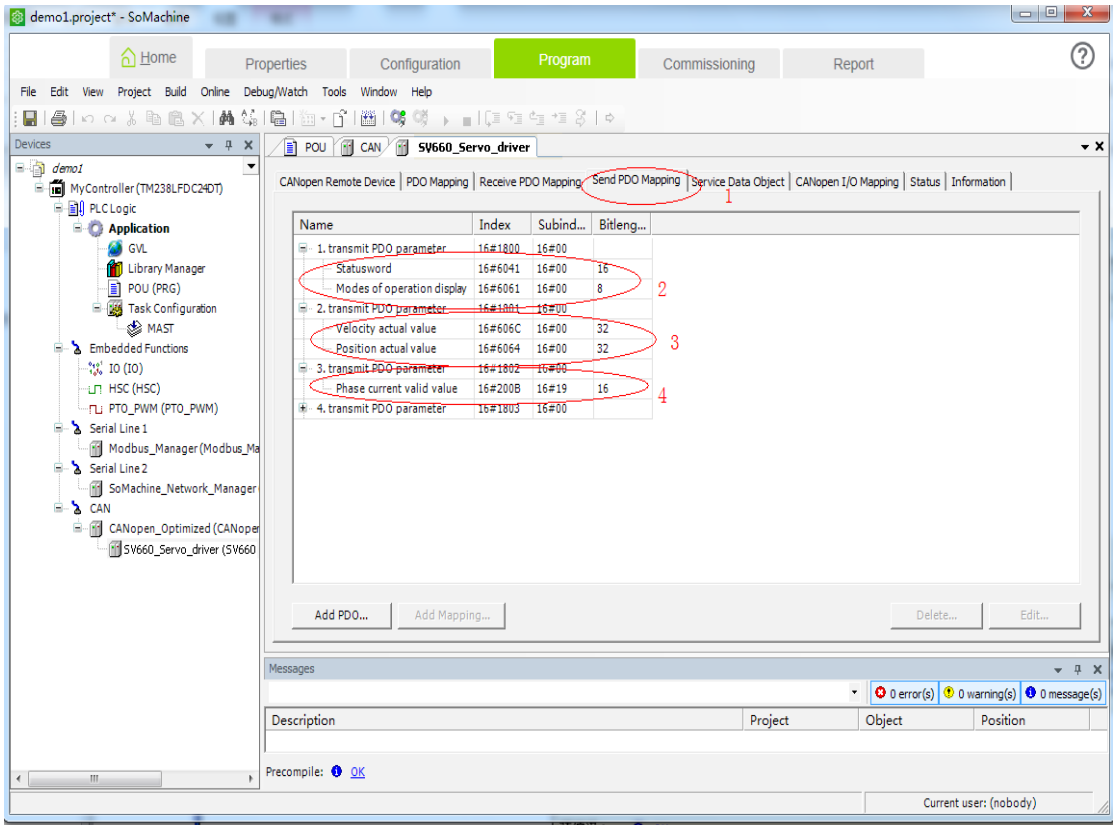
17) A dialog box pops out. Select a proper mapping object according to *"Table 7-1 PDO mapping allocation"*.



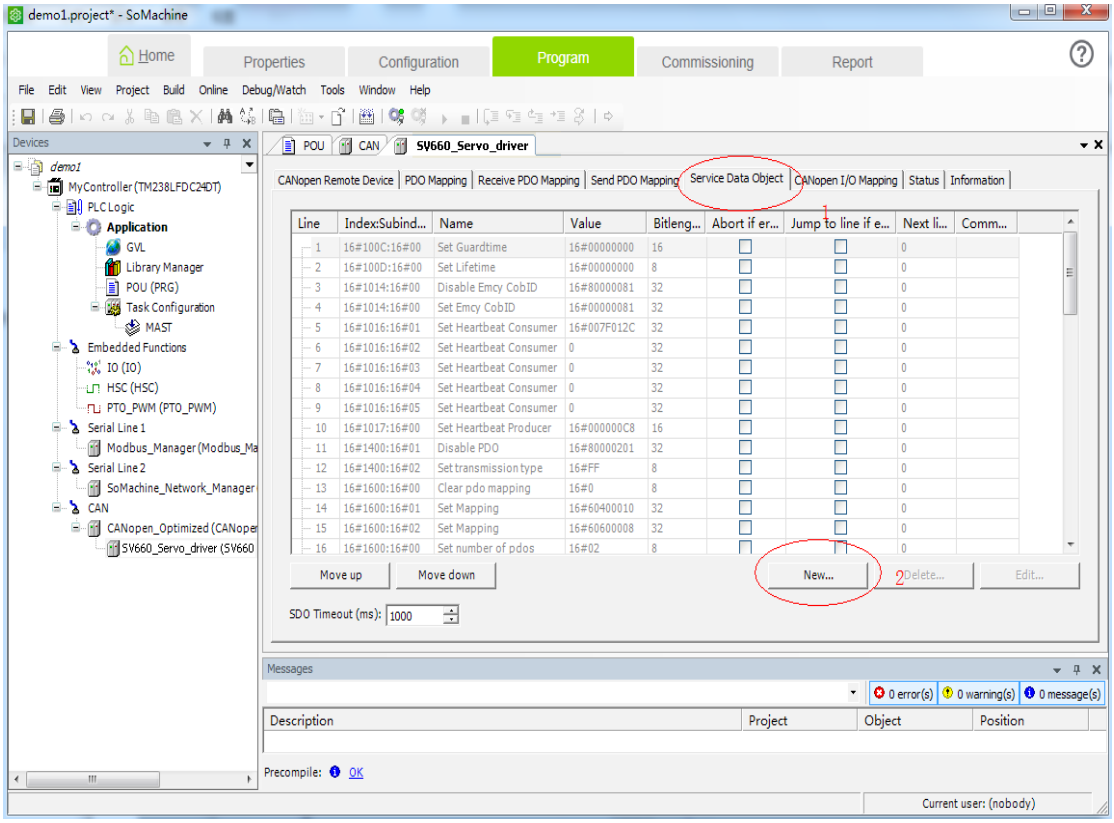
18) After the mapping is added, the RPDO mapping is as follows.



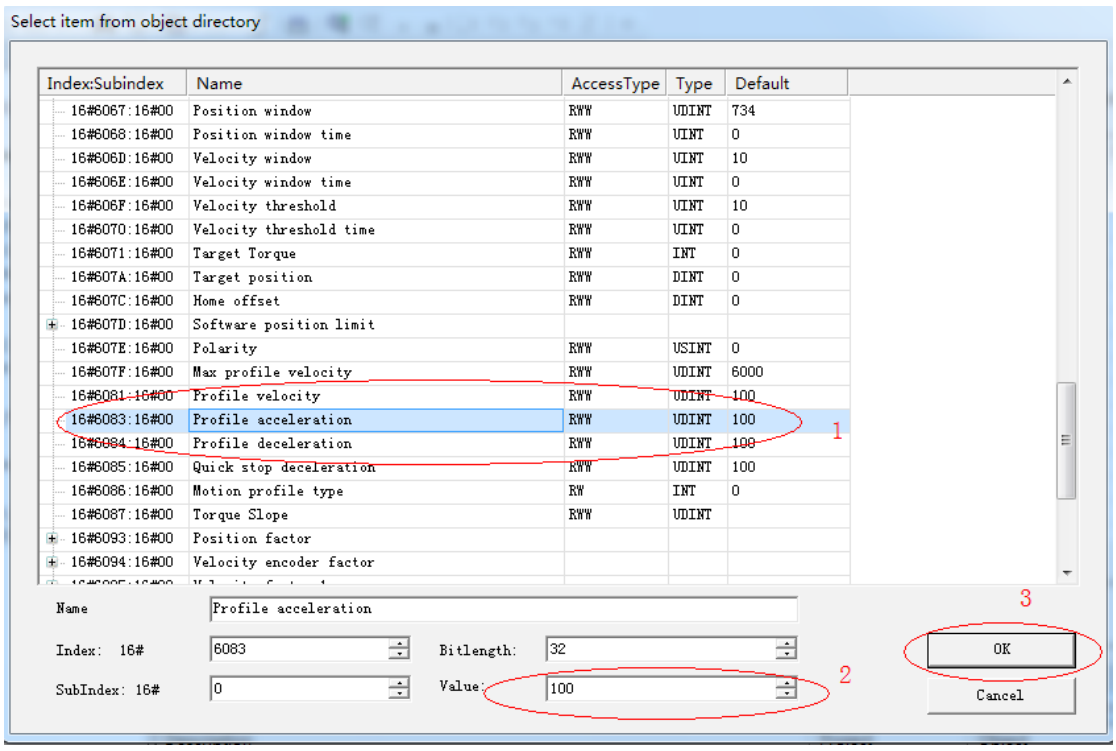
19) Similarly, click **Send PDO Mapping** and perform configuration according to *"Table 7-1 PDO mapping allocation"*. Configurations are shown as follows.



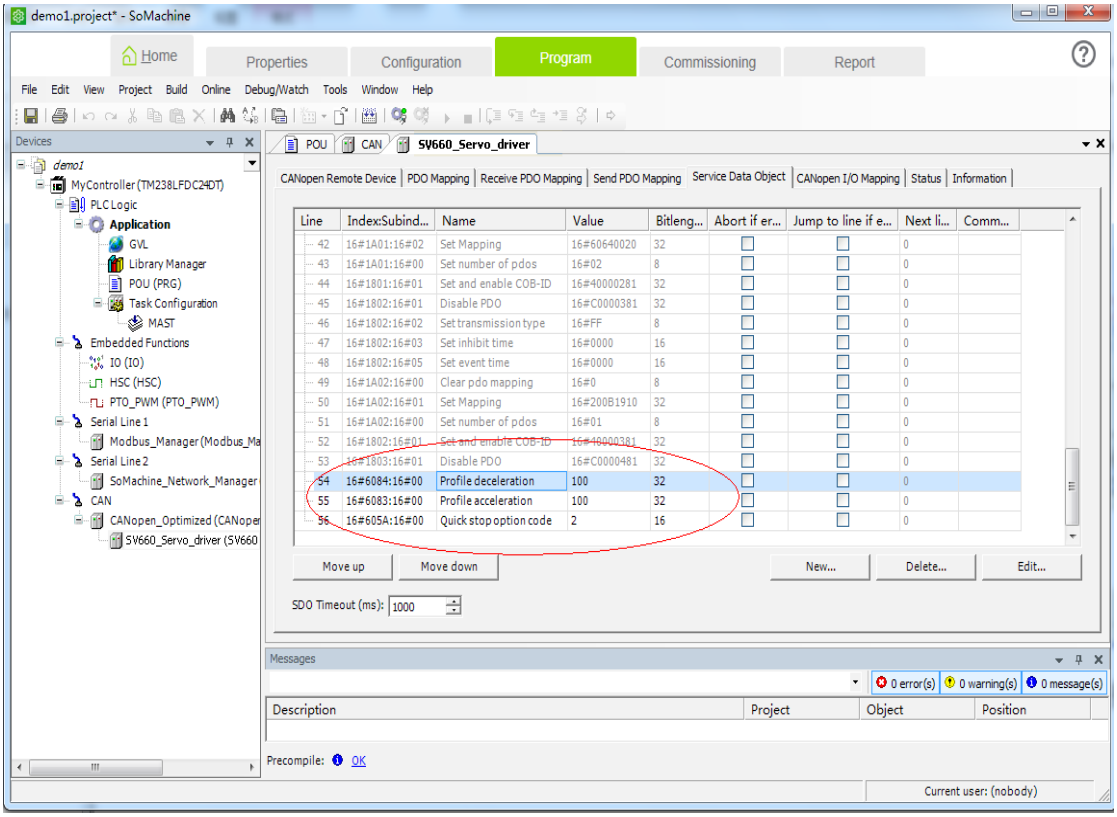
20) Click **Service Data Object** and **New** to add the SDO needed (optional, if default values are used, steps 20 to 22 can be omitted).



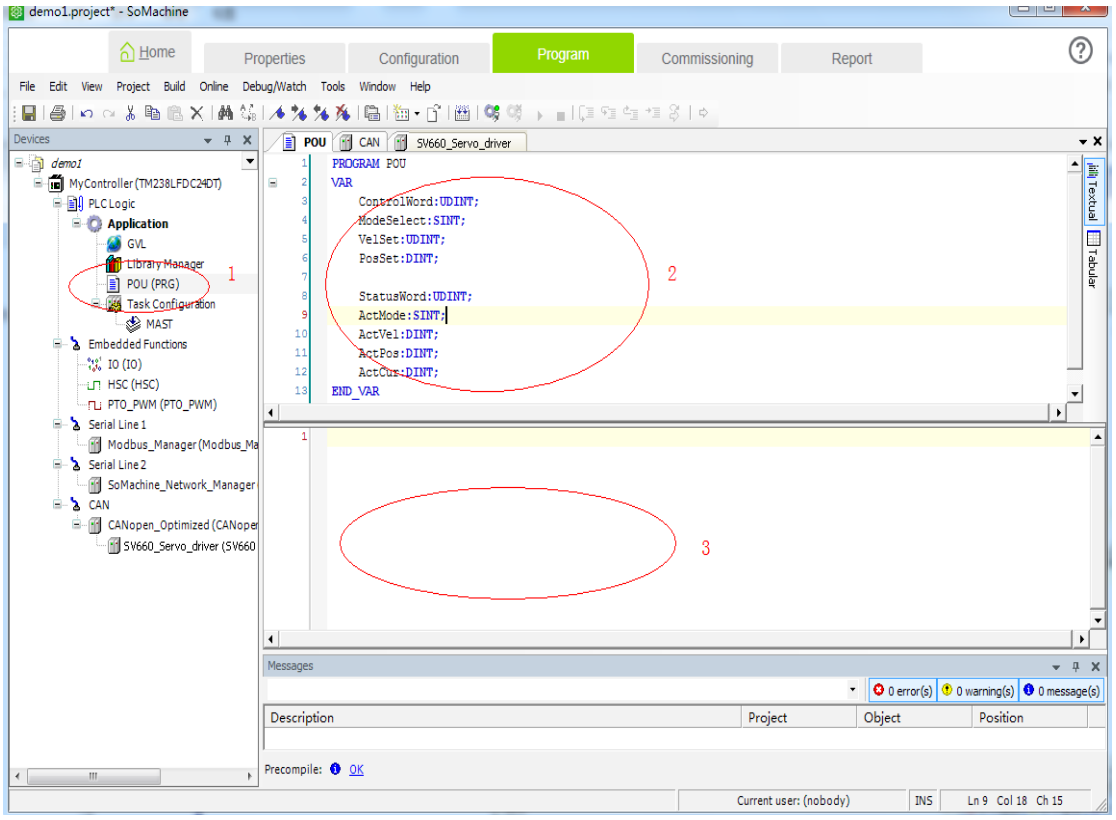
21) Select the corresponding SDO in the list. You can modify the value and click **OK** (optional).



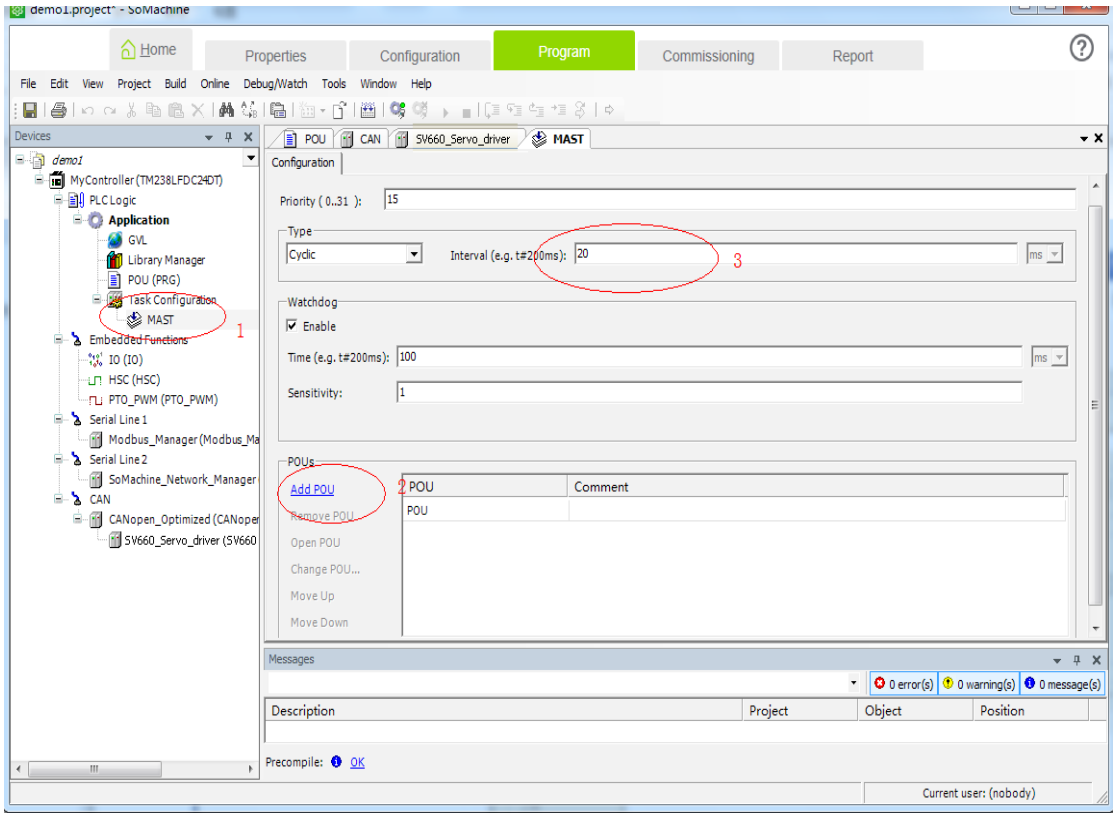
22) The newly added SDO is shown as below (optional).



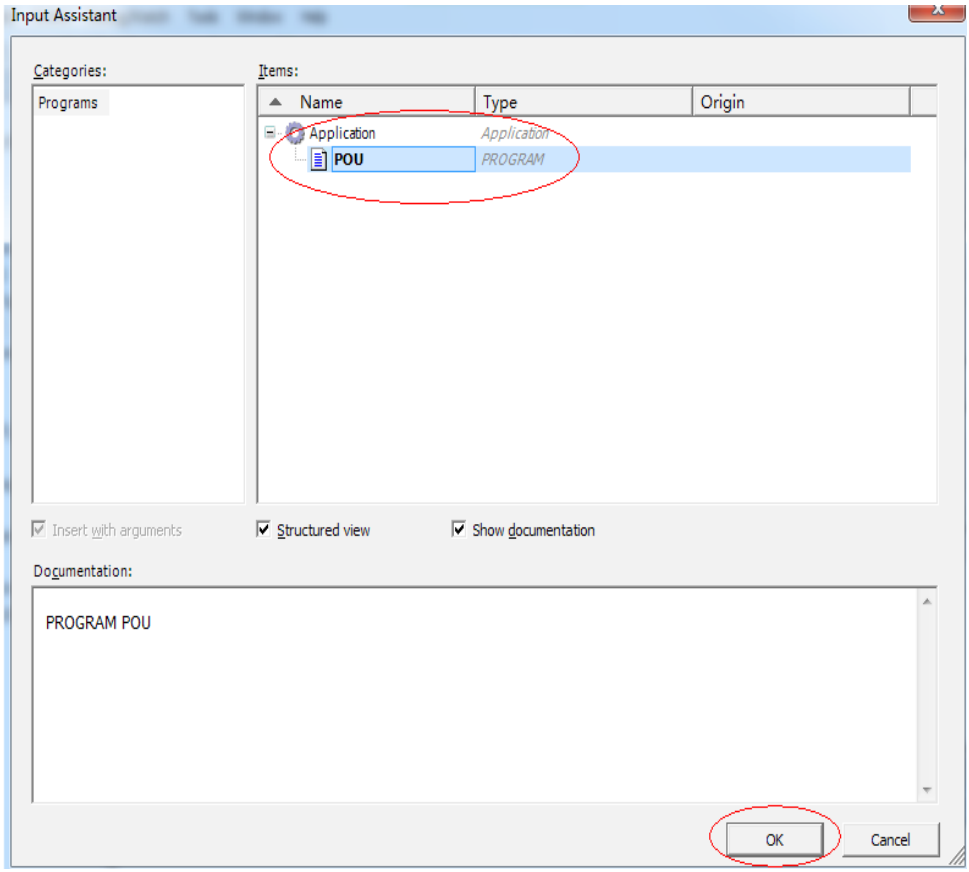
23) Double-click **POU (PRG)** on the left, add the variable definition to the position marked by **2**, and add the PLC program logic to the position marked by **3**. Finally, click **Edit** or press **F11**. If no error occurs, process to the next step.



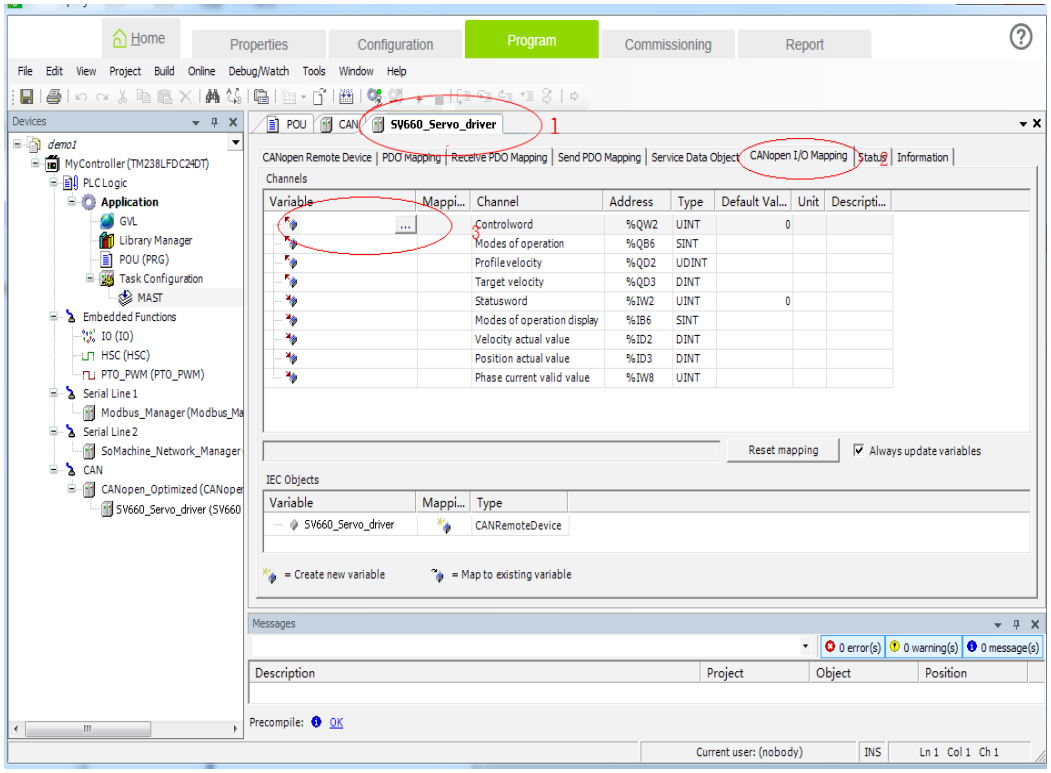
24) Double-click **MAST** and add the POU. You can also set the program cycle interval at the same time.



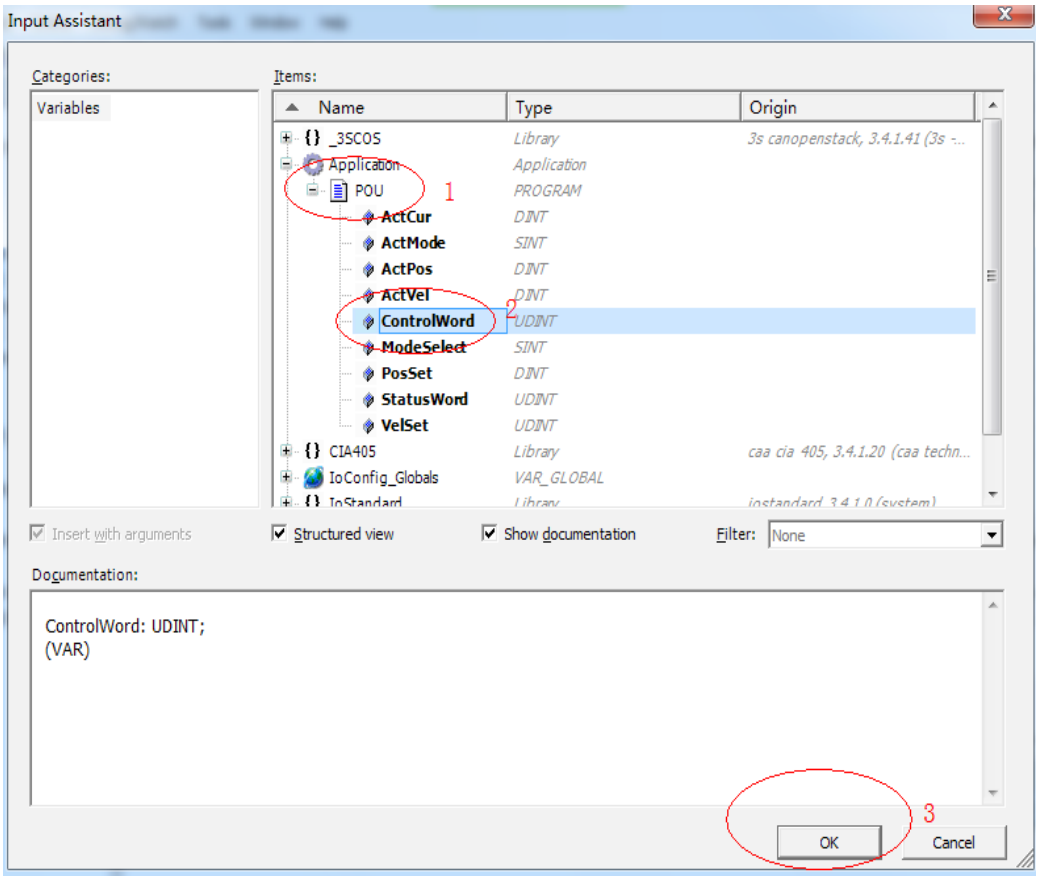
25) Select the POU added based on the following dialog box and click **OK**.



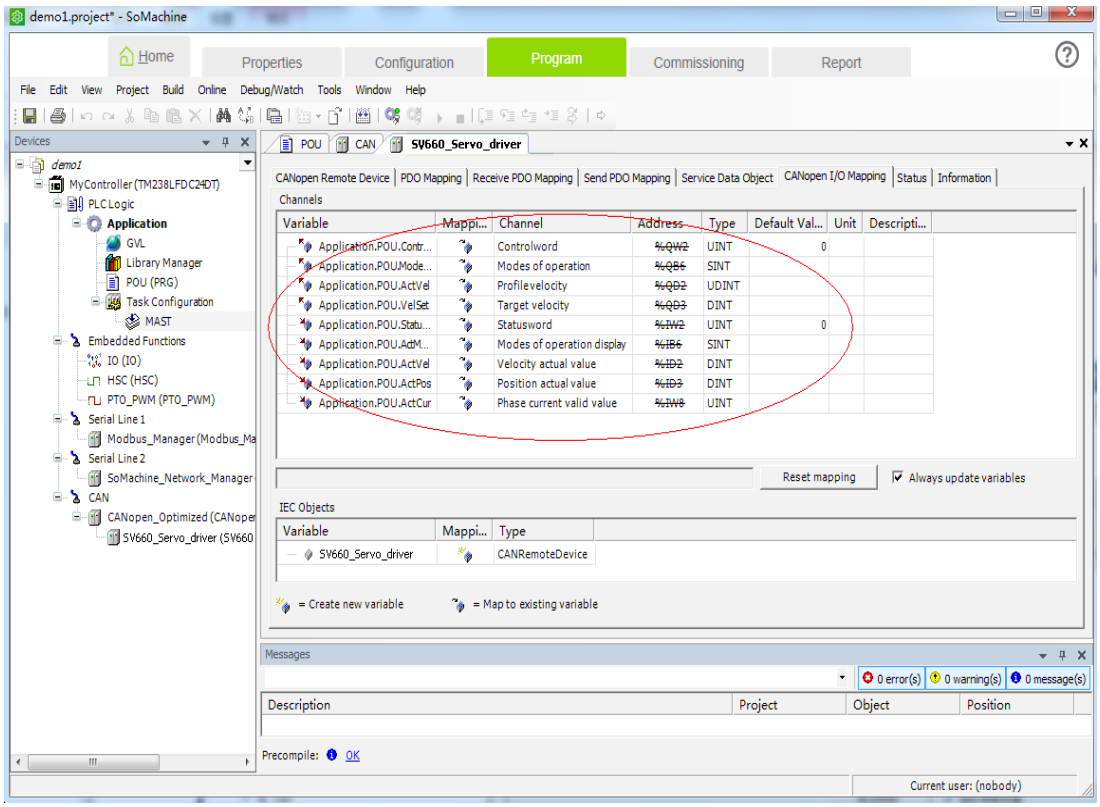
26) Select **CANopen I/O Mapping** under **SV660C_Servo_driver** and double-click the variable to display the **...** button, and then click the **...** button.



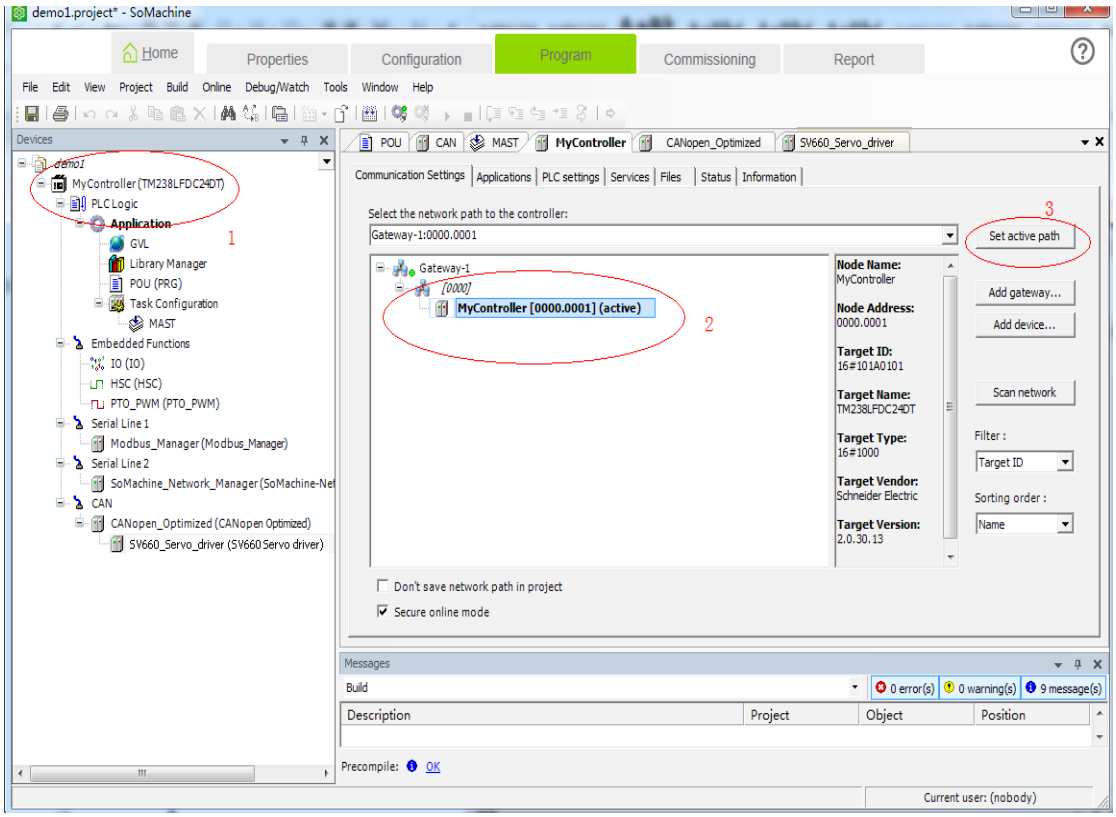
27) Select the PLC-defined variable based on the following steps.



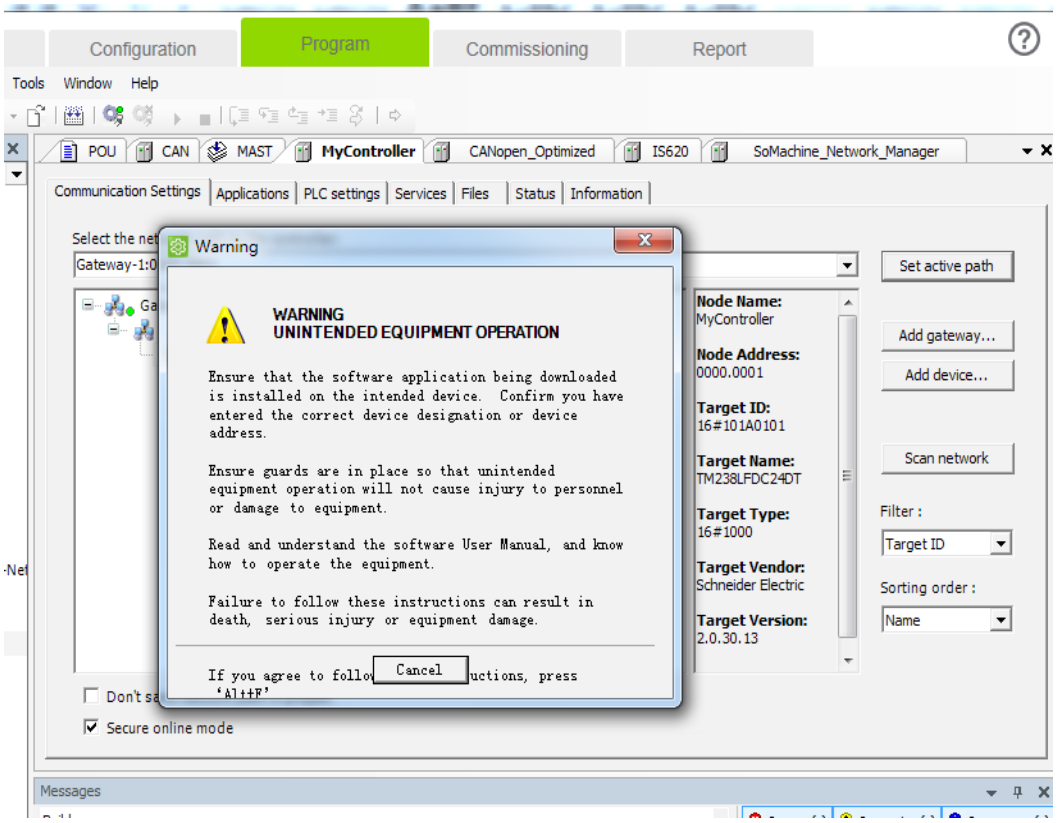
28) Add other variables in the similar way, and the mapping is shown below.



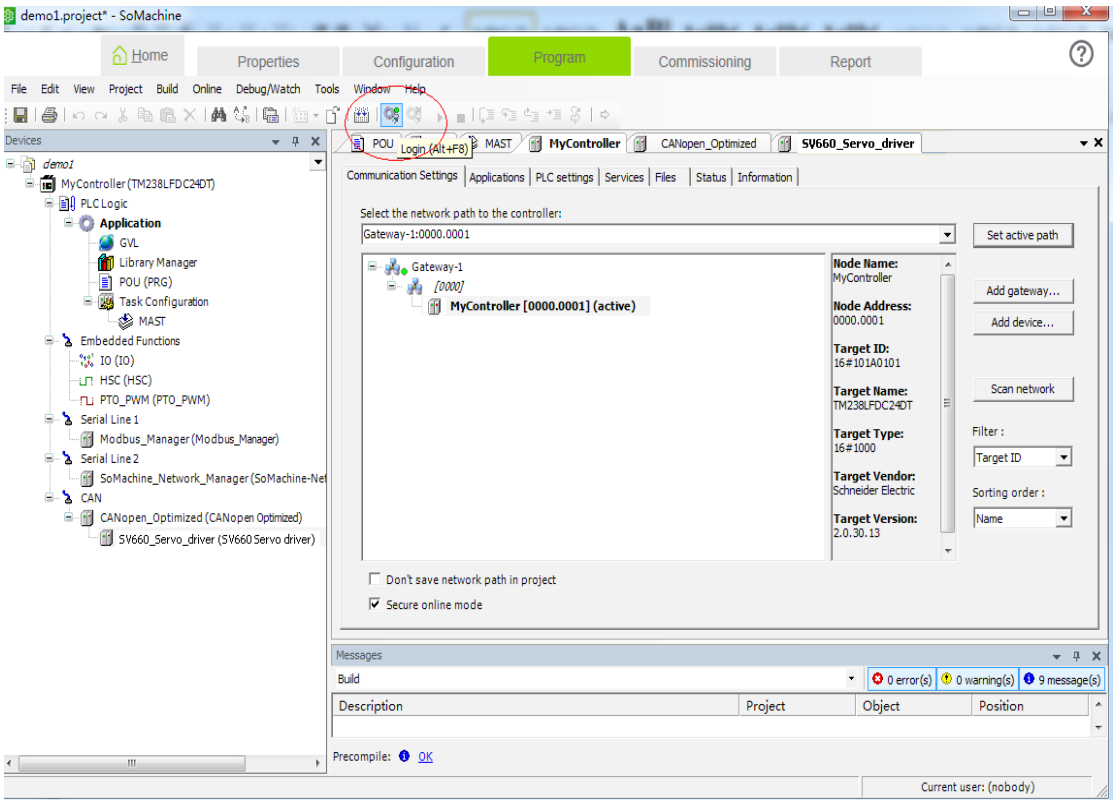
29) Double-click the master name on the left. Select **MyController** and click **Set active path** on the right.



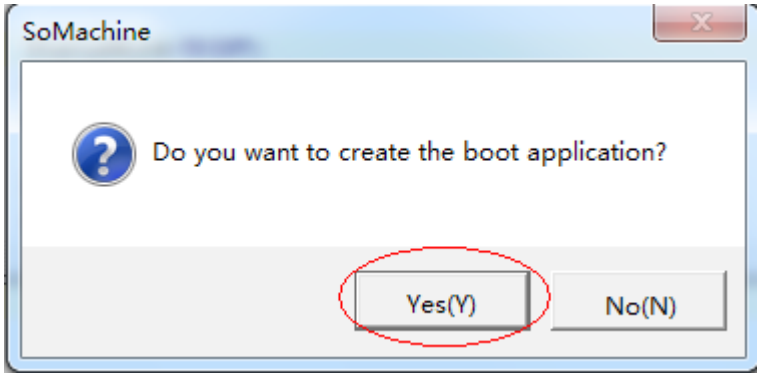
30) The following warning pops out. Press **Alt+F** according to the instructions.



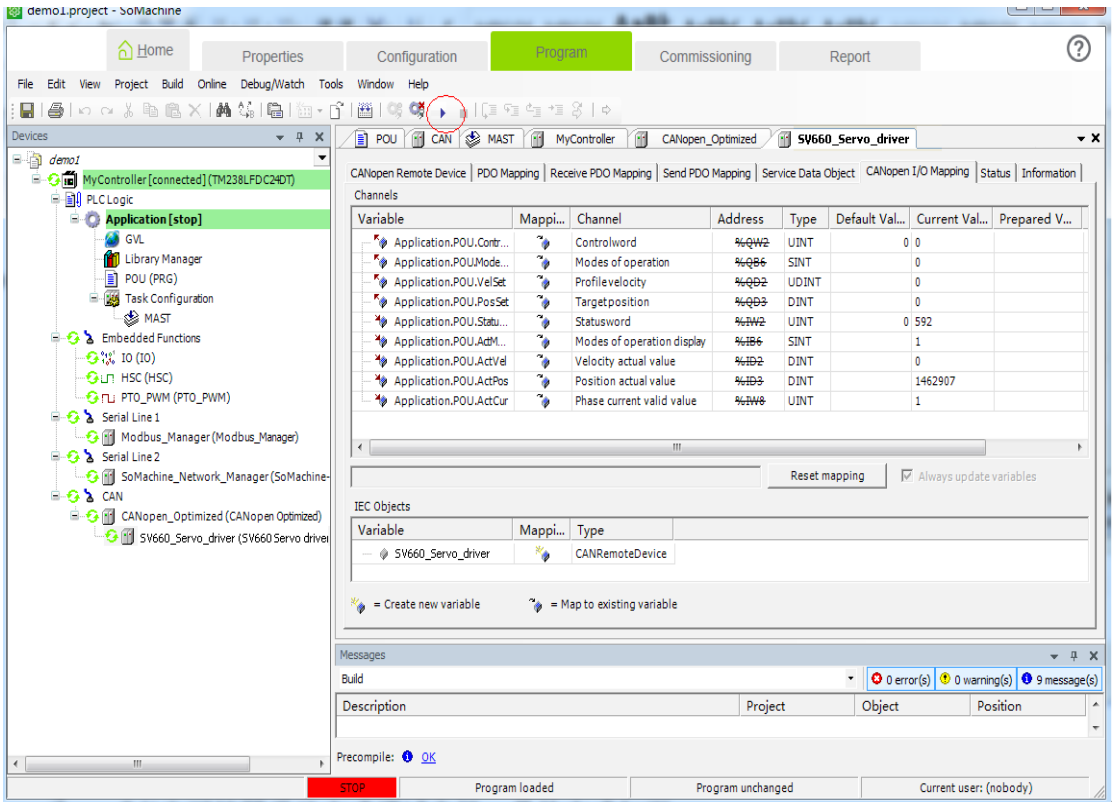
31) Click the icon circled out or select **Online** > **Login** or press **Alt+F8**.



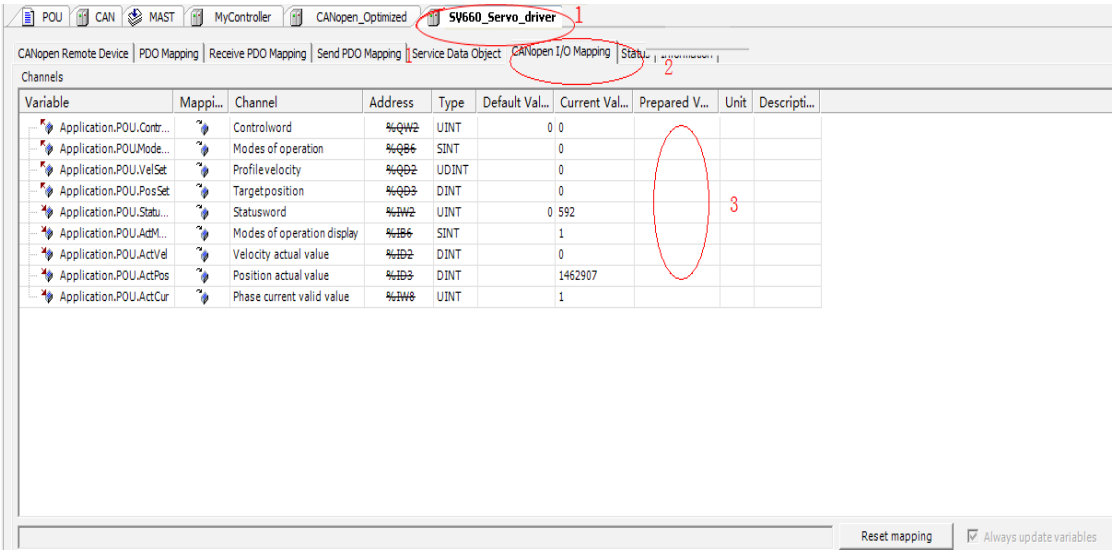
32) Click **Yes** in the dialog box popped out.



33) After download is done, click the small triangle icon circled out or click **Online** > **Start** or press **F5** to start the PLC program written by users. The motor operates in the mode defined by users.



34) You can also perform motor commissioning manually according to the following steps.
Select **CANopen I/O Mapping** under **SV660_Servo_driver** and enter the value needed in the **Prepared V...** column. Next, click **Debug/Watch** > **Forced Value** or press **F7** to modify the variable manually.

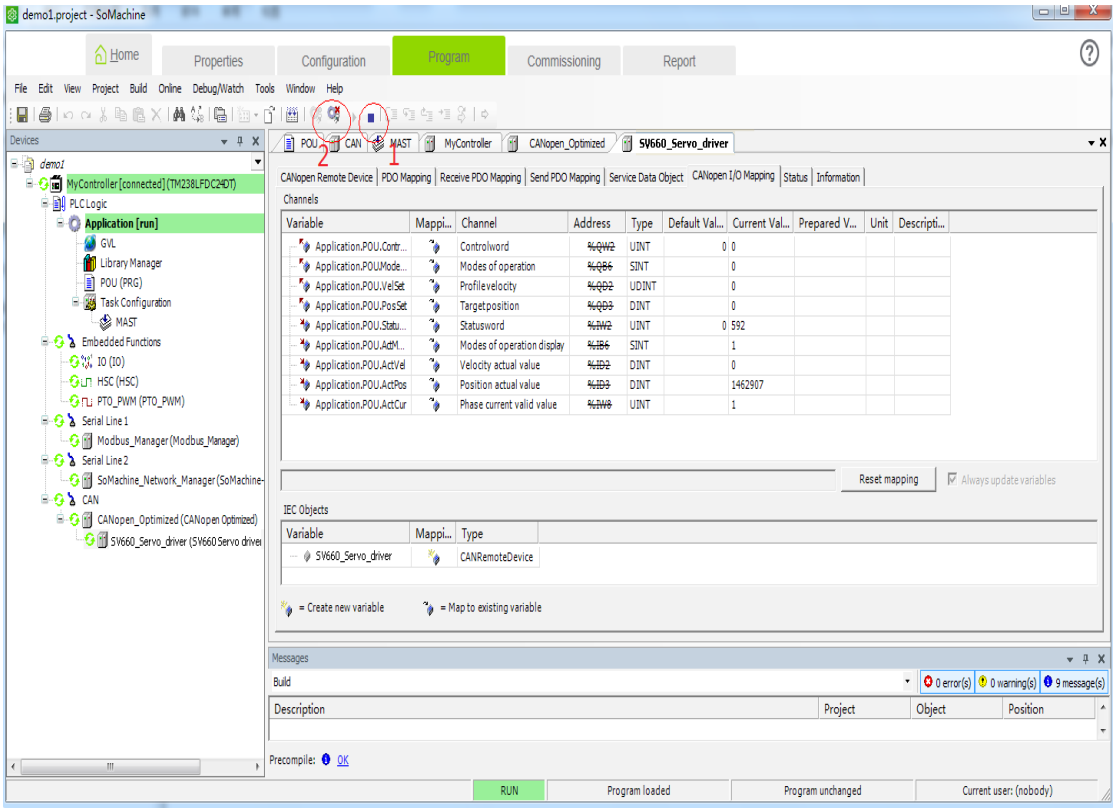


35) Set 6060h to 1, 6081h to 100, and 607Ah to 10485760 (10 revolutions) and set 6040h to 6, 7, 47(0x2f), and 63(0x3f) in sequence to start the motor.

NOTE

- The "Forced value" command must be executed for each written value of a variable. You can execute the "Forced value" command once after entering all the values for different variables.
- When a new position or velocity reference is required, write the new reference and set 6040h to 47(0x2f) and 63(0x3f) in sequence. The motor runs to the position according to the new reference no matter whether the previous reference is executed.
- To stop the motor, set 6040h to 0.
- Do not enter values manually. In the toolbar, select **Debug/Watch** > **Release Values** or press **Alt** + **F7**, and variables will be executed according to the PLC program logic.

36) Execute **1** marked in the following figure, or select **Online** > **Stop** in the toolbar or press **Shift** + **F8** to stop the PLC program. Click **2** in the following figure, or select **Online** > **Exit** or press **Ctrl** + **F8** to exit from the online function of the programming example.



7.2 Connecting SV660C Servo Drive to Beckhoff CANopen Master

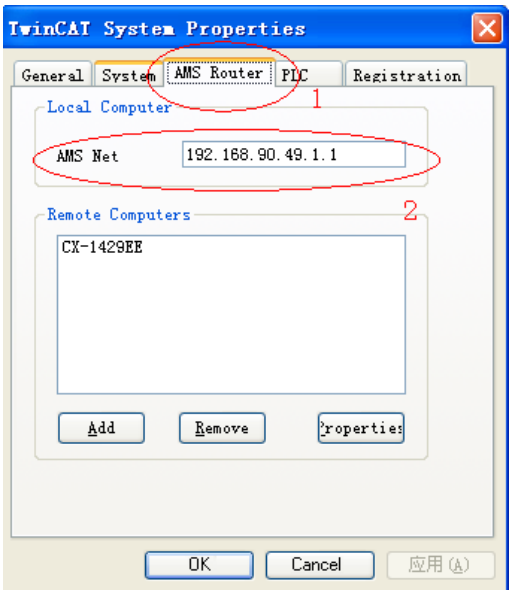
Allocate PDOs according to Table 7-2 in the position control mode.

1) As the PDO mapping configuration of Beckhoff master is complicated, configure the PDO mapping manually before connecting the network. You can change the mapping through modifying parameters based on the following table.

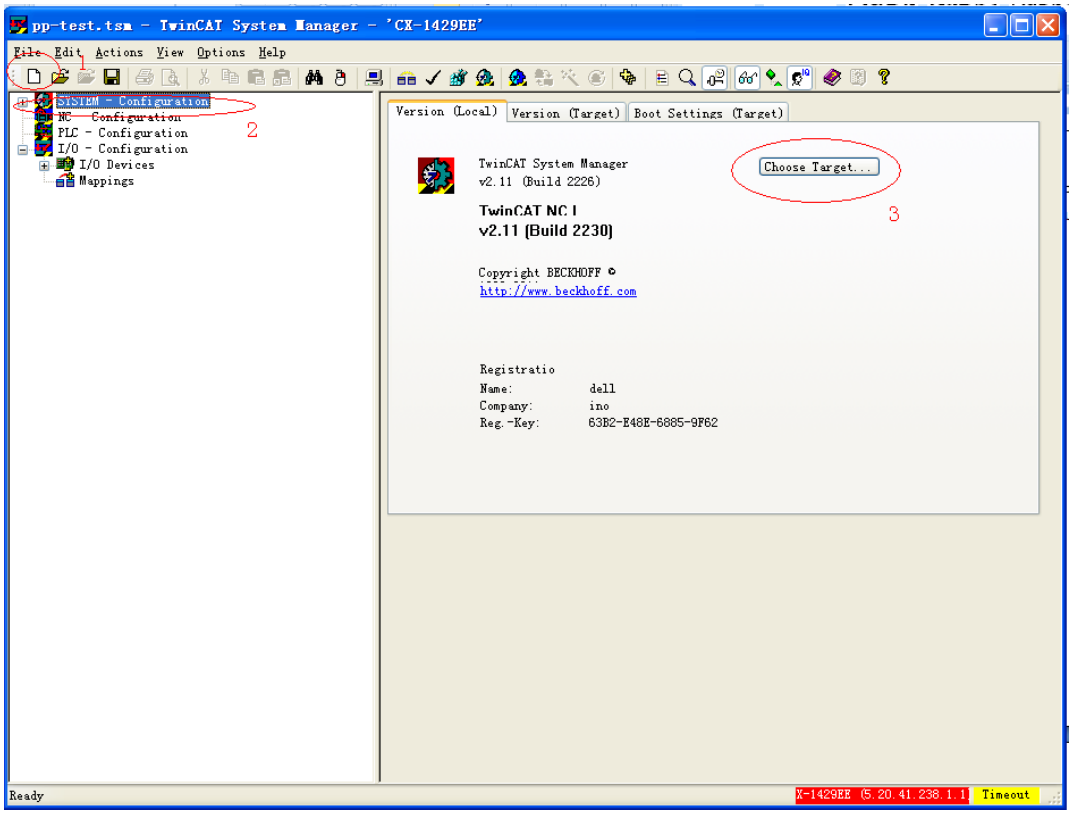
Table 7-2 Example of PDO mapping of Beckhoff master

Parameter	Object	Mapping Object	Input
H2D-32	1600h-00h	Number of RPDO1 mapping objects	2
H2D-33	1600h-01h	6040h-00h	60400010h
H2D-35	1600h-02h	6060h-00h	60600008h
H2D-49	1601h-00h	Number of RPDO2 mapping objects	2
H2D-50	1601h-01h	6081h-00h	60810020h
H2D-52	1601h-02h	607Ah-00h	607A0020h
H2E-20	1A00h-00h	Number of TPDO1 mapping objects	2
H2E-21	1A00h-01h	6041h-00h	60410010h
H2E-23	1A00h-02h	6061h-00h	60610008h
H2E-37	1A01h-00h	Number of TPDO2 mapping objects	2
H2E-38	1A01h-01h	606Ch-00h	606C0020h
H2E-40	1A01h-02h	6064h-00h	60640020h
H2E-54	1A02h-00h	Number of TPDO3 mapping objects	1
H2E-55	1A02h-01h	200Bh-19h	200B1910h
H2E-57	1A02h-02h	-	0

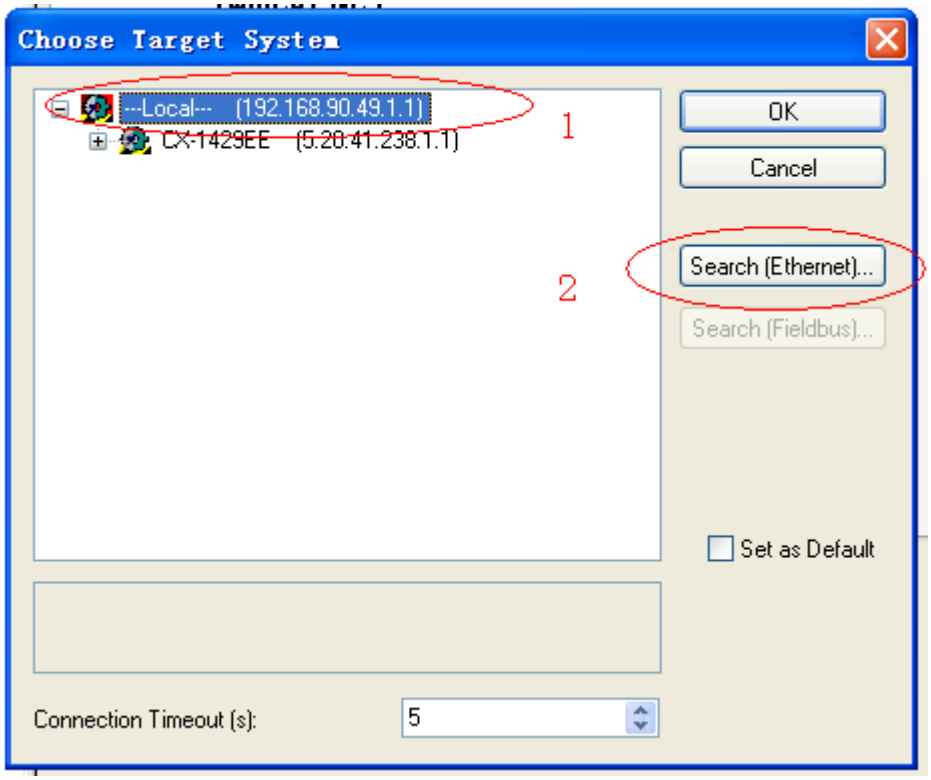
2) Perform tests by taking the EL6751 CANopen module connected to Beckhoff CX9020 as the master. Ensure the IP address of CX9020 and that of the PC are in the same LAN. The first four digits of **TwinCAT System Properties** > **AMS Router** > **AMS Net** of Beckhoff TwinCAT software must be the same with that of the IP address of the PC.



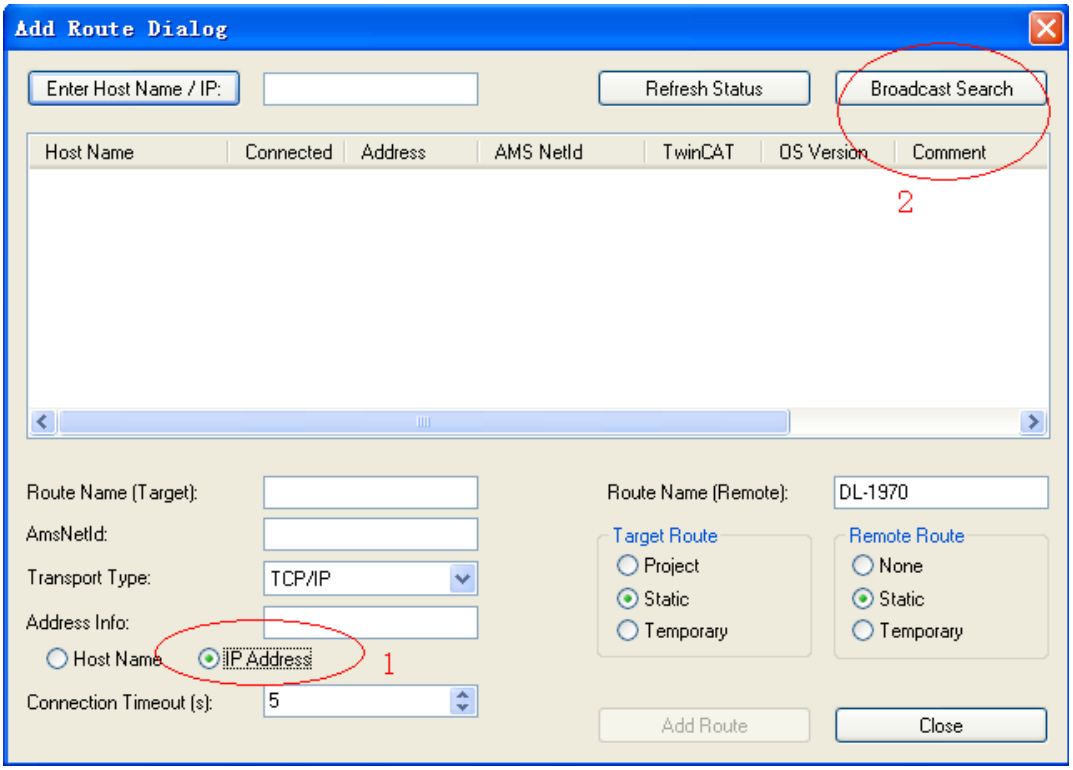
3) Open **TwinCAT System Manager** and create an empty project. Click **SYSTEM - Configuration** on the left and click **Choose Target...** on the right.



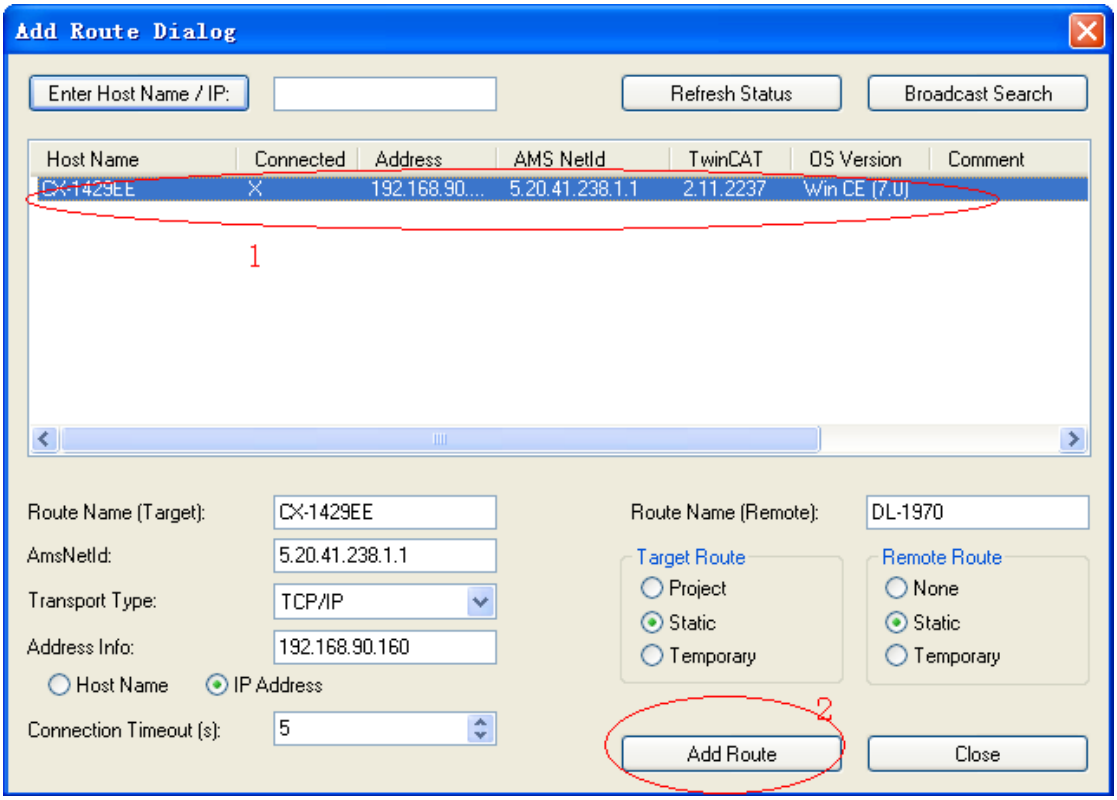
4) In the dialog box popped out, select ···**local**··· and click **Search (Ethernet)...**.



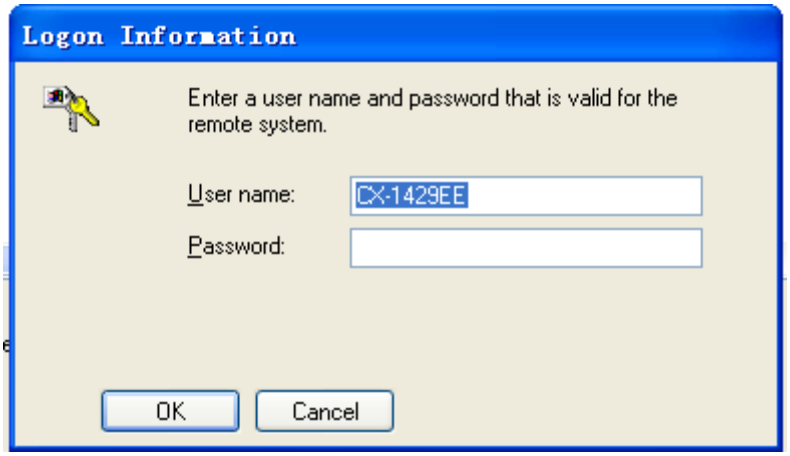
5) In the dialog box popped out, select **IP Address** as indicated by **1** and click **Broadcast Search** as indicated by **2**.



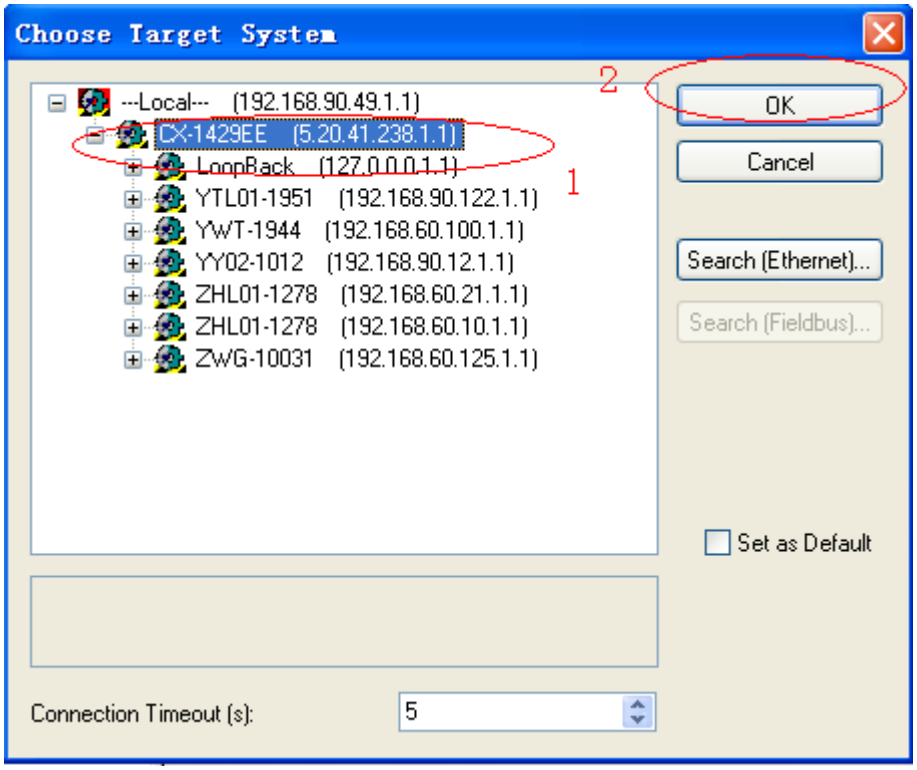
6) The master is displayed. Select the master and click **Add Route**.



7) In the dialog box popped out, the account is the same with the **Host Name** and the password is empty. Click **OK**.

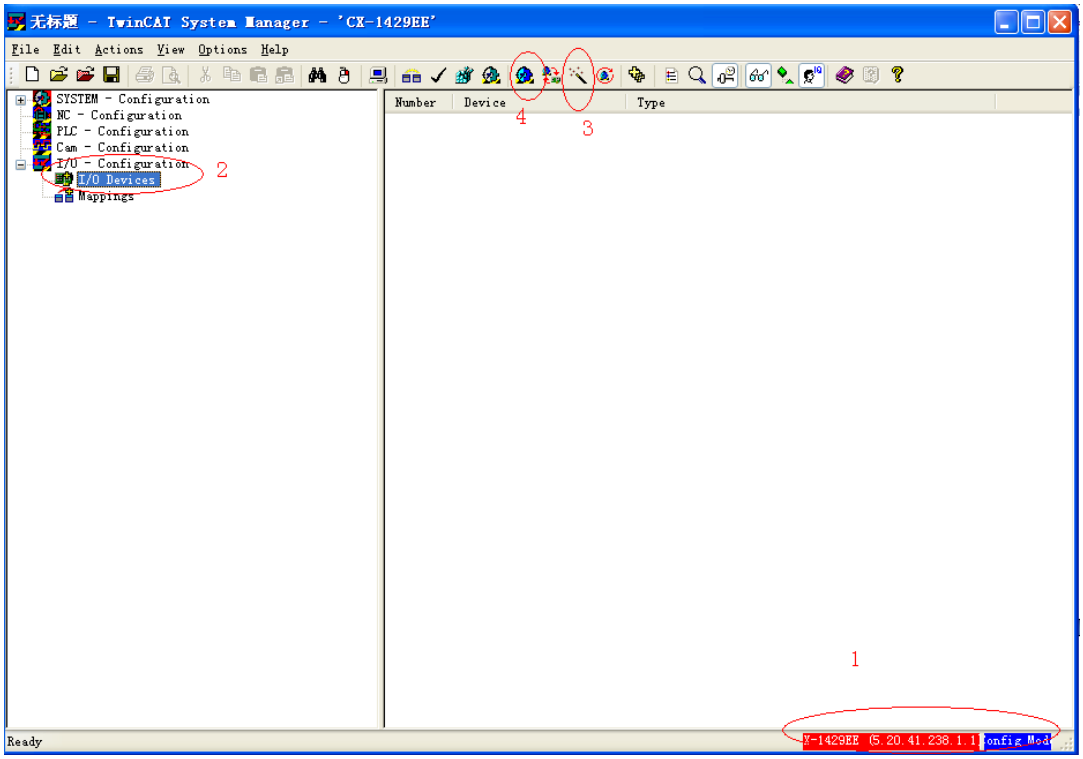


8) Click **Close** in the interface shown in Step 6. Click **+** in the **Choose Target System** dialog box, select the master and click **OK**.

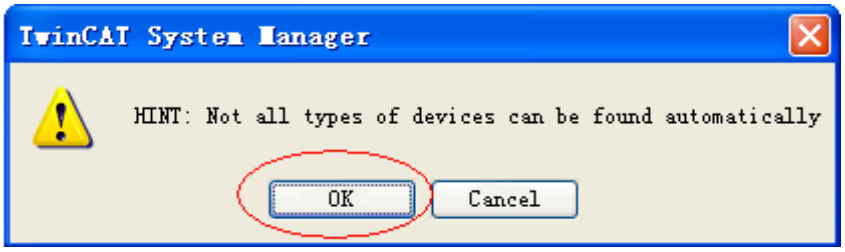


9) The master (in red) can be seen in the lower right corner of the window, which is in the configuration state (in blue). If the master is in the operating state (in green), click the icon indicated by **4** to switch to the configuration mode and perform the next step.

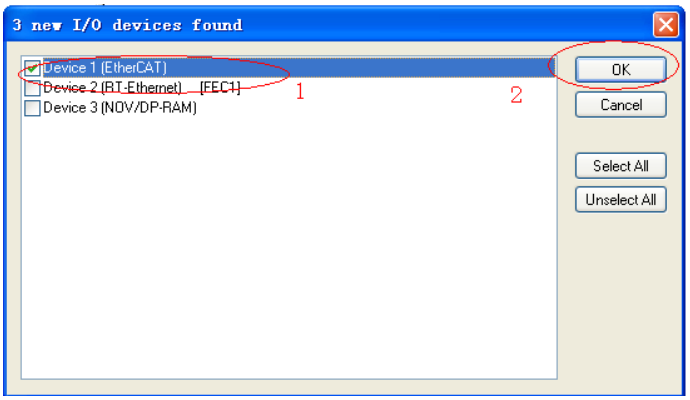
Select **I/O Devices** on the left and click the icon indicated by **3** or right-click **I/O Devices** and select **Scan Devices**.



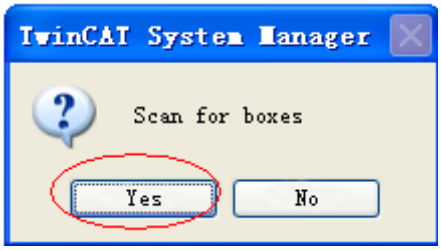
10) Click **OK** in the dialog box popped out.



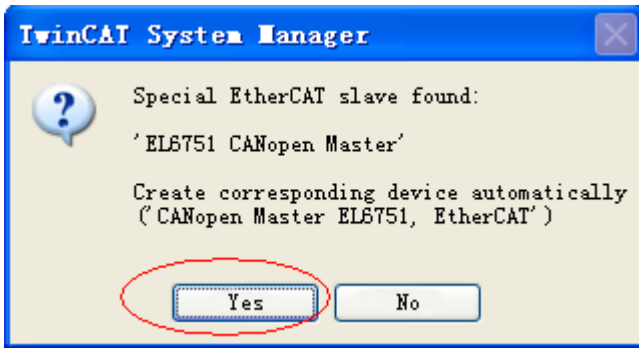
11) Check **Device 1 (EtherCAT)** and click **OK** in the dialog box popped out.



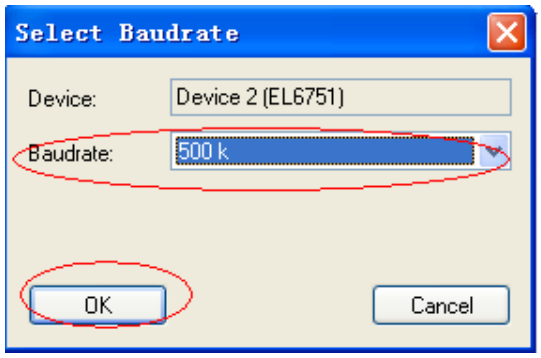
12) Click **Yes** in the dialog box asking whether to scan for boxes.



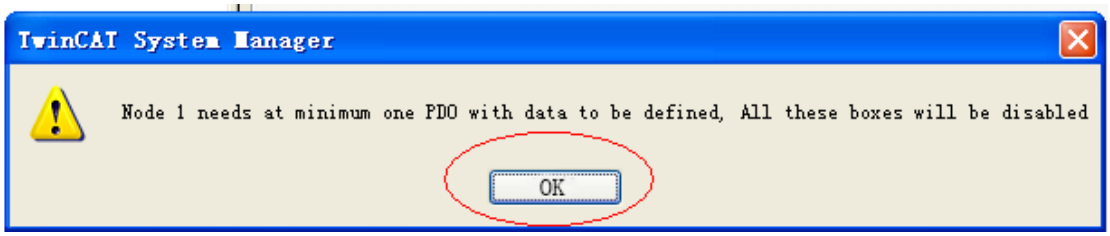
13) Click **Yes** in the dialog box asking whether to create EL6751 master.



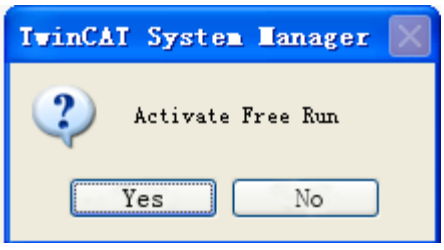
14) Select the baud rate (default: 500 Kbps) and click **OK**. The master starts device searching, which may take a while.



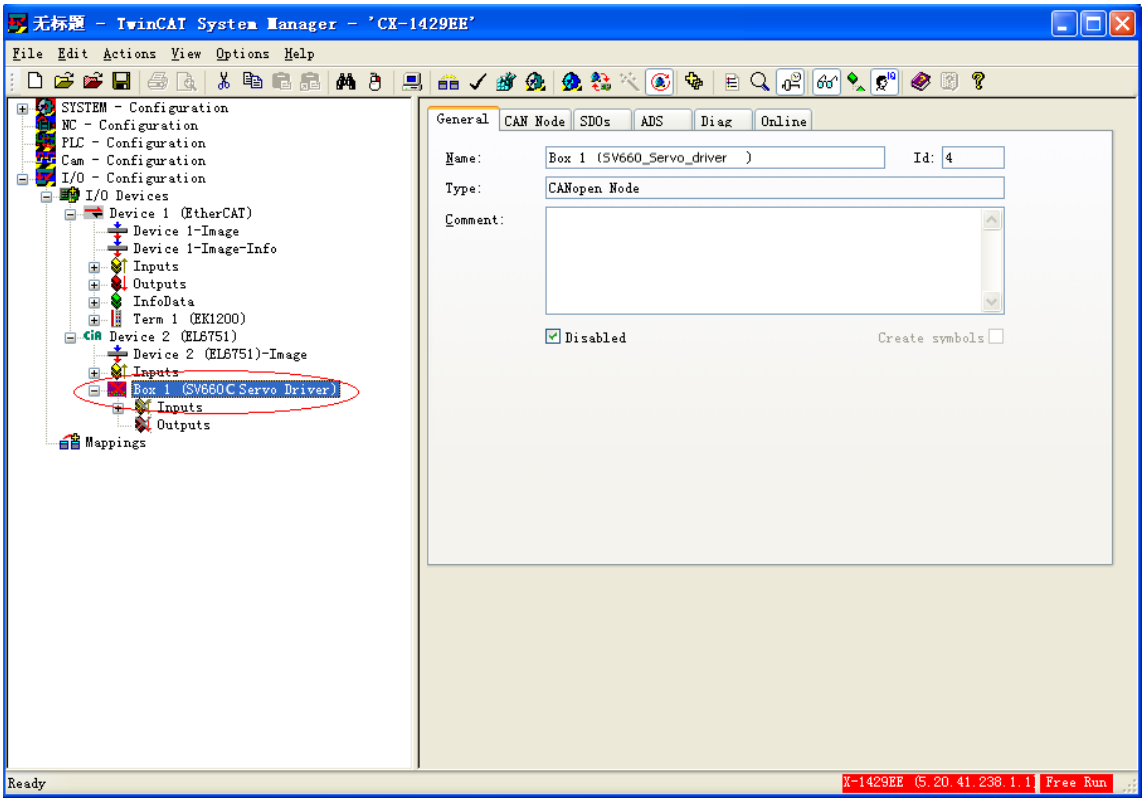
15) After the searching is done, click **OK** in the warning dialog box popped out.



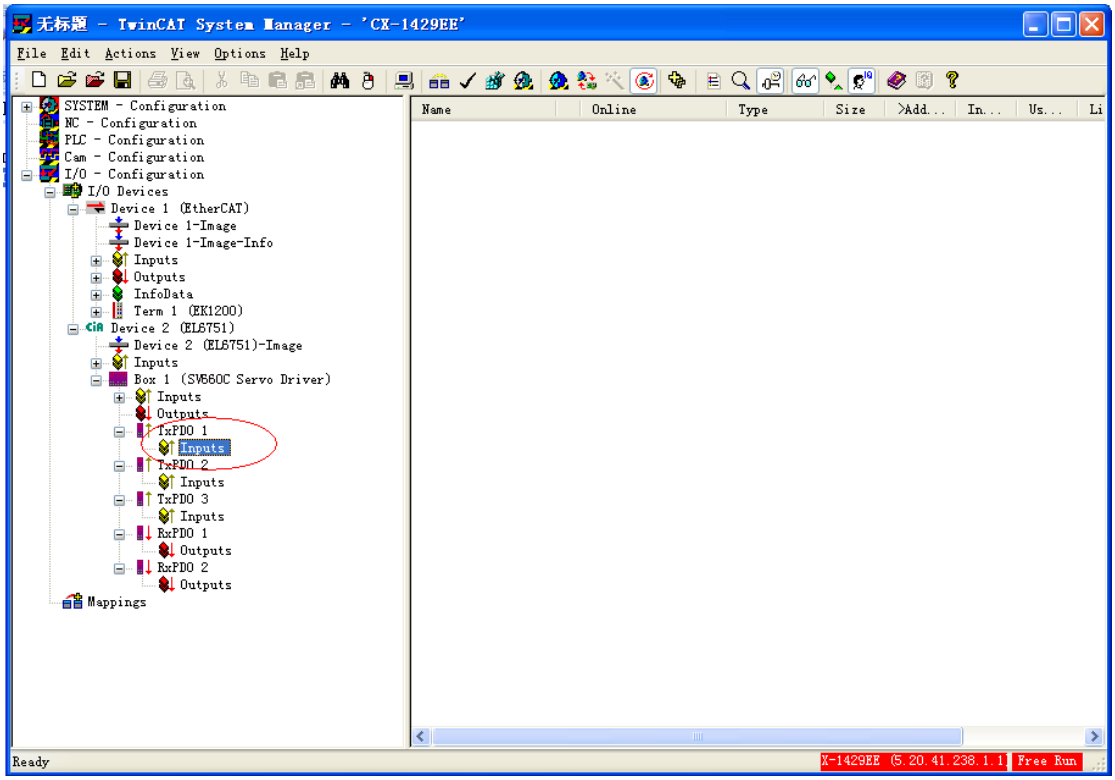
16) Click **Yes** in the dialog box asking whether to activate free run.



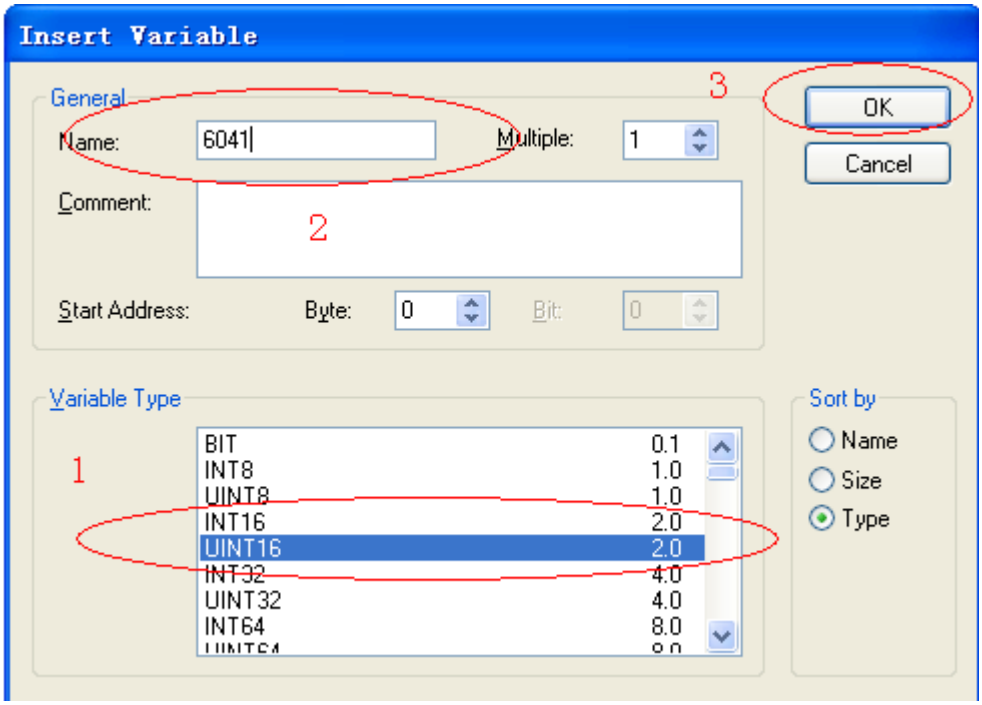
17) The Box of SV660C servo drive is now displayed on the left. Select the box and right-click to insert three TPDOs and two RPDOs and right-click **Disabled**. (SV660C servo drives not configured with the termination resistor cannot be scanned by the master.)



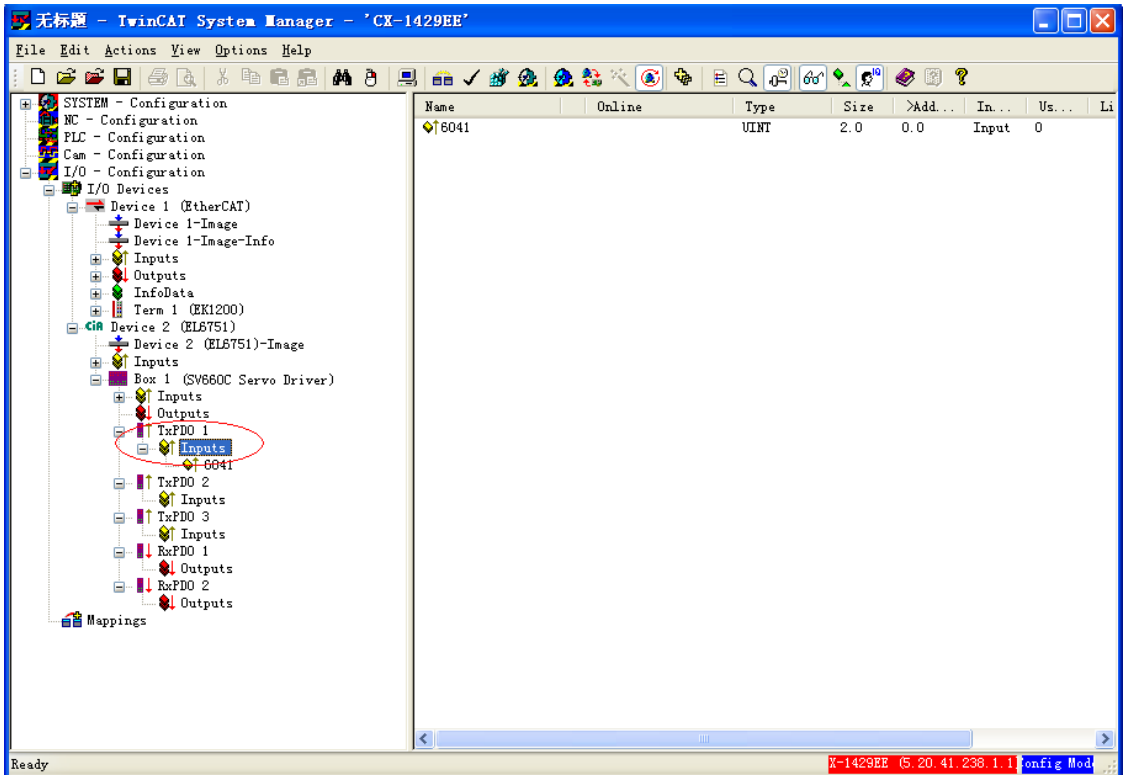
18) The following interface is displayed after preceding step is done. Select **TxPDO1** > **Inputs**, and right-click to insert the variable.



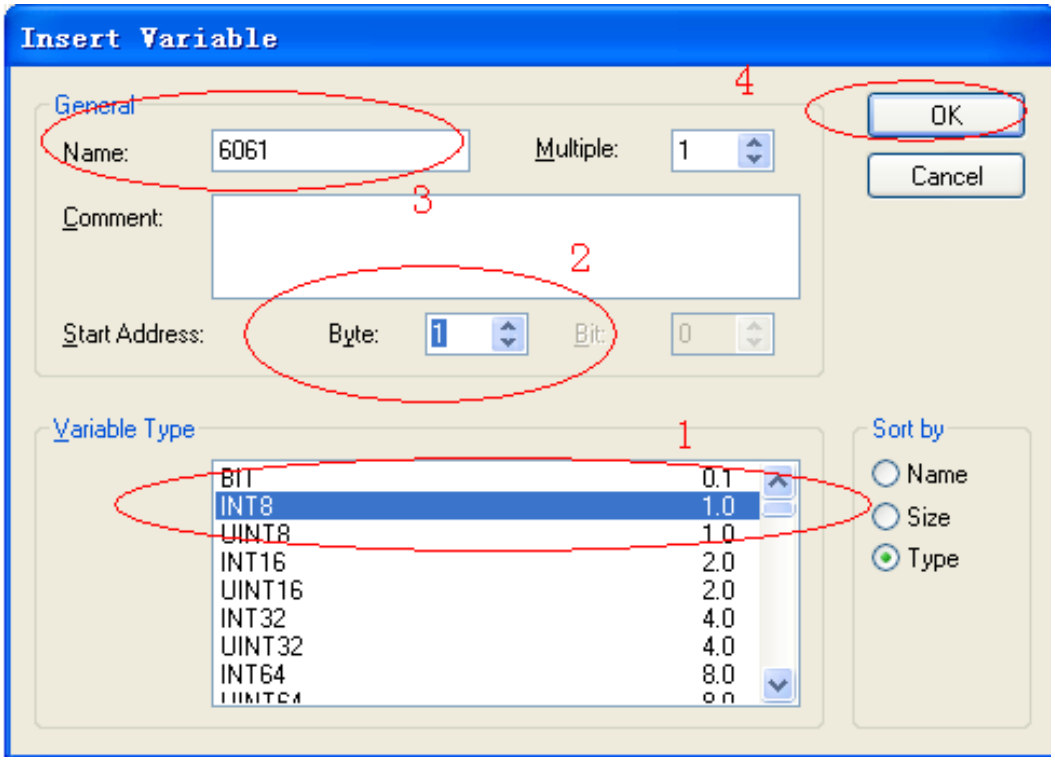
19) Map different variables in each PDO based on table B-1. TPDO1 maps 6041h-00 and 6061h-00. Insert the first variable 6041h, select **UINT16** in the **Variable Type**, enter a proper name, and click **OK**.



20) Now, 6041h is added to TPDO1. Right-click **Inputs**, select **Insert Variable**, and insert the second variable.

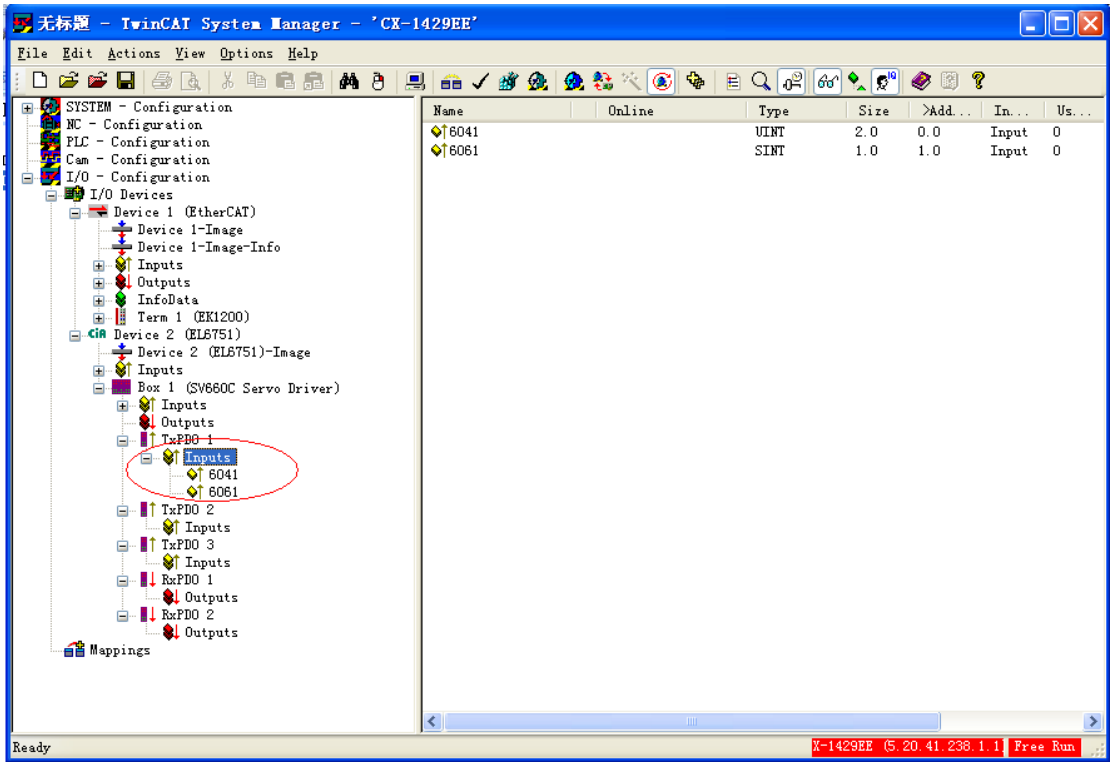


21) Select **INT8** (see the attribute in the object dictionary) in the **Variable Type** of 6061h. Enter a large value in the **Start Address** > **Byte** to prevent 6061h from being inserted before 6041h. Enter a proper name and click **OK**.

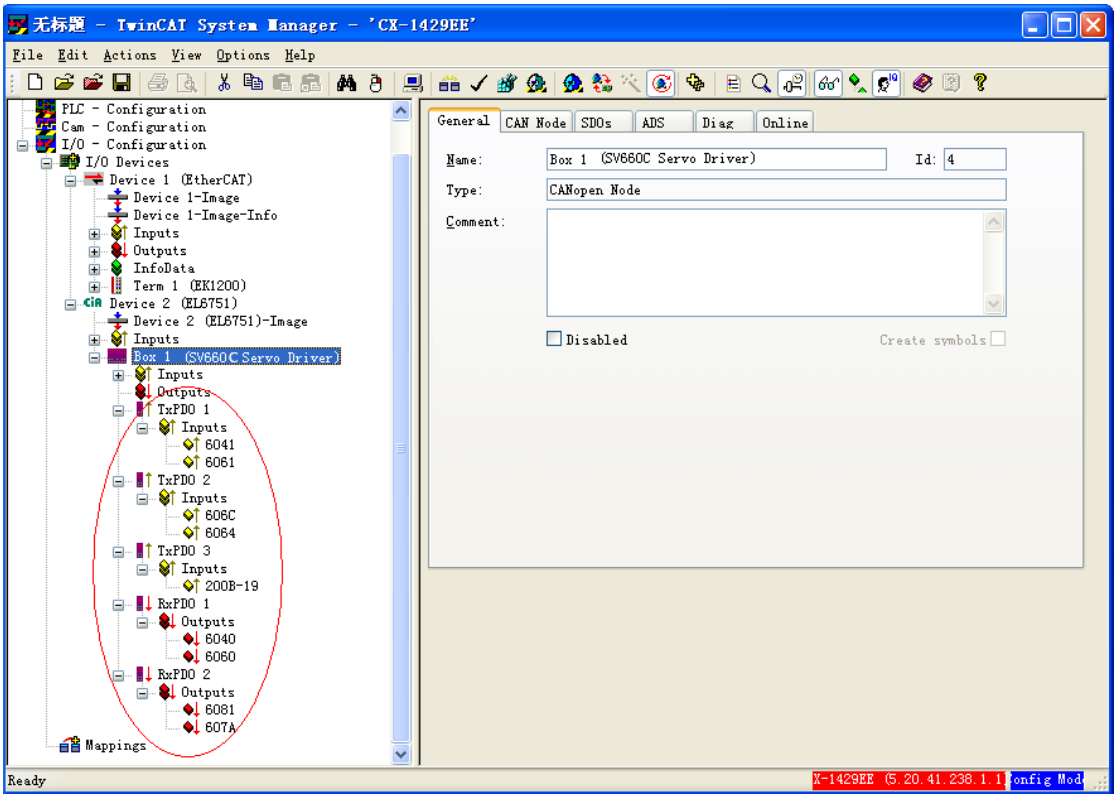


22) Now, two objects are added to TPDO1. Note that the insertion sequence of the two variables must be consistent with Table B-1. If the insertion sequence is inconsistent, you need to delete the second variable, re-insert a variable, and enter a large value in the position marked by **2** in the preceding figure.

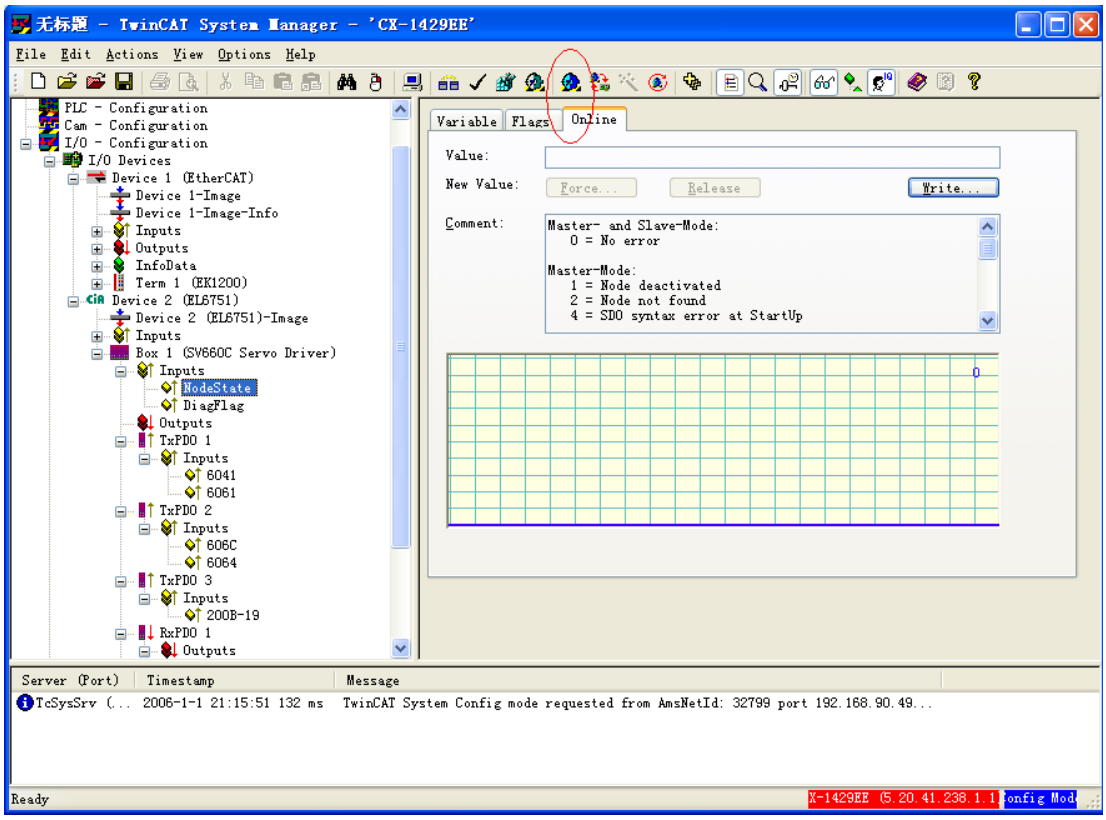
When the insertion sequence is correct, select **TxPDO1** > **Inputs**, right-click, and select **Recalc Address** to allocate addresses. This step must be performed. Otherwise, address chaos may occur.



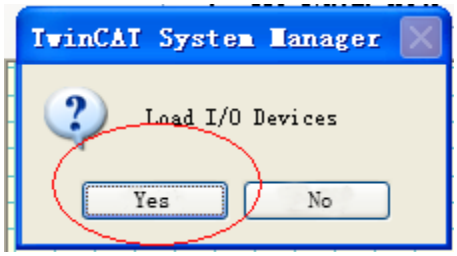
23) Repeat steps 18 to 22 for other PDOs. Add corresponding mapping variables based on Table B-1. The interface after variables are added is shown below.



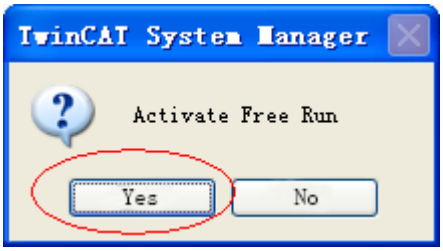
24) Click the icon circled out in the following figure or press **Shift** + **F4**.



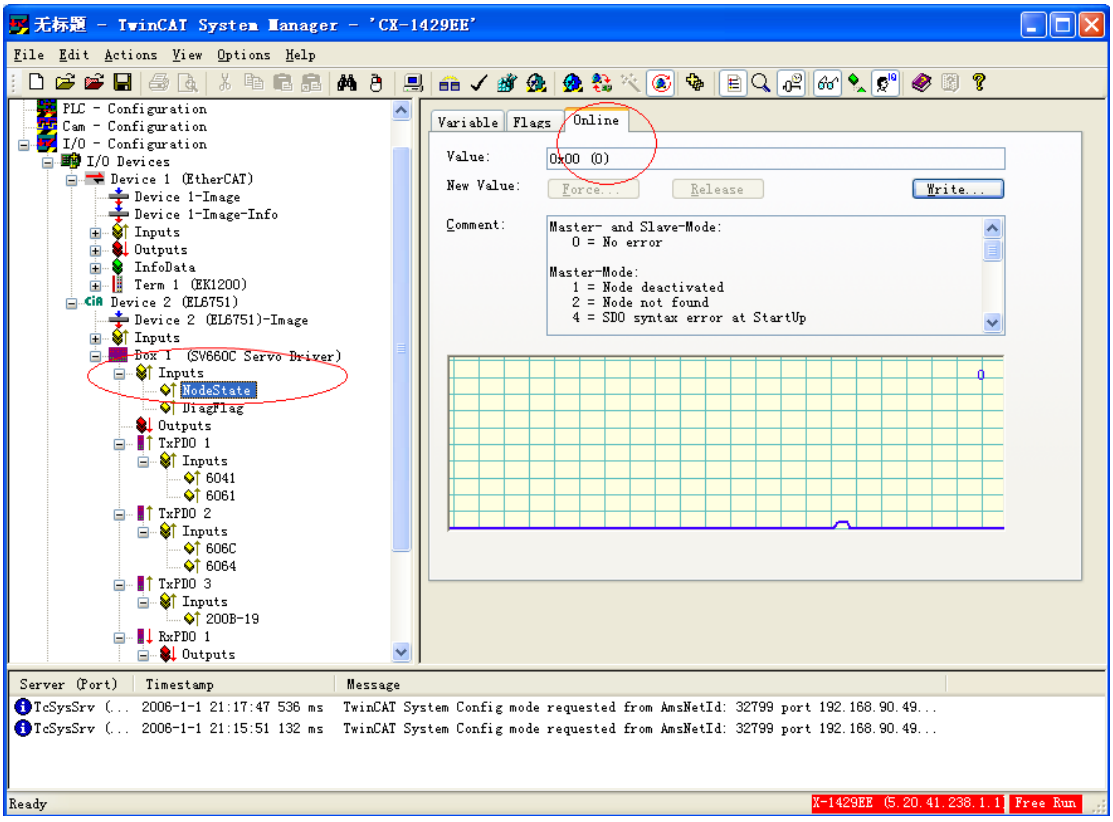
25) Click **Yes** in the following dialog box.



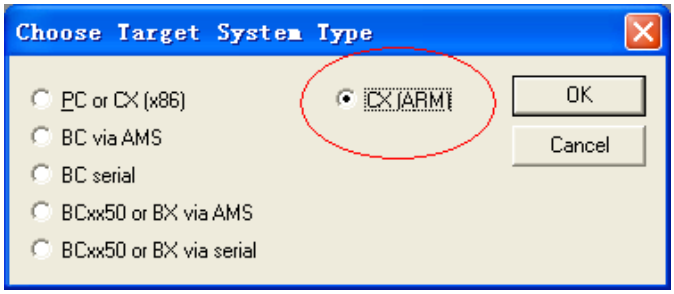
26) Select **Yes** in the dialog box asking whether to activate free run.



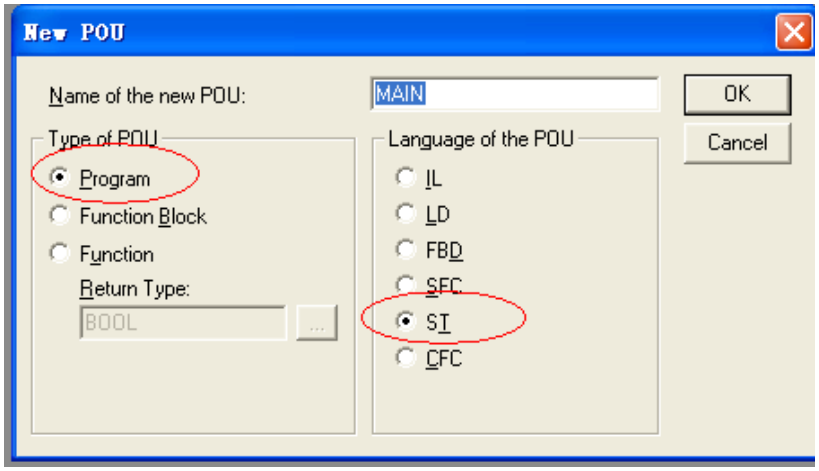
27) Select the Box of SV660C and select **Inputs** > **NodeState**. The node state in **Online** is 0, indicating the node is in a normal condition.



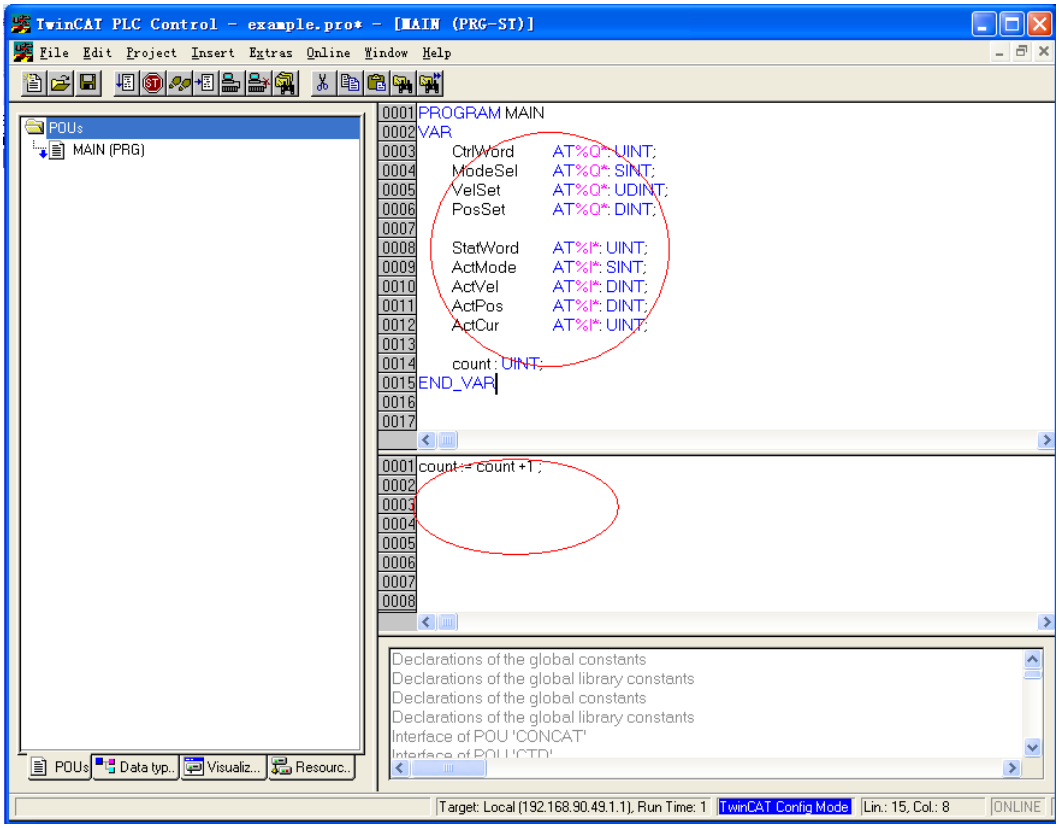
28) Open TwinCAT PLC Control, create a new project and select **CX(ARM)** in the dialog box popped out.



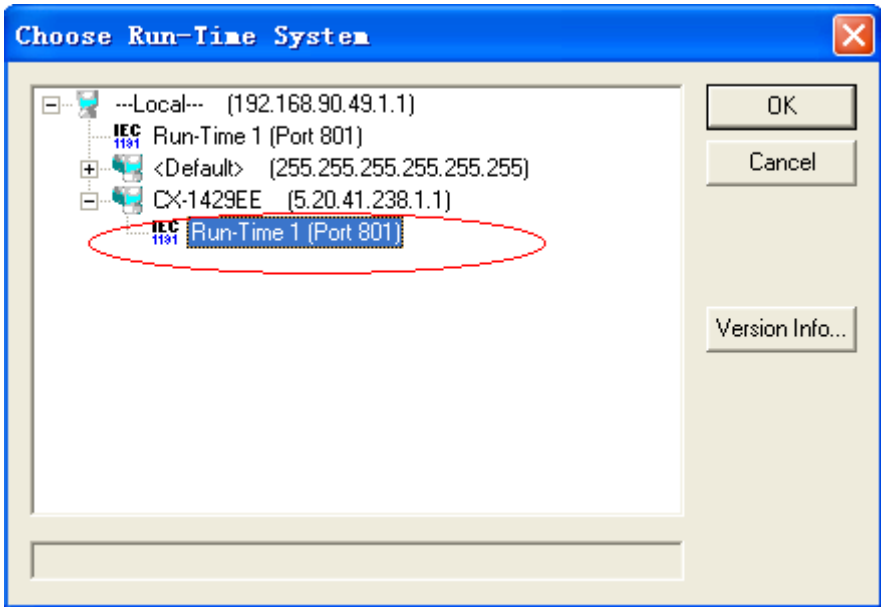
29) Select as indicated below in the dialog box popped out.



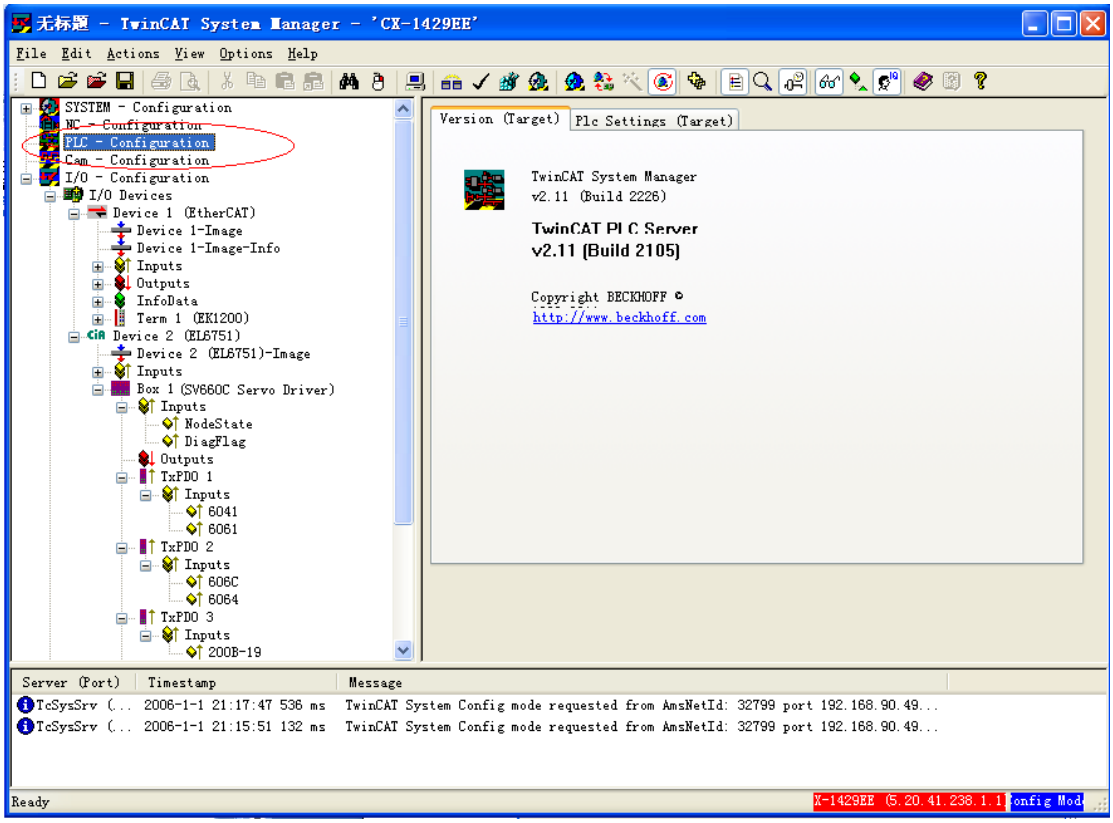
30) Input corresponding variable definition and the PLC logic.



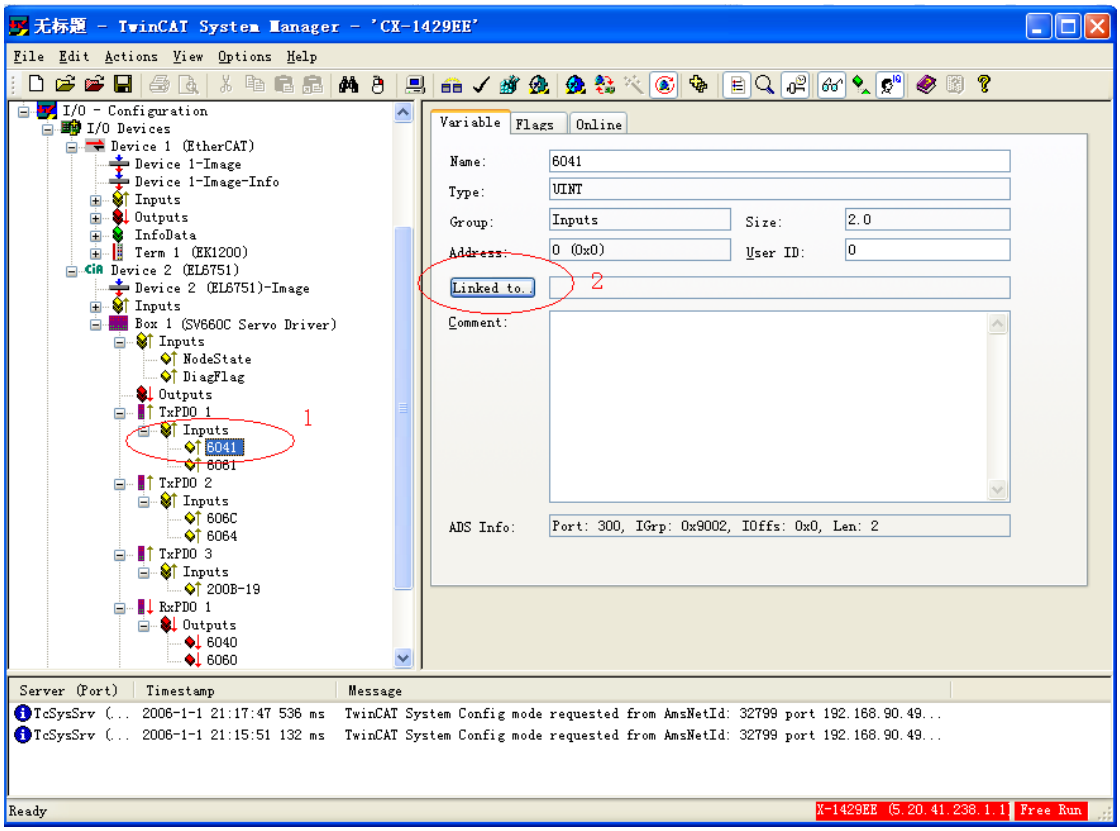
31) In the toolbar, select **Online** > **Choose Run-Time System**. Select the corresponding master port in the dialog box popped out and click **OK**.



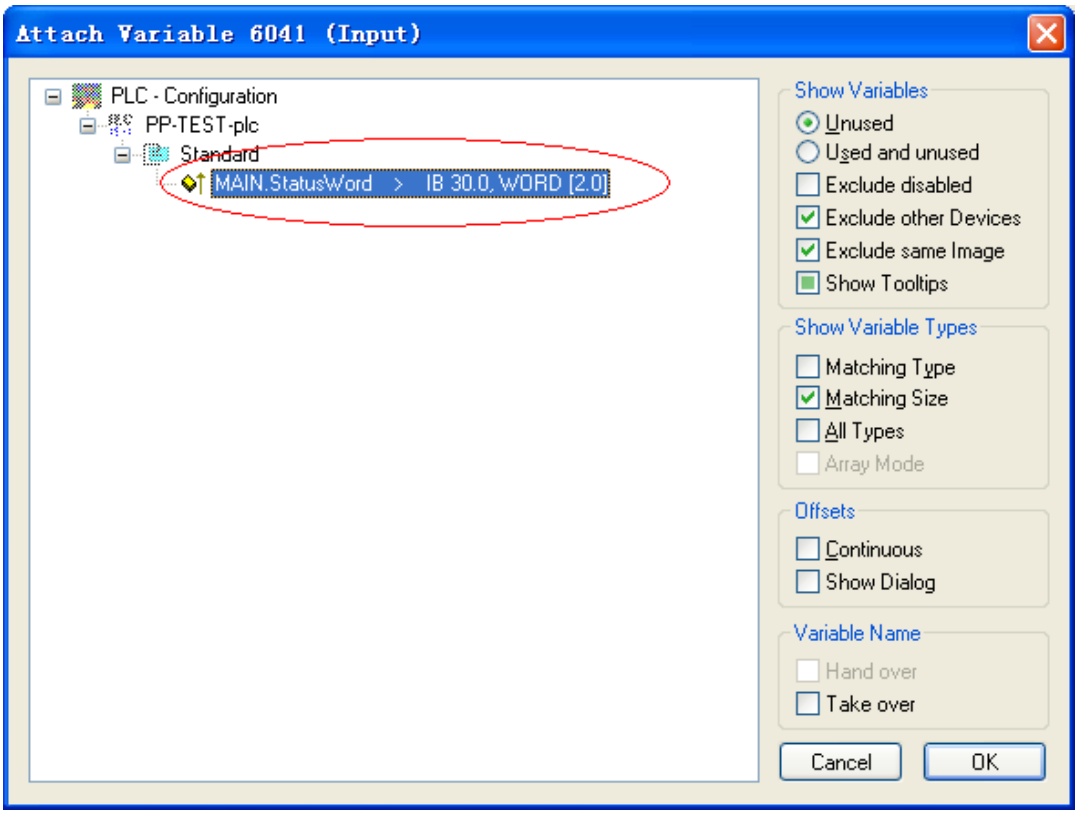
32) In the TwinCAT System Manager, select **PLC - Configuration** on the left, right-click, and select **Append PLC Project...** to select the PLC program (.tpy). created.



33) After the PLC program is added, select the PDO variable and click **Linked to** or double-click the variable to link the variable to the PLC program.

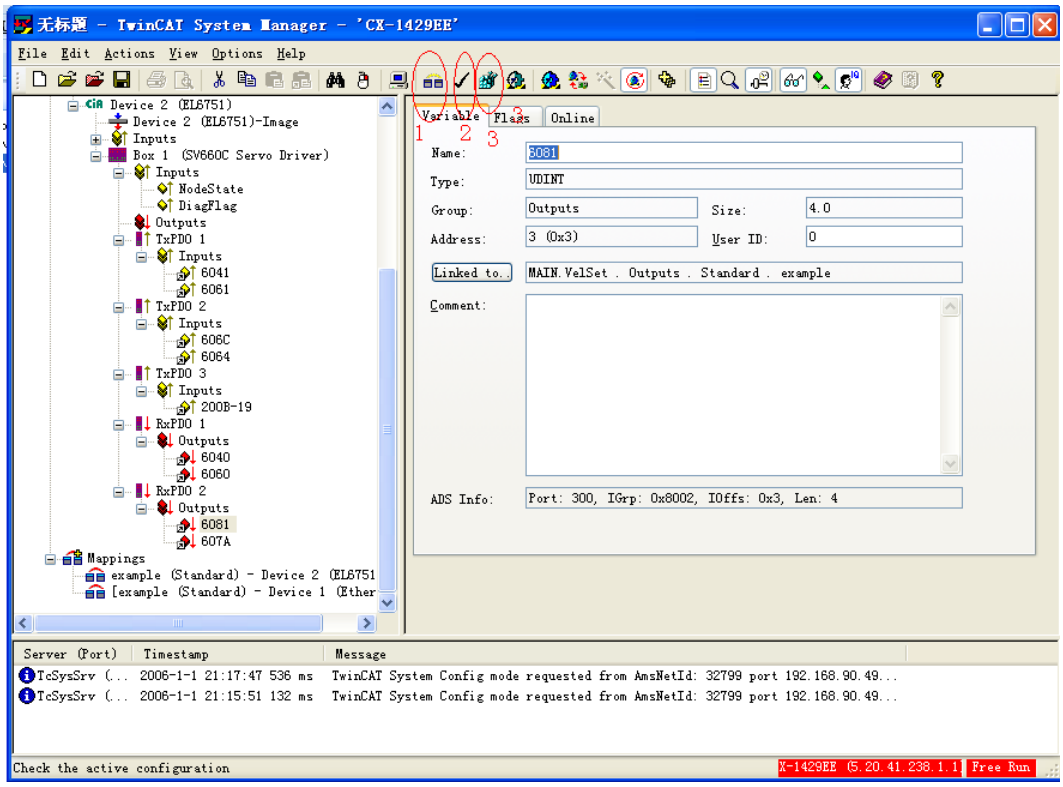


34) Select the corresponding PLC variable and click **OK**.

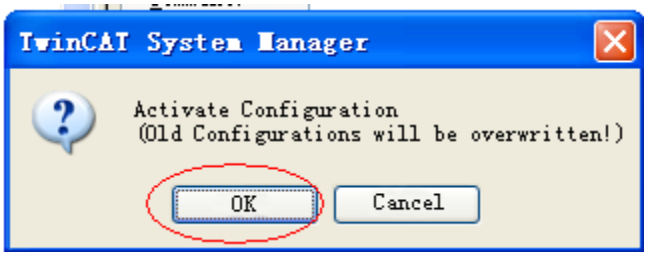


35) After the variable is linked, a small arrow pointing upper right appears at the bottom left of the variable name icon. The following figure on the left shows the variable linked and that on the right shows the variable not linked.

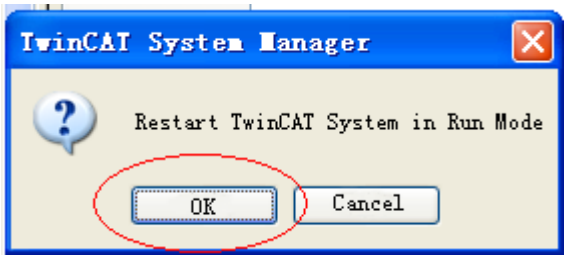
36) Click **Generate mapping**, **Check Configuration**, and **Activate Configuration** in sequence, as circled out by **1**, **2**, and **3** in the following figure.



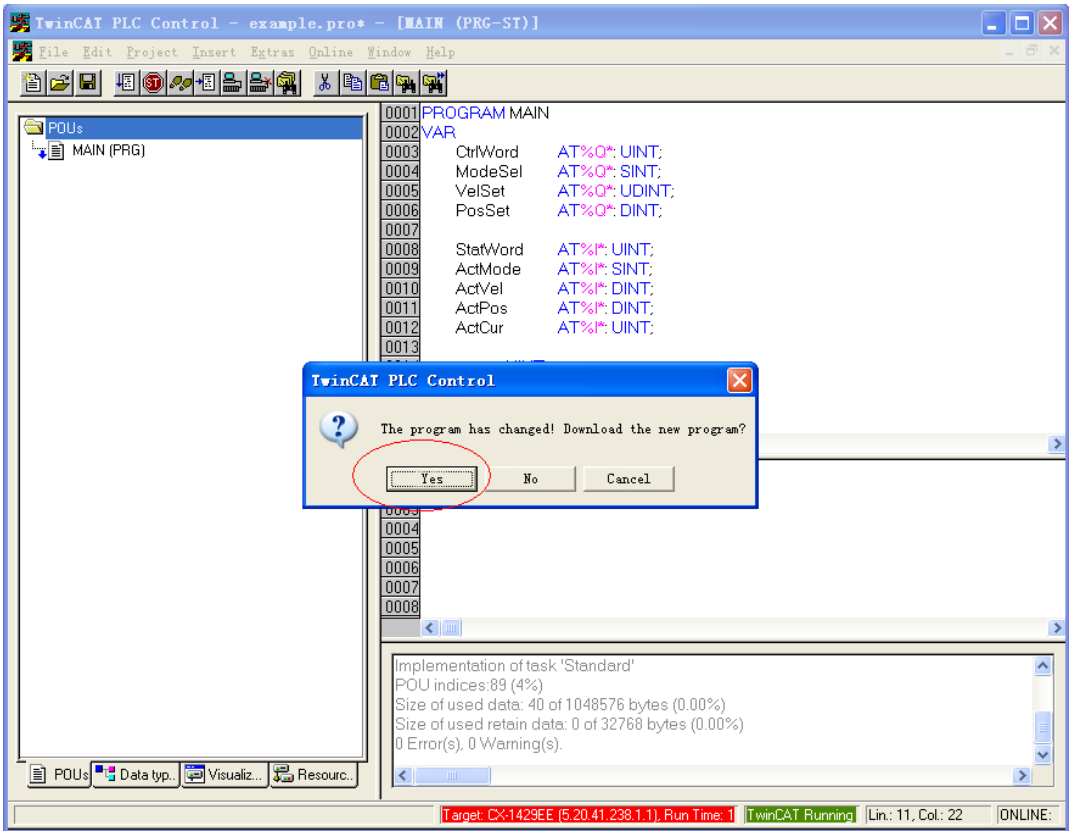
37) Click **OK** to activate configuration.



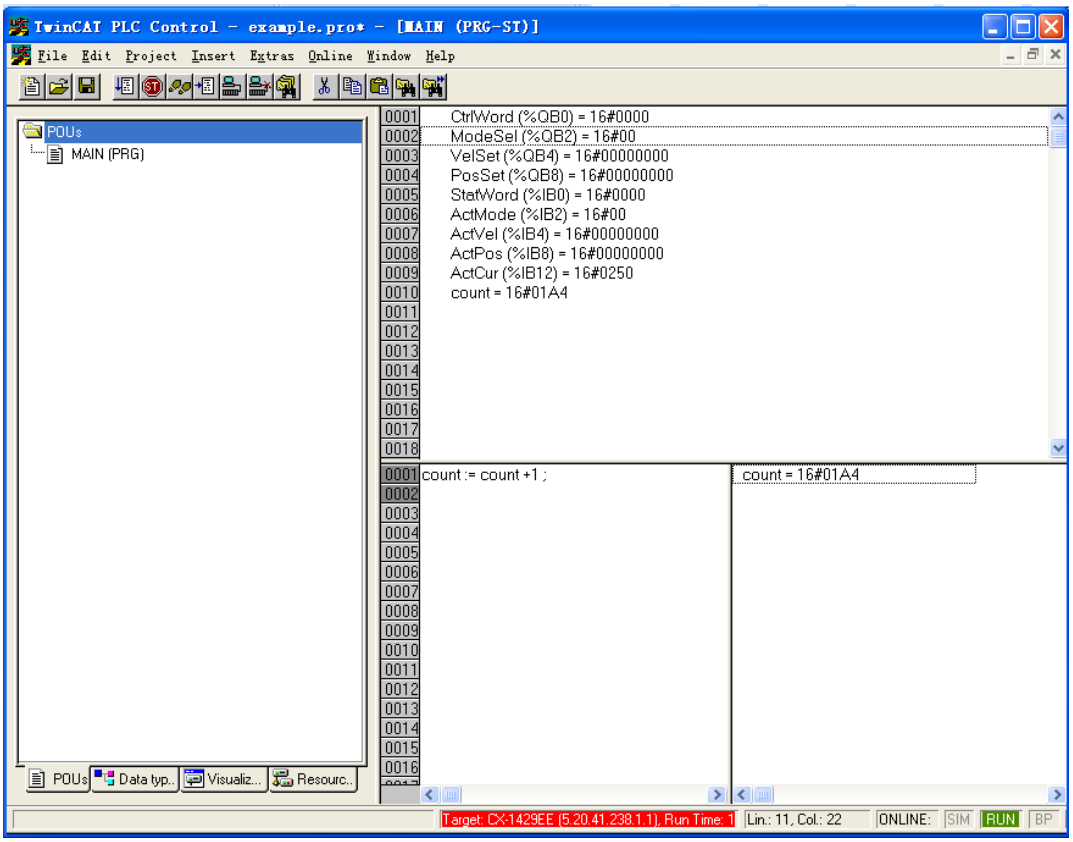
38) Click **OK** to restart TwinCAT system in run mode.



39) Open the project created before **TwinCAT PLC Control** is established and select **Online** > **Login** or press **F11** to display the dialog box asking whether to download the new program.

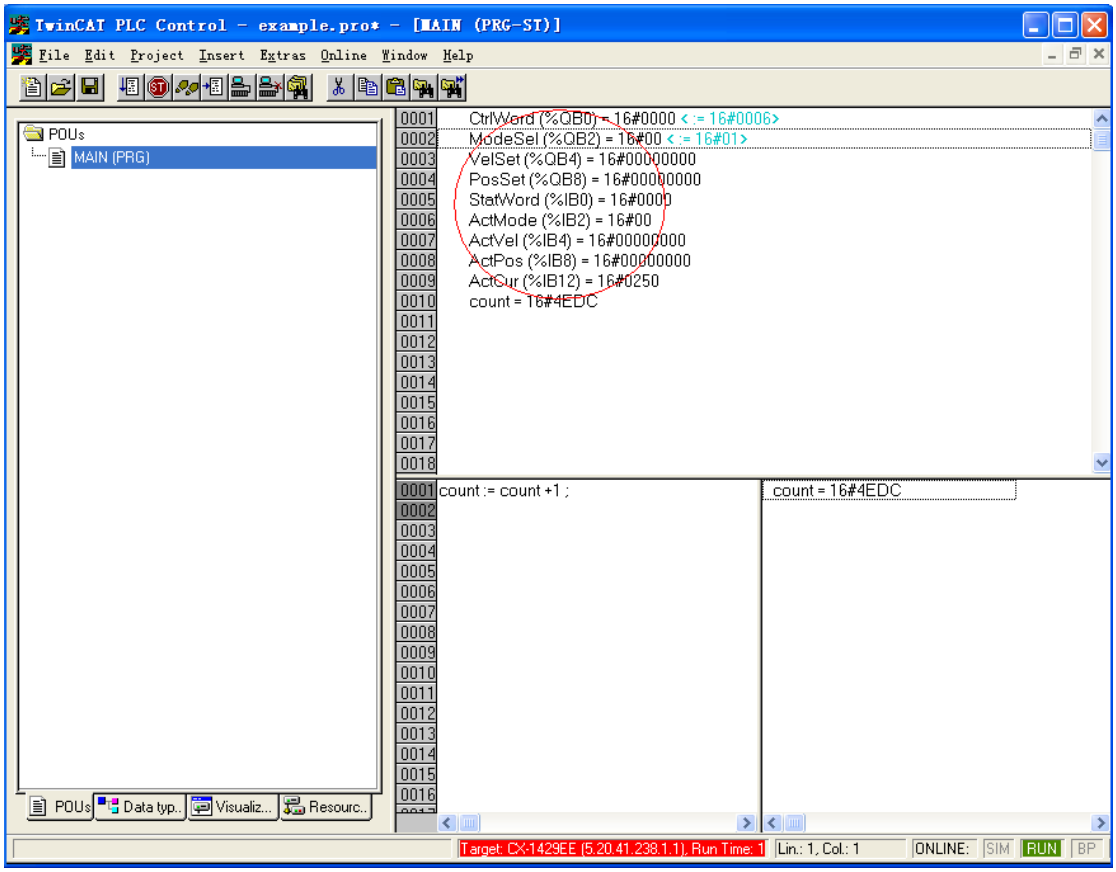


40) Select **Online** > **Run** or press **F5** to run the user PLC program.

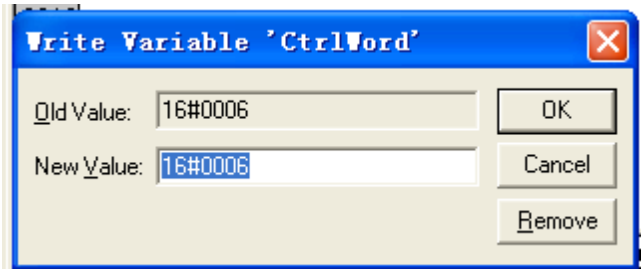


41) You can perform write commissioning forcibly through the manual mode. The commissioning method is similar to that of the Schneider master.

Double-click variables circled out in the following figure and input values.



42) Input the value and click **OK**.



The value entered is displayed in the square brackets behind the original variable. Select **Online** > **Forced Value** or press **F7** to write the value forcibly.

Set 6060h to 1, 6081h to 100, and 607Ah to 10485760 (10 revolutions) and set 6040h to 6, 7, 47(0x2f), and 63(0x3f) in sequence to make the motor run.

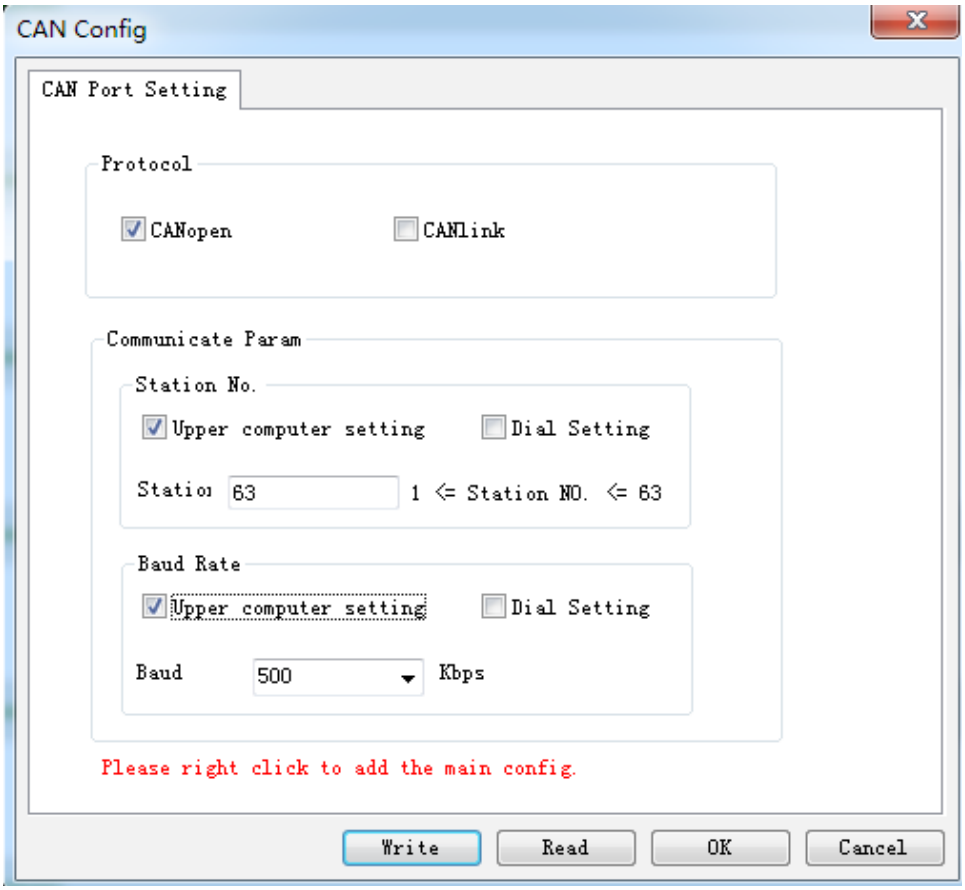
NOTE

- The "Forced value" command must be executed for each written value of a variable. You can execute the "Forced value" command once after entering all the values for different variables.
- When a new position or velocity reference is required, write the new reference and set 6040h to 47(0x2f) and 63(0x3f) in sequence. The motor runs to the position according to the new reference no matter whether the previous reference is executed.
- To stop the motor, set 6040h to 0.
- Do not enter values manually. In the toolbar, select **Online** > **Release Force** or press **Shift** + **F7**, and variables will be executed according to the PLC program logic.

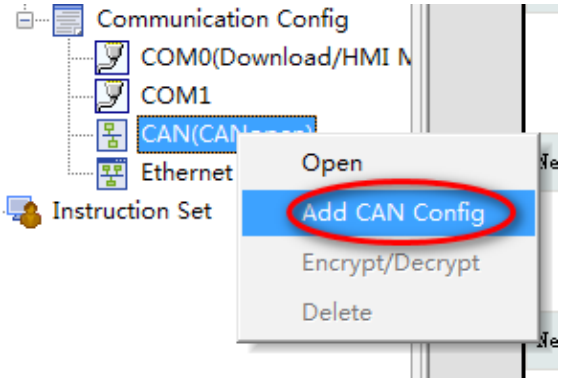
43) To stop executing the user PLC program, select **Online** > **Stop** in the toolbar. To continue editing the PLC program or exit, select **Online** > **Logout**.

7.3 Connecting SV660C Servo Drive to Inovance H3U CANopen Master

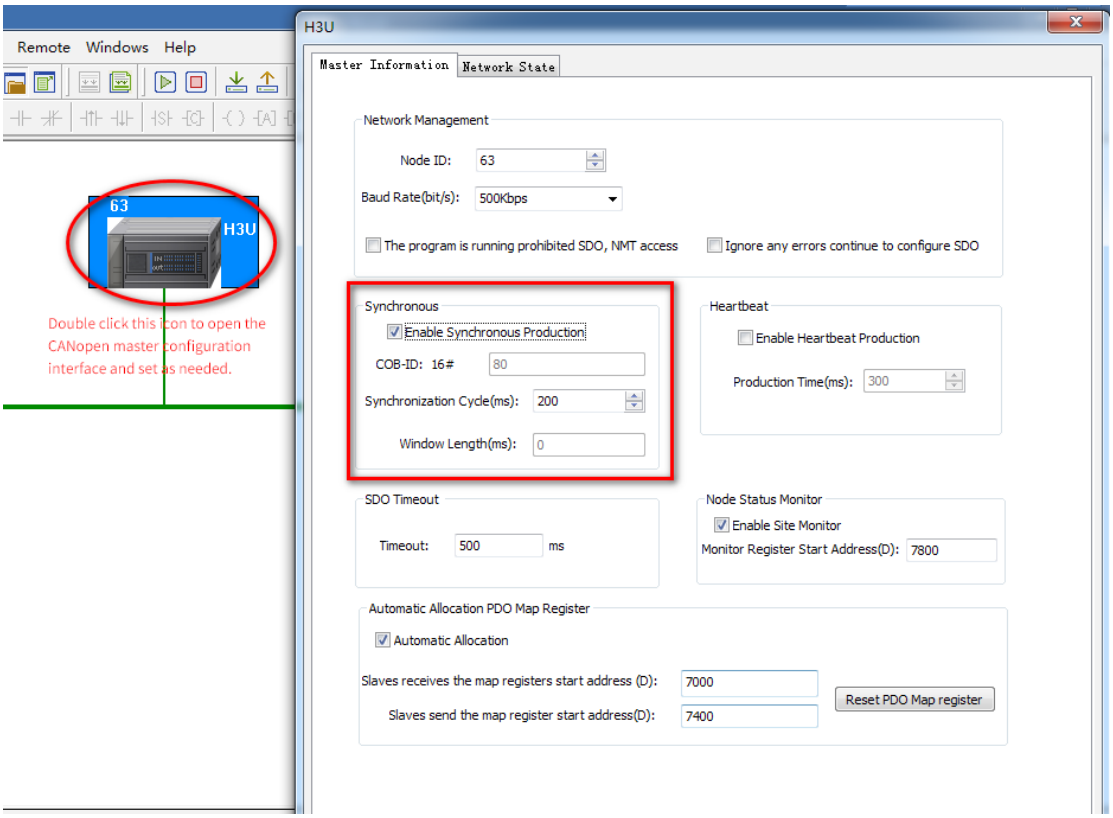
1) First, open AutoShop and double-click **CAN** protocol type in the communication port of the project management interface. Next, select the CANopen master and set **Station No.** and **Baud Rate** of the master.



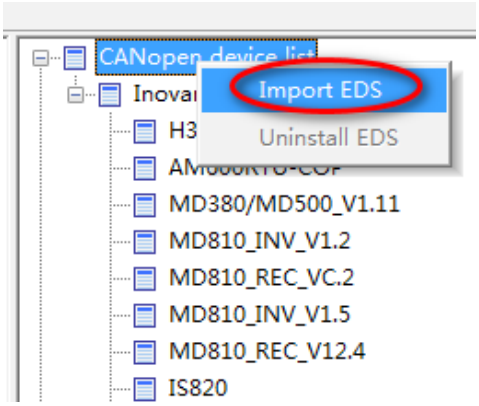
2) Right-click **CAN(CANopen)** and select **Add CAN Config** in the context menu.



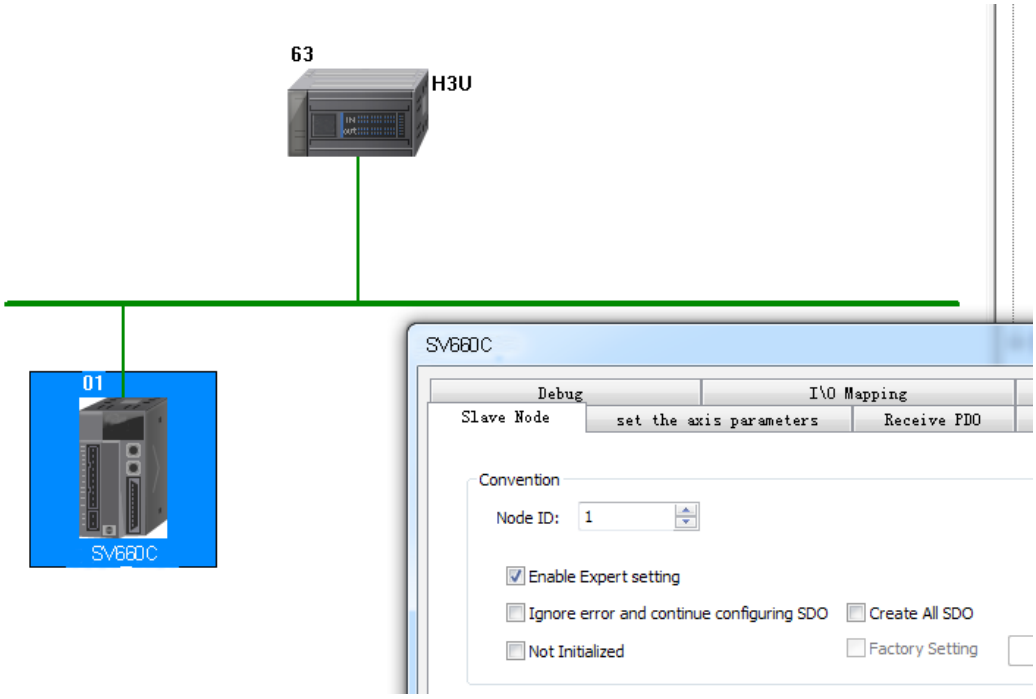
3) You can see the H3U master icon in the CANopen configuration interface. Double-click this icon to open the master configuration interface, in which you can set parameters such as synchronization and heartbeat. H3U axis-control commands control the servo drive in the PDO communication mode. The PDO adopts synchronous mode by default when the SV660C servo drive is working with an H3U master. Therefore, you need to check **Enable Synchronous Production** in this interface and set the synchronization cycle (15 ms for 8 axes generally) as needed. For other servo drive models, this option also needs to be checked if the PDO also adopts synchronous mode.



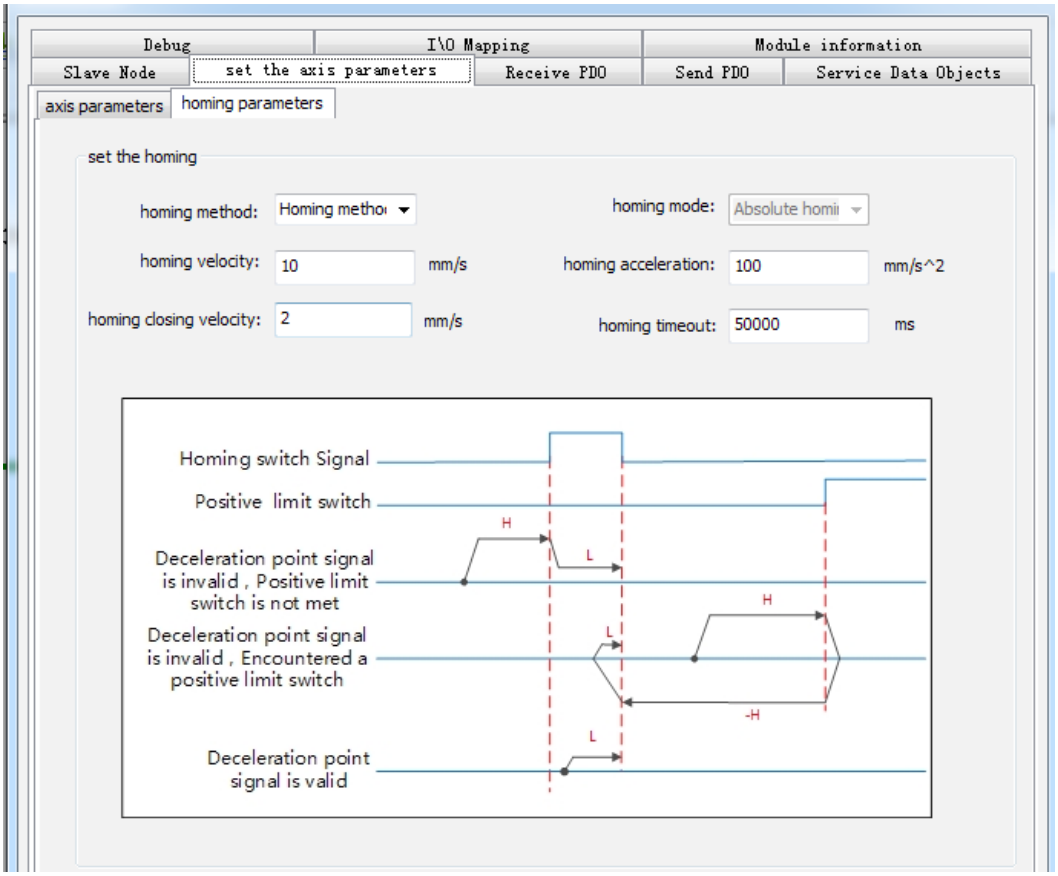
4) If the EDS files needed is not in the CANopen device list, add the device ESD needed. To do this, select the CANopen device list and right-click. In the context menu popped out, select **Import EDS**, and in the dialog box popped out, select the EDS device file needed and click **Open**. The device added will be displayed in the CANopen device list on the right.



5) Double-click the SV660C in the CANopen device list to add the slave device and CANopen slave. Then, double-click the SV660C icon in the configuration to open the slave configuration parameter list.



6) The **set the axis parameters** interface is shown as follows, which include **axis parameters** interface and **homing parameters** interface.

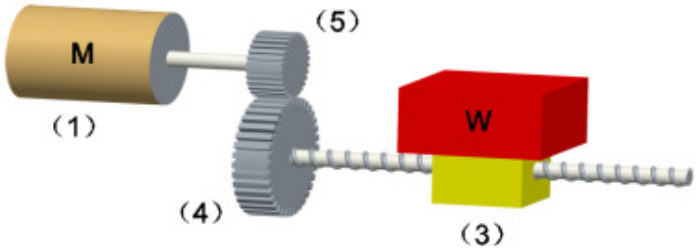


■ Axis parameters

For devices without reducers, set the gear ratio to 1:1. Set the pulses per motor revolution and distance per motor revolution correctly. The calculation formula is as follows.

$$\text{Number of pulses} = \frac{\text{Pulses per motor revolution (1)}}{\text{Travel distance per motor revolution (3)}} \times \text{travel distance (in display unit)}$$

Applications with reducers are shown as follows.



The calculation formula for devices with reducers is as follows.

$$\text{Number of pulses} = \frac{\text{Pulses per motor revolution (1)} \times \text{Motor gear ratio (5)}}{\text{Travel distance per motor revolution (3)} \times \text{Working gear ratio (4)}} \times \text{travel distance (in display unit)}$$

■ Homing parameters

The range of the homing method is 1 to 35. The calculation formula for parameters and object dictionaries of the homing speed, homing acceleration, and homing proximity speed is shown as follows.

$$\text{Object dictionary value} = \frac{\text{Pulses per motor revolution (1)} \times \text{Motor gear ratio (5)}}{\text{Travel distance per motor revolution (3)} \times \text{Working gear ratio (4)}} \times \text{setpoint in the software tool (in display unit)}$$

The relation between preceding parameters and object dictionaries is as follows.

Index	Subindex	Data Type	Description	Unit
6098h	0	SINT	Homing method	-
6099h	1	UDINT	Speed during search for switch	Reference unit/s
6099h	2	UDINT	Speed during search for zero	Reference unit/s
609Ah	0	UDINT	Homing acceleration	Reference unit/s^2
60E6h	0	USINT	Homing method	-

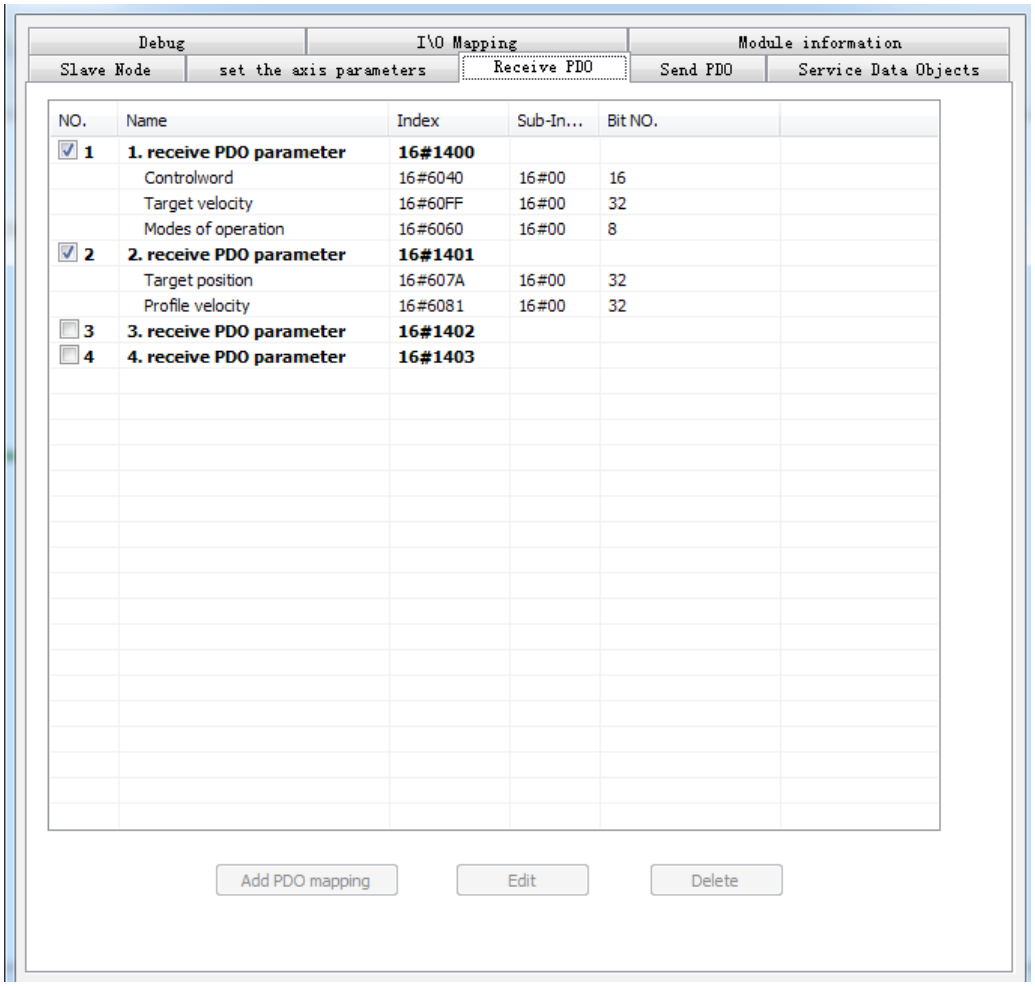
7) The object dictionaries needing to be operated in CANopen 402 motion control commands interact with the slave in the PDO mode. These object dictionaries, which include 6040h (Control word), 6041h (Status word), 6060h (Modes of operation), 6061h (Modes of operation display), 6081h (Profile velocity), 607Ah (Target position), 60FFh (Target velocity), 6064h (Position actual value), and 606Ch (Velocity actual value), must be configured as required below. Otherwise, axis configuration failure may occur during calling axis control commands.

NOTE

It is recommended to configure the PDO communication to synchronous mode to prevent frame loss caused by interference during communication. The synchronous mode requires synchronous production to be enabled in the master configuration. To ensure communication stability, ensure the network load rate is below 70%.

$$\text{Network load rate} = \frac{328 \times \text{Number of axes} + 79}{\text{Baud rate} \times \text{Synchronous cycle}} \times 100\%$$

Configure the RPDOs.

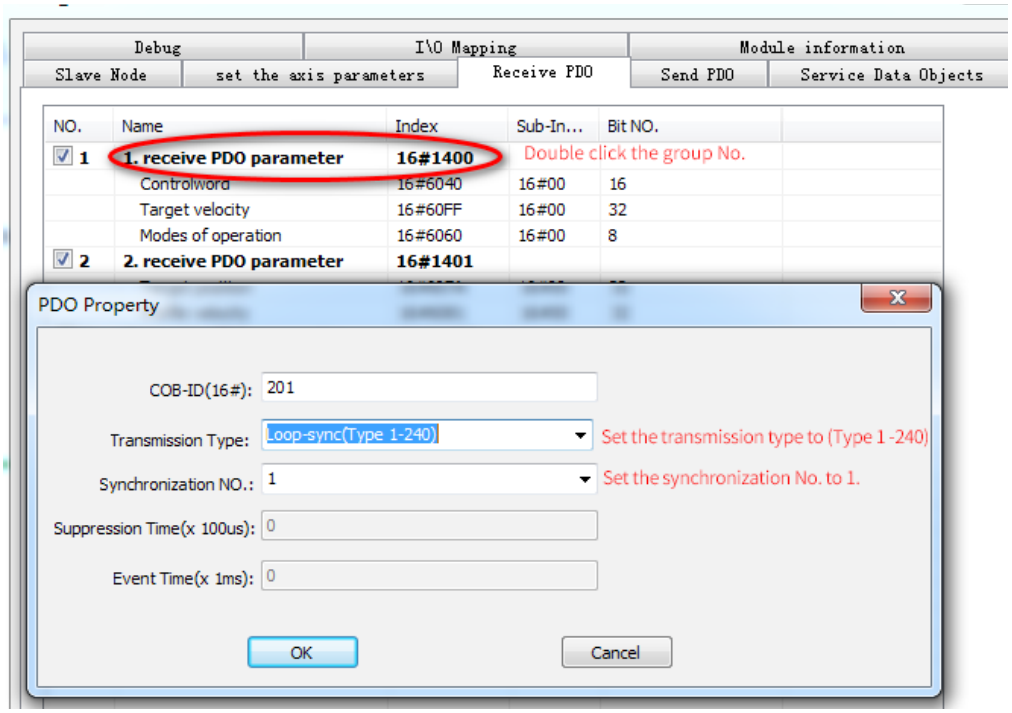


Configure the RPDOs in the following sequence.

Index	Subindex	Name
6040h	0	Control word
60FFh[1]	0	Target velocity
6060h	0	Modes of operation
607Ah	0	Target position
6081h	0	Target velocity

[1] When MCMOVVEL and MCJOG are not in use，this object dictionary can be replaced by other object dictionaries with a length of 0x20.

It is recommended to use synchronous mode for PDO communication. The method for setting synchronous PDO communication of the slave is as follows.



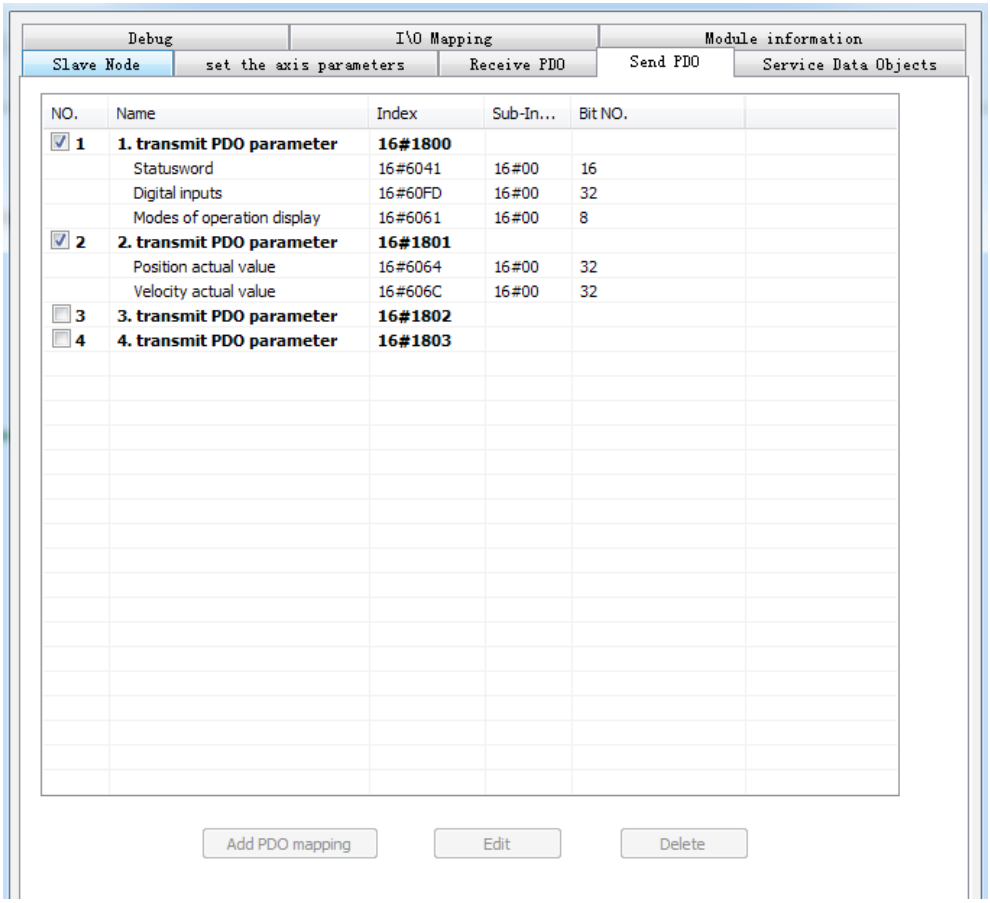
NOTE

① : Double-click the group No. to display a dialog box.

② : Set the transmission type to "Type1-240".

③ : Set the synchronization NO. to 1.

Configure the TPDOs.



Configure the TPDOs in the following sequence.

Index	Subindex	Name
6041h	0	Status word
60FDh[1]	0	Digital inputs
6061h	0	Modes of operation
6064h[2]	0	Position actual value
606ch	0	Velocity actual value

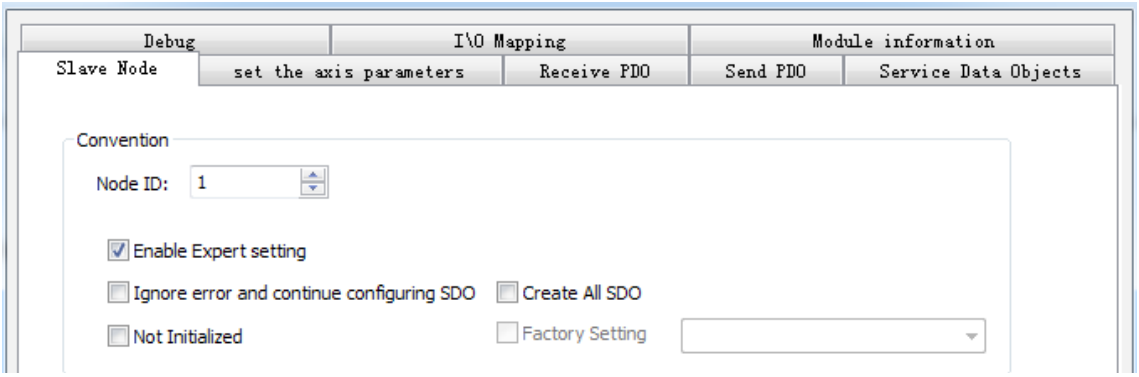
[1] This object dictionary can be replaced by other object dictionaries with a length of 0x20.

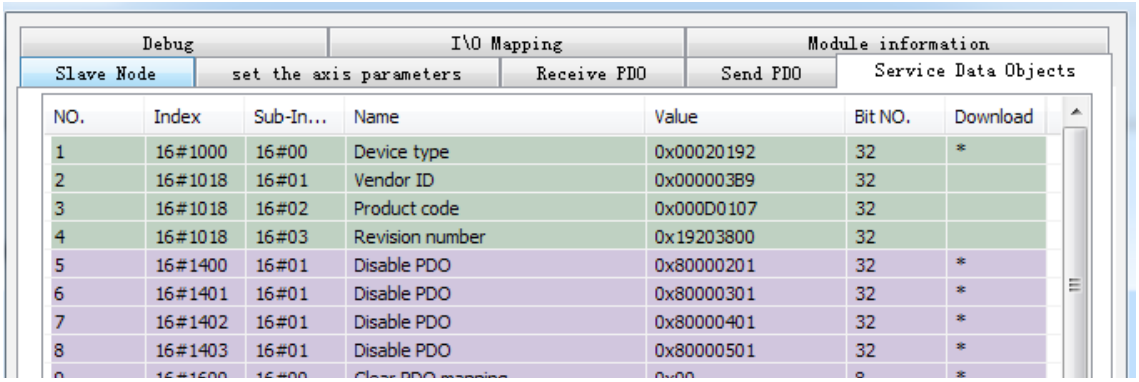
[2] This object can be replaced by 60FCh.

The setting mode of TPDOs is similar to that of RPDOs.

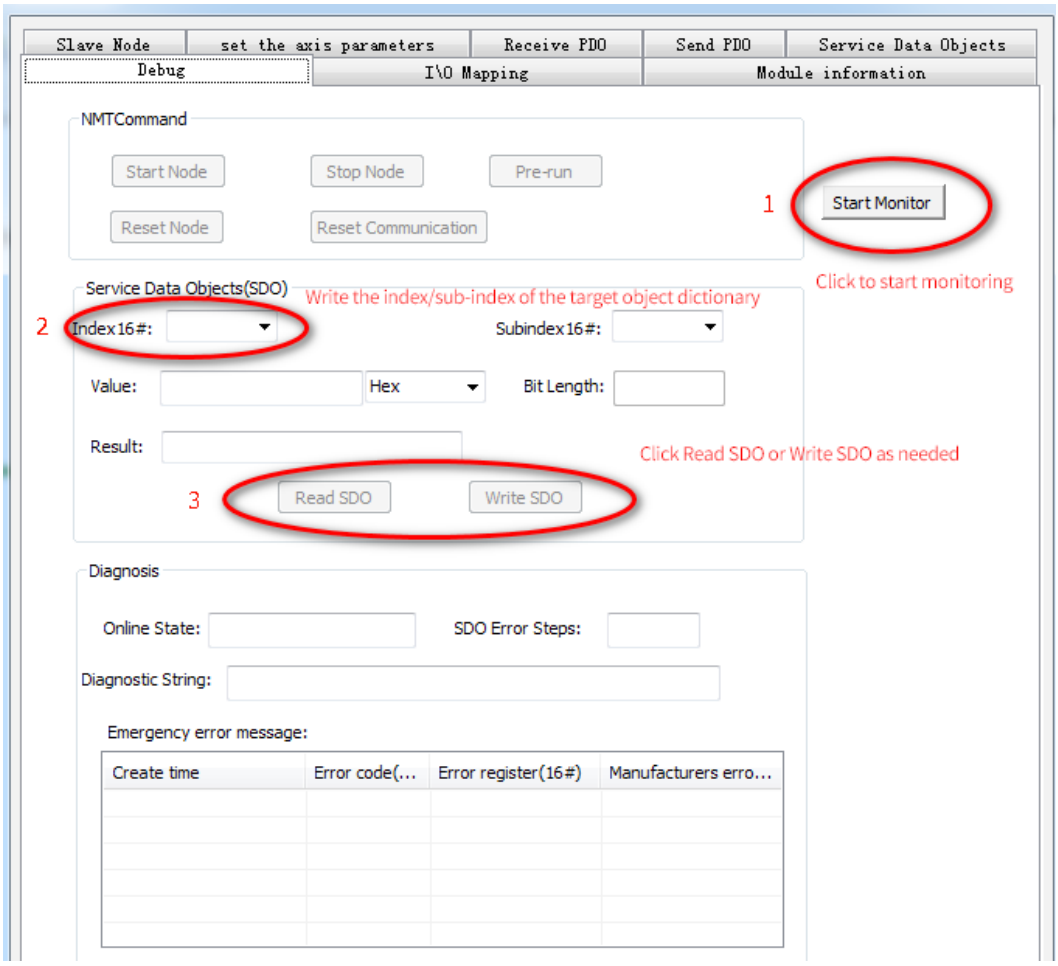
The EDS is configured based on the preceding sequence by default. Pay attention to the preceding configuration sequence when adding new objects. A wrong sequence will cause failure of H3U axis control commands. The preceding configuration sequence does not apply to PLCs made by other manufacturers.

8) Download the CANopen configuration to H3U. The H3U starts slave configuration based on the pre-defined configurations. The configuration is performed based on the object dictionaries listed in the Servo Data Object interface. To view this list, check **Enable Expert setting** in the Slave Node interface first.





Monitoring on the device state and reading/writing of the slave object dictionaries are available during commissioning.



NOTE

Step 1: Click **Start Monitor**.

Step 2: Write the index/subindex of the object dictionary to be operated in the input box of **Index16#**.

Step 3: Click **Read SDO** or **Write SDO** as needed.

Shenzhen Inovance Technology Co., Ltd.
Add.: Building E, Hongwei Industry Park, Liuxian Road, Baocheng No. 70 Zone, Bao'an District, Shenzhen
Tel: +86-755-2979 9595
Fax: +86-755-2961 9897
http: //www.inovance.com

Suzhou Inovance Technology Co., Ltd.
Add.: No. 16 Youxiang Road, Yuexi Town, Wuzhong District, Suzhou 215104, P.R. China
Tel: +86-512-6637 6666
Fax: +86-512-6285 6720
http: //www.inovance.com

19011302A02