INOVANCE

Application Manual - CANopen Communication

IS620P Series

Servo Drive

User Guide

A01

Data code 19010699

Foreword

First of all, thank you for purchasing Inovance IS620P series of servo drive configured with the CANopen field bus function.

Based on the general IS620P servo drive of Inovance, the IS620P-CANopen product adds the CANopen bus communication function, which covers all products of the series. After the CANopen communication function is added, the product can be connected to the high-speed CANopen communication network and implement bus control on site.

This manual introduces applications related to the CANopen function. For other general functions, see the *IS620P Series Servo Design and Maintenance User Manual*. Contact our technical personnel if you have any question during use of CANopen.

Inovance commits itself to continuously improving the servo drive. Therefore, this document is subject to change without notice.

Notes
◆ To describe product details, this manual provides diagrams showing the status without a housing or safety cover. Before using this product, install the housing or safety cover as required and refer to the instructions in this manual.
◆ Diagrams in this document are used only for function description. The product structure shown in the diagrams may be different from the structure of the product that you purchase.
◆ When the product is upgraded or the specifications change, this manual will be updated in a timely manner to improve its accessibility and accuracy.
◆ If you need to purchase this manual in case that the original copy is damaged or lost, contact our local agent or our customer service center.
◆ If you have any questions regarding the usage of this product, contact our customer service center.
◆ Service hotline: 400-777-1260

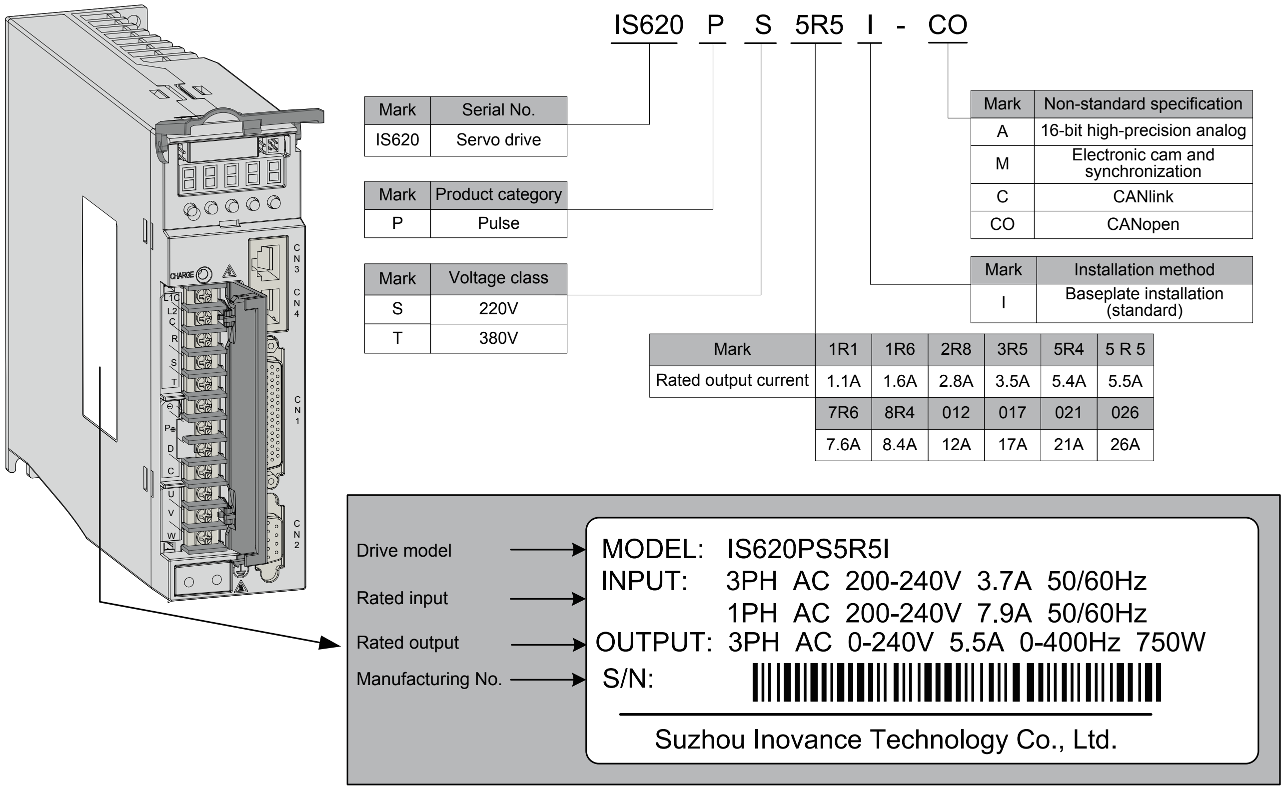
Table of Contents

Chapter 1 Product Information

Chapter 1 Product Information

1.1 Nameplate and Model of Servo Drive

Figure 1-1 Naming and nameplate description of servo drive



1.2 Integrated Performance Parameters

Table 1-1 Integrated 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 number of sites	63
CAN frame length	0 to 8 bytes
Application layer CAN frame type	Data frame, remote frame
Terminal matching resistance	120 Ω
Supported sub-protocol	CiA-301 V4.02: CANopen application layer and communication protocol DSP-402 V2.0: driver and motion control sub-protocol

Item	Description
Supported service	NMT: network management system SDO: service data object PDO: process data object Device monitoring: including node protection and heartbeat SYNC: including synchronization generator and synchronous receiving, which is used in PDO transmission
PDO transmission type	Time trigger, event trigger, synchronous trigger
Number of supported PDOs	4 RPDOs, 4 TPDOs
SDO transmission mode	Accelerated SDO transmission, fragmented SDO transmission
Supported servo drive mode	Profile position mode Profile speed mode Homing mode Interpolated position mode

The CANopen communication function of the IS620P series of servo drive supports seven baud rates. The communication distance and baud rate are related to communication cables.

Table 1-2 Supported baud rates

Baud rate (bps)	1M	500K	250K	125K	100K	50K	20K
Length (m)	25	100	250	500	500	1,000	1,000

Table 1-3 Relationship between CAN communication transmission distance, rate, and number of 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

For CAN communication, cables of different diameters have little impact on the transmission distance. However, cable diameters must be large. Table 1-4 lists the transmission distance between two nodes under different cable diameters and rates.

Table 1-4 Relationship between cable diameters and transmission distance

Cable diameter	500 Kbps	1 Mbps
3×0.3 mm^2	95 m	30 m
3×0.5 mm^2	95 m	30 m
3×0.75 mm^2	100 m	30 m

Memo NO. __________

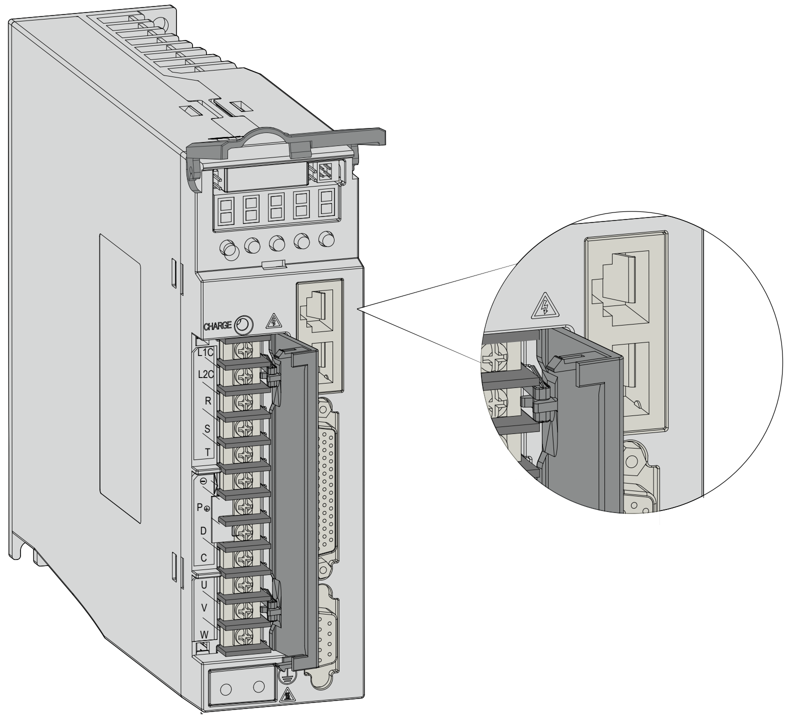
Date / /

Chapter 2 Wiring

Chapter 2 Wiring

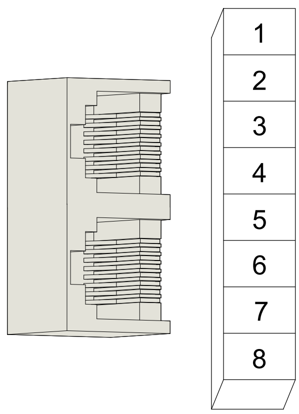
The two RJ45 terminals of the ISP620 servo drive are CANopen communication ports and CN3 and CN4 ports of general products. Figure 2-1 shows the ports.

Figure 2-1 CANopen communication ports of the IS620P servo drive



Pins of the two ports are internally connected. Table 2-1 lists definitions of the pins. CAN interface connectors are configured with at least the CANH, CANL, and CGND pins.

Table 2-1 Pin definitions of communication signal connectors

Pin No.	Pin	Description	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 transmit end, which is connected to the receiving end of the host computer	
7	RS232-RXD	RS232 receiving end, which is connected to the transmit end of the host computer	
8	GND	Ground	
Housing	PE	Shielding	

Connecting CGND greatly helps improve the anti-interference performance of CAN ports.

2.1 CAN Communication Cable

1) CAN cable for communicating with PLC

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

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

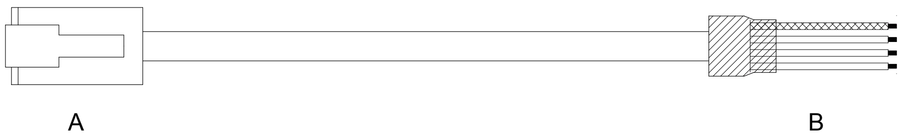


Table 2-2 Pin connections of the communication cable (model: S6-L-T02-2.0) between the PLC and the servo drive

RJ45 on 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 network layer)	Housing		PE (shield network layer)	Housing

2) CAN communication cable for multiple drives connected in parallel

The following figure shows the connecting cable (model: S6-L-T01-0.3) for multiple drives connected in parallel under CAN communication:

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

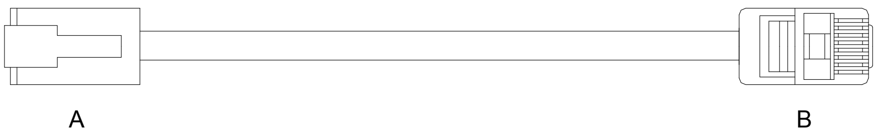
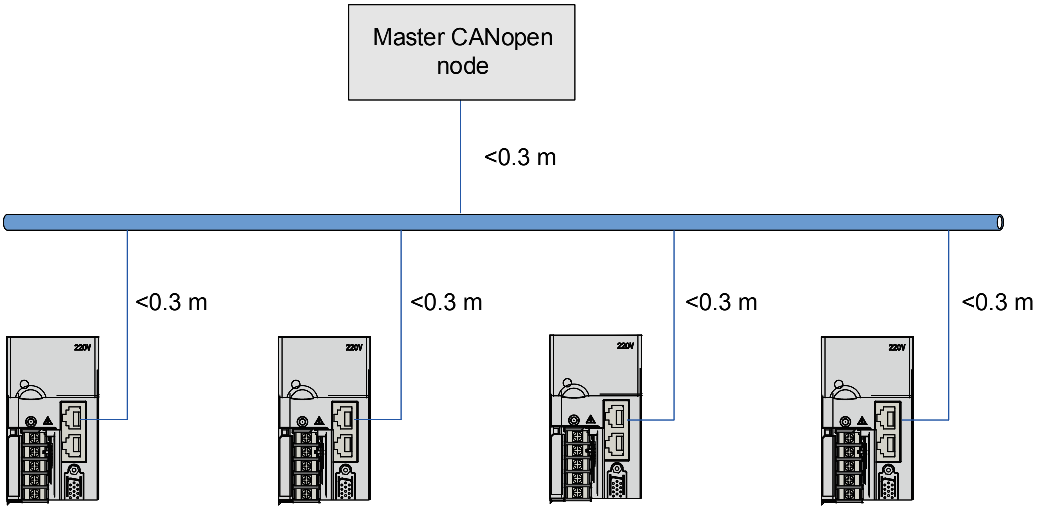


Table 2-3 Pin connections of the communication cable (model: S6-L-T01-0.3) for multiple drives connected in parallel

RJ45 on Servo Drive Side (A)			RJ45 on 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 network layer)	Housing		PE (shield network layer)	Housing

2.2 CAN Communication Bus and Multi-node Connection Mode

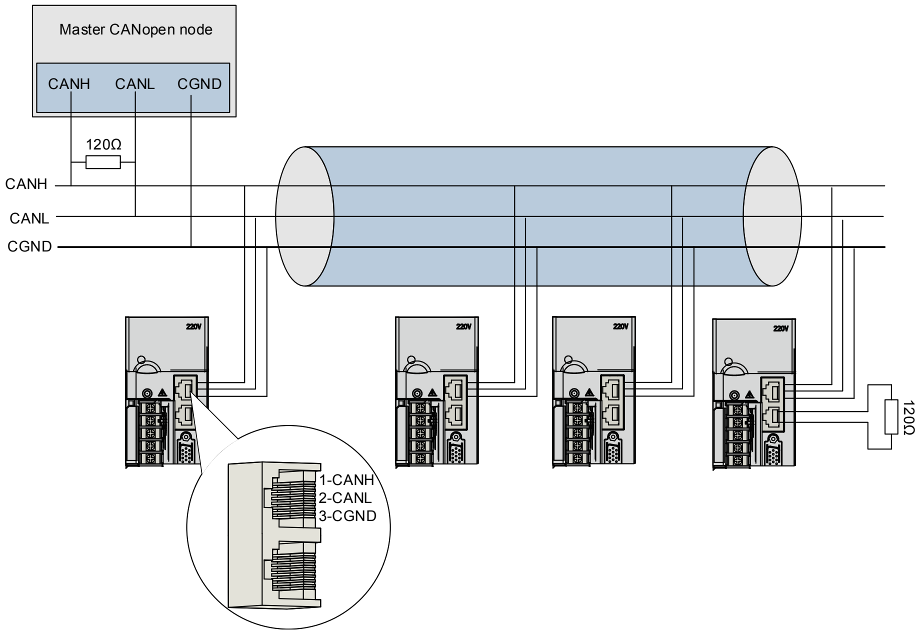
Figure 2-4 CAN communication network topology



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

CAN transmitters and receivers are mounted on the bus. Each branch must be shorter than 0.3 m. Otherwise, reflection is caused and communication problems occur.

Figure 2-5 Schematic diagram of CANopen cable

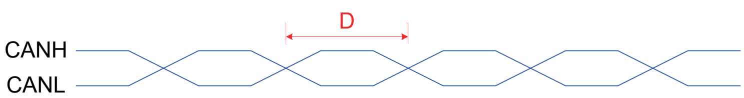


- It is recommended that a shielded twisted cable is connected to the bus. A 120 Ω terminal matching resistor is connected to each end of the bus to prevent signal reflection. The shield layer generally uses reliable single-point grounding.
- Use a multimeter to measure the resistance between CANH and CANL to determine whether the receiving resistance on site is correct. The normal resistance value is around 60 Ω (two resistors are connected in parallel).
- Up to 64 devices can be mounted under the bus.
- When CAN devices communicate over a long distance, CGND of different CAN circuits must be mutually connected to ensure that the reference potential of different communication devices is the same.

2.3 Twisted Pair Cables Recommended for CAN Communication Cables

- The CAN communication network recommends using twisted pair cables, which can better resist high-frequency magnetic field noise interference and reduce external radiation of cables. Figure 2 shows the schematic diagram of a twisted pair cable.

Figure 2-6 Schematic diagram of a twisted pair cable



- The torque D of a twisted pair cable should be smaller than 2 cm. Smaller torque indicates better anti-interference effect.
- During short-distance low-speed communication, a twisted pair shielded cable can be used to enhance the anti-interference capability. Both ends of the shield layer are connected to the PE.
- During long-distance high-speed communication, shielded cables are not recommended. This is because large capacitance exists between the shield layer and the signal cable, which cause delay of transmission signals.

2.4 Recommended Connection Modes of Different Cables

Figure 2-7 Recommended connection mode 1

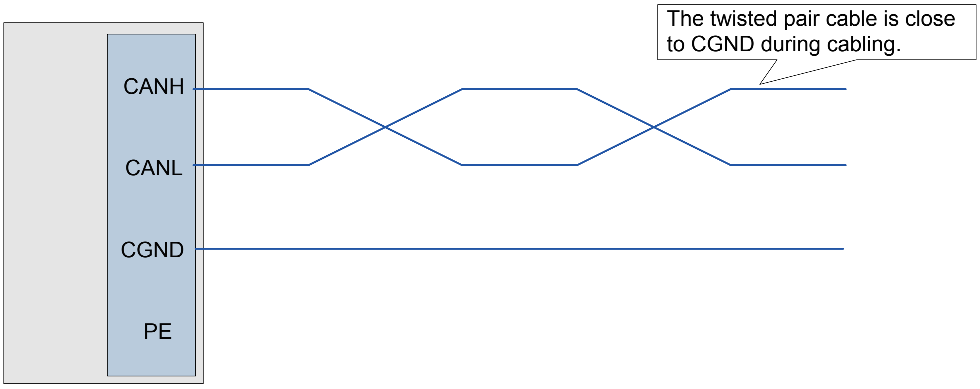


Figure 2-8 Recommended connection mode 2

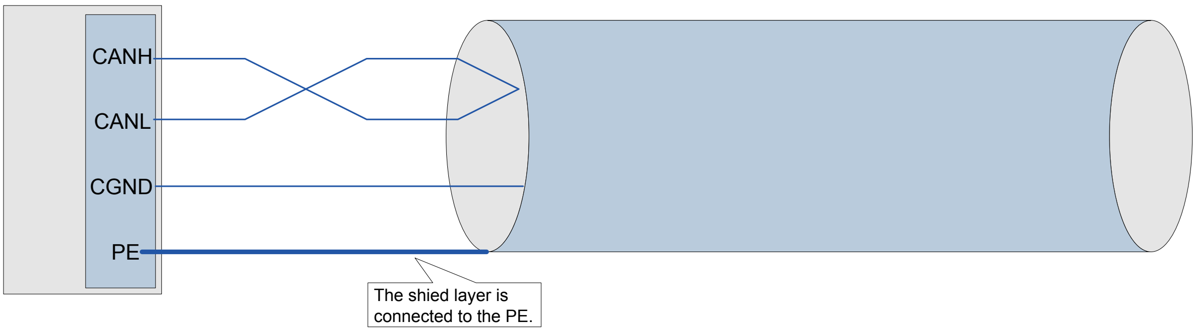


Figure 2-9 Recommended connection mode 3

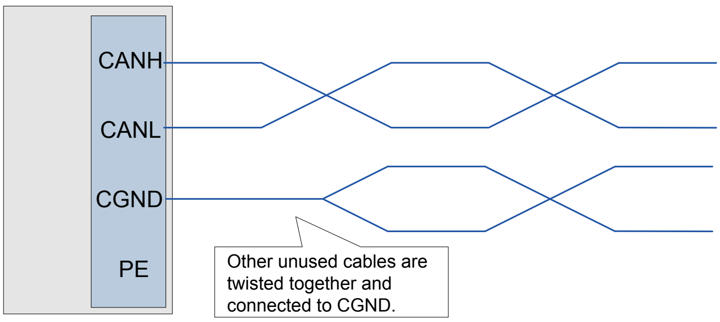


Figure 2-10 Recommended connection mode 4

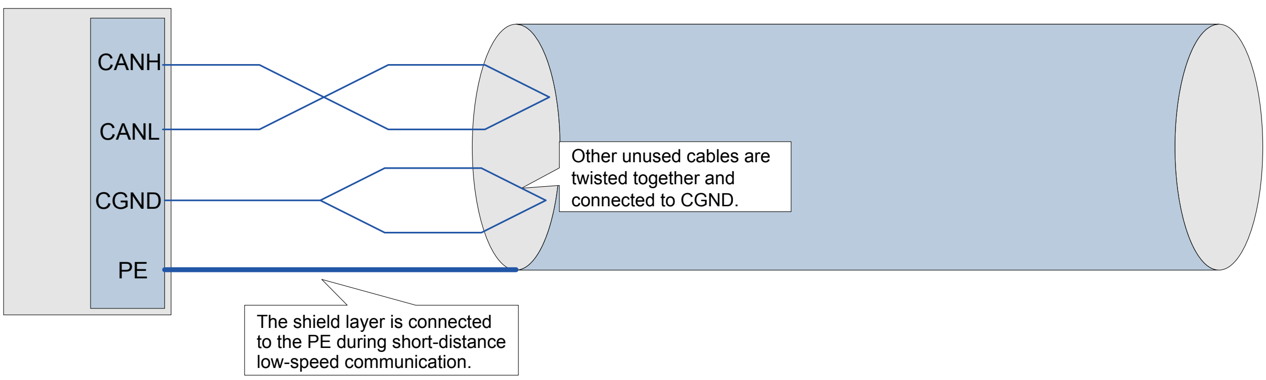
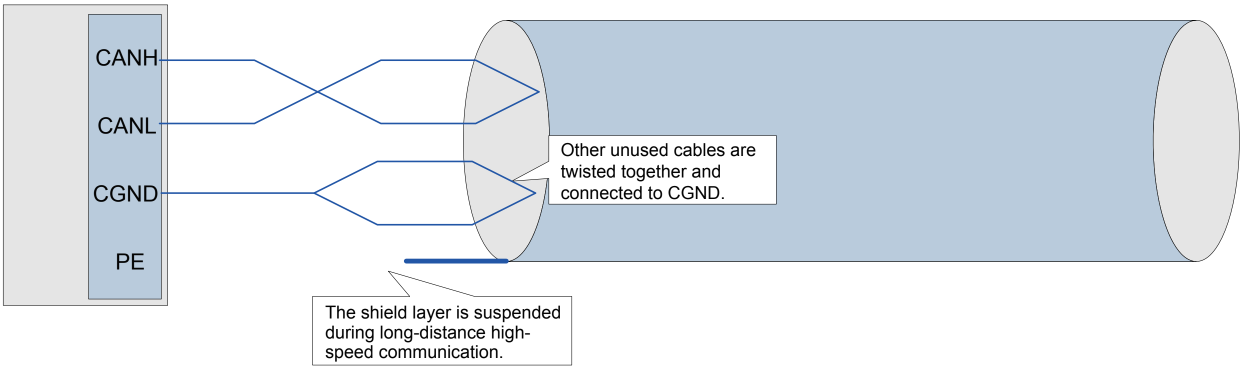


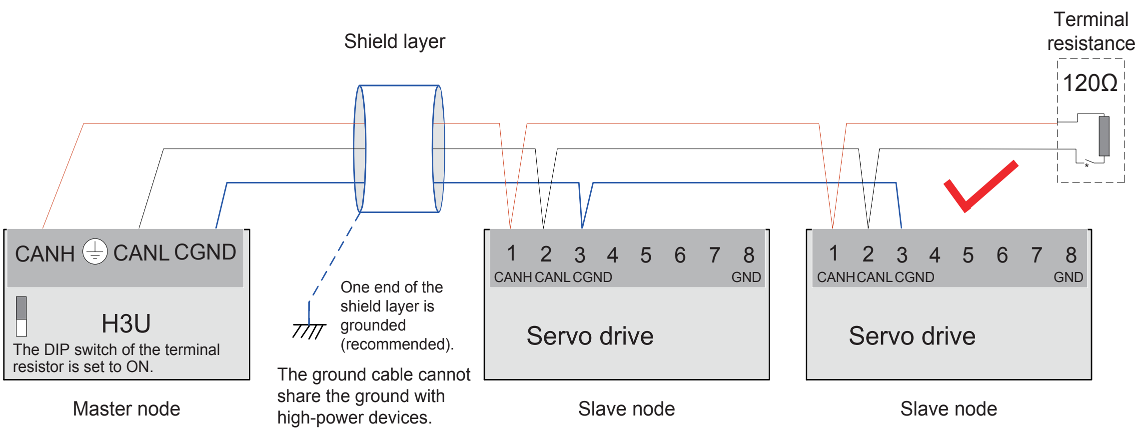
Figure 2-11 Recommended connection mode 5



2.5 Precautions for Grounding during CAN Communication

When CAN communication is used, the CGND terminal of the host computer must be connected to the CGND terminal of the servo drive, as shown in Figure 2-12.

Figure 2-12 Correct CAN connection method

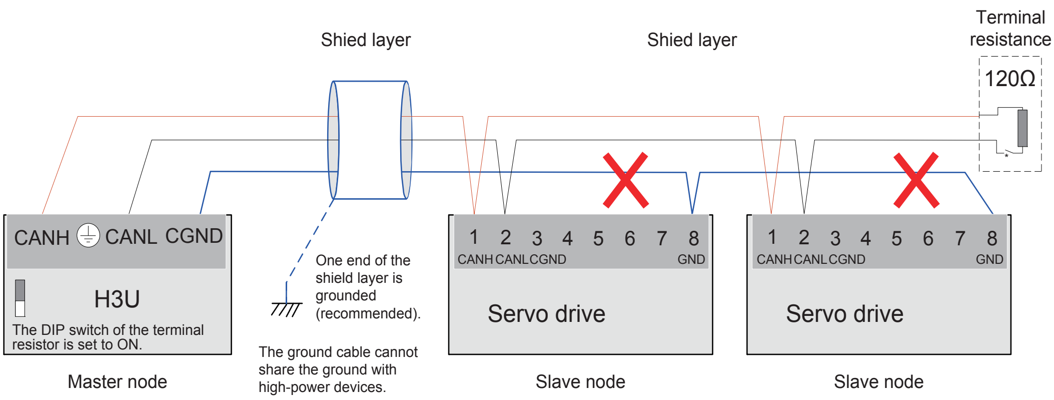


Note: 1. A CAN communication terminal resistor is embedded in the PLC and therefore the corresponding DIP switch must be set to ON.

2. It is recommended that the shield layer is grounded at one end.

Do not connect the CGND terminal of the host computer to the CGND terminal of the servo drive. Otherwise, the devices are damaged.

Figure 2-13 Wrong CAN connection method

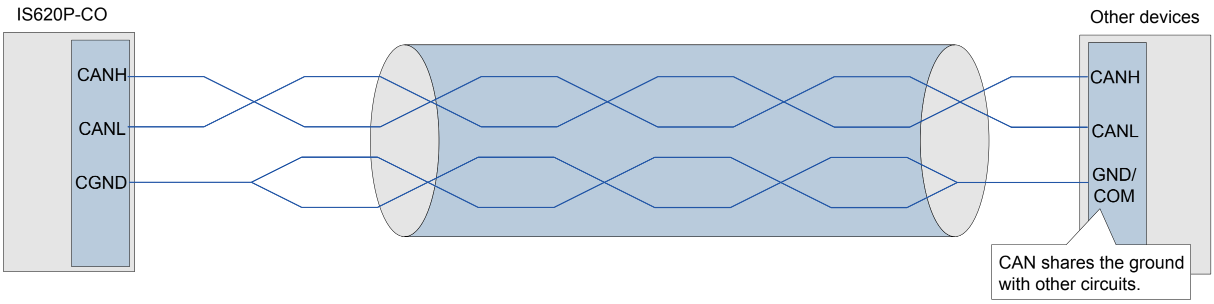


2.6 Description of Wiring of Other Devices without External CGND Port

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 CGND of Inovance devices, as shown in Figure 2-8.

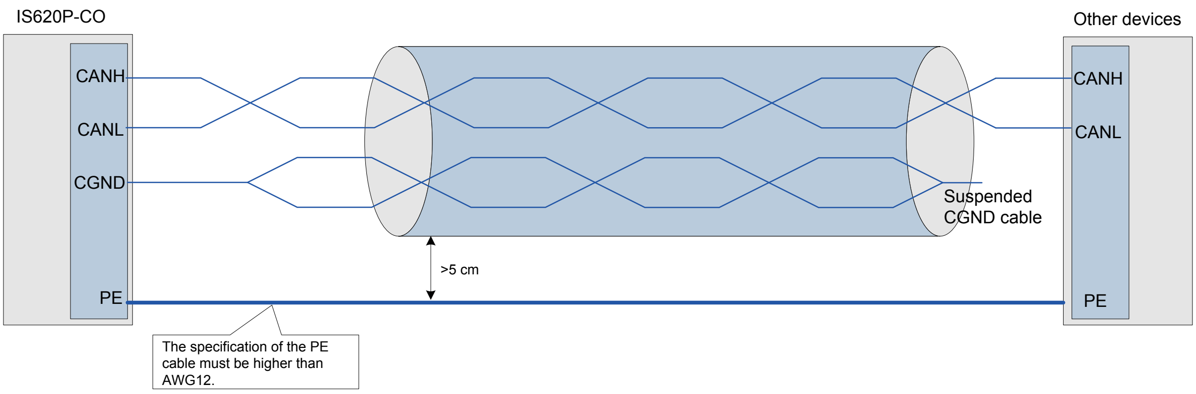
Figure 2-14 Connection mode for sharing the ground with other circuits



2.6.2 No CGND for CAN and Other Ports of Devices

CGND is not connected to any cable. A cable that is not smaller than AWG12 is used to connect PEs of devices. The cable is more than 5 cm away from the CAN communication cable, as shown in Figure 2-9.

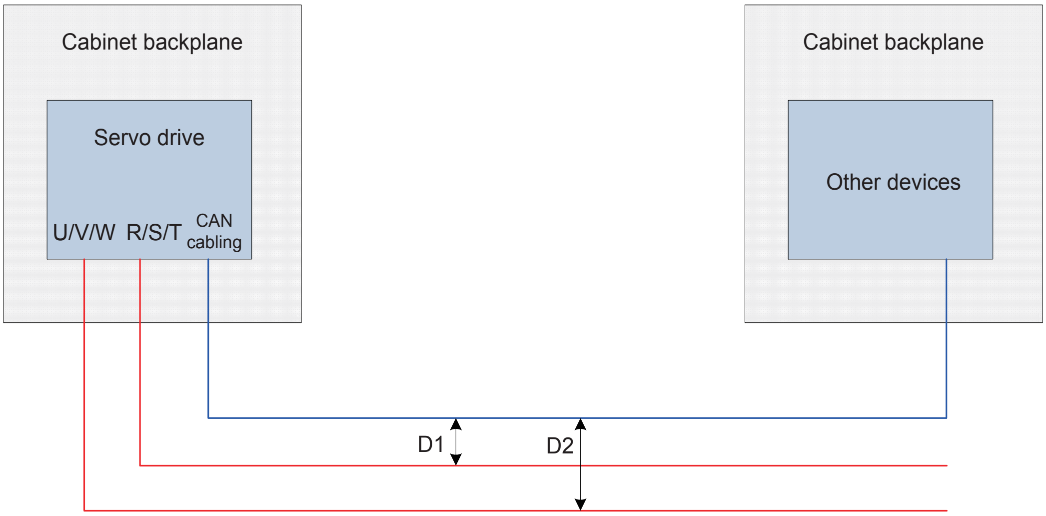
Figure 2-15 CAN of other devices without port for connecting GND



2.7 Recommended Layout of CAN Communication Cables

CAN communication is susceptible to interference. If field layout is close to interference sources, problems may occur.

Figure 2-16 Recommended cabling mode



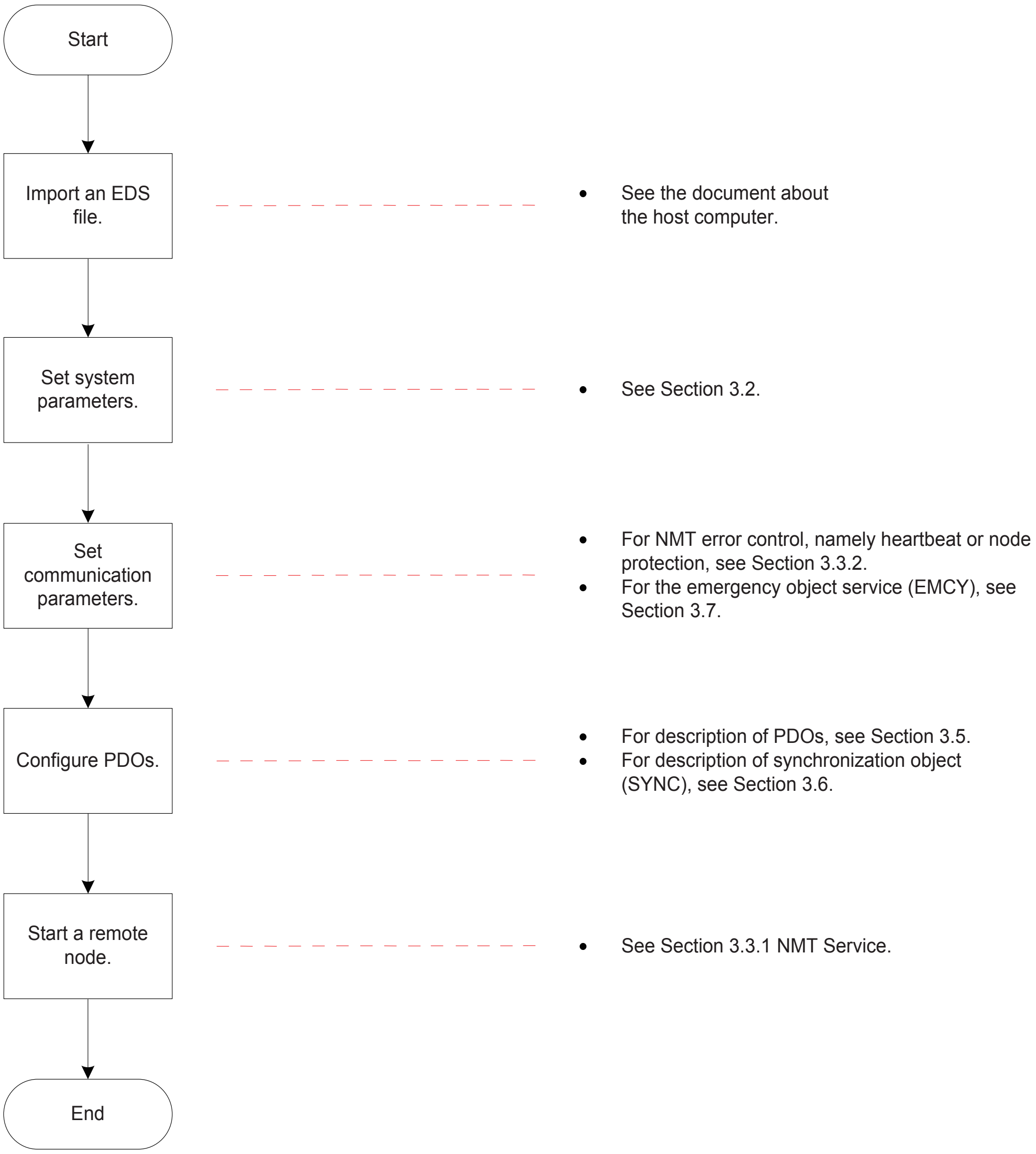
- Interference cables and CAN cables should be deployed along the vertical direction. During parallel cabling, the distance D1 between the R/S/T cable and the CAN signal cable must be longer than 20 cm and the distance D2 between the U/V/W cable and the CAN signal cable must be longer than 50 cm. If interference cables are deployed closely along the backplane of the cabinet, the distance between the CAN communication cable and the backplane of the cabinet must be longer than 1 cm.
- After cables are led out of the cabinet, the R/S/T power cable, U/V/W power cable, and CAN communication cable are deployed respectively in three cable troughs. The distance L3 between cable troughs must be longer than 20 cm. If interference cables and CAN communication cables are deployed in the same cable trough, the preceding principles are followed for the distance between the cables.

Chapter 3 Communication Network Configuration

Chapter 3 Communication Network Configuration

The following figure shows the CANopen use and setting flowchart:

Figure 3-1 CANopen use and setting flowchart



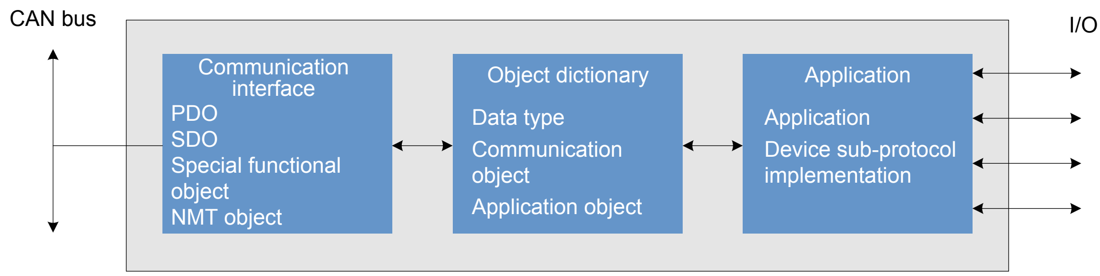
NOTE

Note 1: For details on how to use SDOs, see "3.1 Overview of the CANopen Protocol" on Page 20.

3.1 Overview of the CANopen Protocol

CANopen is an application layer protocol of the network transmission system based on CAN serial bus. It complies with the ISO/OSI standard model. Different devices in the network exchange data through the object dictionary or objects. The master node obtains or modifies data in the object dictionaries of other nodes through PDOs or SDOs. Figure 3-2 shows the CANopen device model.

Figure 3-2 Schematic diagram of CANopen device model



3.1.1 Object Dictionary

Object dictionary is the most important part in device specifications. It is an ordered set of parameters and variables and includes all parameters about device description and device network status. A group of objects can be accessed in an ordered and pre-defined way through the network.

The CANopen protocol adopts an object dictionary with a 16-bit index and an 8-bit index. Table 3-1 describes the structure of the object dictionary.

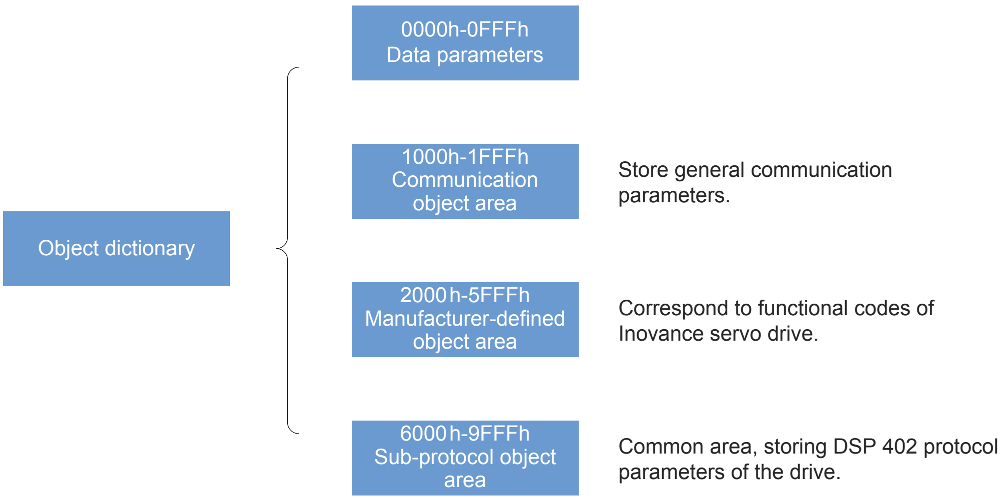
Table 3-1 Structure of the object dictionary

Index	Object
000	Not used
0001h—001Fh	Static data type (standard data type, for example, Boolean and Integer16)
0020h—003Fh	Complex data type (predefined structure consisting of simple types, for example, PDOCommPar and SDOParmeter)
0040h—005Fh	Complex data type specified by the manufacturer
0060h—007Fh	Static data type specified by the device sub-protocol
0080h—009Fh	Complex data type specified by the device sub-protocol
00A0h—0FFFh	Reserved
1000h—1FFFh	Communication sub-protocol area (for example, device type, error register, and number of supported PDOs)
2000h—5FFFh	Sub-protocol area specified by the manufacturer (for example, functional code mapping)
6000h—9FFFh	Standard device sub-protocol area (for example, DSP-402 protocol)
A000h—FFFFh	Reserved

3

Communication Net work Configuration

Figure 3-3 Structure of CANopen object dictionary



The mapping between functional codes of Inovance servo drive and the object dictionary is as follows:

Object dictionary index = 0x2000 + Functional code group No.

Object dictionary sub-index = Hexadecimal of offset in the functional code group + 1

Example:

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

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

Example:

For example, the object 607Dh configured with software position limit describes the minimum position limit and the maximum position limit. The object is defined as follows:

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

Index	Sub-index	Description	Meaning
607Dh	00h	Number of sub-indexes for software absolute position limit	Quantity of object data, not including the object
607Dh	01h	Minimum software absolute position limit	Minimum position limit (in absolute position mode)
607Dh	02h	Maximum software absolute position limit	Maximum position limit (in absolute position mode)

3.1.2 Common Communication Objects

1) 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 nodes in the network. It implements node status control, error control, and node startup.

2) SDO

- An SDO includes a receiving SDO (R-SDO) and a transmitting SDO (T-SDO).
- By using indexes and sub-indexes, SDOs enable clients to access entries in the object dictionary of devices.
- SDO is implemented through multi-domain CMS objects in the CAL and allows transmitting data of any length. When the data size exceeds four bytes, the data is segmented into several packets.
- The protocol confirms the service type and generates a response for each message. An SDO request and a response packet always contain eight bytes.

3) PDO

- A PDO includes a receiving PDO (RPDO) and a transmitting PDO (TPDO).
- A PDO is used to transmit real-time data from one creator to one or multiple receivers. The length of transmitted data ranges from one to eight bytes.
- Each CANopen device has eight default PDO channels, that is, four TPDO channels and four RPDO channels.
- PDOs support synchronous transmission and asynchronous transmission. Which transmission mode is used is determined by PDO communication parameters.
- The content of a PDO message is predefined and is determined by PDO mapping parameters.

4) Synchronization object (SYNC)

A synchronization object is a packet periodically broadcast by the CANopen master station to the CAN bus and is used to provide basic network clock signals. Each device determines whether to use the object to synchronize with other network devices based on its configuration.

5) Emergency packet (EMCY)

In the case of a communication failure or application failure, an emergency packet is sent.

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, namely a 4-bit functional code and a 7-bit node address. Table 3-3 describes the COB-ID.

Table 3-3 Composition of COB-ID

10	9	8	7	6	5	4	3	2	1	0
Functional code				Node ID						

Each CANopen communication object has its default COB-ID, which can be read through SDO. Some COB-IDs can be modified through SDO. Table 3-4 lists COB-IDs.

Table 3-4 COB-IDs

Communication Object	Functional Code	Node Address	COB-ID	Object Index
Network management	0000b	0	0h	-
Synchronization object	0001b	0	80h	1005h, 1006h
Emergency packet 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
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
Network management error control	1110b	1 to 127	700h + Node ID	1016h, 1017h

Example:

The COB-ID of TPDO2 of slave station 4 is 284h (=280h+4).

3.2 System Settings

Related functional codes of the IS620P servo drive must be set so that the servo drive can access the CANopen field bus network correctly.

Table 3-5 Functional codes in system settings

Functional Code		Name	Range	Factory Default
H02	00	Control mode selection	0: Speed mode 1: Position mode 2: Torque mode 3: Speed mode and torque mode 4: Position mode and speed mode 5: Position mode and torque mode 6: Position mode, speed mode, and torque mode 8: CANopen mode	8
H0C	00	Servo axis address	1 to 127	1
H0C	08	CAN communication rate selection	0: 20K 1: 50K 2: 100K 3: 125K 4: 250K 5: 500K 6: 1M 7: 1M	5
H0C	13	Storage to EEPROM	0: Not save 1: Save functional code 2: Save parameter 402 3: Save all parameters	0

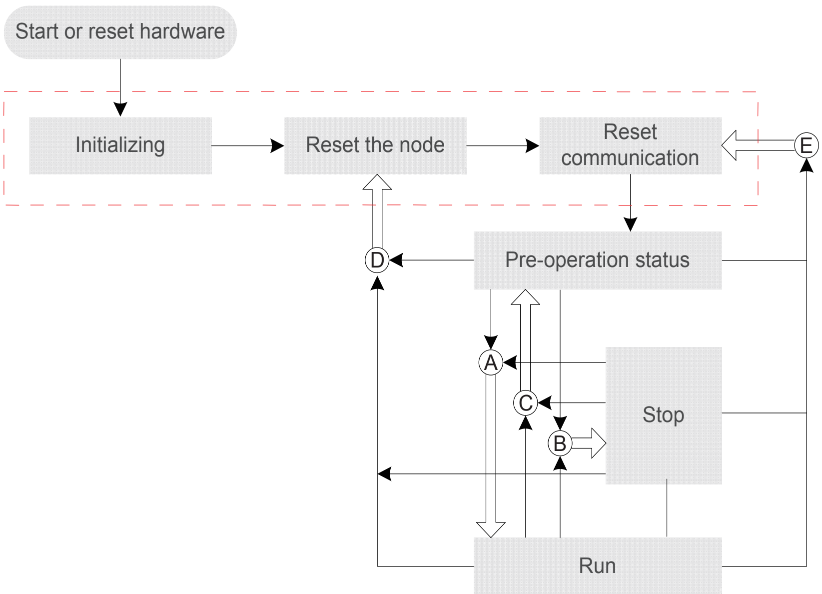
3.3 Network Management System

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

3.3.1 NMT Service

CANopen works according to the state machine specified by the protocol. Some data is automatically converted inside and some data must be converted by the master NMT node by sending packets.

Figure 3-4 NMT state machine



In Figure 3-4, conversion marked with a letter is implemented through NMT packets and only the master NMT node can send NMT control packets. Table 3-6 describes the packet format.

Table 3-6 NMT packet format

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

The COB-ID of an NMT packet is permanently “0x000”.

The data area consists of two bytes. The first byte is a command word that indicates the control role of the frame. Table 3-7 describes the command word.

The second byte is the CANopen node address. When the byte is 0, the byte is a broadcast message and all slave devices in the network are valid.

Table 3-7 Command words of an NMT packet

Command Word	Conversion Code	Description
0x01	A	Instruction for starting a remote node
0x02	B	Instruction for stopping a remote node

Command Word	Conversion Code	Description
0x80	C	Instruction for entering the pre-operation status
0x81	D	Instruction for resetting a node
0x82	E	Instruction for resetting communication

After being powered on, the device automatically enters the initialization status, including initializing, resetting node, and resetting communication. The device is initializing and loading parameters of modules. After the node is reset, the area defined by the object dictionary manufacturer and the sub-protocol area are restored to the values saved previously. After communication is reset, communication parameters in the object dictionary are restored to the values saved previously.

Later, the device sends Boot-up and automatically enters the operation status. The main configuration nodes are in this status.

After configuration is complete, the node needs to send an NMT packet to enter the operation status. When CANopen is working properly, CANopen is in the operation status. All modules should work properly.

When the master NMT node sends a stop node packet, the device enters the stop status. In CANopen communication, only the NMT module is working properly.

Table 3-8 lists CANopen services under various NMT status.

Table 3-8 CANopen services under various NMT status

Service	Pre-operation	Operation	Stop
PDO	No	Yes	No
SDO	Yes	Yes	No
Synchronization object (SYNC)	Yes	Yes	No
Emergency packet (EMCY)	Yes	Yes	No
NMT	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 detect the status of devices, including node protection, life protection, and heartbeat.

NOTE

Note: 1. Life protection and heartbeat cannot be used at the same time.

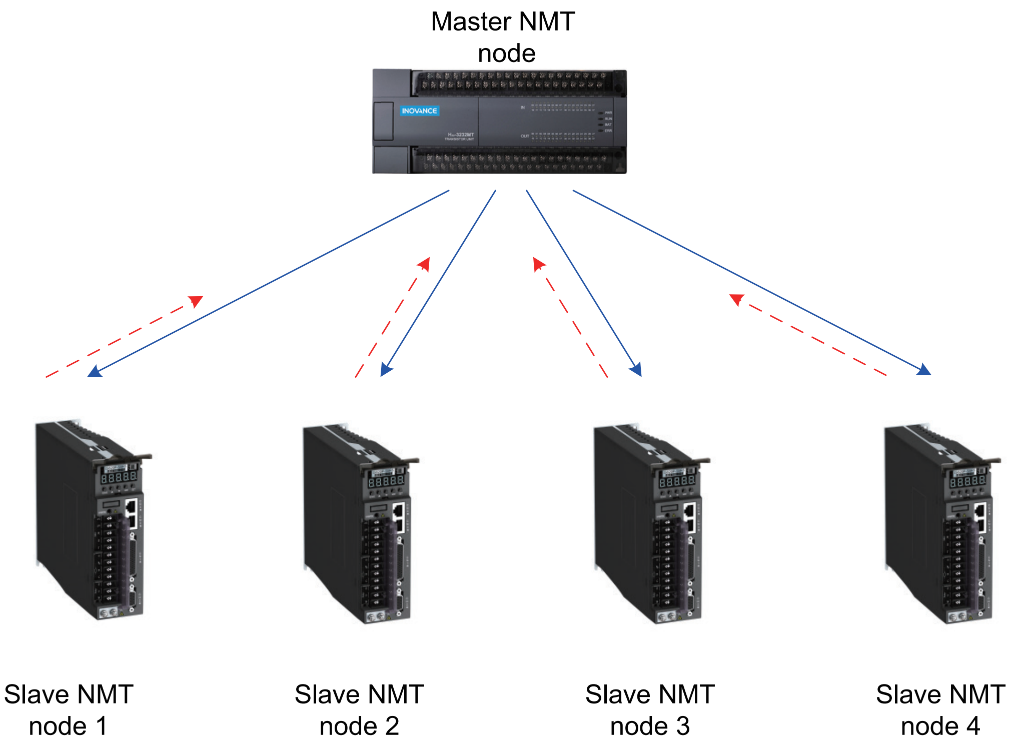
2. The intervals of node protection, life protection, and heartbeat should not be set to smaller values in case network load is increased.

1) Node/life protection

In the node protection status, the master NMT node periodically queries the status of slave nodes. In the life protection status, slave nodes monitor the status of the master node through the interval of the received remote frame used to monitor slave nodes. Node protection complies with the master/slave model. A response must be provided for each remote frame.

The objects related to node/life protection include the protection time 100Ch and life factor 100Dh. The value of 100Ch is the remote frame interval (unit: ms) in the node protection status under normal conditions. The product of 100Ch and 100Dh decides the latest time of query by the master node. Normally, node protection can be implemented. When 100Ch and 100D of a node are non-zero values and a node protection request frame is received, life protection is activated.

Figure 3-5 Schematic diagram of node protection



As shown in Figure 3-5, the master node sends a node protection remote frame at the interval of 100Ch. Slave nodes must respond to the remote frame. Otherwise, slave nodes are considered to be disconnected.

If slave nodes do not receive a node protection remote frame from the master node within the time 100Ch×100Dh, the master node is considered to be disconnected.

Table 3-9 describes the remote frame sent by the master NMT node.

Table 3-9 Node protection remote frame

COB-ID	RTR
0x700+Node_ID	1

Table 3-11 describes the response packet returned by slave NMT nodes. The data segment is a status word consisting of one byte.

Table 3-10 Node protection response packet

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

Table 3-11 Description of status of the response packet

Data Bit	Description
bit7	It must be set to 0 or 1 alternatively.
bit6-bit0	4: Stopped 5: Operation status 127: Pre-operation status

NOTE

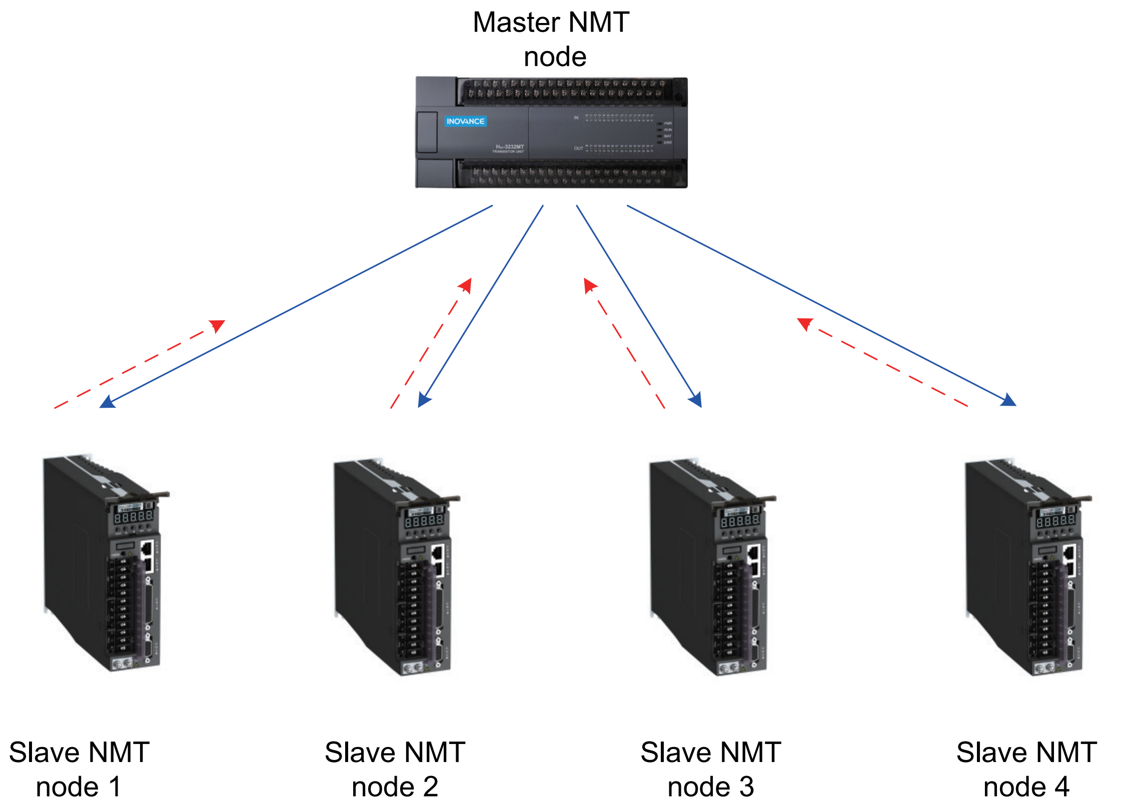
It is recommended that the protection time 100C should be longer than 10 ms and the life factor must be greater than or equal to 2.

2) Heartbeat

The heartbeat mode adopts the producer/consumer model. The CANopen device can send heartbeat packets based on the interval (unit: ms) set by the producer heartbeat interval object 1017h. In the network, there is always a node configured with the consumer heartbeat function, which monitors the producer based on the consumer time set by the object 1016h. Once the producer heartbeat is not received from the corresponding node within the consumer heartbeat time, the node is considered to encounter a failure.

After the producer heartbeat interval 1017h is configured, the node heartbeat function is activated and a heartbeat packet starts to be generated. After a valid sub-index is configured for consumer heartbeat 1016h and a heartbeat frame is received from the corresponding node, monitoring starts.

Figure 3-6 Heartbeat diagram



The master node sends a heartbeat packet based on the producer time. If slave nodes that monitor the master node do not receive the heartbeat packet within the time of 1016h sub-index, the master node is considered to be disconnected. The time of a 1016h sub-index must be longer than or equal to the producer time of the master node multiplied by 1.8. Otherwise, a message indicating that slave nodes consider the master node to be disconnected may be reported.

Slave nodes send a heartbeat packet at the interval of 1017h. If the master node that monitors the slave nodes or anther slave node does not receive the heartbeat packet within the consumer time, the slave nodes are considered to be disconnected. If 1017h multiplied by 1.8 is smaller than or equal to the consumer time of the master node that monitors the slave nodes, a message indicating that the slave nodes are disconnected may be reported.

Table 3-12 describes the format of a heartbeat packet. The data segment includes only one byte. The most significant bit is permanently set to 0 and other bits are consistent with the status of the response packet.

Table 3-12 Heartbeat packet

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

The IS620P servo drive is both a heartbeat producer and a heartbeat consumer. It can serve as the heartbeat consumer of up to five different nodes. It is recommended that the heartbeat producer time should be set to a value not smaller than 20 ms, and the consumer heartbeat time should be set to a value that is not smaller than 40 ms and is more than 1.8 times of the producer heartbeat time.

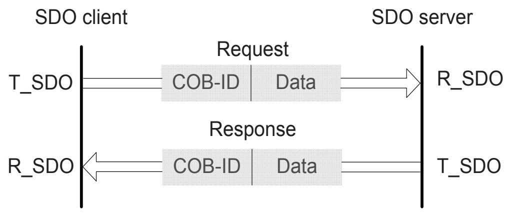
3.4 Service Data Object

An SDO is associated with the object dictionary through object index and sub-index. Based on the SDO, you can read the object content in the object dictionary or modify object data when conditions allow.

3.4.1 SDO Transmission Framework

SDO transmission 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 SDO requires at least two CAN packets and the CAN identifiers of the two CAN packets are different. Figure 3-8 shows the SDO transmission model.

Figure 3-7 Data exchange between an SDO client and the SDO server



3.4.2 SDO Transmission Packet

An SDO can be transmitted using data consisting of not more than four bytes or using data consisting of more than four bytes. For the former, the accelerated SDO transmission mode is adopted; for the latter, the segmented or block transmission mode is adopted. The IS620P servo drive supports only accelerated SDO transmission and segmented transmission.

An SDO transmission packet consists of a COB-ID and a data segment. It can be seen from Table 3-4 that the COB-IDs of the T_SDO packet and R_SDO packet are different.

The data segment adopts the little endian mode, that is, less significant bits are arranged in front of significant bits. The data segments of all SDO packets must consist of eight bytes. Table 3-13 describes the format of an SDO transmission packet.

Table 3-13 Format of an SDO transmission packet

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

The command code specifies the transmission type and transmitted data length of the SDO; the index and sub-index indicate the position of the SDO in the list; the data indicates the value of the SDO.

1) Writing SDO transmission packets in accelerated mode

If data consisting of not more than four bytes is read or written, accelerated SDO transmission is adopted. Depending on the read/write mode and data length, transmission packets are different. Table 3-14 describes an SDO packet that is written in accelerated mode.

Table 3-14 SDO packet that is written in accelerated mode

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

NOTE

"-" indicates data exists but is not considered. It is recommended that 0 is written.

Example:

If the slave station ID is 4, write the speed value 60FFh-00 in speed mode by using an SDO. The value that is written is 1000, namely 0x3E8. The packet sent by the master station is as follows. (All data is in hexadecimal.)

Table 3-15 Example of a packet sent by the master station

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 packet:

Table 3-16 Packet returned by the servo drive if the write operation is normal

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

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

Table 3-17 Packet returned if the type of data that is written does not match

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

2) Reading SDO transmission packets in accelerated mode

When an SDO packet consisting of not more than four bytes is read, the accelerated mode is adopted. Table 3-18 describes the SDO packet read in accelerated mode.

Table 3-18 Format of SDO packet read in accelerated mode

<table>
<tr><th colspan="2"></th><th>COB-ID</th><th>0</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th></tr>
<tr><td colspan="2">Client →</td><td>600h+Node_ID</td><td>40h</td><td colspan="2">Index</td><td>Sub-index</td><td>-</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td rowspan="5">← Server</td><td rowspan="4">Normal</td><td rowspan="5">580h+Node_ID</td><td>43h</td><td colspan="2" rowspan="5">Index</td><td rowspan="5">Sub-index</td><td colspan="4">Data</td></tr>
<tr><td>47h</td><td colspan="3">Data</td><td>-</td></tr>
<tr><td>4bh</td><td colspan="2">Data</td><td>-</td><td>-</td></tr>
<tr><td>4fh</td><td>Data</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td>Abnormal</td><td>80h</td><td colspan="4">Abort code</td></tr>
</table>

Example:

If the slave station ID is 4, read the maximum rotational speed limit H06-07 of the functional code by using an SDO, that is, the SDO is 0x2006-08. The packet sent by the master station is as follows. (All data is in hexadecimal.)

Table 3-19 Example of a packet sent by the master station

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 rotational speed is 6000 rpm, that is, 0x1770. Normally, the following packet is returned:

Table 3-20 Example of a packet returned for the maximum rotational speed

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

If the command word that is written does not match, an invalid command word error is returned, in which the fault code is 0x05040001. The packet is as follows:

Table 3-21 Packet returned if the command word that is written does not match

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

3) Reading SDO transmission packets in segmented mode

If an SDO consists of more than four bytes, the SDO is read in segmented mode. The structure of a packet transmitted in segmented mode is similar to the structure of a packet transmitted in accelerated mode. The start frame is the same as the frame in accelerated transmission. Table 3-22 describes the structure of a start packet that is transmitted.

Table 3-22 Structure of an SDO start packet that is 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 (bit 6) of the command code sends 0 or 1 alternatively. This rule must be maintained so that slave nodes can respond to the packet. Table 3-23 describes the packet structure during transmission.

Table 3-23 Structure of a packet 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 segmented mode includes the last frame identifier and valid data length of the last frame.

Table 3-24 describes the structure of the transmitted packet.

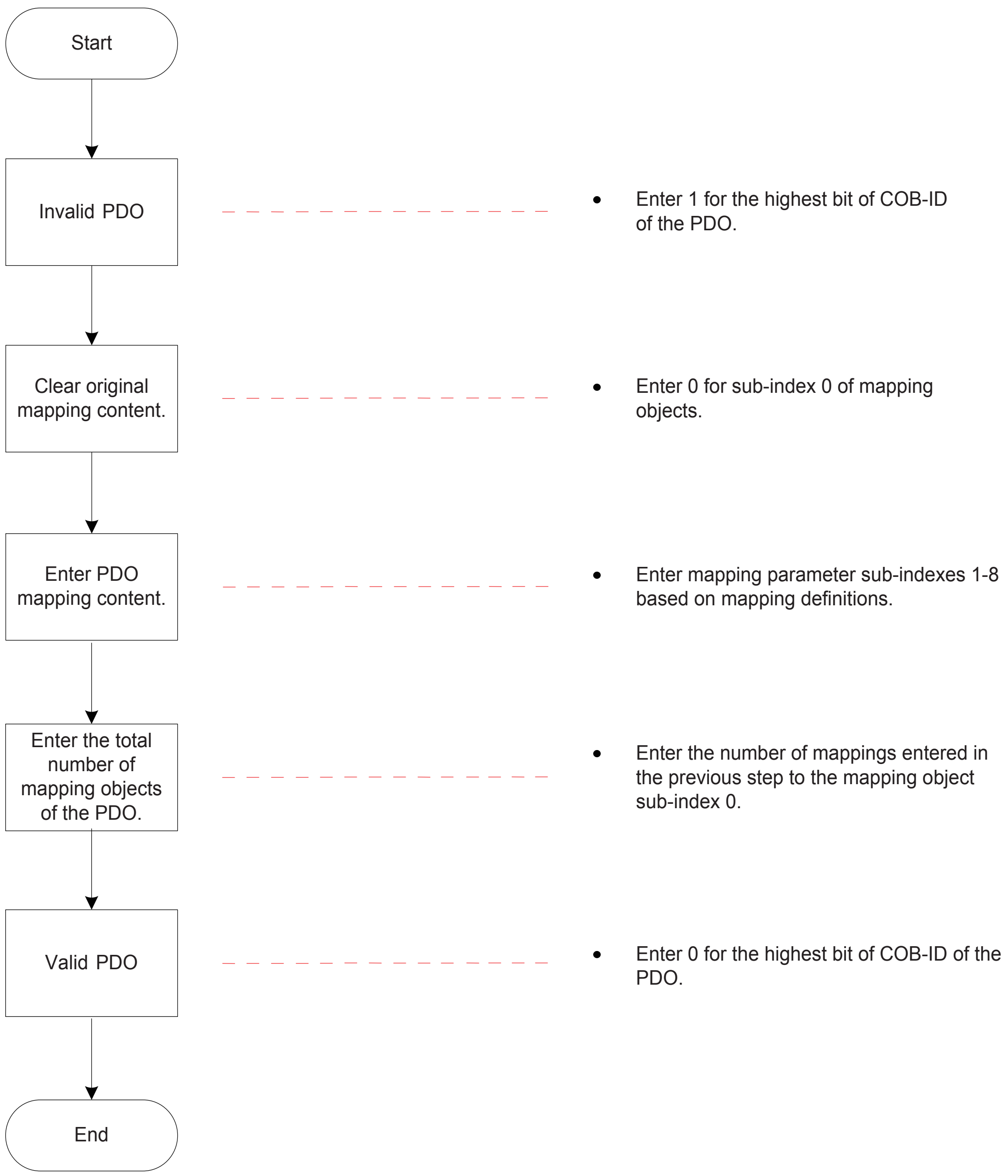
		COB-ID	0	1	2	3	4	5	6	7
Client →		600h+Node_ID	60h/0x70h	Index		Sub-index	-	-	-	-
← Server	Normal	580h+Node_ID	01h/11h	Data						
			03h/13h	Data						-
			05h/15h	Data					-	-
			07h/17h	Data				-	-	-
			09h/19h	Data			-	-	-	-
			0Bh/1Bh	Data		-	-	-	-	-
			0Dh/1Dh	Data	-	-	-	-	-	-
	Abnormal		80h	Index		Sub-index	Abort code			

3.5 Process Data Object

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

Figure 3-8 shows the PDO mapping configuration flowchart.

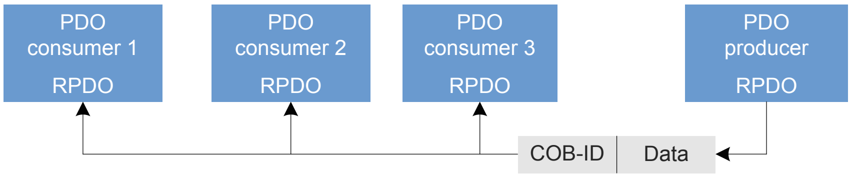
Figure 3-8 PDO mapping configuration flowchart



3.5.1 PDO Transmission Framework

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 more consumers on the network based on the COB-ID.

Figure 3-9 shows the transmission model.



At present, in the IS620P servo drive, CANopen communication only supports point-to-point PDO transmission.

3.5.2 PDO

Depending on receiving and transmitting, PDOs can be divided into RPDOs and TPDOs. The final PDO transmission mode and content are determined by communication parameters and mapping parameters. The IS620P servo drive uses four RPDOs and four TPDOs to transmit PDOs. Table 3-25 lists related PDOs.

Table 3-25 PDOs of IS620P servo drive

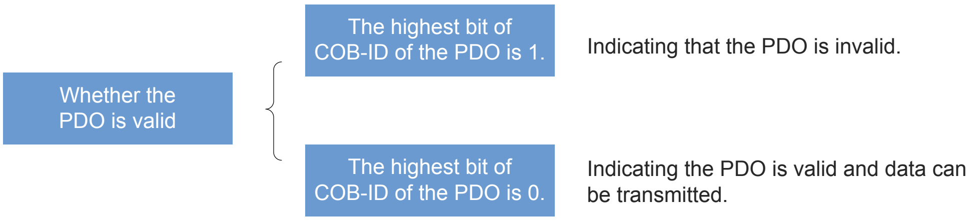
Description		COB-ID	Communication Object	Mapping Object
RPDO	1	200h + Node_ID	1400h	1600h
	2	300h + Node_ID	1401h	1601h
	3	400h + Node_ID	1402h	1602h
	4	500h + Node_ID	1403h	1603h
TPDO	1	180h + Node_ID	1800h	1A00h
	2	280h + Node_ID	1801h	1A01h
	3	380h + Node_ID	1802h	1A02h
	4	480h + Node_ID	1803h	1A03h

3.5.3 PDO Communication Parameters

1) CAN identifiers of PDOs

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

Figure 3-10 Description of PDO validity



The IS620P servo drive only supports point-to-point PDO transmission. Therefore, the less significant seven bits of the COB-ID must be the station address of the node.

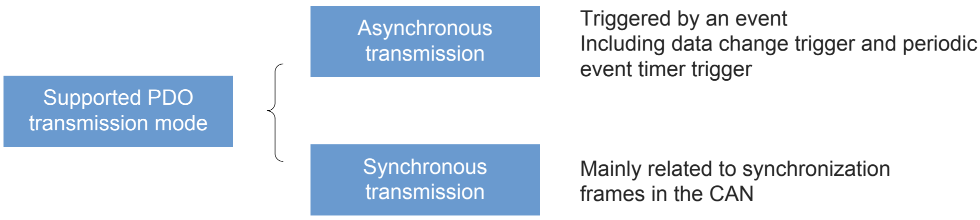
Example:

For the node whose station ID is 4, when TPDO3 is invalid, its COB-ID should be 80000384h. When 384h is written for the COB-ID, it indicates that the PDO is activated.

2) PDO transmission type

The PDO transmission type is on the sub-index 02 of communication parameters (RPDO: 1400h-1403h, TPDO: 1800h-1803h) and decides the mode in which the PDO is transmitted. For details, see "4.4 Overview of Drive Mode" on Page 63.

Figure 3-11 Supported PDO transmission mode



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

Table 3-26 Methods for triggering TPDO and RPDO

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

- When the transmission type of a TPDO is 0, if mapping data is changed and a synchronous frame is received, the TPDO is sent.
- When the transmission type of a TPDO is a value in the range 1 to 240 and a corresponding number of synchronous frames are received, the TPDO is sent.
- When the transmission type of a TPDO is 254 or 255, if mapping data is changed or the event timer expires, the TPDO is sent.
- When the transmission type of an RPDO is a value in the range 0 to 240, once a synchronous frame is received, the latest data of the RPDO is updated to the application; when the transmission type of an RPDO is 254 or 255, the received data is directly updated to the application.

3) Disabled time

Disabled time (unit: us) is set for TPDOs and is stored on the sub-index 03 of communication parameters (1800h to 1803h) to prevent the CAN from being continuously occupied by PDOs with lower priorities. After the parameter (unit: us) is set, the transmission interval of one TPDO should not be shorter than the time corresponding to the parameter.

Example:

If the disabled time of TPDO2 is 300, the transmission interval of TPDOs is not shorter than 30 ms.

4) Event timer

For TPDOs that are transmitted in asynchronous mode (the transmission type is 254 or 255), an event timer is defined and is on the sub-index 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 in the interval of the event timer, the TPDO is triggered and the event timer is immediately reset.

3.5.4 PDO Mapping Parameters

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

Table 3-27 Definitions of PDO mapping parameters

Bits	31	...	16	15	...	8	7	...	0
Meaning	Index			Sub-index			Object length		

The index and sub-index 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 Mapping between object length and object bit length

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

Example:

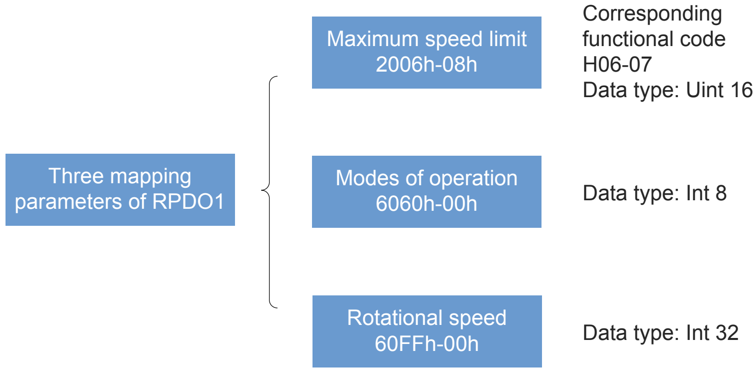
The mapping parameter of the 16-bit command word 6040h-00 is 60400010h.

The following describes the mapping of PDOs by using examples:

Example:

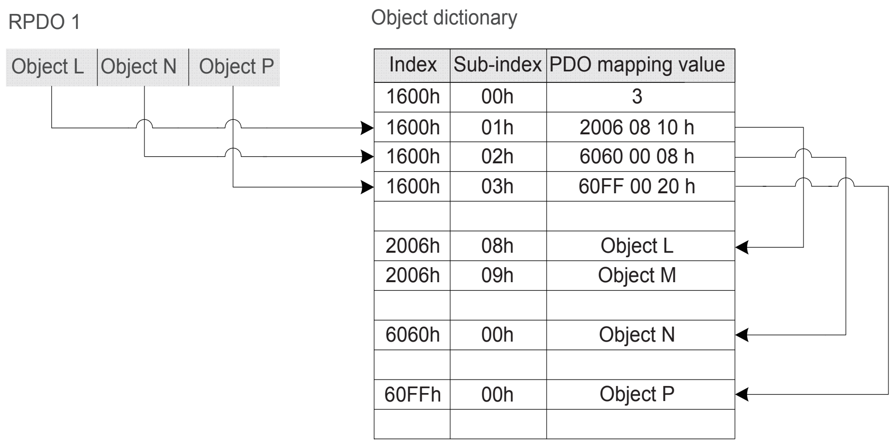
RPDO1 maps three parameters, that is:

Figure 3-12 Example of PDO1 mapping



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

Figure 3-13 Mapping of RPDO

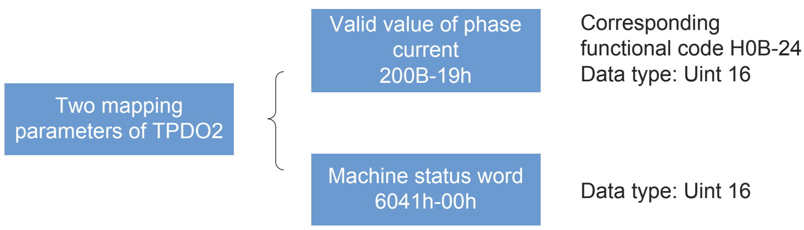


The mapping mode of TPDOs is the same as that of RPDOs but the direction is opposite. An RPDO decodes the input based on the mapping, but a TPDO encodes the output based on the mapping.

Example:

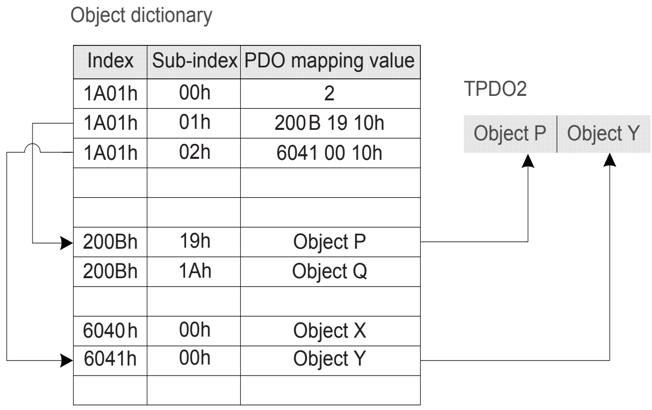
TPDO2 maps two parameters, that is:

Figure 3-14 Example of TPDO2 mapping



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

Figure 3-15 Mapping of TPDO

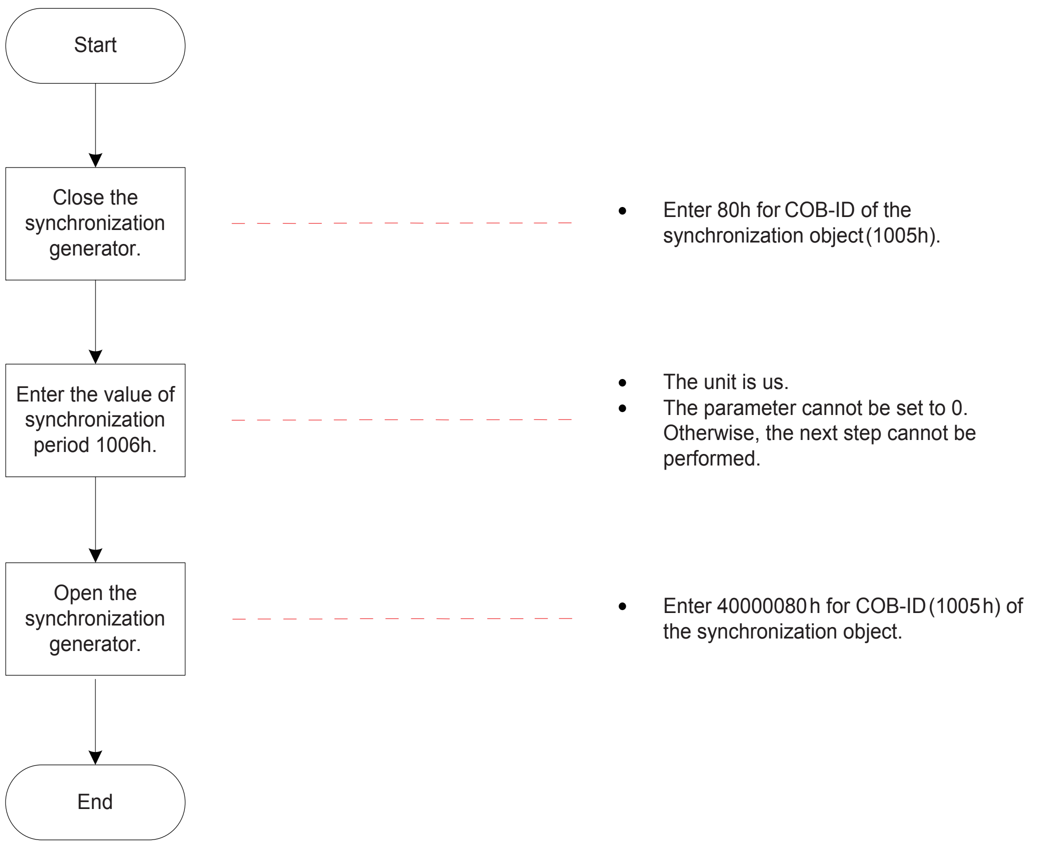


3.6 Synchronization Object (SYNC)

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

Figure 3-16 shows the synchronization generator configuration flowchart.

Figure 3-16 Synchronization generator configuration flowchart



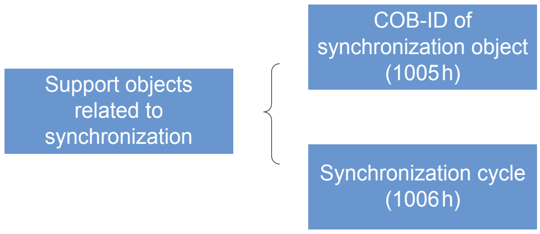
NOTE

The IS620P servo drive does not support the synchronization generator whose cycle is shorter than 500 us. It is recommended that the cycle 1 ms is not used.

3.6.1 Synchronization Generator

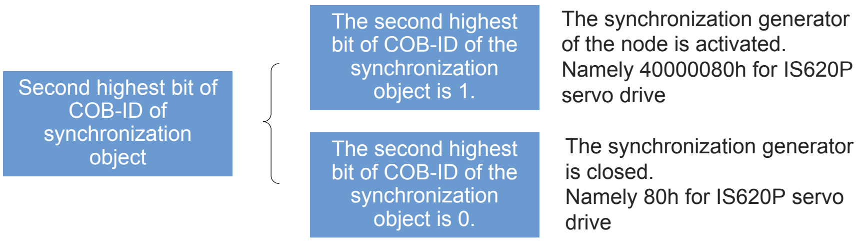
The IS620P servo drive is both a synchronization consumer and a synchronization producer. The supported objects related to synchronization are the synchronization object COB-ID (1005h) and synchronization cycle (1006h).

Figure 3-17 Description of supported objects related to synchronization



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

Figure 3-18 Activating the synchronization generator

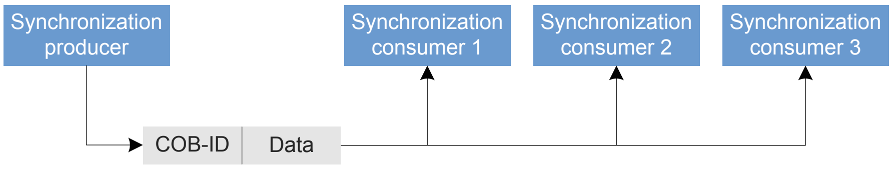


The synchronization cycle (unit: us) is only used for the synchronization generator. It indicates the interval in which a node generates a synchronization object.

3.6.2 Synchronization Object Transmission Framework

Similar to transmission of PDOs, synchronization objects are transmitted, complying with the producer/consumer model. The synchronization producer sends a synchronous frame, and other nodes in the CAN network can receive the synchronous frame as consumers without providing any feedback. In one CAN, only one activated synchronization generator is allowed. Figure 3-19 shows the transmission framework of synchronization objects.

Figure 3-19 Synchronization transmission framework



The transmission of synchronization PDOs is closely related to the synchronous frame.

- For an RPDO, so long as the PDO is received, the received PDO is updated to the application in the next synchronization.
- A synchronization TPDO can be transmitted in cyclic synchronization mode or acyclic synchronization mode.

Figure 3-20 Description of synchronization TPDO

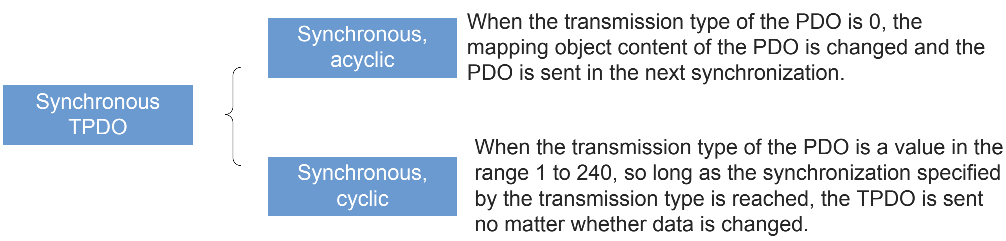
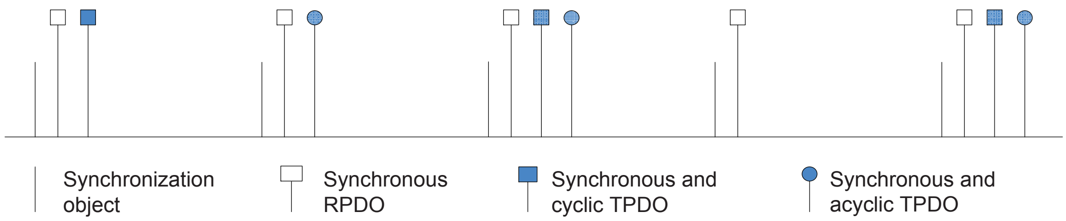


Figure 3-21 shows the synchronous transmission model.

Figure 3-21 Synchronous transmission model



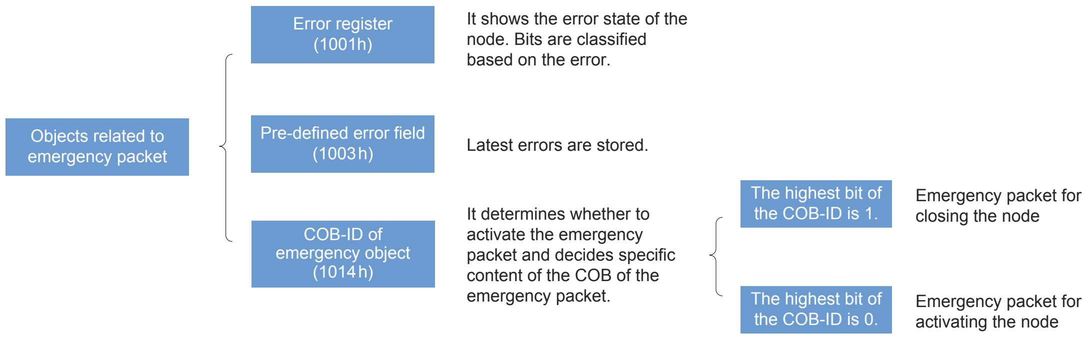
Example:

The transmission type of RPDO1 is 0; the transmission type of RPDO2 is 5; the transmission type of TPDO1 is 0; the transmission type of TPDO2 is 20. Once RPDO1 and RPDO2 receive the PDO, RPDO1 and RPDO2 update the PDO data to the corresponding application in the next synchronization; once the mapping data of TPDO1 is changed, TPDO1 is sent in the next synchronization. After TPDO2 experiences 20 synchronization operations, the PDO is sent no matter whether data is changed.

3.7 Emergency Object Service (EMCY)

When an error occurs in a CANopen node, the node sends an emergency packet according to the standardization mechanism. The emergency packet complies with the producer/consumer model. After the node fault is sent, other nodes in the CAN may choose to handle the fault. As the emergency packet producer, the IS620P servo drive does not process emergency packets of other nodes.

Figure 3-22 Description of objects related to emergency packet



When a node becomes faulty, the error register and the predefined error code must be updated no matter whether the emergency object is activated. Table 3-29 describes an emergency packet.

Table 3-29 Description of an emergency packet

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 communication becomes abnormal, the error code is consistent with the one required by DS301 and the auxiliary byte is 0.
- When the error described in the DSP402 sub-protocol occurs in the servo drive, the error code is consistent with the one required by DS402 and corresponds to the 603Fh object and the auxiliary byte is extra description.
- When an error specified by the user occurs in the servo drive, the error code is 0xFF00 and the auxiliary byte displays the error code specified by the user.

For the definitions of the error code and auxiliary byte, see "Chapter 5 Troubleshooting" on Page 136.

Memo NO. ___________

Date / /

Chapter 4 Motion Mode

Chapter 4 Motion Mode

4.1 Setting Conversion Factors

The direct user of the drive is the motor. Therefore, default units are motor units, for example:

- Motor displacement unit: p (pulse)
- Motor speed unit: rpm (round/minute)
- Motor acceleration unit: rpm/ms (for example, 10 rpm/ms indicates the motor is accelerated to 1000 rpm in 100 ms)

For the sake of convenience, users often use the actual load displacement, speed, and acceleration units, for example:

- Load displacement unit: mm
- Load speed unit: mm/s
- Load acceleration unit: mm/s^2

If motor units are inconsistent with user units, errors may occur during running of the motor. Therefore, before running the servo drive, correctly set conversion factors, through which proportional relations are established between motor units and user units.

4.1.1 6093h: Position Factor

The position factor indicates the motor displacement (unit: p) corresponding to the load displacement of one user unit.

The position factor is defined by the numerator 6093-1h and denominator 6093-2h. It can be used to establish a proportional relation between a load displacement (in user unit) and a motor displacement (in motor unit).

Motor displacement = Load displacement x Position factor

The motor is connected to load through a reducer and another mechanical transmission mechanism. Therefore, the position factor is related to the mechanical reduction ratio, parameters related to mechanical dimensions, and motor resolution. The position factor is calculated as follows:

Index 6093h	Name	Position Factor			Data Structure	ARR	Data Type	Uint32
	Accessibility	RW	Mapping	YES	Data Range	OD Data Range	Factory Default	OD Default Value
The position factor is used to establish a proportional relation between the specified load displacement and the motor displacement. Note: The range of position factor is: 0.001 × Encoder resolution/10000 to 4000 x Encoder resolution/10000. Outside the range, the Er.B03 error occurs in the drive.								

Sub-index 0	Name	Number of Entries			Data Structure	-	Data Type	Uint8
	Accessibility	RO	Mapping	NO	Data Range	2	Factory Default	2

<table>
<tr><td rowspan="2">Sub-index 1</td><td>Name</td><td colspan="3">Numerator</td><td>Data Structure</td><td>-</td><td>Data Type</td><td>Uint32</td></tr>
<tr><td>Accessibility</td><td>RW</td><td>Mapping</td><td>YES</td><td>Data Range</td><td>0 to 4294967295</td><td>Factory Default</td><td>1</td></tr>
<tr><td colspan="9">Set the product of the motor resolution and the reduction ratio.</td></tr>
</table>

<table>
<tr><td rowspan="2">Sub-index 2</td><td>Name</td><td colspan="3">Feed Constant</td><td>Data Structure</td><td>-</td><td>Data Type</td><td>Uint32</td></tr>
<tr><td>Accessibility</td><td>RW</td><td>Mapping</td><td>YES</td><td>Data Range</td><td>0 to 4294967295</td><td>Factory Default</td><td>1</td></tr>
<tr><td colspan="9">Set each feed of the bearing axle.
The feed of the bearing axle is related to each feed of the load and parameters of the mechanical transmission mechanism.</td></tr>
</table>

Example:

For a ball screw:

- Each feed of the load: fc: 40 mm
- Screw lead pB = 10 mm/r
- Reduction ratio n = 1:5
- Inovance 20-bit bus-based motor resolution P = 1048576 (p/r)

The position factor is calculated as follows:

1) Set each feed of the bearing axle.

$$\text{Load axis feed} = \frac{\text{Load feed fc}}{\text{Lead pB}} = \frac{40\text{ mm}}{10\text{ mm/r}} = 4(\text{r})$$

2) Position factor:

$$\text{Position factor} = \frac{\text{Motor resolution P}}{\text{Load axis feed} \times \text{Deceleration ratio n}} = \frac{1048576}{4\times(1/5)} = \frac{5242880}{4} = \frac{1310720}{1}$$

Therefore, 6093-1h = 1310720, 6093-2h = 1. It means that when the load displacement is 1 mm, the motor displacement is 1310720p.

NOTE

The values of 6093-1h and 6093-2h must go through fraction reduction until no common divisor is left and final values are obtained.

4.1.2 6094h: Velocity Encoder Factor

The velocity encoder factor indicates the motor displacement per second (unit: p/s) corresponding to the load speed of one user unit.

The velocity encoder factor is defined by the numerator 6094-1h and denominator 6094-2h. It can be used to establish a proportional relation between the load speed (in user unit) and the motor displacement per second (p/s).

Motor displacement per second = Load speed x Velocity encoder factor

The motor is connected to load through a reducer and another mechanical transmission mechanism. Therefore, the velocity encoder factor is related to the mechanical reduction ratio, parameters related to mechanical dimensions, and motor resolution. It can be calculated as follows:

$$\text{Velocity encoder factor} = \frac{\frac{\text{Motor velocity (rpm)}}{60} \times \text{Motor resolution}}{\text{Load speed}}$$

Index 6094h	Name	Velocity Encoder Factor			Data Structure	ARR	Data Type	Uint32
	Accessibility	RW	Mapping	YES	Data Range	OD Data Range	Factory Default	OD Default Value
The velocity encoder factor is used to establish a proportional relation between the load speed specified by the user and motor position increment.								

Sub-index 0	Name	Number of Entries			Data Structure	-	Data Type	Uint8
	Accessibility	RO	Mapping	NO	Data Range	2	Factory Default	2

Sub-index 1	Name	Numerator			Data Structure	-	Data Type	Uint32
	Accessibility	RW	Mapping	YES	Data Range	0 to 4294967295	Factory Default	1048576
Set the product of the motor velocity (r/s) and motor resolution.								

Sub-index 2	Name	Divisor			Data Structure	-	Data Type	Uint32
	Accessibility	RW	Mapping	YES	Data Range	0 to 4294967295	Factory Default	60
Set the load speed.								

Example:

For a ball screw:

- Load speed: 40 mm/s
- Screw lead pB = 10 mm/r
- Reduction ratio n = 1:5
- Inovance 20-bit bus-based motor resolution P = 1048576 (p/r)

Therefore, the velocity encoder factor is calculated as follows:

1) Motor velocity:

$$\text{Motor velocity} = \frac{\text{Load speed}}{\text{Lead pB/ Deceleration ratio}}$$

$$= \frac{40\ \text{mm/s}}{10\ \text{mm/r/(1/5)}}$$

$$= 20\ \text{(r/s)}$$

2) Velocity encoder factor:

$$\text{Velocity encoder factor} = \frac{\text{Motor velocity} \times \text{Motor resolution P}}{\text{Load speed}}$$

$$= \frac{20\ \text{r/s} \times 1048576\ \text{p/r}}{40\ \text{mm/s}}$$

$$= \frac{524288}{1}$$

Therefore, 6094-1h = 524288, 6094-2h = 1. It means that when the load speed is 1 mm/s, the motor displacement per second is 524288p.

NOTE

The values of 6094-1h and 6097-2h must go through fraction reduction until no common divisor is left and final values are obtained.

4.1.3 6095h: Velocity Factor 1

Velocity factor 1 indicates the motor velocity (unit: rpm) corresponding to the load speed of one user unit.

Velocity factor 1 is defined by the numerator 6095-1h and denominator 6095-2h. It can be used to establish a proportional relation between the motor velocity (rpm) and the load speed (user unit).

Load speed = Motor velocity x Velocity factor 1

The motor is connected to load through a reducer and another mechanical transmission mechanism. Therefore, velocity factor 1 is related to the mechanical reduction ratio and parameters related to mechanical dimensions. It can be calculated as follows:

$$\text{Velocity factor 1} = \frac{\text{Load speed}}{\text{Motor velocity (rpm)}}$$

Index 6095h	Name	Velocity Factor 1			Data Structure	ARR	Data Type	Uint32
	Accessibility	RW	Mapping	YES	Data Range	OD Data Range	Factory Default	OD Default Value
Velocity factor 1 is used to establish a proportional relation between the load speed and the motor velocity.								

Sub-index 0	Name	Number of Entries			Data Structure	-	Data Type	Uint8
	Accessibility	RO	Mapping	NO	Data Range	2	Factory Default	2

Sub-index 1	Name	Numerator			Data Structure	-	Data Type	Uint32
	Accessibility	RW	Mapping	YES	Data Range	0 to 4294967295	Factory Default	1
Set the load speed.								

Sub-index 2	Name	Divisor			Data Structure	-	Data Type	Uint32
	Accessibility	RW	Mapping	YES	Data Range	0 to 4294967295	Factory Default	1
Set the motor velocity (rpm).								

Example:

For a ball screw:

- Load speed unit: mm/s
- Motor velocity: 1200 rpm
- Screw lead pB = 10 mm/r
- Reduction ratio n = 1:5

Therefore, velocity factor 1 is calculated as follows:

1) Load speed:

$$\text{Load speed} = \text{Motor velocity} \times \text{Deceleration ratio} \times \text{Lead}$$
$$= \frac{1200\ \text{rpm}}{60\ \frac{\text{s}}{\text{min}}} \times (1/5) \times 10\ \text{mm/r}$$
$$= 40\ \text{mm/s}$$

2) Velocity factor 1:

$$\text{Velocity factor 1} = \frac{\text{Load speed}}{\text{Motor velocity}}$$
$$= \frac{40\ \text{mm/s}}{1200\ \text{rpm}}$$
$$= \frac{1}{30}$$

Therefore, 6095-1h = 1, 6095-2h = 30. It means that when the load speed is 1 mm/s, the motor velocity is 30 rpm.

NOTE

The values of 6095-1h and 6095-2h must go through fraction reduction until no common divisor is left and final values are obtained.

4.1.4 6097h: Acceleration Factor

The acceleration factor indicates the motor displacement increment per second (unit: p/s^2) corresponding to the load acceleration of one user unit.

The acceleration factor is defined by the numerator 6097-1h and denominator 6097-2h. It can be used to establish a proportional relation between the load speed (in user unit) and displacement increment per second (p/s^2).

Motor displacement increment per second = Load speed x Acceleration factor

The motor is connected to load through a reducer and another mechanical transmission mechanism. Therefore, the acceleration factor is related to the mechanical reduction ratio, parameters related to mechanical dimensions, and motor resolution. It can be calculated as follows:

$$\text{Acceleration factor} = \frac{\dfrac{1000 \times \text{Motor acceleration (rpm/ms)}}{60} \times \text{Motor resolution}}{\text{Load acceleration}}$$

Index 6094h	Name	Acceleration Factor			Data Structure	ARR	Data Type	Uint32
	Accessibility	RW	Mapping	YES	Data Range	OD Data Range	Factory Default	OD Default Value
The acceleration factor is used to establish a proportional relation between the load acceleration specified by the user and motor position increment per second.								

Sub-index 0	Name	Number of Entries			Data Structure	-	Data Type	Uint8
	Accessibility	RO	Mapping	NO	Data Range	2	Factory Default	2

Sub-index 1	Name	Numerator			Data Structure	-	Data Type	Uint32
	Accessibility	RW	Mapping	YES	Data Range	0 to 4294967295	Factory Default	1048576000
Set the product of the motor acceleration (r/s^2) and motor resolution.								

Sub-index 2	Name	Divisor			Data Structure	-	Data Type	Uint32
	Accessibility	RW	Mapping	YES	Data Range	0 to 4294967295	Factory Default	60
Set load acceleration.								

Example:

For a ball screw:

- Load acceleration: 40 mm/s^2
- Screw lead pB = 10 mm/r
- Reduction ratio n = 1:5
- Inovance 20-bit bus-based motor resolution P = 1048576 (p/r)

Therefore, the acceleration factor is calculated as follows:

1) Motor acceleration:

$$\text{Motor acceleration} = \frac{\text{Load acceleration}}{\text{Lead pB/Deceleration ratio}}$$

$$= \frac{40\ \text{mm/s}^2}{10\ \text{mm/r/(1/5)}}$$

$$= 20\ (\text{r/s}^2)$$

2) Accelerator factor:

$$\text{Acceleration factor} = \frac{\dfrac{1000 \times \text{Motor acceleration (rpm/ms)}}{60} \times \text{Motor resolution}}{\text{Load acceleration}}$$

$$= \frac{20\ \text{r/s}^2 \times 1048576\ \text{p/r}}{40\ \text{mm/s}^2}$$

$$= \frac{524288}{1}$$

Therefore, 6097-1h = 524288, 6097-2h = 1. It means that when load acceleration is 40 mm/s^2, the motor position increment per second is 524288 p/s.

NOTE

The values of 6097-1h and 6097-2h must go through fraction reduction until no common divisor is left and final values are obtained.

4.1.5 607Eh: Polarity

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

Index 607Eh	Name	Polarity			Data Structure	VAR	Data Type	Uint8
	Accessibility	RW	Mapping	YES	Data Range	OD Data Range	Factory Default	0

Set the polarity of position or velocity references.

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Position reference polarity	Velocity reference polarity	Torque reference feature	NA	NA	NA	NA	NA

Bit7=1 indicates the position reference X(-1) reverses the motor in standard position mode or interpolated position mode.

Bit6=1 indicates the speed reference (60FFh)×(-1) reverses the motor in velocity mode.

Bit5=1 indicates the torque reference (6071h)×(-1) reverses the motor in torque mode.

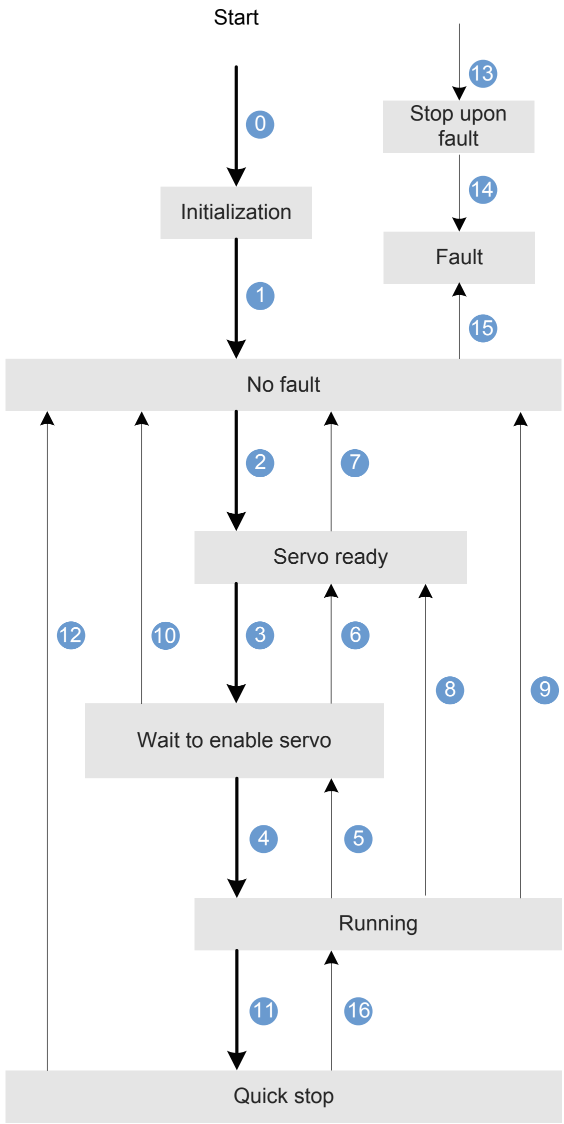
NA: not defined

4.2 Servo Status Control

4.2.1 CiA402 State Machine

The IS620P CANopen servo drive runs in the specified status only when it is instructed according to the flow defined in CiA402.

Figure 4-1 CiA402 state machine switching diagram



The states are described in the following table.

Table 4-1 Status description

Status	Description
Initialization	Initialization of the servo drive and internal self-check are complete. Parameters of the driver cannot be set and the drive function cannot be implemented.
No fault	No fault exists in the servo drive or the fault is eliminated. Parameters of the servo drive can be set.
Servo ready	The servo drive is ready and "rdy" is displayed on the panel. Parameters of the servo drive can be set.
Wait to enable serve	The servo drive waits for enabling of servo and "rdy" is displayed on the panel. Parameters of the servo drive can be set.
Running	The servo drive is running properly and a servo mode is enabled; the motor is powered on and starts to work when the reference is not 0. Only parameters whose attributes are "running change" can be set.
Quick stop	The quick stop function is activated and the servo drive is implementing the quick stop function. Only parameters whose attributes are "running change" can be set.
Stop upon fault	A fault occurs and the servo drive in performing the stop process. Only parameters whose attributes are "running change" can be set.
Fault	The stop process is complete and all drive functions are disabled. Parameters of the servo drive can be modified to eliminate the fault. For faults that can be reset, after parameters are modified, reset the faults through the control word 6040h=0x80.

Control command and status switching:

Table 4-2 Relationship between status switching and control commands

CiA402 Status Switching		Control Word 6040h	Bit0 to Bit9*1 of Status Word 6041h
0	Power-on → Initialization	Natural transition, control command not required	0x0000
1	Initialization → No fault	Natural transition, control command not required If an error occurs during initialization, the servo drive directly goes to status 13.	0x0250
2	No fault → Ready	0x06	0x0231
3	Ready → Wait to enable servo	0x07	0x0233
4	Wait to enable servo → Running	0x0F	0x0237
5	Running → Wait to enable servo	0x07	0x0233
6	Wait to enable servo → Ready	0x06	0x0231
7	Ready → No fault	0x00	0x0250
8	Running → Ready	0x06	0x0231

CiA402 Status Switching		Control Word 6040h	Bit0 to Bit9*1 of Status Word 6041h
9	Running → No fault	0x00	0x0250
10	Wait to enable servo → No fault	0x00	0x0250
11	Running → Quick stop	0x02	0x0217
12	Quick stop → No fault	Set 605A to a value in the range 0 to 3. Natural transition is performed after stop and no control command is required.	0x0250
13	→ Stop upon fault	Once a fault occurs in any status other than "fault", the servo drive automatically switches to the status of stop upon fault without any control command.	0x021F
14	Stop upon fault → Fault	Natural transition is performed after stop upon fault and no control command is required.	0x0218
15	Fault → No fault	0x80 The rising edge of bit7 is valid. If bit7 is 1, other control commands are invalid.	0x0250
16	Quick stop → Running	Set 605A to a value in the range 5 to 7. After the stop process is complete, 0x0F is sent.	0x0237

NOTE

*1: Bit10 to bit15 (bit14 is meaningless) of status word 6041h are related to the running status of the servo drive in different modes and are set to 0 in the preceding table. For specific status of the bits, see all drive modes.

4.2.2 Control Word 6040h

Index 6040h	Name	Control Word				Data Structure	VAR	Data Type	Uint16
	Accessibility	RW	Mapping	YES	Relevant Mode	All	Data Range	0 to 65535	Factory Default	0

Set control commands:

bit	Name	Description
0	Servo ready	0: Disabled 1: Enabled
1	Switch on	0: Disabled 1: Enabled
2	Quick stop	0: Enabled 1: Disabled
3	Running	0: Disabled 1: Enabled
4 to 6	-	Related to drive modes.
7	Fault reset	Fault reset is implemented for faults and warnings that can be reset. ■ The rising edge of bit7 is valid. ■ If bit7 is 1, other control commands are invalid.
8	Halt	Not supported
9 to 10	NA	Reserved
11 to 15	Defined by the manufacturer	Reserved

◆ Note:

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

■ The meanings of bit0 to bit3 and bit7 are the same in each mode of the servo drive. The servo drive switches to the preset status according to the CiA402 state machine only when control words are sent in sequence. Each command corresponds to one status.

■ The meanings of bit4 to bit6 vary according to the drive modes. For details, see control commands in different modes.

4.2.3 Status Word 6041h

Index 6041h	Name	Status Word				Data Structure	VAR	Data Type	Uint16
	Accessibility	RO	Mapping	TPDO	Relevant Mode	All	Data Range	0 to 65535	Factory Default	-

Show the status of the servo drive.

bit	Name	Description
0	No fault	-
1	Wait to enable servo	-
Two	Running	-
3	Fault	-
4	Switch on	-
5	Quick stop	-
6	Servo ready	-
7	Warning	-
8	Defined by the manufacturer	Reserved
9	Remote control	0: In a mode other than CANopen mode, some IS620P standard software functions can be used. 1: CANopen remote control mode
10	Target reached	0: The target position or velocity is not reached. 1: The target position or velocity is reached.
11	Software internal position limit	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.
12 to 13		Related to drive modes.
14	NA	Reserved
15	Homing completed	0: Homing is not performed or complete. 1: Homing is complete. This bit is unrelated to the current status of the drive.

◆ Note:

■ All bits in the control word work together to show the current status of the servo drive. One bit is meaningless if it is set separately.

■ The meanings of bit0 to bit9 are the same in each mode of the servo drive. After control commands in 6040h are sent in sequence, the servo drive shows a certain status.

■ The meanings of bit12 to bit13 vary according to the drive modes. For details, see control commands in different modes.

■ The meanings of bit10, bit11, and bit15 are the same in each mode of the servo drive and indicate the status after a control mode is implemented.

4.2.4 Stop Mode

IS620P-CANopen supports five stop modes.

- Servo enabled stop
- Servo stop upon fault
- Stop beyond limit
- Emergency stop
- Quick stop
- Halt: not supported

1) Servo enabled stop

When servo enabled stop occurs, the stop mode is decided by the functional code H02-05 (object dictionary 2002-06h), which is the same as the IS620P standard.

2) Servo stop upon fault

When a fault or warning occurs, the servo drive automatically enters the status of stop upon fault. The stop mode is decided 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 IS620P standard.

3) Stop beyond limit

When stop beyond limit occurs, the stop mode is decided by the functional code H02-07 (object dictionary 2002-08h), which is the same as the IS620P standard.

4) Emergency stop

The servo drive supports two emergency stop modes:

- Using DI function 34 (FunIN.34: Emergency Stop), which is the same as the IS620P standard.
- Using an auxiliary function: H0D-05 (object dictionary 200D-06h). This is the same as the IS620P standard.

5) Quick stop

When the control word 6040h is 0x02 in the non-faulty status, the servo drive implements quick stop in a mode selected from 605A.

Index 605Ah	Name	Quick Stop Option Code				Data Structure	VAR	Data Type	Int16
	Accessibility	RW	Mapping	NO	Relevant Mode	All	Data Range	0 to 7	Factory Default	2

Set the quick stop mode.

Value	Stop Mode
0	Free stop. The free running status is maintained.
1	Ramp stop based on the deceleration set in 6084h. After stop, the free running status is maintained.
2	Ramp stop based on the deceleration set in 6085h. After stop, the free running status is maintained.
3	Torque stop for emergency stop set in 2007-10h (functional code: H07-15). After stop, the free running status is maintained.
4	NA
5	Ramp stop based on the deceleration set in 6084h. After stop, the position locked status is maintained.
6	Ramp stop based on the deceleration set in 6085h. After stop, the position locked status is maintained.
7	Torque stop for emergency stop set in 2007-10h (functional code: H07-15). After stop, the position locked status is maintained.

6) Halt

Bit8 (Halt) of the control word 6040 indicates the halt function and is not supported. After the command is entered, the drive maintains the current status and does not make any response.

4.3 Trial Running Steps

Step	Operation	Description
1	Confirm installation	Perform installation according to requirements in the appendix (try not to install the motor on the machine). For details, see the *IS620P Series Servo Design and Maintenance User Manual*.
2	Confirm connecting cables	Connect the cable for the encoder, power cable for the motor, and terminal cables. For details, see the *IS620P Series Servo Design and Maintenance User Manual*.
3	Confirm the supply voltage	Ensure that the power input meets specification requirements of the servo drive.
4	Confirm communication parameter settings	Confirm system settings in "3.2 System Settings" on Page 24.
5	Confirm the motor model	Ensure that the motor matches the servo drive model.
6	Power on the servo drive	Ensure that no alarm is reported during power-on.
7	Set parameters	Set related objects. For details, see "4.4 Overview of Drive Mode" on Page 63.
8	Perform trial run	In profile velocity mode, the specified low-speed commands run properly. For detail, see "4.8 Profile Velocity Mode" on Page 122.
9	Adjust parameters	Adjust parameters related to gain. In this step, check waveforms through the oscilloscope in the background and adjust related gain.
10	Run the servo drive	-

4.4 Overview of Drive Mode

The IS620P-CANopen supports four drive modes, which are defined in the object dictionary 6502h.

1) Supported drive modes in 6502H

Index 6502h	Name	Supported Drive Modes					Data Structure	VAR	Data Type	Uint32
	Accessibility	RO	Mapping	TPDO	Relevant Mode	-	Data Range	0 to 4294967295	Factory Default	0x0000006D

It indicates the supported drive modes.

bit	Description	Support 0: Not supported 1: Supported
0	Profile position (PP) mode	1
1	vl	0
2	Profile velocity (PV) mode	1
3	tq	1
4	NA	0
5	Homing (HM) mode	1
6	Interpolated position (IP) mode	1
7 to 15	NA	Reserved
16 to 31	Defined by the manufacturer	Reserved

If the CANopen device supports the object dictionary 6502h, you can learn the drive modes supported by the servo drive from 6502h.

The pre-operation mode of the servo drive can be set via the object dictionary 6060h. The current operation mode of the servo drive can be viewed in the object dictionary 6061h.

2) Modes of operation (6060h)

Index 6060h	Name	Modes of Operation				Data Structure	VAR	Data Type	Int8
	Accessibility	RW	Mapping	YES	Relevant Mode	All	Data Range	0 to 7	Factory Default	0

Select modes of operation:

bit	Description	Description
0	NA	Reserved
1	Profile position (PP) mode	For parameter settings, see "4.5 Profile Position Mode" on Page 66.
2	NA	Reserved
3	Profile velocity (PV) mode	For parameter settings, see "4.8 Profile Velocity Mode" on Page 122.
4	Profile torque (PT) mode	For parameter settings, see "4.9 Profile Torque Mode" on Page 132.
5	NA	Reserved
6	Homing mode	For parameter settings, see "4.6 Homing Mode" on Page 83.
7	IP mode	For parameter settings, see "4.7 Interpolated Position Mode" on Page 113.

■ If an unsupported operation mode is selected through an SDO, a SDO error is returned.

■ If an unsupported operation mode is selected through a PDO, the change of the operation mode is invalid.

3) Modes of operation display (6061h)

Index 6061h	Name	Modes of Operation Display				Data Structure	VAR	Data Type	Int8
	Accessibility	RO	Mapping	TPDO	Relevant Mode	All	Data Range	0 to 7	Factory Default	-

Display the actual operation mode:

bit	Description	Description
0	NA	Reserved
1	Profile position (PP) mode	For parameter settings, see "4.5 Profile Position Mode" on Page 66.
2	NA	Reserved
3	Profile velocity (PV) mode	For parameter settings, see "4.8 Profile Velocity Mode" on Page 122.
4	Profile torque (PT) mode	For parameter settings, see "4.9 Profile Torque Mode" on Page 132.
5	NA	Reserved
6	Homing mode	For parameter settings, see "4.6 Homing Mode" on Page 83.
7	IP mode	For parameter settings, see "4.7 Interpolated Position Mode" on Page 113.

4) Precautions for mode switching:

- When the servo drive in any status switches from the profile position mode to another mode, the position references not executed in profile position mode are discarded.
- When the servo drive in any status switches from the profile velocity mode to another mode, the servo drive performs ramp stop (stop deceleration 6084h) before entering another mode.
- When the servo drive is running in homing mode, the servo drive cannot switch to another mode. After homing is complete or is interrupted (fault or S-ON off), the servo drive can switch to another mode.
- When the servo drive in any status switches from the interpolated position mode to another mode, the position references not executed in interpolated position mode are discarded.

4.5 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 transition between references can be modified in real time. The profile position mode is often used in point-to-point positioning, and the operation cure is planned by the servo drive. The servo drive completes control over the position, speed, and torque inside.

4.5.1 Control Block Diagram

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

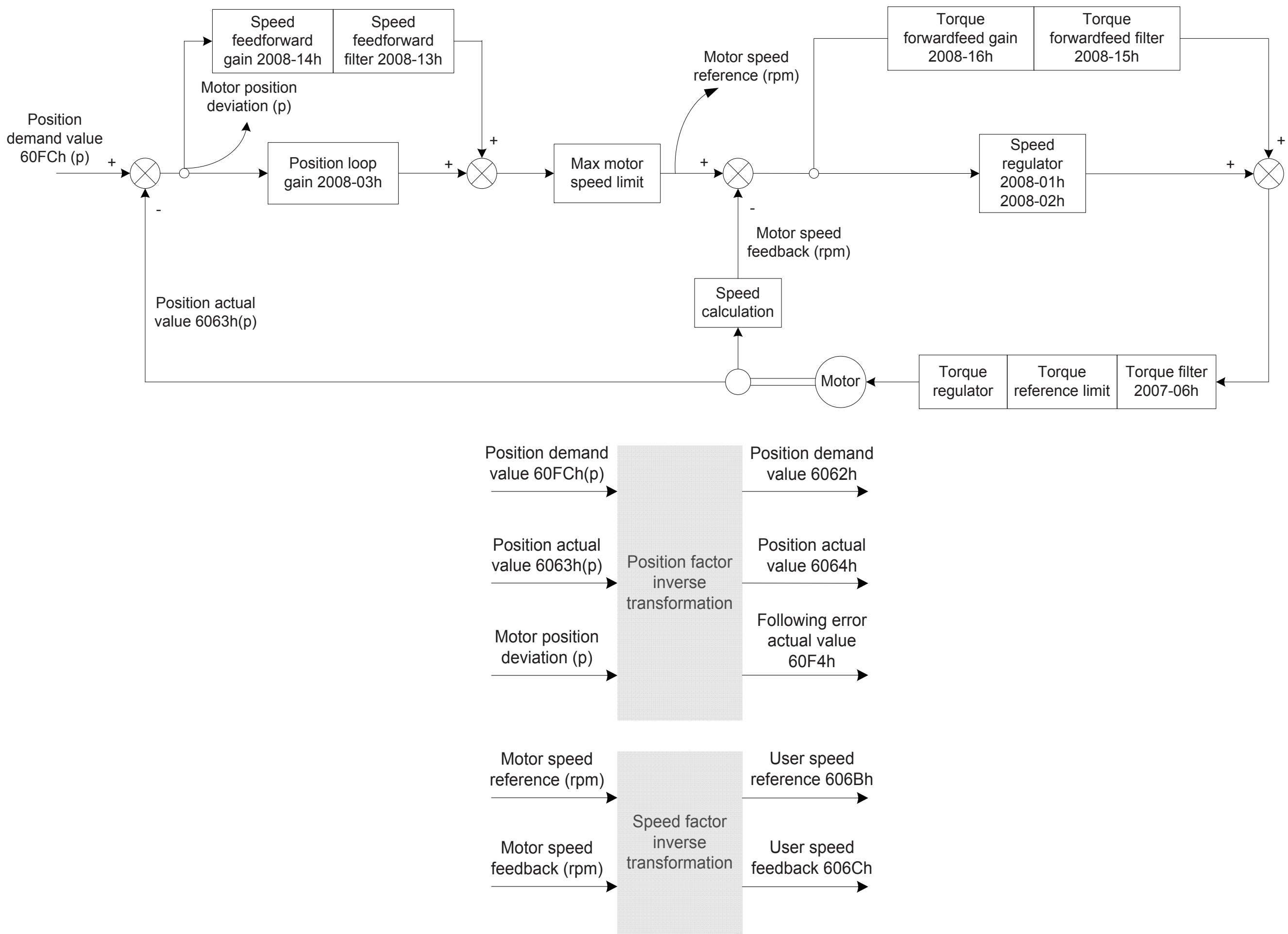
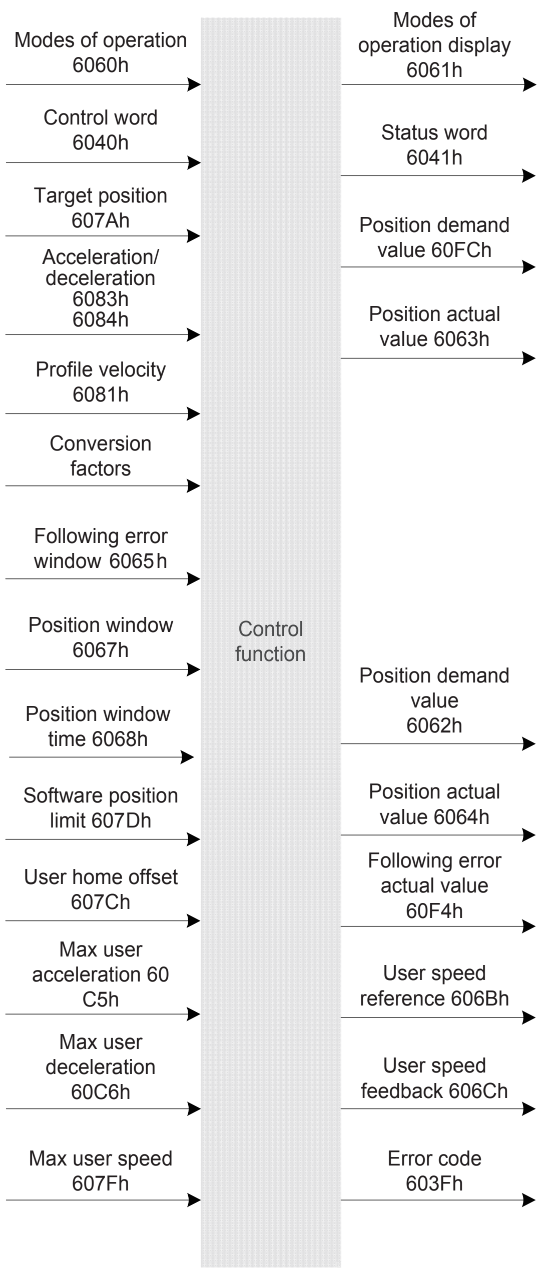


Figure 4-3 Input and output objects in profile position mode



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

References of the host computer are entered in user units and are called references in the drive unit after they go through limiting and conversion.

Figure 4-4 Unit conversion

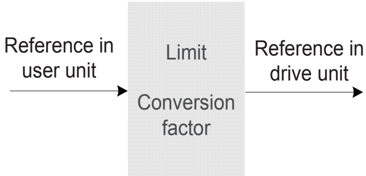
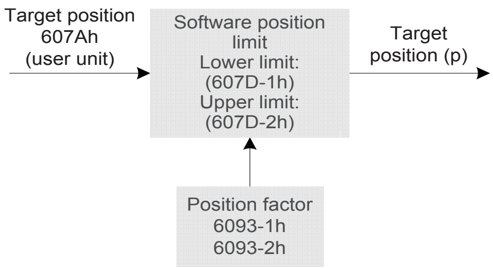


Figure 4-5, Figure 4-6, and Figure 4-7 show processing of the drive for the target position, profile velocity, and profile acceleration and deceleration.

By setting 0x200A-02h, you can check the absolute position limit of the user position reference and position feedback. By default, 200A-02h is 2, that is, after homing is complete and the reference zero position of mechanical operation is known, software position limit check is performed for the user target position 607Ah and user position feedback 6064h.

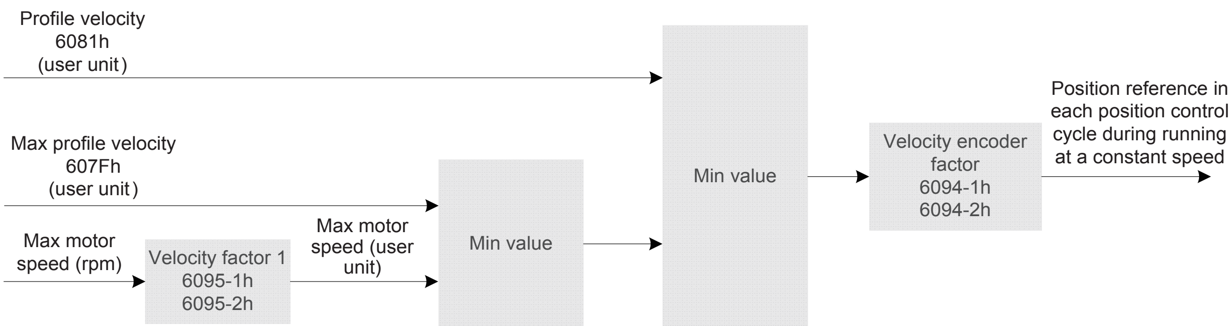
When the position reference exceeds the internal software position limit, bit11 of the status word 6041h is set to 1 and the drive runs by using the limit as the target position. After reaching the target position, the drive stops and provides a prompt. A reverse reference can make the drive exit the limit status and clear bit11 of 6041h. When external DI limit switch and internal software position limit are valid at the same time, the limit status is determined by the external DI limit switch.

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



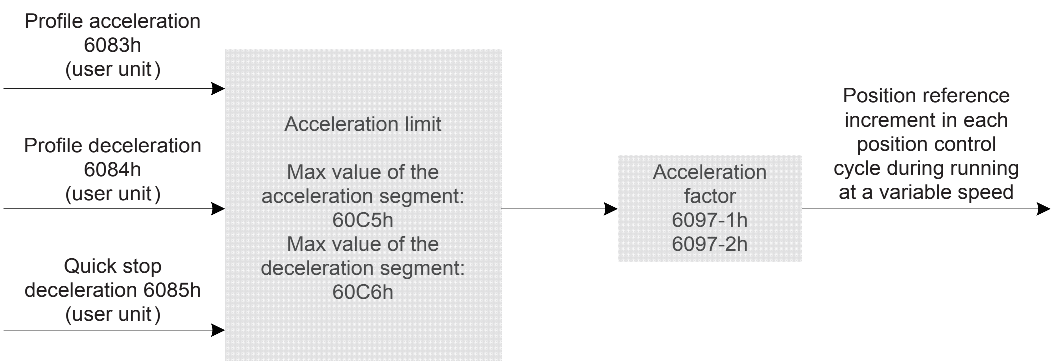
The profile velocity 6081h is used to set the maximum speed during running of the displacement reference. It cannot exceed the maximum velocity 607Fh set by the user and the maximum motor speed after conversion. Figure 4-6 shows the block diagram.

Figure 4-6 Profile velocity 6081h: speed limit



Profile acceleration 6083h and profile deceleration 6084h are used to set acceleration and deceleration during running of the displacement reference. The values cannot exceed the maximum acceleration 60C5h and deceleration 60C6h set by the user. Figure 4-7 shows the block diagram.

Figure 4-7 Profile acceleration and deceleration limit



4.5.2 Relevant Object Setting

1) Positioning completed

Index 6067h	Name	Position Window				Data Structure	VAR	Data Type	Uint32
	Accessibility	RW	Mapping	YES	Relevant Mode	pp/hm/ip	Data Range	0 to 4294967295	Factory Default	734 p

Sub-index: 00
When the position deviation 60F4h of the user unit is smaller than 6068h and time reaches this value, bit10 of 6041h is 1.
When either condition is not met, the position window is invalid.

Index 6068h	Name	Position Window Time				Data Structure	VAR	Data Type	Uint16
	Accessibility	RW	Mapping	YES	Relevant Mode	pp/hm/ip	Data Range	0 to 65535	Factory Default	0 ms

Sub-index: 00
When the position deviation 60F4h of the user unit is smaller than 6068h and time reaches this value, bit10 of 6041h is 1.
When either condition is not met, the position window is invalid.

2) Detection for Following Error Window

Index 6065h	Name	Following Error Window				Data Structure	VAR	Data Type	Uint32
	Accessibility	RW	Mapping	YES	Relevant Mode	pp/hm/ip	Data Range	0 to 4294967295	Factory Default	3145728 p

Sub-index: 00
When the position deviation is larger than this value, Er.B00 occurs and bit13 of the status word 6041h is set to 1.

Index	Sub-index	Name	Accessibility	Mapping	Data Type	Unit	Data Range	Factory Default
603Fh	00h	Error code	RO	TPDO	Uint16	-	0–65535	-
6040h	00h	Control word	RW	YES	Uint16	-	0–65535	0
6041h	00h	Status word	RO	TPDO	Uint16	-	0–65535	-
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	User 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	User unit	-2^{31} to $(2^{31}-1)$	-
6065h	00h	Following error window	RW	YES	Uint32	User unit	0 to $(2^{32}-1)$	3145728
6067h	00h	Position window	RW	YES	Uint32	User 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	rpm	-2^{31} to $(2^{31}-1)$	-
606Ch	00h	Velocity actual value	RO	TPDO	Int32	rpm	-2^{31} to $(2^{31}-1)$	-
607Ah	00h	Target position	RW	YES	Int32	User unit	-2^{31} to $(2^{31}-1)$	0
607Dh	01h	Min software position limit	RW	YES	Int32	User unit	-2^{31} to $(2^{31}-1)$	-2^{31}
	02h	Max software position limit	RW	YES	Int32	User unit	-2^{31} to $(2^{31}-1)$	$2^{31}-1$
607Ch	00h	Home offset	RW	YES	Int32	User unit	-2^{31} to $(2^{31}-1)$	0
6081h	00h	Profile velocity	RW	YES	Uint32	rpm	0 to $(2^{32}-1)$	100
6083h	00h	Profile acceleration	RW	YES	Uint32	rpm/ms	0 to $(2^{32}-1)$	100
6084h	00h	Profile deceleration	RW	YES	Uint32	rpm/ms	0 to $(2^{32}-1)$	100
6093h	01h	Position factor numerator	RW	YES	Uint 32	-	0 to $(2^{32}-1)$	1
	02h	Position factor denominator	RW	YES	Uint 32	-	1 to $(2^{32}-1)$	1
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	Accessibility	Mapping	Data Type	Unit	Data Range	Factory 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 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

4.5.3 Control Commands in PP Mode

Table 4-2 Relationship between status switching and control commands

CiA402 Status Switching		Control Word 6040h	Bit0 to Bit9*1 of Status Word 6041h
0	Power-on → Initialization	Natural transition, control command not required	0x0000h
1	Initialization → No fault	Natural transition, control command not required If an error occurs during initialization, the servo drive directly goes to status 13.	0x0250h
2	No fault → Ready	0x06h	0x0231h
3	Ready → Wait to enable servo	0x07h	0x0233h
4	Wait to enable servo → Running	0x0Fh	0x0237h
5	Running → Wait to enable servo	0x07h	0x0233h
6	Wait to enable servo → Ready	0x06h	0x0231h
7	Ready → No fault	0x00h	0x0250h
8	Running → Ready	0x06h	0x0231h
9	Running → No fault	0x00h	0x0250h
10	Wait to enable servo → No fault	0x00h	0x0250h
11	Running → Quick stop	0x02h	0x0217h
12	Quick stop → No fault	Set 605A to a value in the range 0 to 3. Natural transition is performed after stop and no control command is required.	0x0250h
13	→ Stop upon fault	Once a fault occurs in any status other than "fault", the servo drive automatically switches to the status of stop upon fault without any control command.	0x021Fh
14	Stop upon fault → Fault	Natural transition after stop at fault, control command not required	0x0218h
15	Fault → No fault	0x80h The rising edge of bit7 is valid. If bit7 is 1, other control commands are invalid.	0x0250h
16	Quick stop → Running	Set 605A to a value in the range 5 to 7. After the stop process is complete, 0x0F is sent.	0x0237h

NOTE

*1: Bit10 to bit15 (bit14 is meaningless) of status word 6041h are related to the running status of the servo drive in different modes and are set to 0 in the preceding table. For specific status of the bits, see all drive modes.

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

Index 6040h	Name	Control Word					Data Structure	VAR	Data Type	Uint16
	Accessibility	RW	Mapping	YES	Relevant Mode	All	Data Range	0 to 65535	Factory Default	-

It sets the control commands in PP mode.

Control Word 6040h					
Bit	bit7-15	bit6	bit5	bit4	bit0-bit3
Name	-	Position reference Type	Position reference Update mode *1	Enabled new position reference (Valid edge change)	-
Set value	See Table 4-2 Relationship between status switching and control commands.	-	-	-	See Table 4-2 Relationship between status switching and control commands.
Description	For details, see "6.5.3 Details of Parameters Defined by Sub-protocols" on Page 182.	0: Target position 607Ah is an absolute position reference. 1: Target position 607Ah is a relative position reference.	0: Not update immediately 1: Update immediately	0 to 1 A new position reference is enabled in advance. However, whether the reference can be enabled successfully is determined by the servo status. 1 to 0 Bit12 of the control word 6041h is cleared in advance. However, whether bit12 is cleared successfully is determined by the servo status.	For details, see "6.5.3 Details of Parameters Defined by Sub-protocols" on Page 182.

NOTE

Note 1: *1. When the servo drive meets certain conditions and the displacement reference is updated, the two attributes of the reference, namely change mode and reference type, are locked and cannot be modified during running of the displacement reference. Other attributes can be modified in immediate change mode.

Note 2: The attributes of a displacement reference includes: acceleration 6083, deceleration 6084, maximum velocity 6081, target position 607A, reference change mode 6040 bit5, and reference type 6040 bit6.

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

Index 6041h	Name	Status Word					Data Structure	VAR	Data Type	Uint16
	Accessibility	RO	Mapping	TPDO	Relevant Mode	All	Data Range	0 to 65535	Factory Default	-

It indicates the status of the servo drive in PP mode.

State word 6041h							
Bit	bit15	bit14	bit13	bit12	bit11	bit10	bit0-bit9
Name	Homing completed	NA	Position deviation status	Position reference receiving supported	Software internal setting exceeding limit	Target reached	-
Set value	-	-	For details, see Table 4-2. Table 4-2 Relationship between status switching and control commands	-	-	-	For details, see Table 4-2. Table 4-2 Relationship between status switching and control commands
Description	0: Homing is not performed or homing is not complete. 1: Homing is complete and the reference point is found.	Reserved	0: The position deviation is within the following error window (6065h). 1: The position deviation exceeds the following error window (6065h).	0: The servo drive can receive a new displacement reference. 1: The servo drive cannot receive a new displacement reference.	0: The position reference does not reach the software position limit (607Dh). 1: The position reference or feedback reaches the software position limit (607Dh). *1	0: The target position is not reached. 1: The target position is reached. *2	For details, see "6.5.3 Details of Parameters Defined by Sub-protocols" on Page 182.

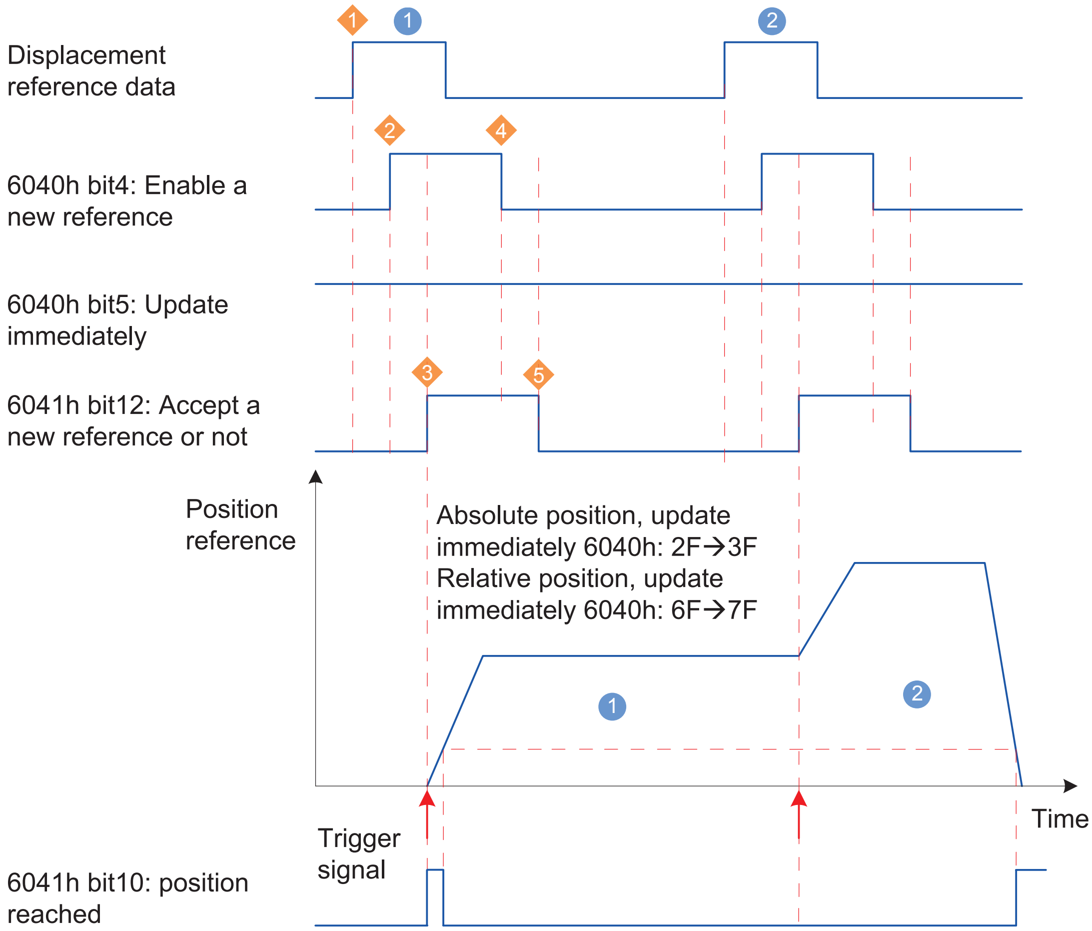
NOTE

Note 1: *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 Details of Parameters Defined by Sub-protocols" on Page 182.

Note 2: *2. When the position deviation is within the position window 6067h and the time reaches the value set by 6068h, the target position is reached. If either condition is not met, the target position is not reached.

a) Time sequence 1: Update immediately

Figure 4-8 Time sequence and motion profile 1 in the mode of update immediately



NOTE

Note: 1. A trigger signal needs to be sent again when any parameter of the displacement reference is modified.

Note 2: ① The host computer modifies other attributes of the displacement reference (profile acceleration/deceleration 6083h, profile deceleration 6084h, maximum velocity 6081h, and target displacement 607Ah) as required.

Note 3: ② The host computer changes bit4 of 6040h to 1 from 0, prompting the slave node that a new displacement reference needs to be enabled.

Note 4: ③ After receiving the rising edge of 6040h bit4, the slave node judges whether to receive the new displacement reference.

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

In the mode of update immediately, the servo drive immediately executes the new displacement reference once it receives a new displacement reference (bit12 of 6041h is changed to 1 from 0).

Note 5: ④ After bit12 of the status word 6041h received by the host computer from the slave node is changed into 1, the host computer issues the displacement reference data and changes bit4 of 6040h to 0 from 1, indicating there is no new position reference currently.

Because the edge change of 6040h bit4 is valid, this operation does not interrupt the displacement reference being executed.

Note 6: ⑤ After detecting that bit4 of 6040h is changed to 0 from 1, bit12 of 6041h can be set to 0 from 1, indicating the slave node is ready to receive a new displacement reference.

In the mode of update immediately, when the slave node detects that bit4 of 6040h is changed to 0 from 1, the slave node always sets bit12 of 6041h to 0.

If a new displacement reference ② is received when the current displacement reference ① is being executed, the displacement reference not executed in ① is not discarded. For a relative position reference, after a new displacement reference is complete, total displacement increment = target position increment 607Ah of ① + target position increment 607Ah of ② . For an absolute position reference, after a new position reference is complete, user absolute position = target position 607Ah of ② .

Example:

Example: two position references, mode of update immediately, absolute position reference

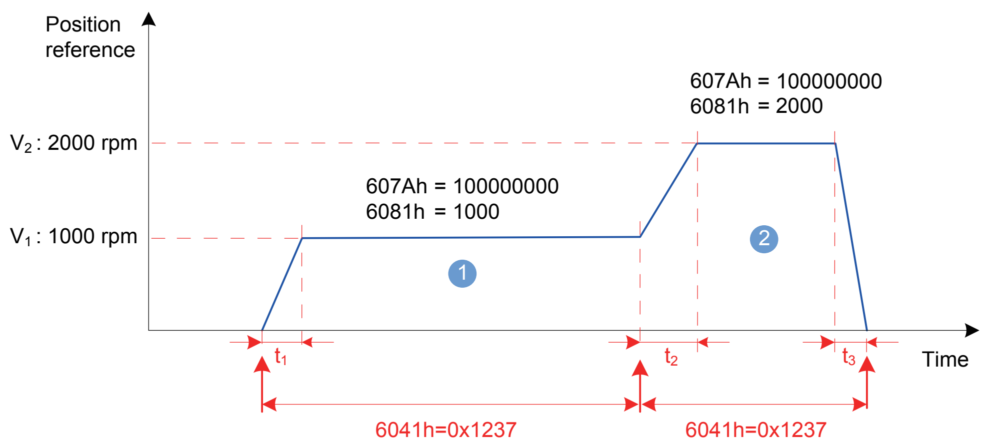
Displacement reference ① :

- Target position 607A = 100000000
- 6081 = 1000 rpm

Displacement reference ② :

- Target position 607A = 100000000
- 6081 = 2000 rpm

Figure 4-9 Time sequence and motion profile 2 in the mode of update immediately



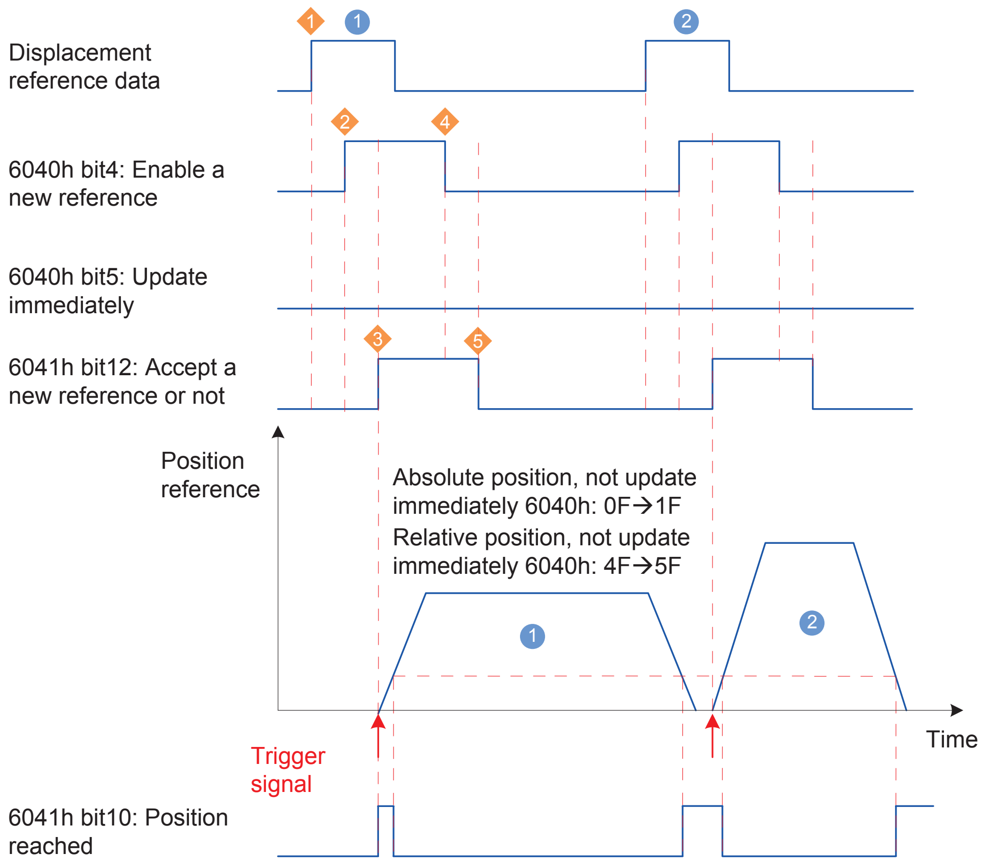
If bit12 of 6041h=0	Step 5	6041h=0x1637
Step 3	6040h=0x2F	Position reached
6040h=0x2F	6041h=0x0237	
Step 4	Step 6	
6040h=0x3F	6040h=0x3F	
	6041h=0x1237	

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

SN	Control Command 6040h	Status of 6041h	Description
1	0x06	0x0231	The drive is ready to receive a new reference.
2	0x07	0x0233	The drive is ready to receive a new reference and the servo can be enabled.
3	0x2F	0x0637	A new reference can be received and the servo is enabled (because no other position references are received before the displacement reference ① is executed, the target position is considered to be 0 and bit12 of 6041h is 1 in the target position).
4	0x3F	0x1237	The drive already receives a reference and is executing the reference. The servo does not reach the target position.
◆ If the target position 607Ah remains unchanged, the velocity 6081h needs to be modified. Perform the following operation when the displacement reference is not positioned.			
5	0x2F	0x0237	Bit12 of 6041h is released and the servo drive can receive a new reference again. The current reference is being executed and the target position is not reached.
6	0x3F	0x1237	The drive already receives a reference and is executing the reference. The target position is not reached.
◆ If a new target position 607Ah does not need to be entered and parameters of the current position reference do not need to be modified, wait until the current position reference is complete. After positioning is complete, current user position 6063h = 607Ah and status word 6041h = 0x1637. ◆ If a new target displacement needs to be entered and smooth transition between positions is required, repeat operations 5 and 6 before positioning of the current position reference is complete.			
7	0x3F	0x1637	The target position is not reached.

b) Time sequence 2: Not update immediately

Figure 4-10 Time sequence and motion profile 1 in the mode of not update immediately



NOTE

Note: 1. A trigger signal needs to be sent again when any parameter of the displacement reference is modified.

Note 2: ① The host computer modifies other attributes of the displacement reference (profile acceleration/deceleration 6083h, profile deceleration 6084h, maximum velocity 6081h, and target displacement 607Ah) as required.

Note 3: ② The host computer changes bit4 of 6040h to 1 from 0, prompting the slave node that a new displacement reference needs to be enabled.

Note 4: ③ After receiving the rising edge of 6040h bit4, the slave node judges whether to receive the new displacement reference.

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

Note 5: ④ After bit12 of the status word 6041h received by the host computer is changed to 1, displacement reference data can be released and bit4 of 6040h is changed to 0 from 1, indicating there is no new position reference currently.

Because the edge change of 6040h bit4 is valid, this operation does not interrupt the displacement reference being executed.

Note 6: ⑤ After detecting that 6040h bit4 changes from 1 to 0, the drive releases 6041h bit12, indicating it is ready to receive a new position reference. In the mode of not update immediately, the servo drive can receive a new displacement reference only after it completes execution of the previous one. The servo drive immediately executes the new reference once it receives a new reference (bit12 of 6041h is changed to 1 from 0).

Example:

Example: two position references, not update immediately, absolute position reference

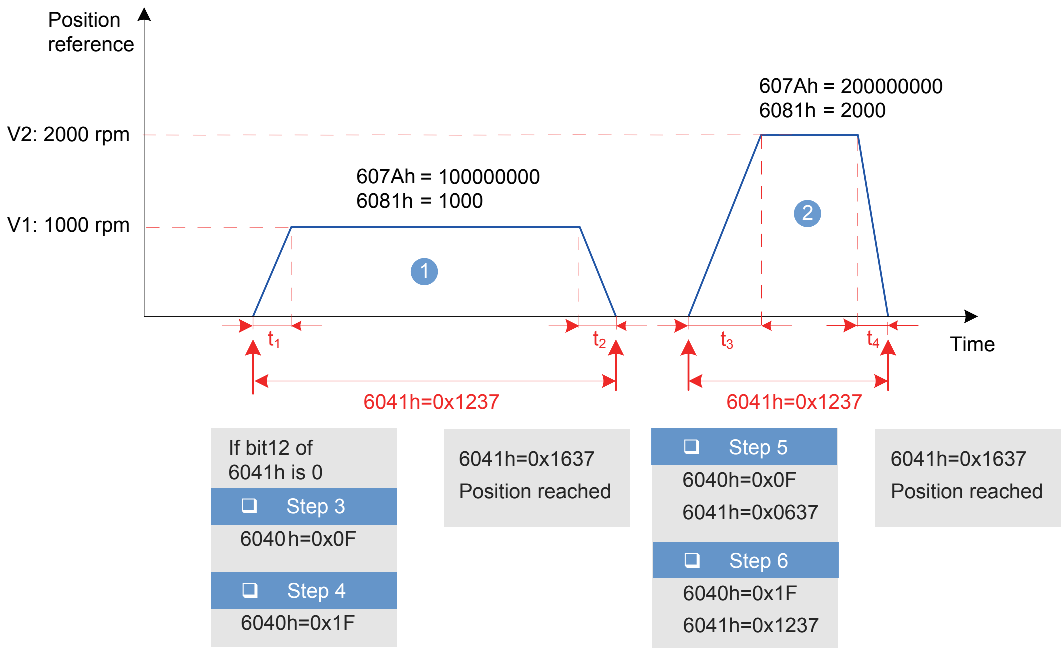
Displacement reference ① :

- Target position 607A = 100000000
- 6081 = 1000 rpm

Displacement reference ② :

- Target position 607A = 200000000
- 6081 = 2000 rpm

Figure 4-11 Time sequence and motion profile 2 in the mode of not update immediately

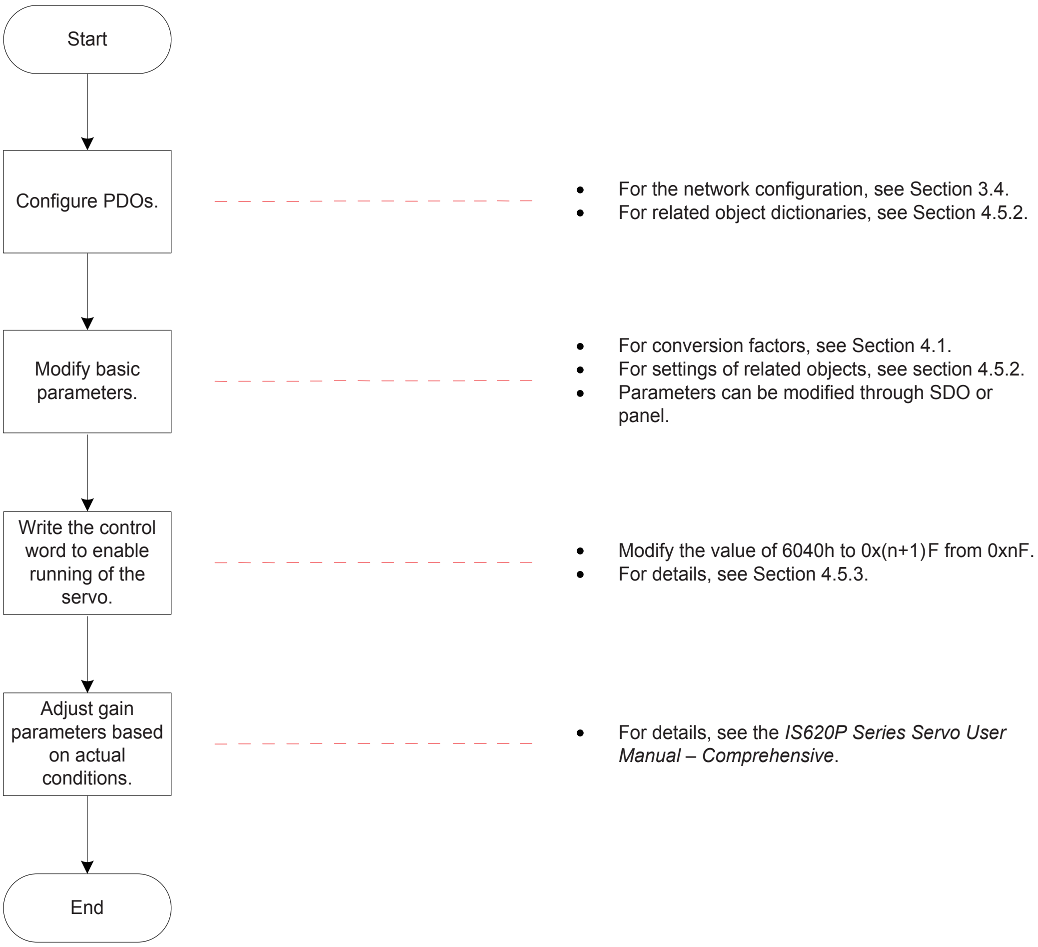


$t_1 = \frac{V_1}{6083h}$ ms $\quad t_2 = \frac{V_1}{6084h}$ ms $\quad t_3 = \frac{V_2}{6083h}$ ms $\quad t_4 = \frac{V_2}{6084h}$ ms

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

4.5.4 Configuration Example

Figure 4-12 Example of PP mode configuration flowchart



Functional Code	Object	Mapping Object	Input Content	Description
H18-34	1600h-00h	Number of RPDO1 mapping objects	2	
H18-35	1600h-01h	6040h-00h	60400010h	The first mapping parameter of RPDO1 is 6040-00h. The parameter is 16 bits long.
H18-37	1600h-02h	6060h-00h	60600008h	The second mapping parameter of RPDO1 is 6060-00h. The parameter is 8 bits long.
H19-02	1601h-00h	Number of RPDO2 mapping objects	2	
H19-03	1601h-01h	607Ah-00h	607A0020h	The first mapping parameter of RPDO2 is 607A-00h. The parameter is 32 bits long.
H19-05	1601h-02h	6081h-00h	60810020h	The second mapping parameter of RPDO2 is 6081-00h. The parameter is 32 bits long.
H19-19	1602h-00h	Number of RPDO3 mapping objects	2	
H19-20	1602h-01h	6083h-00h	60830020h	The first mapping parameter of RPDO3 is 6083-00h. The parameter is 32 bits long.
H19-21	1602h-02h	6084h-00h	60840020h	The second mapping parameter of RPDO3 is 6084-00h. The parameter is 32 bits long.
H1A-24	1A00h-00h	Number of TPDO1 mapping objects	2	
H1A-25	1A00h-01h	6041h-00h	60410010h	The first mapping parameter of TPDO1 is 6041-00h. The parameter is 16 bits long.
H1A-27	1A00h-02h	6061h-00h	60610008h	The second mapping parameter of TPDO1 is 6061-00h. The parameter is 8 bits long.
H1A-41	1A01h-00h	Number of TPDO2 mapping objects	2	
H1A-42	1A01h-01h	6064h-00h	60640020h	The first mapping parameter of TPDO2 is 6064-00h. The parameter is 32 bits long.
H1A-44	1A01h-02h	606Ch-00h	606C0020h	The second mapping parameter of TPDO2 is 606C-00h. The parameter is 32 bits long.

- Set the drive mode 6060h to 0x01 to make the drive work in PP mode.
- Set the target position 607Ah (user unit, default value: 0p).
- Set the constant speed of current displacement reference 6081h (user unit, default value: 100 rpm).
- Set acceleration 6083h (user unit, default value: 100 rpm/ms) and deceleration 6084h (100 rpm/ms) of each displacement reference according to requirements.
- Set the control word 6040h to 0x(n+1)F from 0xnF and enable the servo drive.

Position Reference Type 6040h bit6	Reference Change Change Mode 6040h bit5	6040h	Description
0	0	0x0F → 0x1F	Absolute position, not update immediately
0	1	0x2F → 0x3F	Absolute position, update immediately
1	0	0x4F → 0x5F	Relative position, not update immediately
1	1	0x6F → 0x7F	Relative position, update immediately

Monitoring parameters:

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

For specific operations on different reference types and update types, see "4.5.3 Control Commands in PP Mode" on Page 72.

4.6 Homing Mode

This mode is used to searches for the mechanical home and determine the position relationship 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 complete, the motor stops at the location of mechanical home. The relationship 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 mechanical zero.

In homing mode, the host computer should first selects the homing mode (6098h), set the homing speed (6099-1h and 6099-2h) and homing acceleration (609Ah), and issue the homing trigger signal. The servo drive automatically searches for the mechanical home according to the setting and sets the relative position relationship between the mechanical home and mechanical zero. The servo drive completes control over the position, speed, and torque inside.

4.6.1 Control Block Diagram

Figure 4-12 Control block diagram of the homing mode

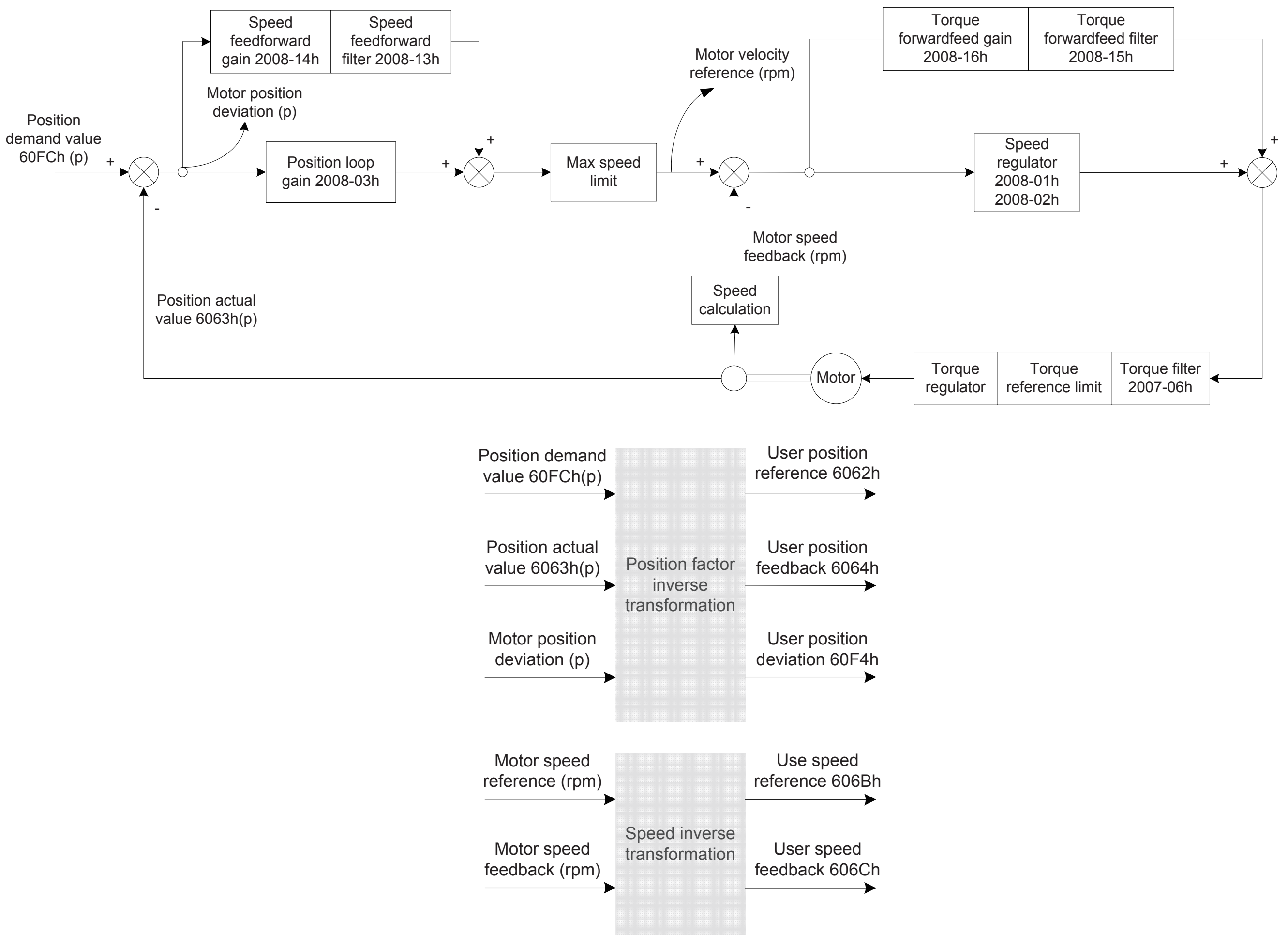
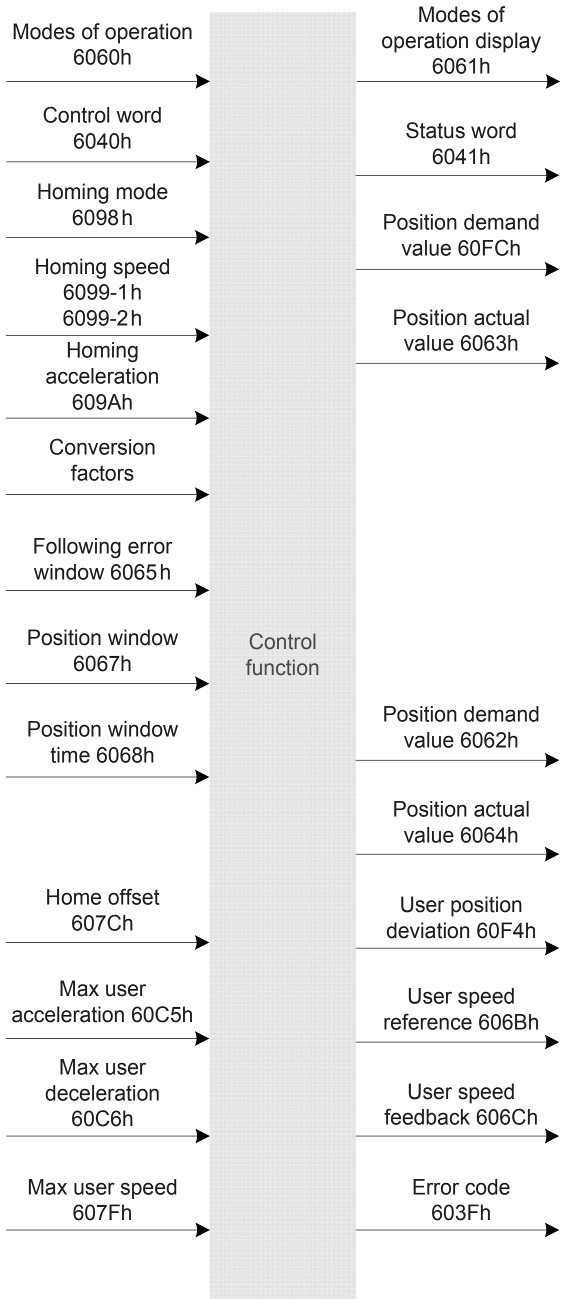


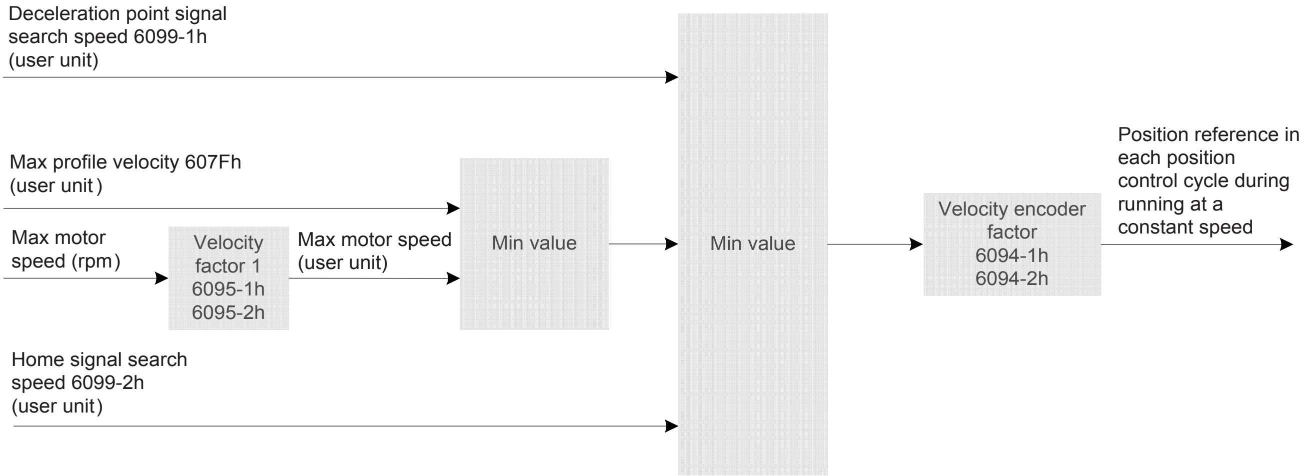
Figure 4-13 Input/output block diagram in homing mode



Figures 4-14 and 4-15 show processing of the servo drive for homing speeds and homing acceleration/ deceleration.

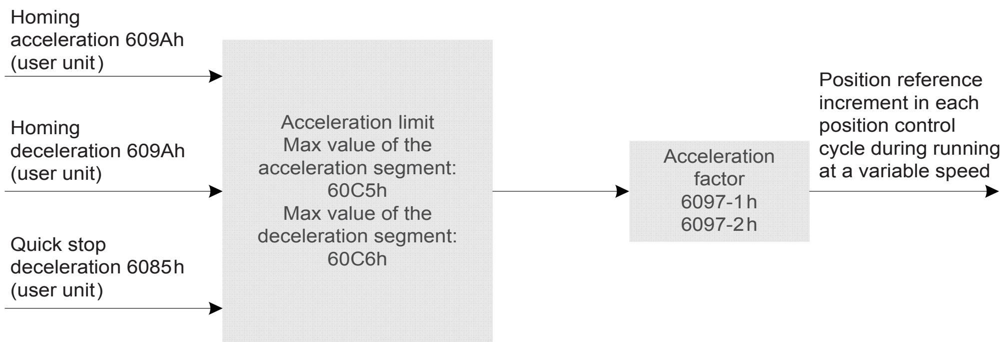
Two speeds are involved during homing. One is the speed during search for switch 6099-1h (user speed unit) and the other is the speed during search for zero 6099-2h (user speed unit). 6099-1h can be set to a large value to prevent homing timeout due to long homing time. 6099-1h should be set to a small value to prevent overshoot during high-speed stop of the servo drive and large deviation of the stop position from the preset mechanical home.

Figure 4-14 Homing speeds 6099h: speed limit



Homing acceleration 609Ah is used during acceleration and deceleration. When quick stop is enabled in homing mode, deceleration is determined by 6085h.

Figure 4-15 Homing acceleration 609Ah limit



4.6.2 Relevant Object Setting

1) Homing timeout

Index 2005h	Name	Time of Home Searching				Data Structure	VAR	Data Type	Uint16
	Accessibility	RW	Mapping	YES	Relevant Mode hm	Data Range	0-65535	Factory Default	50000

Sub-index: 24h
Unit: 10 ms
If homing is not complete within the duration, Er.601 is reported.
This fault can be reset.

2) Positioning complete

Index 6067h	Name	Position Window					Data Structure	VAR	Data Type	Uint32
	Accessibility	RW	Mapping	YES	Relevant Mode	pp/hm/ip	Data Range	0 to 4294967295	Factory Default	734 p

Sub-index: 00
When the position deviation 60F4h of the user unit is smaller than 6068h and time reaches this value, bit10 of 6041h is 1.
When either condition is not met, the position window is invalid.

Index 6068h	Name	Position Window Time					Data Structure	VAR	Data Type	Uint16
	Accessibility	RW	Mapping	YES	Relevant Mode	pp/hm/ip	Data Range	0 to 65535	Factory Default	0 ms

Sub-index: 00
When the position deviation 60F4h of the user unit is smaller than 6068h and time reaches this value, bit10 of 6041h is 1.
The position reached signal is invalid when either of the condition is not met.

3) Detection for Following Error Window

Index 6065h	Name	Following Error Window					Data Structure	VAR	Data Type	Uint32
	Accessibility	RW	Mapping	YES	Relevant Mode	pp/hm/ip	Data Range	0 to 4294967295	Factory Default	3145728 p

Sub-index: 00
When the position deviation is larger than this value, Er.B00 occurs and bit13 of the status word 6041h is set to 1.

4) Homing speed

Index 6099h	Name	Homing Speed				Data Structure	ARR	Data Type	Uint32
	Accessibility	RW	Mapping	YES	Relevant Mode	All	Data Range	OD Data Range	Factory Default	OD Default Value
It sets the speeds used in homing procedure.									

Sub-index 00h	Name	Number of Entries				Data Structure	-	Data Type	Uint8
	Accessibility	RO	Mapping	NO	Relevant Mode	-	Data Range	2	Factory Default	2

Sub-index 01h	Name	Speed during search for switch (Speed During Search for Switch)				Data Structure	-	Data Type	Uint32
	Accessibility	RW	Mapping	YES	Relevant Mode	-	Data Range	0 to 4294967295	Factory Default	100 rpm
It sets the speed during search for the deceleration point signal. The speed can be set to a large value to prevent homing timeout due to long homing time. Note: After finding the deceleration point, the slave node decelerates and shields change of the home signal. To prevent the slave node from reaching the home signal during deceleration, set the position of the deceleration point switch properly to reserve sufficient deceleration distance or increase homing acceleration to shorten the deceleration time.									

Sub-index 02h	Name	Speed During Search for Zero				Data Structure	-	Data Type	Int32
	Accessibility	RW	Mapping	YES	Relevant Mode	-	Data Range	0 to 4294967295	Factory Default	10 rpm
It sets the speed (user speed unit) during search for the home signal. Set this parameter to a small value to prevent overshoot during high-speed stop and large deviation of the stop position from the preset mechanical home.									

5) Homing acceleration

Index 609Ah	Name	Homing Acceleration				Data Structure	VAR	Data Type	Uint32
	Accessibility	RW	Mapping	YES	Relevant Mode	hm	Data Range	0 to 4294967295	Factory Default	100 rpm/ms

It sets acceleration in homing mode. This parameter is used during acceleration and deceleration.

The set value takes effect after homing is enabled.

◆ By default, the object dictionary means that the acceleration (in the unit of rpm/ms) of the motor from 0 rpm to 1000 rpm. It is calculated in the following formula:

$$609Ah(rpm/ms) = \frac{1000rpm}{Xms}$$

X: time used by the motor to decelerate from 1000 rpm to 0 rpm For example, 100 rpm/ms means that X equals 10 ms, that is, the motor decelerates from 1000 rpm to 0 within 10 ms.

4 Motion Mode

Index	Sub-index	Name	Accessibility	Mapping	Data Type	Unit	Range	Factory Default
603Fh	00h	Error code	RO	TPDO	Uint16	-	0–65535	-
6040h	00h	Control word	RW	YES	Uint16	-	0–65535	0
6041h	00h	Status word	RO	TPDO	Uint16	-	0–65535	-
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	User 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	User unit	-2^{31} to $(2^{31}-1)$	-
6065h	00h	Following error window	RW	YES	Uint32	User unit	0 to $(2^{32}-1)$	3145728
6067h	00h	Position window	RW	YES	Uint32	User 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	rpm	-2^{31} to $(2^{31}-1)$	-
606Ch	00h	Velocity actual value	RO	TPDO	Int32	rpm	-2^{31} to $(2^{31}-1)$	-
607Dh	01h	Minimum Software Position Limit	RW	YES	Int32	User unit	-2^{31} to $(2^{31}-1)$	-2^{31}
	02h	Max Software Position Limit	RW	YES	Int32	User unit	-2^{31} to $(2^{31}-1)$	$2^{31}-1$
607Ch	00h	Home Offset	RW	YES	Int32	User unit	-2^{31} to $(2^{31}-1)$	0
6098h	00h	Homing mode	RW	YES	Int8	-	0 to 35	0
6099h	01h	Speed during search for switch	RW	YES	Uint32	rpm	0 to $(2^{32}-1)$	100
	02h	Speed during search for zero	RW	YES	Int32	rpm	0 to $(2^{32}-1)$	10
609Ah	00h	Homing acceleration	RW	YES	Uint32	rpm/ms	0 to $(2^{32}-1)$	100
2005h	24h	Duration limit of homing	RW	YES	Uint16	10 ms	0 to 65535	50000
60F4h	00h	Following error actual value	RO	TPDO	Int32	User unit	-2^{31} to $(2^{31}-1)$	-
60FCh	00h	Position demand value	RO	TPDO	Int32	User unit	-2^{31} to $(2^{31}-1)$	-
2007h	06h	Torque reference filter time constant	RW	YES	Uint16	ms	0 to 30.00	0.79

Index	Sub-index	Name	Accessibility	Mapping	Data Type	Unit	Range	Factory 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 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

4.6.3 Control Commands in Homing Mode

Table 4-2 Relationship between status switching and control commands

CiA402 Status Switching		Control Word 6040h	Bit0 to Bit9*1 of Status Word 6041h
0	Power-on → Initialization	Natural transition, control command not required	0x0000h
1	Initialization → No fault	Natural transition, control command not required If an error occurs during initialization, the servo drive directly goes to status 13.	0x0250h
2	No fault → Ready	0x06h	0x0231h
3	Ready → Wait to enable servo	0x07h	0x0233h
4	Wait to enable servo → Running	0x0Fh	0x0237h
5	Running → Wait to enable servo	0x07h	0x0233h
6	Wait to enable servo → Ready	0x06h	0x0231h
7	Ready → No fault	0x00h	0x0250h
8	Running → Ready	0x06h	0x0231h
9	Running → No fault	0x00h	0x0250h
10	Wait to enable servo → No fault	0x00h	0x0250h
11	Running → Quick stop	0x02h	0x0217h
12	Quick stop → No fault	Set 605A to a value in the range 0 to 3. Natural transition is performed after stop and no control command is required.	0x0250h
13	→ Stop upon fault	Once a fault occurs in any status other than "fault", the servo drive automatically switches to the status of stop upon fault without any control command.	0x021Fh
14	Stop upon fault → Fault	Natural transition after stop at fault, control command not required	0x0218h
15	Fault → No fault	0x80h The rising edge of bit7 is valid. If bit7 is 1, other control commands are invalid.	0x0250h
16	Quick stop → Running	Set 605A to a value in the range 5 to 7. After the stop process is complete, 0x0F is sent.	0x0237h

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

Index 6040h	Name	Control Word				Data Structure	VAR	Data Type	Uint16
	Accessibility	RW	Mapping	YES	Relevant Mode	All	Data Range	0 to 65535	Factory Default	-

It sets the control commands in homing mode.

Control Word 6040h				
Bit	bit7-15	bit5-bit6	bit4	bit0-bit3
Name	-	N/A	Homing enable	-
Set value	See Table 4-2 Relationship between status switching and control commands.	-	-	See Table 4-2 Relationship between status switching and control commands.
Description	For details, see "6.5.3 Details of Parameters Defined by Sub-protocols" on Page 182.	-	0: Homing is not activated. 0→1: Enable homing. 1: Homing is ongoing. 1→0: Interrupt homing. Bit4 must always be 1 during homing.	For details, see "6.5.3 Details of Parameters Defined by Sub-protocols" on Page 182.

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

Index 6041h	Name	Status Word				Data Structure	VAR	Data Type	Uint16
	Accessibility	RO	Mapping	TPDO	Relevant Mode	All	Data Range	0 to 65535	Factory Default	-

It indicates the status of the servo drive in homing mode.

State word 6041h							
Bit	bit15	bit14	bit13	bit12	bit11	bit10	bit0-bit9
Name	Homing completed	NA	Homing error	Homing completed	Software internal setting exceeding limit	Target reached	-
Set value	-	-	For details, see Table 4-2. Table 4-2 Relationship between status switching and control commands	-	-	-	For details, see Table 4-2. Table 4-2 Relationship between status switching and control commands
Description	0: Homing is not performed or homing is not complete. 1: Homing is complete and the reference point is found. This bit is unrelated to the drive mode and status of the drive.	Reserved	0: No error 1: An error occurs in homing.*1	0: Homing is not complete. 1: Homing is complete.	0: The actual position value does not reach the software position limit. 1: The actual position value reaches the software position limit.*2	0: The target position is not reached. 1: The target position is reached. *3	For details, see "6.5.3 Details of Parameters Defined by Sub-protocols" on Page 182.

NOTE

Note 1: *1: When a homing error occurs, Er.601 (homing timeout) occurs in the servo drive. If any error or warning occurs during homing, bit13 of 6041 is set to 1.

Note 2: *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 Details of Parameters Defined by Sub-protocols" on Page 182.

Note 3: *3: When the position deviation is within the position window 6067h and the time reaches the value set by 6068h, the target position is reached. If either condition is not met, the target position is not reached.

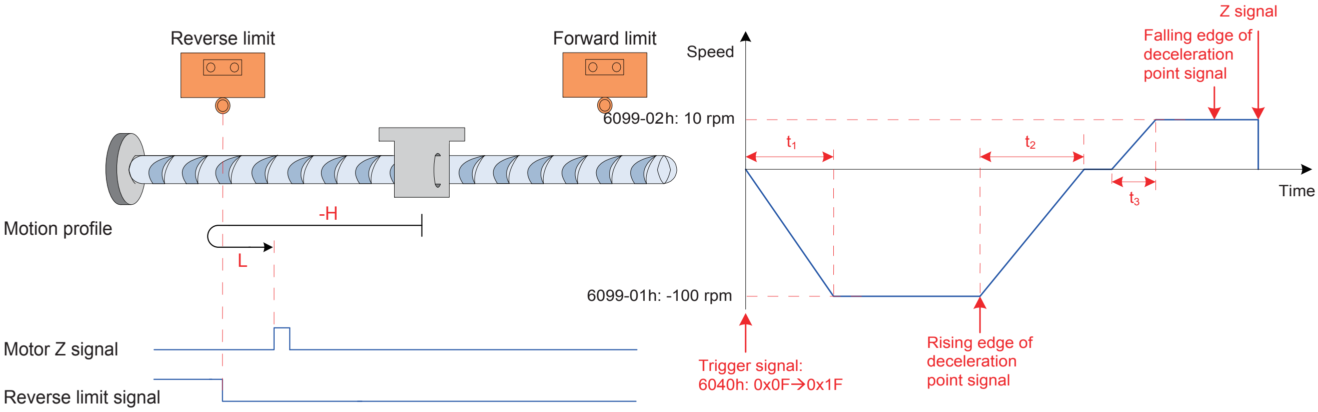
4.6.4 Introduction to the Homing Mode

1) 6098h=1

- Mechanical home: motor Z signal
- Deceleration point: reverse limit switch

a) Invalid deceleration point signal at start of homing

Figure 4-16 Mode ① in which 6098h is 1 and deceleration point signal is invalid



Note 1: In Figure 4-16, "H" indicates search for the deceleration point signal speed 6099-1h and "L" indicates search for the home signal speed 6099-2h.

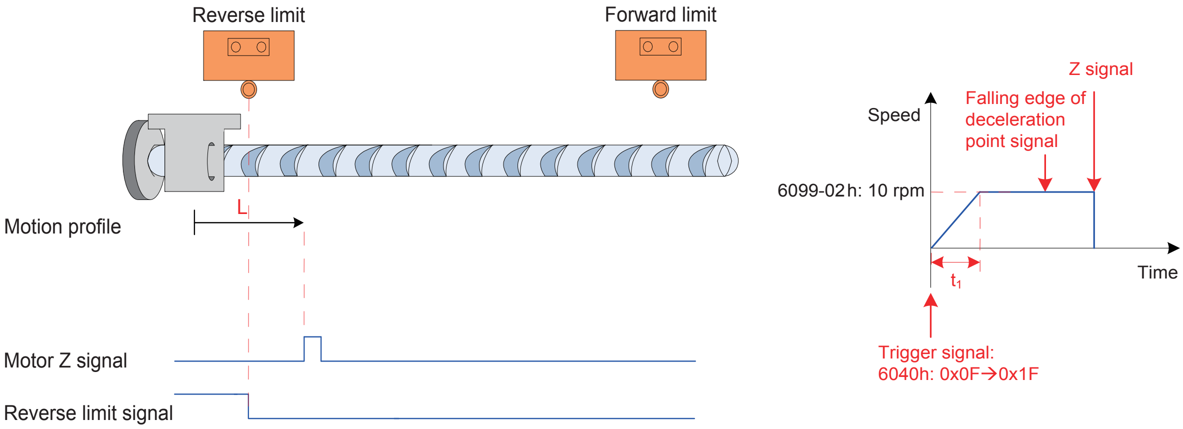
6099-1H=100 rpm, 6099-2h=10 rpm, 609Ah=100 rpm/ms:

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

The N-OT signal is 0 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, runs in the reverse direction, and then runs 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.

b) Valid deceleration point signal at start of homing

Figure 4-17 Mode ② in which 6098h is 1 and the deceleration point signal is valid



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

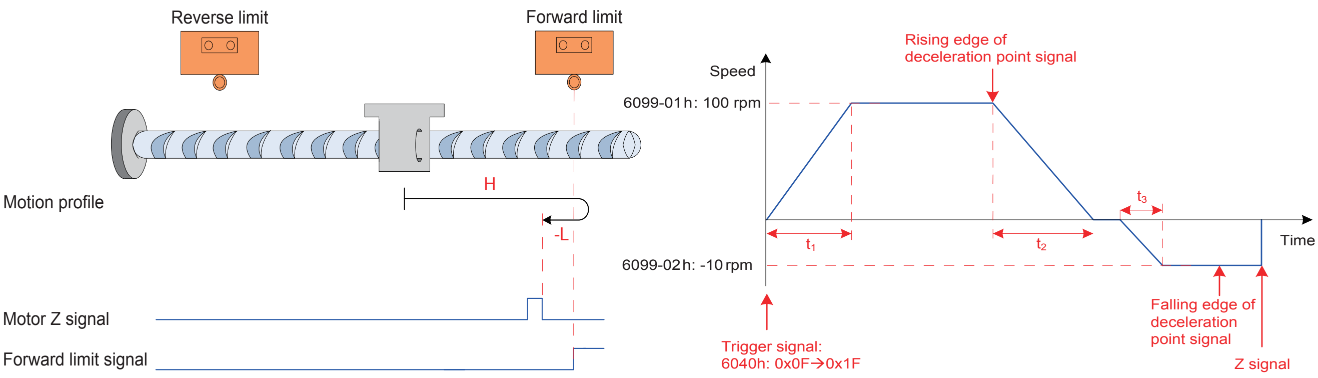
The N-OT signal is 1 initially and the motor directly starts homing in the forward direction at a low speed. After reaching the falling edge of the N-OT signal, the motor stops at the first motor Z signal.

2) 6098h=2

- Home: Z signal
- Deceleration point: forward limit switch

a) Invalid deceleration point signal at start of homing

Figure 4-18 Mode ① in which 6098h is 2 and the deceleration point signal is invalid

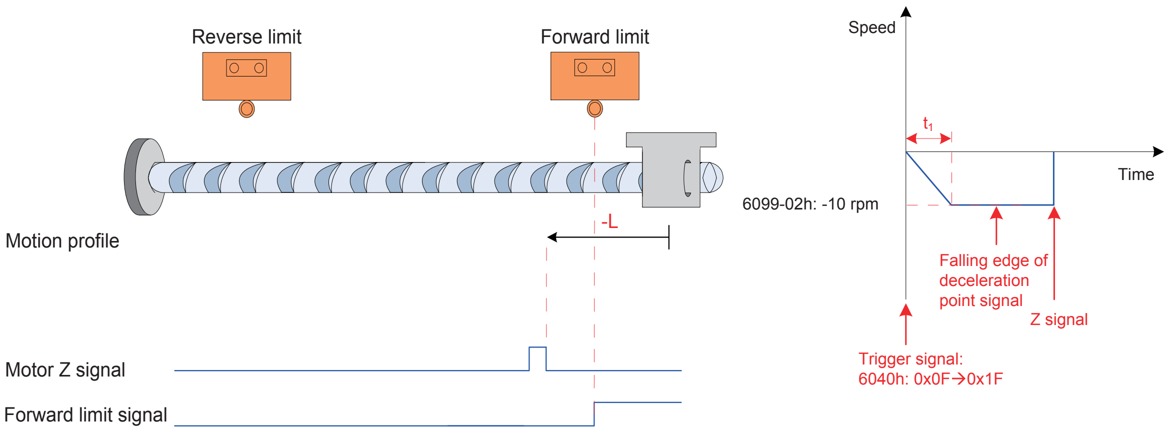


$$t_1 = \frac{6099\text{-}01h}{609Ah}\ ms \quad t_2 = \frac{6099\text{-}01h}{609Ah}\ ms \quad t_3 = \frac{6099\text{-}02h}{609Ah}\ ms$$

The P-OT signal is 0 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 then runs 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.

b) Valid deceleration point signal at start of homing

Figure 4-19 Mode ② in which 6098h is 2 and the deceleration point signal is valid



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

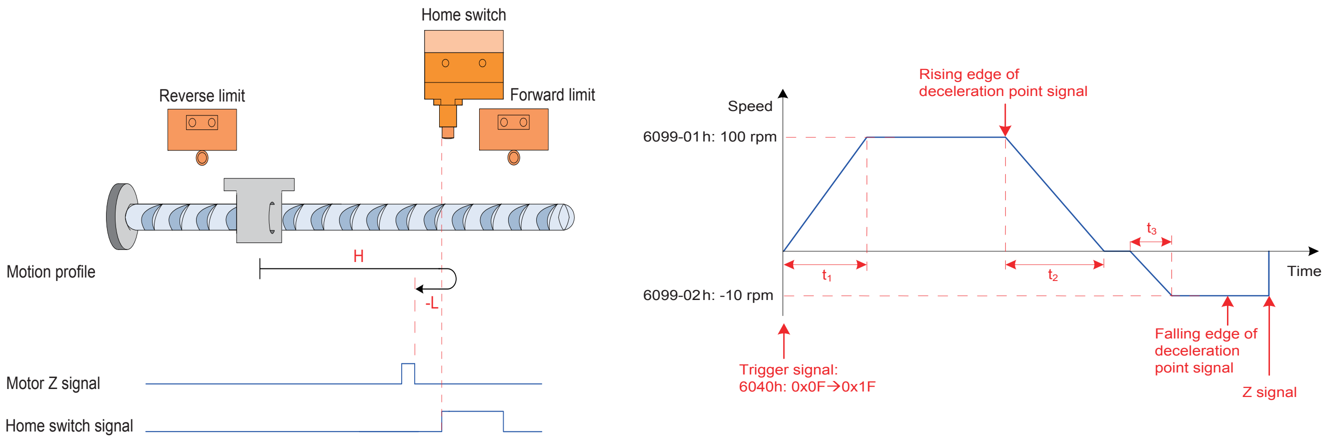
The P-OT signal is 1 initially and the motor directly starts homing in the reverse direction at a low speed. After reaching the falling edge of the P-OT signal, the motor stops at the first motor Z signal.

3) 6098h=3

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

a) Invalid deceleration point signal at start of homing

Figure 4-20 Mode ① in which 6098h is 3 and the deceleration point signal is invalid

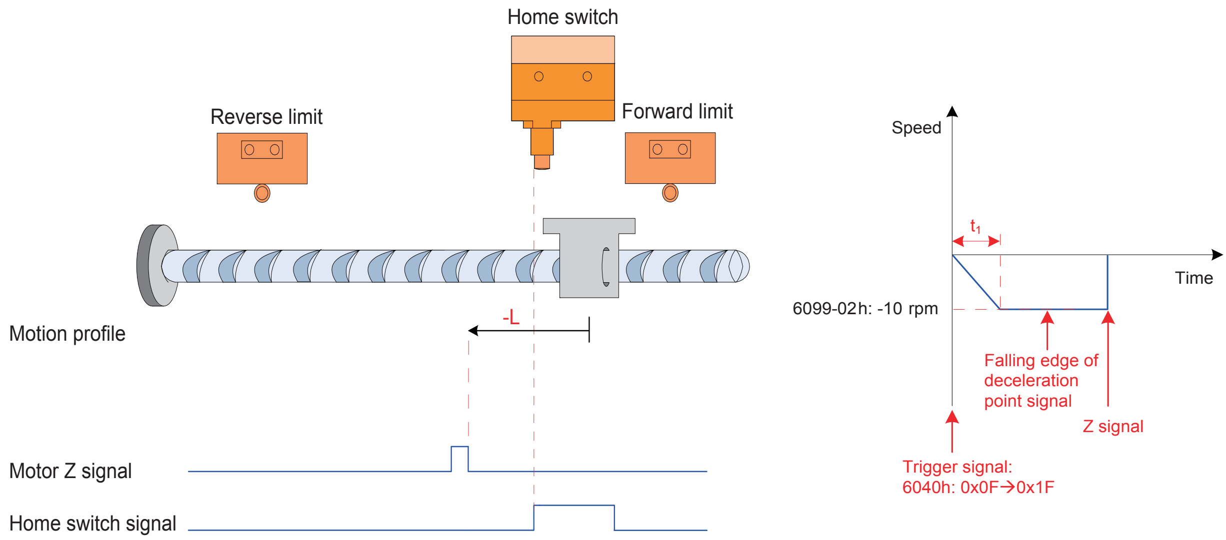


$$t_1 = \frac{6099\text{-}01h}{609Ah}\ \text{ms} \qquad t_2 = \frac{6099\text{-}01h}{609Ah}\ \text{ms} \qquad t_3 = \frac{6099\text{-}02h}{609Ah}\ \text{ms}$$

The HW signal is 0 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 then runs 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.

b) Valid deceleration point signal at start of homing

Figure 4-21 Mode ② in which 6098h is 3 and the deceleration point signal is valid



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

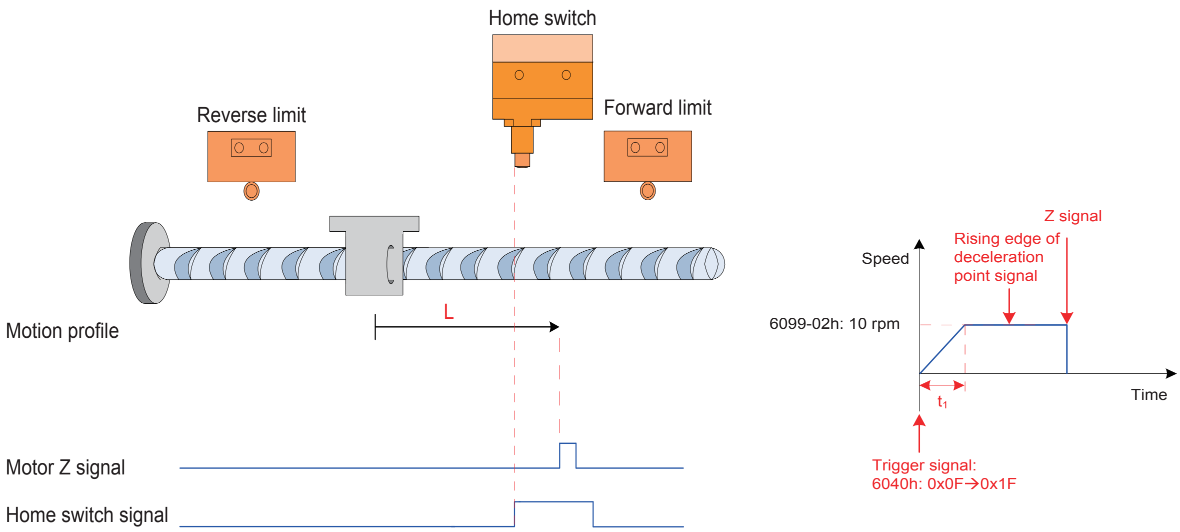
The HW signal is 1 initially and the motor directly starts homing in the reverse direction at a low speed. After reaching the falling edge of the HW signal, the motor stops at the first motor Z signal.

4) 6098h = 4

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

a) Invalid deceleration point signal at start of homing

Figure 4-22 Mode ① in which 6098h is 4 and the deceleration point signal is invalid

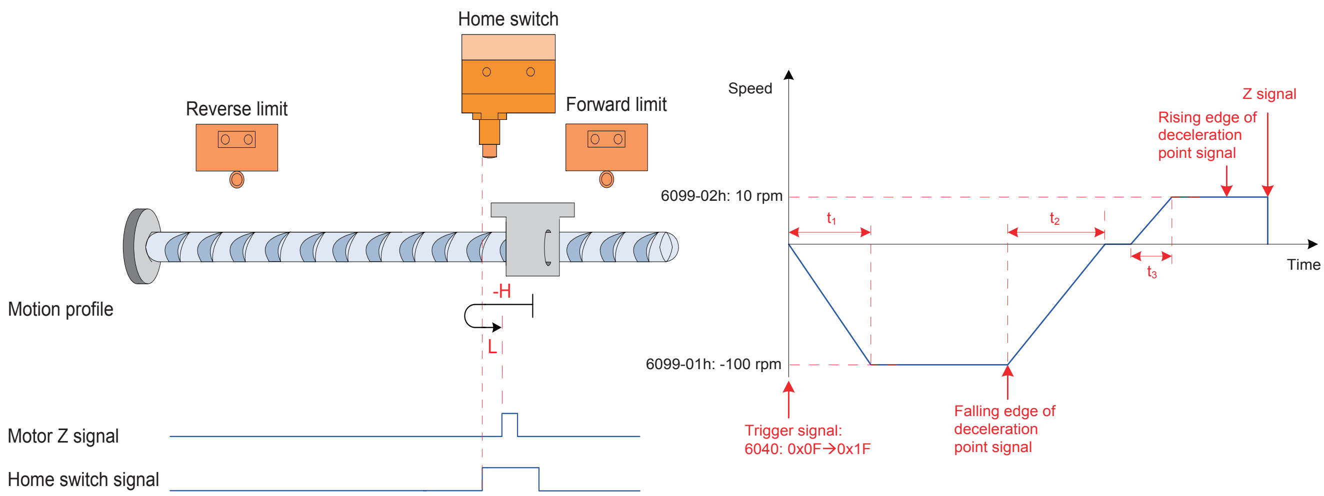


$$t_1 = \frac{6099\text{-}02h}{609Ah} \text{ ms}$$

The HW signal is 0 initially and the motor directly starts homing 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.

b) Valid deceleration point signal at start of homing

Figure 4-23 Mode ② in which 6098h is 4 and the deceleration point signal is valid



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

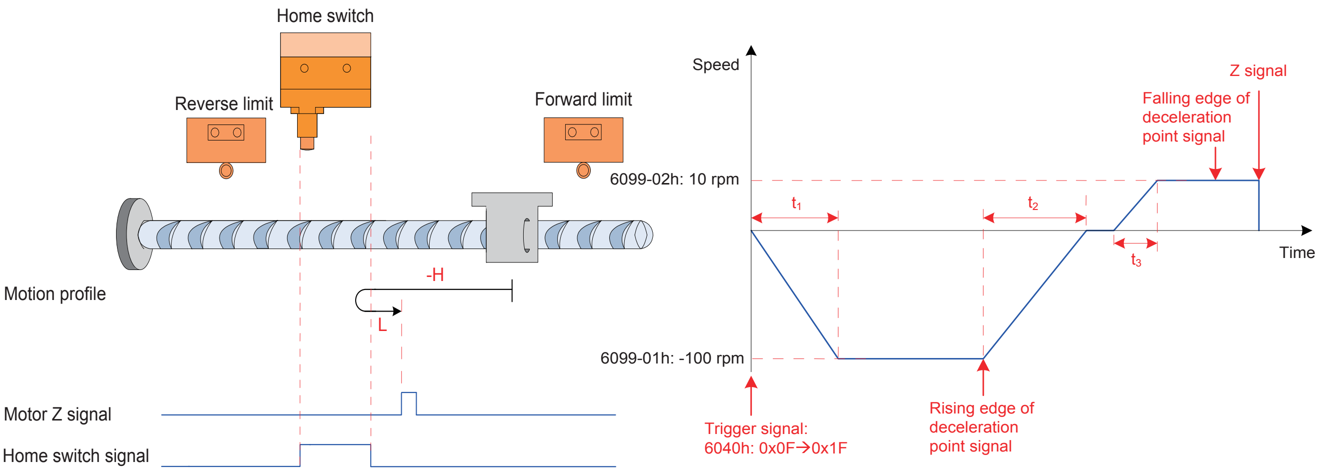
The HW signal is 1 initially and the motor starts homing in the reverse direction at a high speed. After reaching the falling edge of the HW signal, the motor decelerates, runs in the reverse direction, and then runs in the forward direction at a low speed. After reaching the rising edge of the HW signal, the motor stops at the first motor Z signal.

5) 6098h=5

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

a) Invalid deceleration point signal at start of homing

Figure 4-24 Mode ① in which 6098h is 5 and the deceleration point signal is invalid

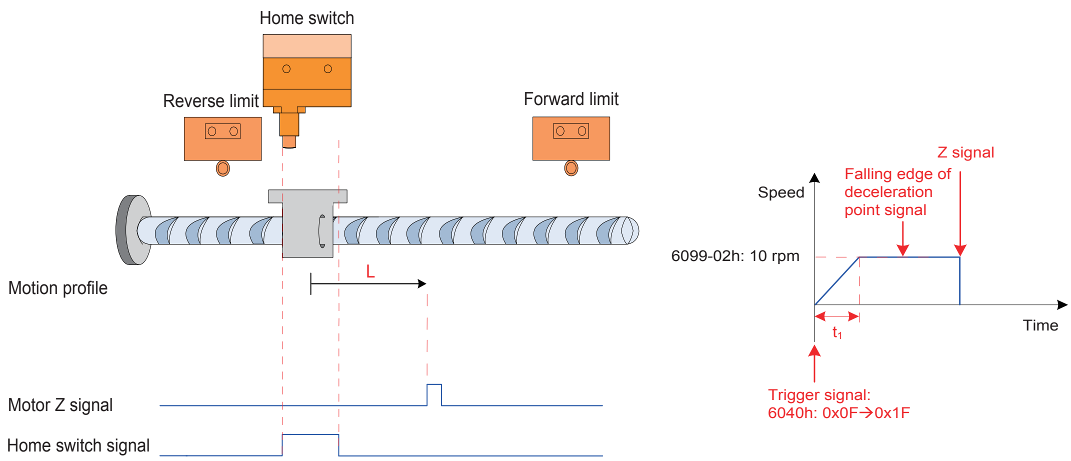


$$t_1 = \frac{6099\text{-}01h}{609Ah}\ ms \quad t_2 = \frac{6099\text{-}01h}{609Ah}\ ms \quad t_3 = \frac{6099\text{-}02h}{609Ah}\ ms$$

The HW signal is 0 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, runs in the reverse direction, and then runs 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.

b) Valid deceleration point signal at start of homing

Figure 4-25 Mode 2 in which 6098h is 5 and the deceleration point signal is valid



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

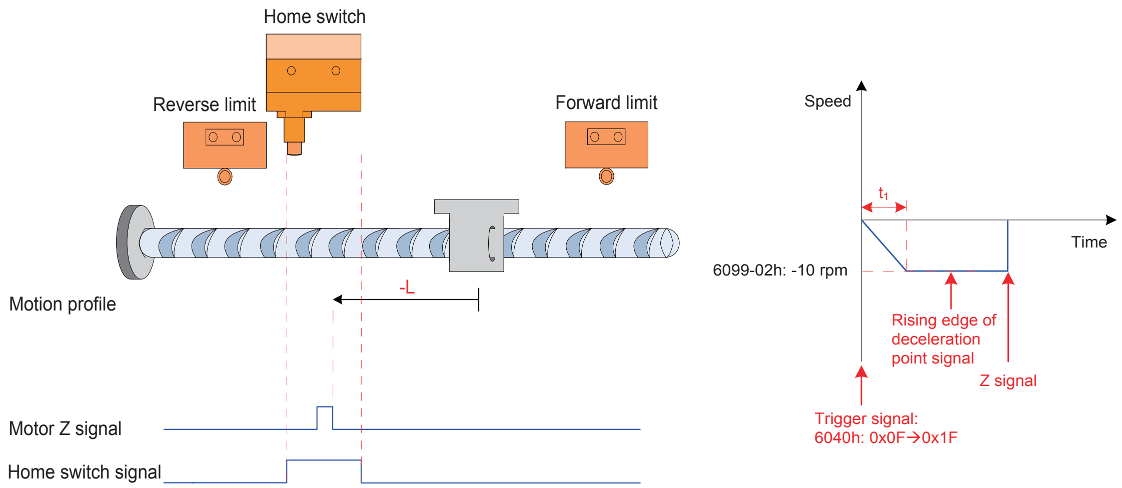
The HW signal is 1 initially and the motor directly starts homing in the reverse direction at a low speed. After reaching the falling edge of the HW signal, the motor stops at the first motor Z signal.

6) 6098h=6

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

a) Invalid deceleration point signal at start of homing

Figure 4-26 Mode ① in which 6098h is 6 and the deceleration point signal is invalid

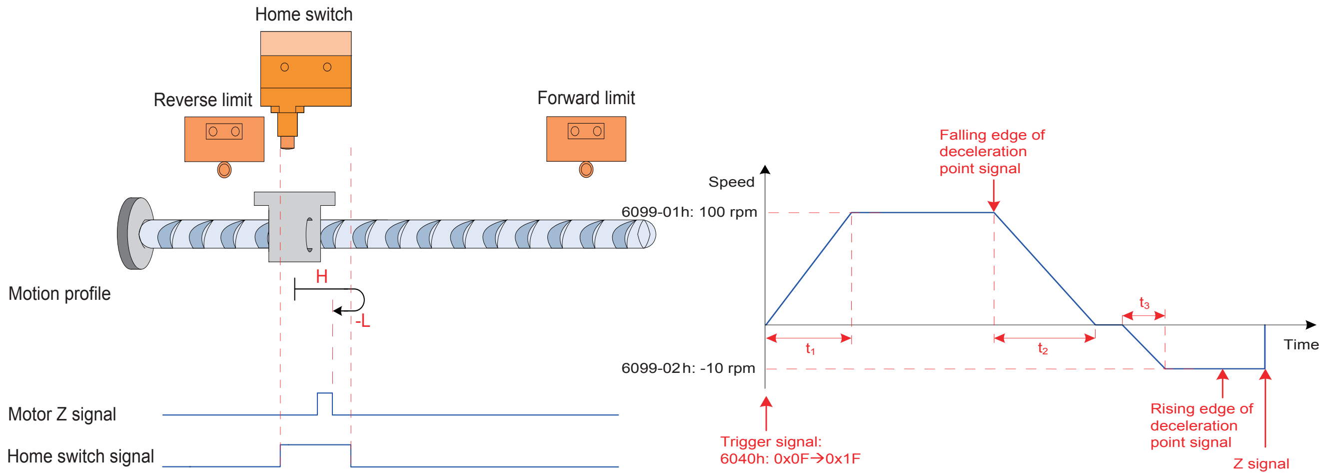


$$t_1 = \frac{6099\text{-}02h}{609Ah} \text{ ms}$$

The HW signal is 0 initially and the motor directly starts homing in the reverse direction at a low speed. After reaching the rising edge of the HW signal, the motor stops at the first motor Z signal.

b) Valid deceleration point signal at start of homing

Figure 4-27 Mode ② in which 6098h is 6 and the deceleration point signal is valid



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

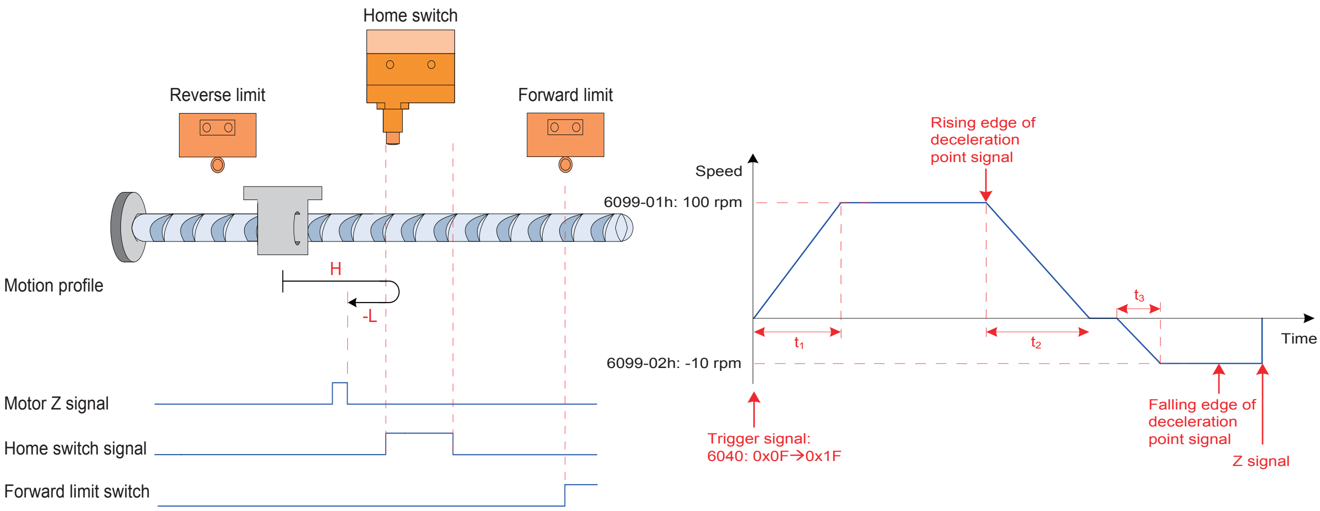
The HW signal is 1 initially and the motor starts homing in the forward direction at a high speed. After reaching the falling edge of the HW signal, the motor decelerates and then runs in the reverse direction at a low speed. After reaching the rising edge of the HW signal, the motor stops at the first motor Z signal.

7) 6098h= 7

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

a) Invalid deceleration point signal at start of homing start, not reaching forward limit switch

Figure 4-28 Mode ① in which 6098 is 7, the deceleration point signal is invalid, and the forward limit switch is not reached

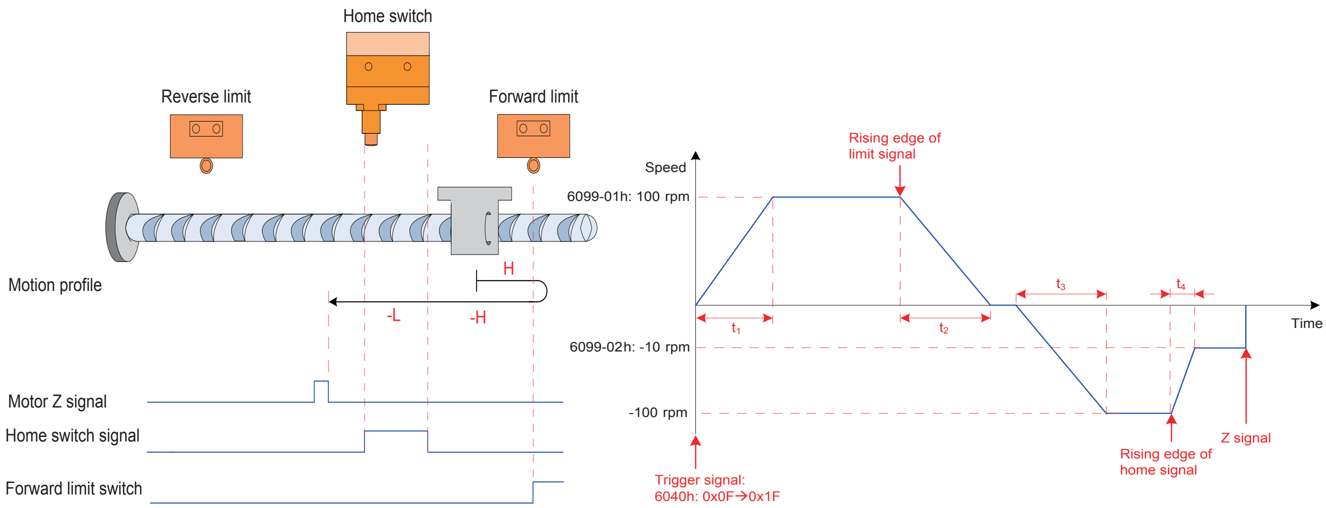


$$t_1 = \frac{6099\text{-}01h}{609Ah}\ ms \qquad t_2 = \frac{6099\text{-}01h}{609Ah}\ ms \qquad t_3 = \frac{6099\text{-}02h}{609Ah}\ ms$$

The HW signal is 0 initially and the motor starts homing in the forward direction at a high speed. If the motor does not reach the limit switch and reaches the rising edge of the HW signal, the motor decelerates and then runs 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.

b) Invalid deceleration point signal at start of homing, reaching the forward limit switch

Figure 4-29 Mode ② in which 6098 is 7, the deceleration point signal is invalid, and the forward limit switch is reached

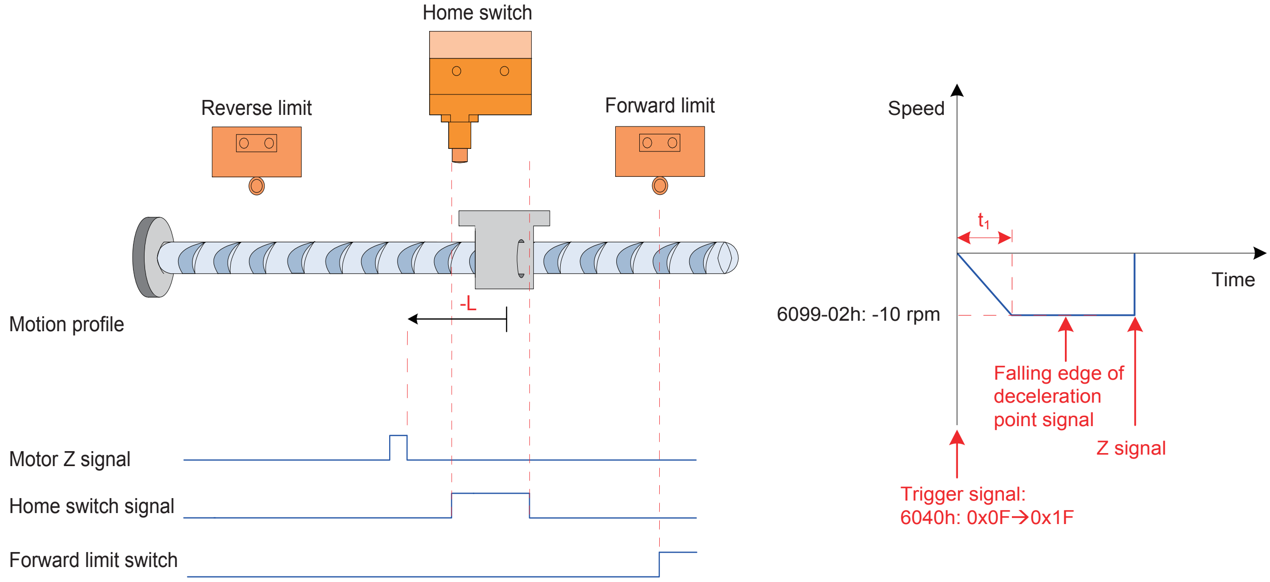


$t_1 = \frac{6099\text{-}01h}{609Ah}$ ms $t_2 = \frac{6099\text{-}01h}{609Ah}$ ms $t_3 = \frac{6099\text{-}01h}{609Ah}$ ms $t_4 = \frac{[6099\text{-}01h] - [6099\text{-}02h]}{609Ah}$ ms

The HW signal is 0 initially and the motor starts homing in the forward direction at a high speed. If the motor reaches the limit switch, the motor automatically runs in the reverse direction at a high speed. After reaching the rising edge of the HW signal, the motor decelerates and continues to run 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.

c) Valid deceleration point signal at start of homing

Figure 4-30 Mode ③ in which 6098 is 7 and the deceleration point signal is valid



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

The HW signal is 1 initially and the motor directly 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.

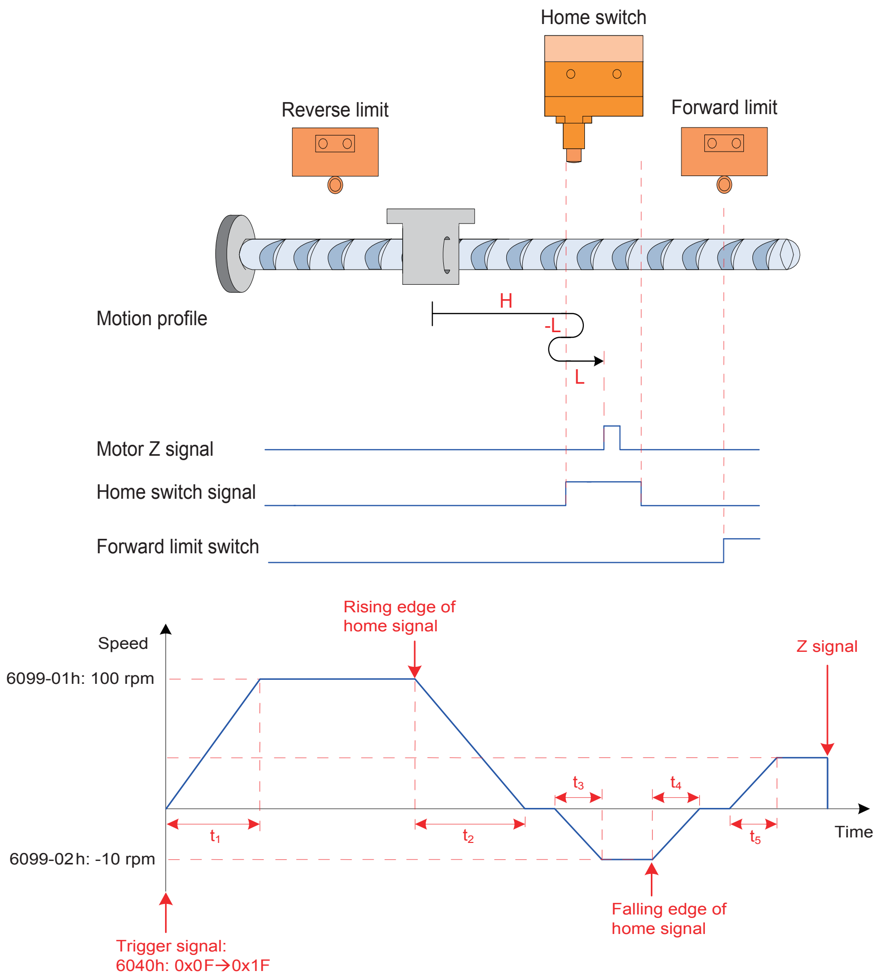
4
Motion Mode

8) 6098h=8

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

a) Invalid deceleration point signal at start of homing start, not reaching forward limit switch

Figure 4-31 Mode ① in which 6098h is 8, the deceleration point signal is invalid, and the forward limit switch is not reached

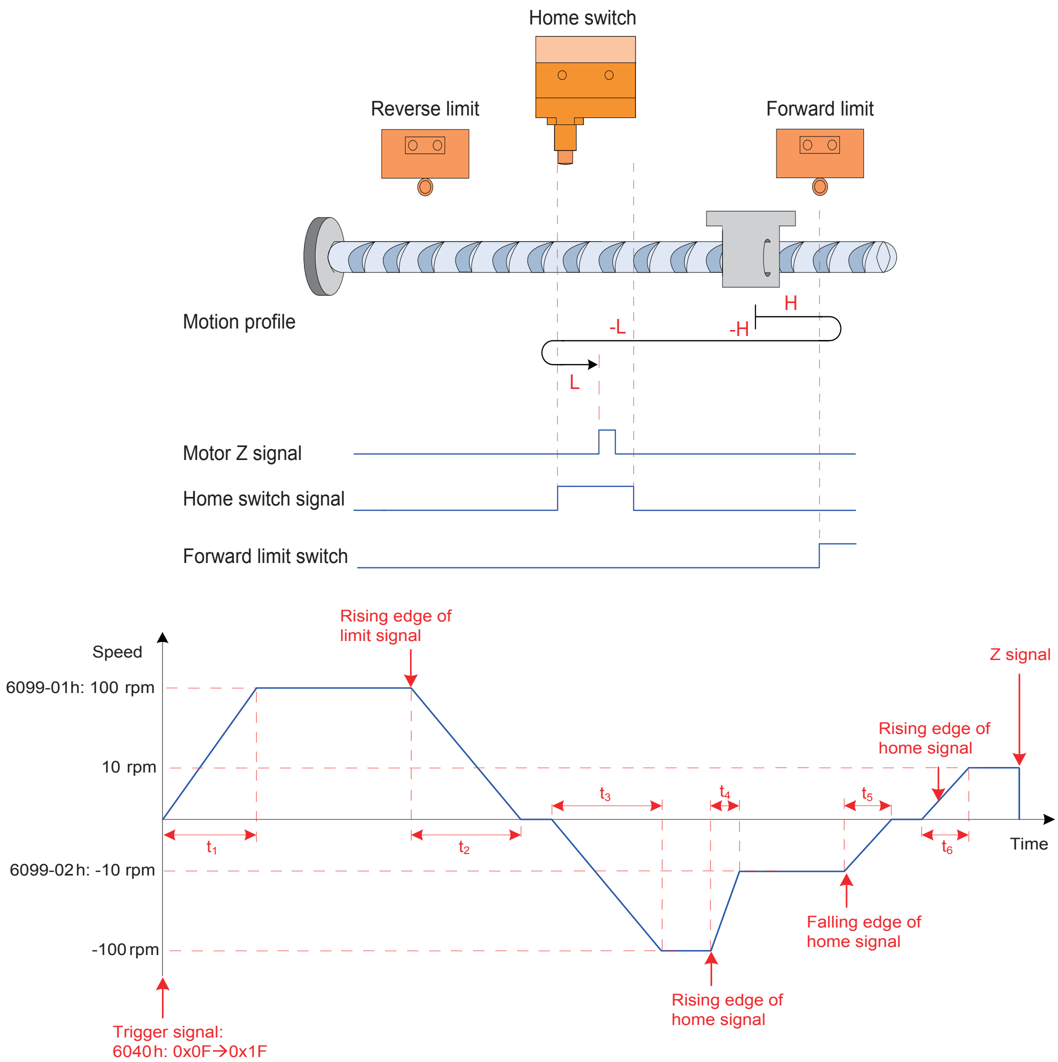


$t_1 = \frac{6099\text{-}01h}{609Ah}$ ms $t_2 = \frac{6099\text{-}01h}{609Ah}$ ms $t_3 = \frac{6099\text{-}02h}{609Ah}$ ms $t_4 = \frac{6099\text{-}02h}{609Ah}$ ms $t_5 = \frac{6099\text{-}02h}{609Ah}$ ms

The HW signal is 0 initially and the motor starts homing in the forward direction at a high speed. If the motor does not reach the limit switch, the motor decelerates and then runs in the reverse direction at a low speed after reaching the rising edge of the HW signal. After reaching the falling edge of the HW signal, the motor runs in the reverse direction and then runs 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.

b) Invalid deceleration point signal at start of homing, reaching the forward limit switch

Figure 4-32 Mode ② in which 6098h is 8, the deceleration point signal is invalid, and the forward limit switch is reached



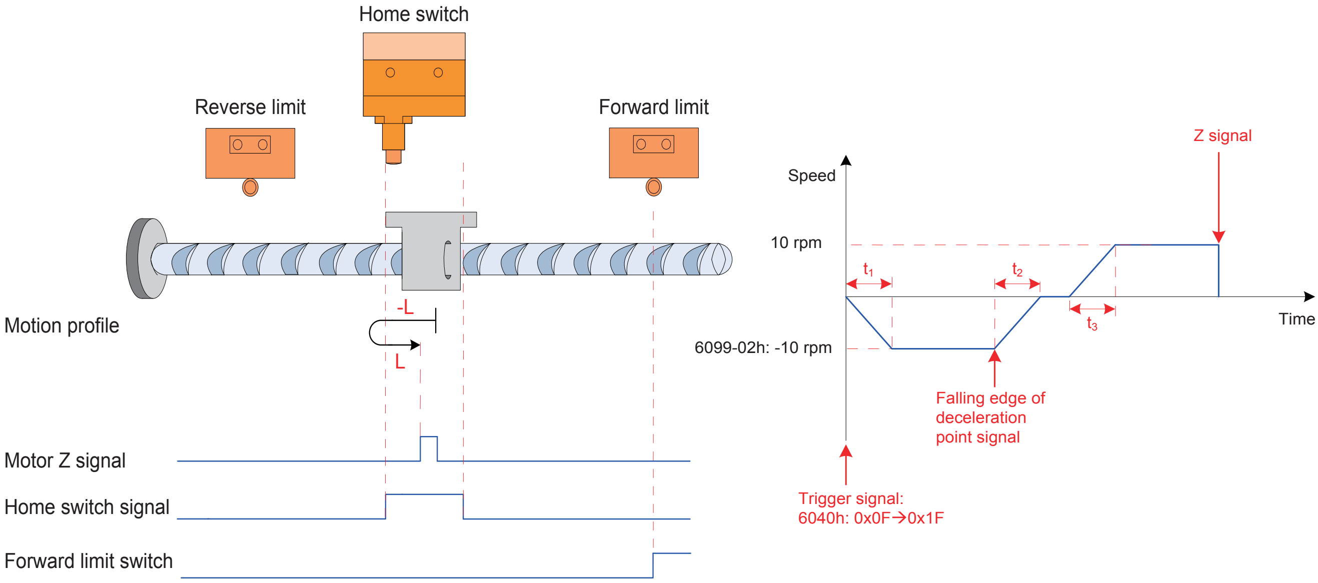
$t_1 = \frac{6099\text{-}01h}{609Ah}$ ms $t_2 = \frac{6099\text{-}01h}{609Ah}$ ms $t_3 = \frac{6099\text{-}01h}{609Ah}$ ms

$t_4 = \frac{[6099\text{-}01h] - [6099\text{-}02h]}{609Ah}$ ms $t_5 = \frac{6099\text{-}02h}{609Ah}$ ms $t_6 = \frac{6099\text{-}02h}{609Ah}$ ms

The HW signal is 0 initially and the motor starts homing in the forward direction at a high speed. If the motor reaches the limit switch, the motor automatically runs in the reverse direction at a high speed. After reaching the rising edge of the HW signal, the motor decelerates and runs in the reverse direction at a low speed. After reaching the falling edge of the HW signal, the motor runs in the reverse direction and then runs 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.

c) Valid deceleration point signal at start of homing

Figure 4-33 Mode ③ in which 6098h is 8 and the deceleration point signal is valid



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

The HW signal is 1 initially and the motor directly starts homing in the reverse direction at a low speed. After reaching the falling edge of the HW signal, the motor runs in the reverse direction and then runs 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.

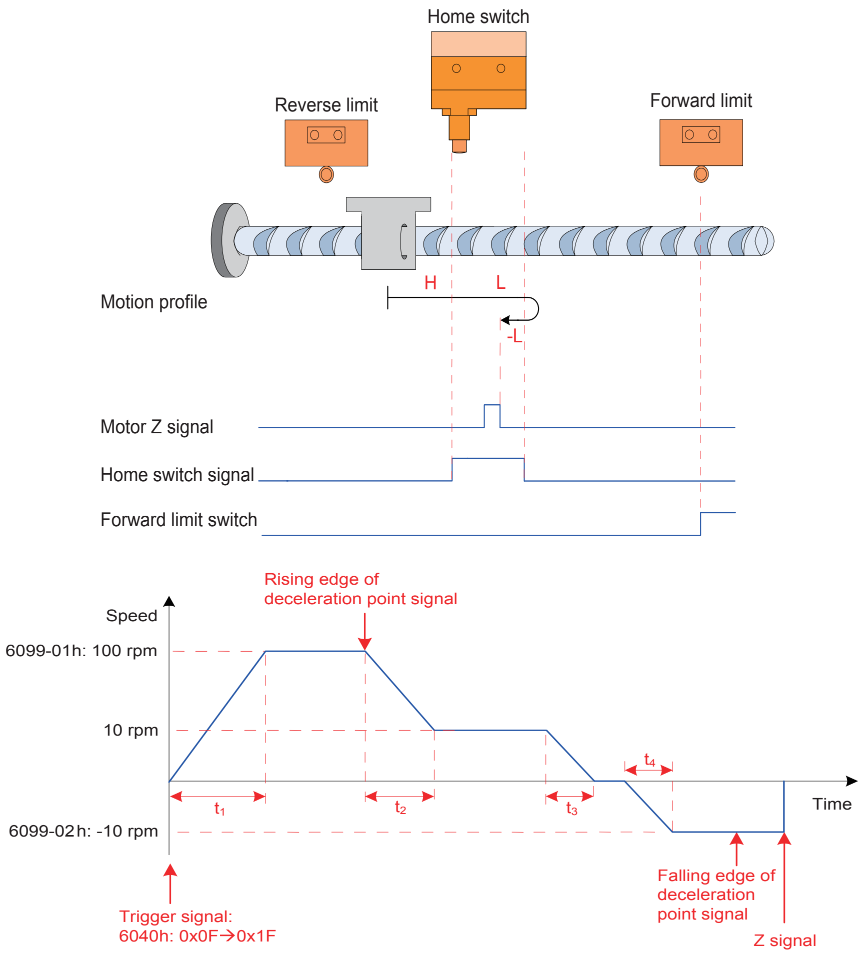
4 Motion Mode

9) 6098h=9

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

a) Invalid deceleration point signal at start of homing start, not reaching forward limit switch

Figure 4-34 Mode 1 in which 6098h is 9, the deceleration point signal is invalid, and the forward limit switch is not reached

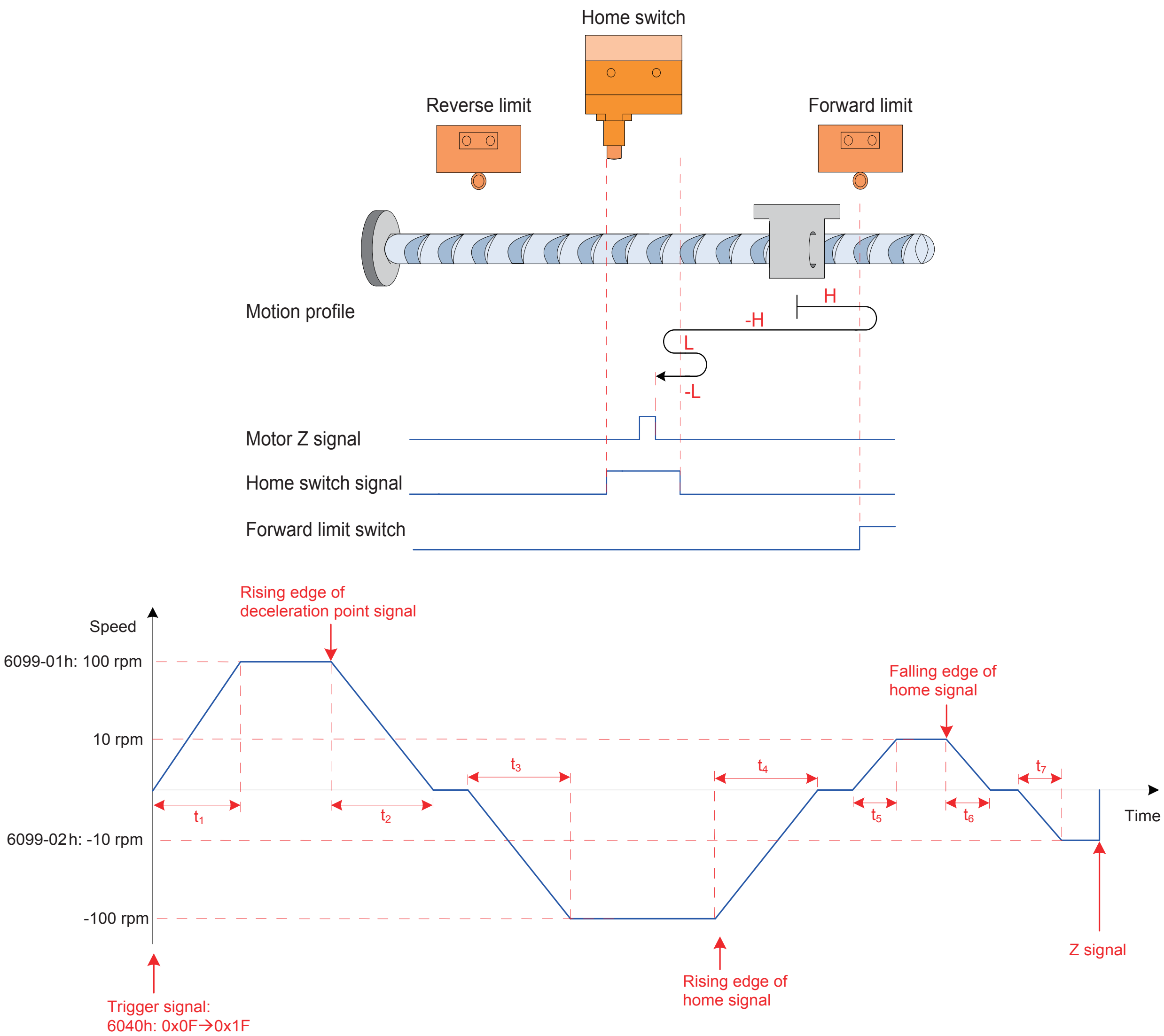


$t_1 = \frac{6099\text{-}01h}{609Ah}$ ms $t_2 = \frac{[6099\text{-}01h] - [6099\text{-}02h]}{609Ah}$ ms $t_3 = \frac{6099\text{-}02h}{609Ah}$ ms $t_4 = \frac{6099\text{-}02h}{609Ah}$ ms

The HW signal is 0 initially and the motor starts homing in the forward direction at a high speed. If the motor does not reach the limit switch, the motor decelerates and then runs in the forward direction at a low speed after reaching the rising edge of the HW signal. After reaching the falling edge of the HW signal, the motor runs in the reverse direction and then runs 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.

b) Invalid deceleration point signal at start of homing, reaching the forward limit switch

Figure 4-35 Mode ② in which 6098h is 9, the deceleration point signal is invalid, and the forward limit switch is reached



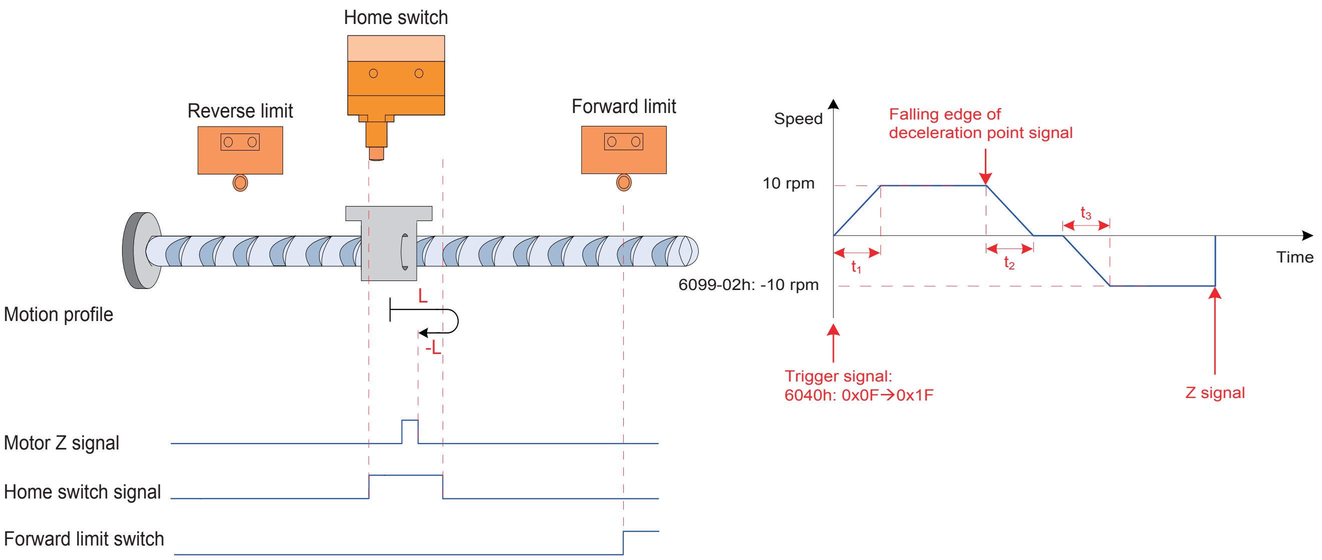
$$t_1 = \frac{6099\text{-}01h}{609Ah} \text{ ms} \quad t_2 = \frac{6099\text{-}01h}{609Ah} \text{ ms} \quad t_3 = \frac{6099\text{-}01h}{609Ah} \text{ ms} \quad t_4 = \frac{6099\text{-}01h}{609Ah} \text{ ms}$$

$$t_5 = \frac{6099\text{-}02h}{609Ah} \text{ ms} \quad t_6 = \frac{6099\text{-}02h}{609Ah} \text{ ms} \quad t_7 = \frac{6099\text{-}02h}{609Ah} \text{ ms}$$

The HW signal is 0 initially and the motor starts homing in the forward direction at a high speed. If the motor reaches the limit switch, the motor automatically runs in the reverse direction at a high speed. After reaching the rising edge of the HW signal, the motor decelerates, runs in the reverse direction, and then runs in the forward direction at a low speed. After reaching the falling edge of the HW signal, the motor runs in the reverse direction at a low speed. After reaching the rising edge of the HW signal, the motor stops at the first motor Z signal.

c) Valid deceleration point signal at start of homing

Figure 4-36 Mode ③ in which 6098h is 9 and the deceleration point signal is valid



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

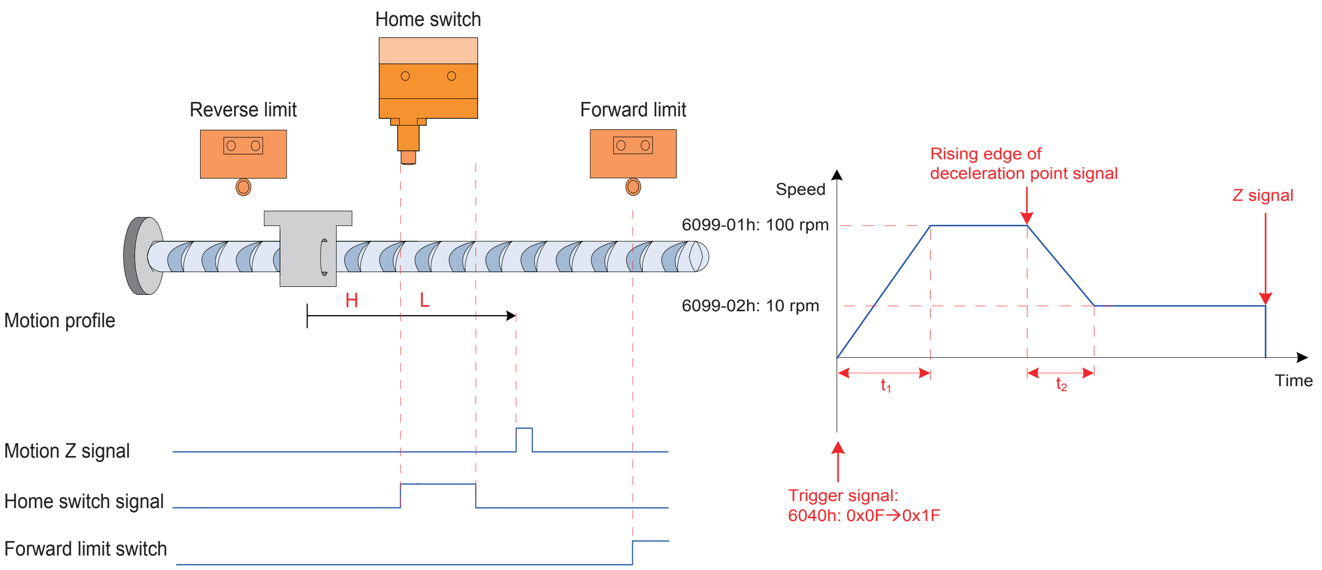
The HW signal is 1 initially and the motor directly starts homing in the forward direction at a low speed. After reaching the falling edge of the HW signal, the motor runs in the forward direction at a low speed. After reaching the rising edge of the HW signal, the motor stops at the first motor Z signal.

10) 6098h =10

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

a) Invalid deceleration point signal at start of homing start, not reaching forward limit switch

Figure 4-37 Mode ① in which 6098h is 10, the deceleration point signal is invalid, and the forward limit switch is not reached

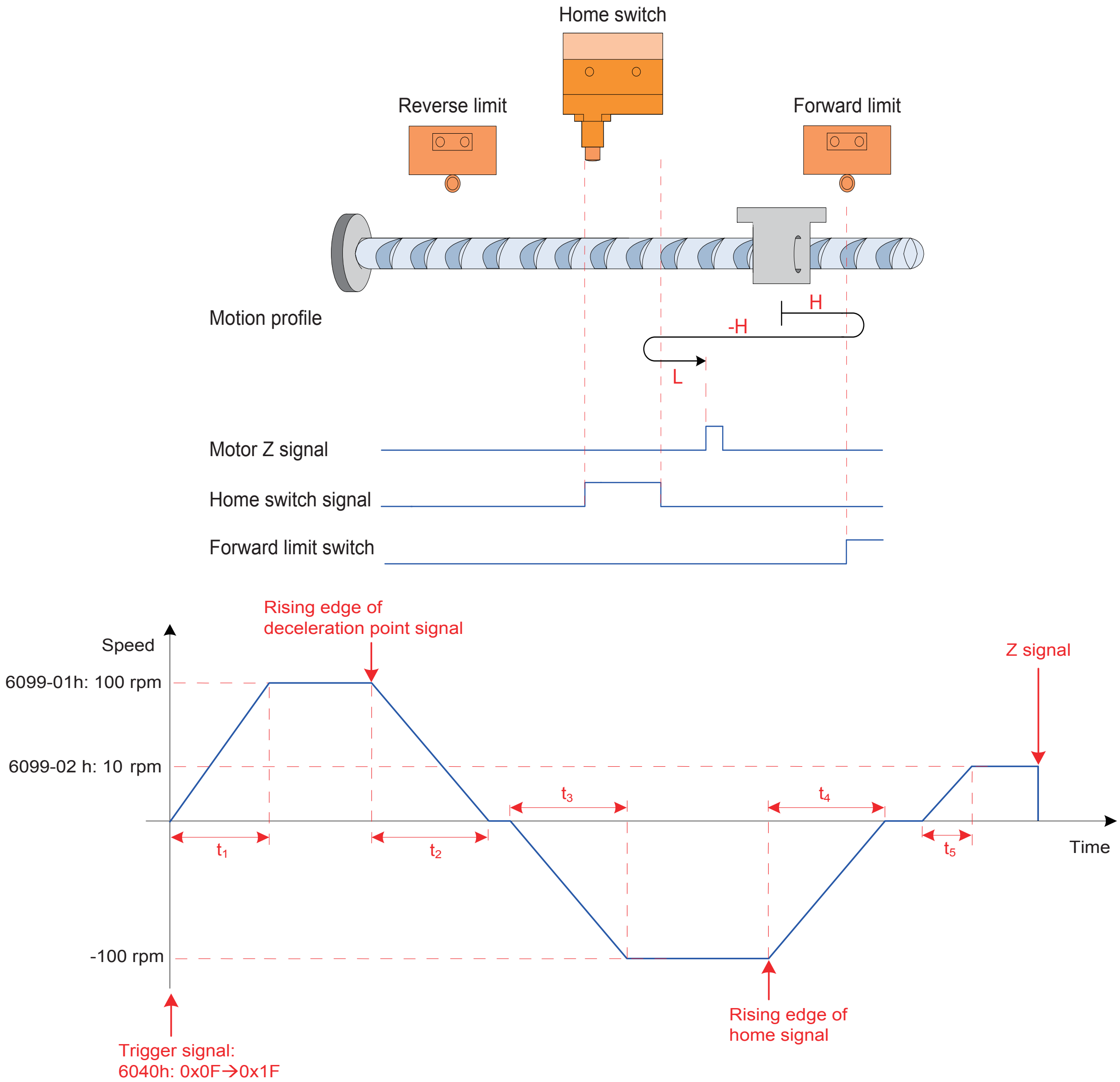


$t_1 = \frac{6099\text{-}01h}{609Ah}$ ms $t_2 = \frac{[6099\text{-}01h] - [6099\text{-}02h]}{609Ah}$ ms

The HW signal is 0 initially and the motor starts homing in the forward direction at a high speed. If the motor does not reach the limit switch, the motor decelerates and then runs in the forward direction at a low speed after reaching the rising edge of the HW signal. After reaching the falling edge of the HW signal, the motor continues to run in the forward direction at a low speed. Later, the motor stops at the first motor Z signal.

b) Invalid deceleration point signal at start of homing, reaching the forward limit switch

Figure 4-38 Mode ② in which 6098h is 10, the deceleration point signal is invalid, and the forward limit switch is reached



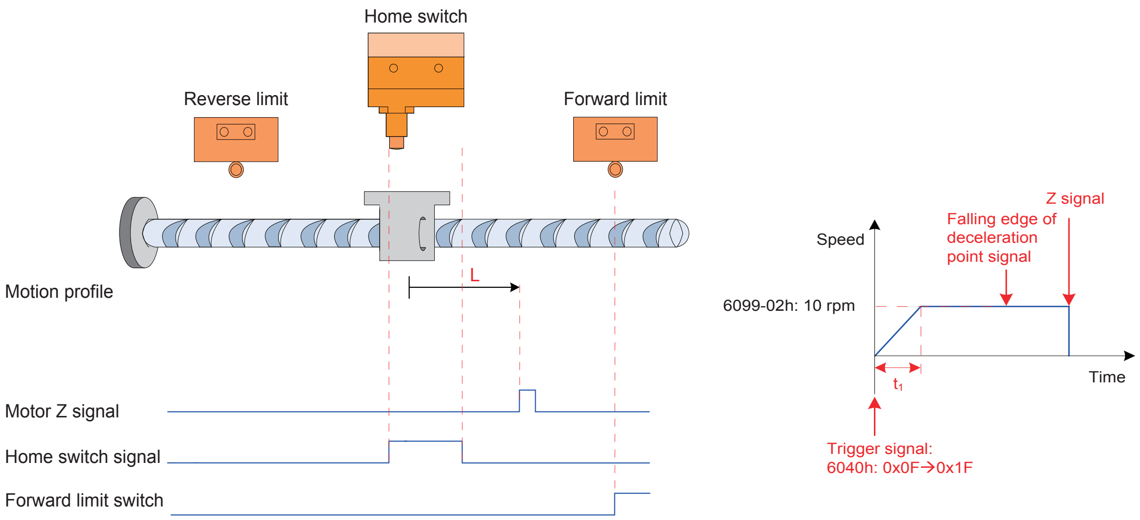
$t_1 = \frac{6099\text{-}01h}{609Ah}$ ms $t_2 = \frac{6099\text{-}01h}{609Ah}$ ms $t_3 = \frac{6099\text{-}01h}{609Ah}$ ms

$t_4 = \frac{6099\text{-}01h}{609Ah}$ ms $t_5 = \frac{6099\text{-}02h}{609Ah}$ ms

The HW signal is 0 initially and the motor starts homing in the forward direction at a high speed. If the motor reaches the limit switch, the motor automatically runs in the reverse direction at a high speed. After reaching the rising edge of the HW signal, the motor decelerates and continues to run 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.

c) Valid deceleration point signal at start of homing

Figure 4-39 Mode ③ in which 6098h is 10 and the deceleration point signal is valid



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

The HW signal is 1 initially and the motor directly 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 profile of 6098h = 7 to 10, opposite in the initial running direction

12) 6098h = 17 to 30

Same profile as that of 6098 = 1 to 14, without the step of searching for motor Z signal. The motor stops immediately at the following home signal.

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

4 Motion Mode

13) 6098h = 31 to 32

This mode is not defined in CiA402. It can be used for extension.

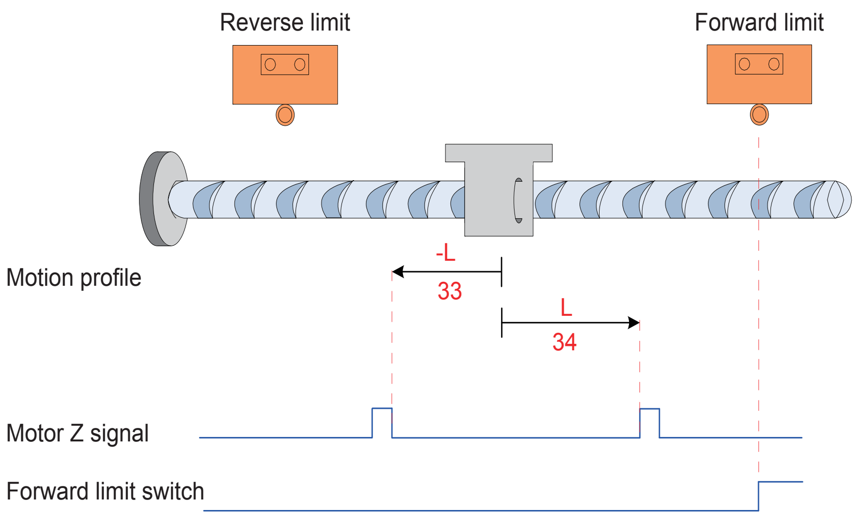
14) 6098h = 33 and 34

- Home: Z signal
- Deceleration point: None

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

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

Figure 4-40 6098h=33 or 34



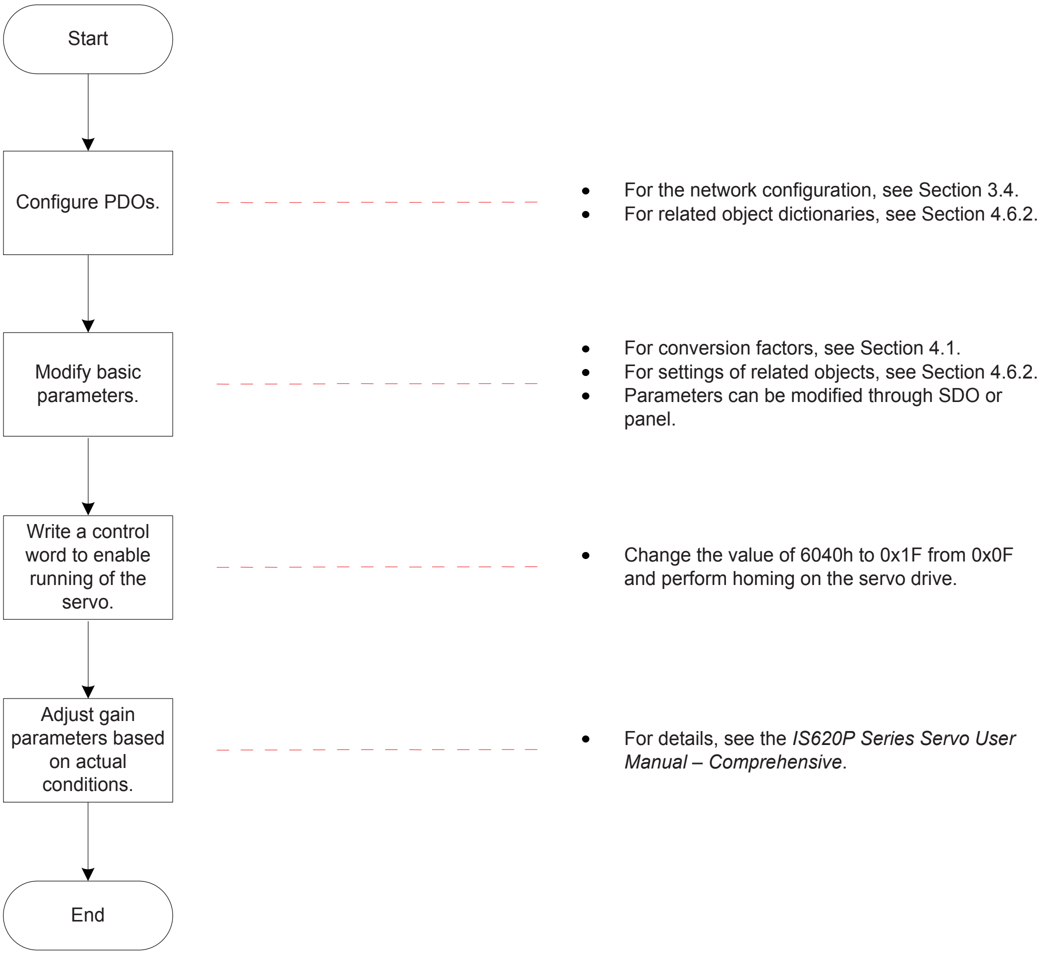
15) 6098h = 35

The homing mode is 35 and homing is triggered with the current position as the mechanical home (control word 6040h: 0x0F → 0x1F).

Position actual value 6064h = 607C

4.6.5 Configuration Example

Figure 4- Example of configuration flowchart in homing mode



Functional Code	Object	Mapping Object	Input Content	Description
H18-34	1600h-00h	Number of RPDO1 mapping objects	2	
H18-35	1600h-01h	6040h-00h	60400010h	The first mapping parameter of RPDO1 is 6040-00h. The parameter is 16 bits long.
H18-37	1600h-02h	6060h-00h	60600008h	The second mapping parameter of RPDO1 is 6060-00h. The parameter is 8 bits long.
H19-02	1601h-00h	Number of RPDO2 mapping objects	2	
H19-03	1601h-01h	6098h-00h	60980008h	The first mapping parameter of RPDO2 is 6098-00h. The parameter is 8 bits long.
H19-05	1601h-02h	609Ah-00h	609A0020h	The first mapping parameter of RPDO2 is 609A-00h. The parameter is 32 bits long.
H19-19	1602h-00h	Number of RPDO3 mapping objects	2	
H19-20	1602h-01h	6099h-01h	60990120h	The first mapping parameter of RPDO3 is 6099-01h. The parameter is 32 bits long.

Functional Code	Object	Mapping Object	Input Content	Description
H19-21	1602h-02h	6099h-02h	60990220h	The second mapping parameter of RPDO3 is 6099-02h. The parameter is 32 bits long.
H1A-24	1A00h-00h	Number of TPDO1 mapping objects	2	
H1A-25	1A00h-01h	6041h-00h	60410010h	The first mapping parameter of TPDO1 is 6041-00h. The parameter is 16 bits long.
H1A-27	1A00h-02h	6061h-00h	60610008h	The second mapping parameter of TPDO1 is 6061-00h. The parameter is 8 bits long.
H1A-41	1A01h-00h	Number of TPDO2 mapping objects	2	
H1A-42	1A01h-01h	6064h-00h	60640020h	The first mapping parameter of TPDO2 is 6064-00h. The parameter is 32 bits long.
H1A-44	1A01h-02h	606Ch-00h	606C0020h	The second mapping parameter of TPDO2 is 606C-00h. The parameter is 32 bits long.

- Set the drive mode 6060h to 0x06 to make the drive run in homing mode.
- Set the homing mode 6098h (user unit, default value: 0 p).
- Set the speed during search for switch 6099-01h (user unit, default value: 100 rpm) and speed during search for zero 6099-02h (user unit, default value: 10 rpm).
- Set the homing acceleration 609A-00h (user unit, default value: 100 rpm/ms).
- Set the control word 6040h to 0x1F from 0x0F so that the drive performs the homing operation.

Monitoring parameters:

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

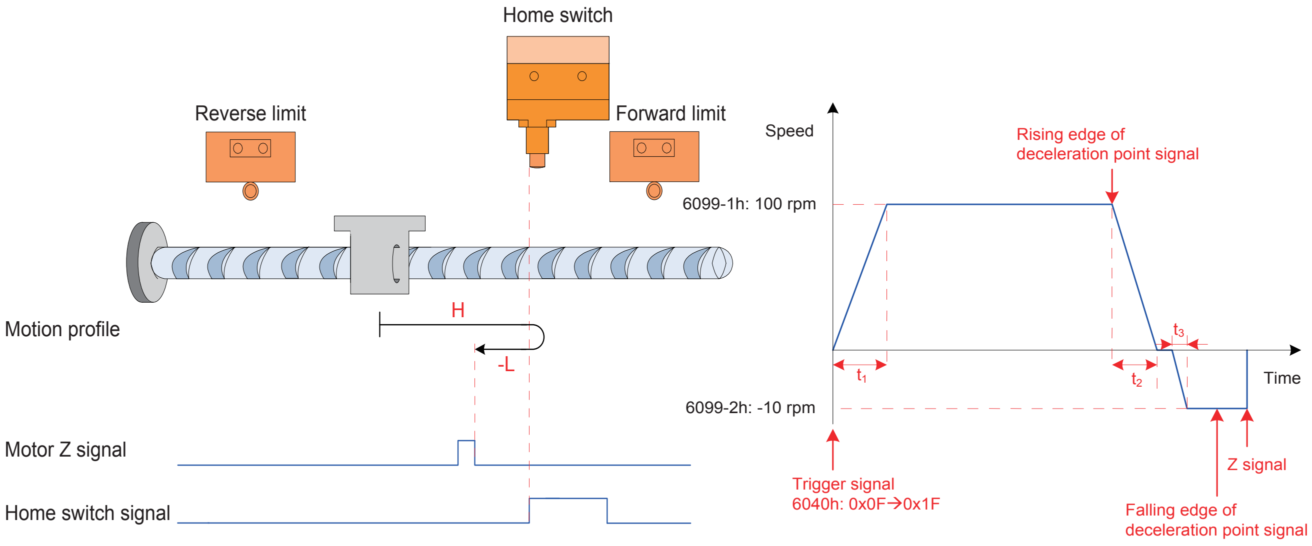
Example:

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

- Speed during search for switch: 6099-1h = 100 rpm
- Speed during search for zero: 6099-2h = 10 rpm
- Homing acceleration: 609Ah = 100 rpm/ms

SN	Control Command 6040h	Status of 6041h	Description
1	0x06	0x0231	Servo ready
2	0x07	0x0233	Ready, wait to switch on
3	0x0F	0x0637	Homing not started, target position reached
4	0x1F	0x9637	Homing completed, target position reached

Figure 4-41 Description of case in which 6060h is 0x06 and 6098h is 3

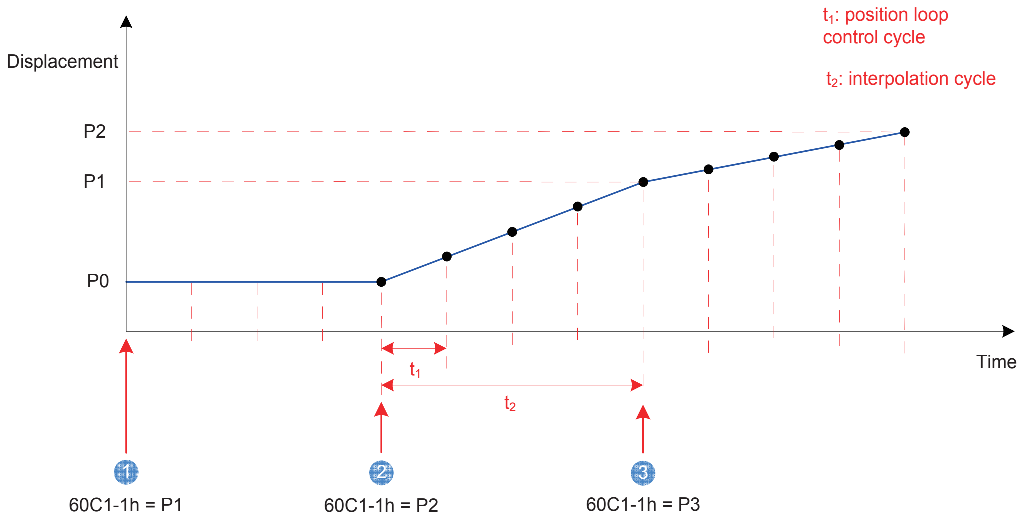


$t_1 = \frac{(6099\text{-}1h)}{609Ah}$ ms $\quad t_2 = \frac{(6099\text{-}1h)}{609Ah}$ ms $\quad t_3 = \frac{(6099\text{-}2h)}{609Ah}$ ms

4.7 Interpolated Position Mode

The interpolated position mode can implement synchronization of a multi-shaft servo drive or single-shaft servo drive. When the servo drive is not enabled, after the host computer sets the interpolated position mode, the displacement profile is planned in advance according to actual application requirements. When the servo drive is running, different absolute position points on the displacement profile are periodically sent to the slave node. The slave node synchronously receives the displacement reference, subdivides and evenly sends the displacement reference increment based on the position loop control cycle. The servo drive completes control over the position, speed, and torque inside.

Figure 4-42 Displacement profile of single-shaft linear interpolation motor



NOTE

Note: 1: ① The current absolute position of the servo motor is P0. After receiving the first absolute position reference P1, the servo motor starts to plan the first displacement profile.

Note: 2: ② The current absolute position of the servo motor is P0. The servo motor starts to move towards the first absolute position P1. After receiving the second absolute position reference P2, the servo motor starts to plan the second displacement profile.

Note: 3: ③ The servo motor reaches the first absolute position P1 and starts to move towards the second absolute position P2. After receiving the third absolute position reference P3, the servo motor starts to plan the third displacement profile.

Note 4: t_1 - position loop control cycle, which is determined by the servo drive internally.

Note 5: t_2 - interpolation cycle, which is set in the object dictionary 60C2h. IS620P supports the synchronization cycle in the range 1 ms to 20 ms. When a synchronization cycle beyond the range is set, the synchronization cycle is set to a limited value.

Note 6: P0/P1/P2 - absolute position. An absolute position reference is sent through 60C1-1h. The interpolated position mode supports only absolute position references.

Note 7: The displacement reference increments in each synchronization cycle are P1-P0 and P2-P1.

4.7.1 Control Block Diagram

Figure 4-43 Control block diagram of the interpolated position mode

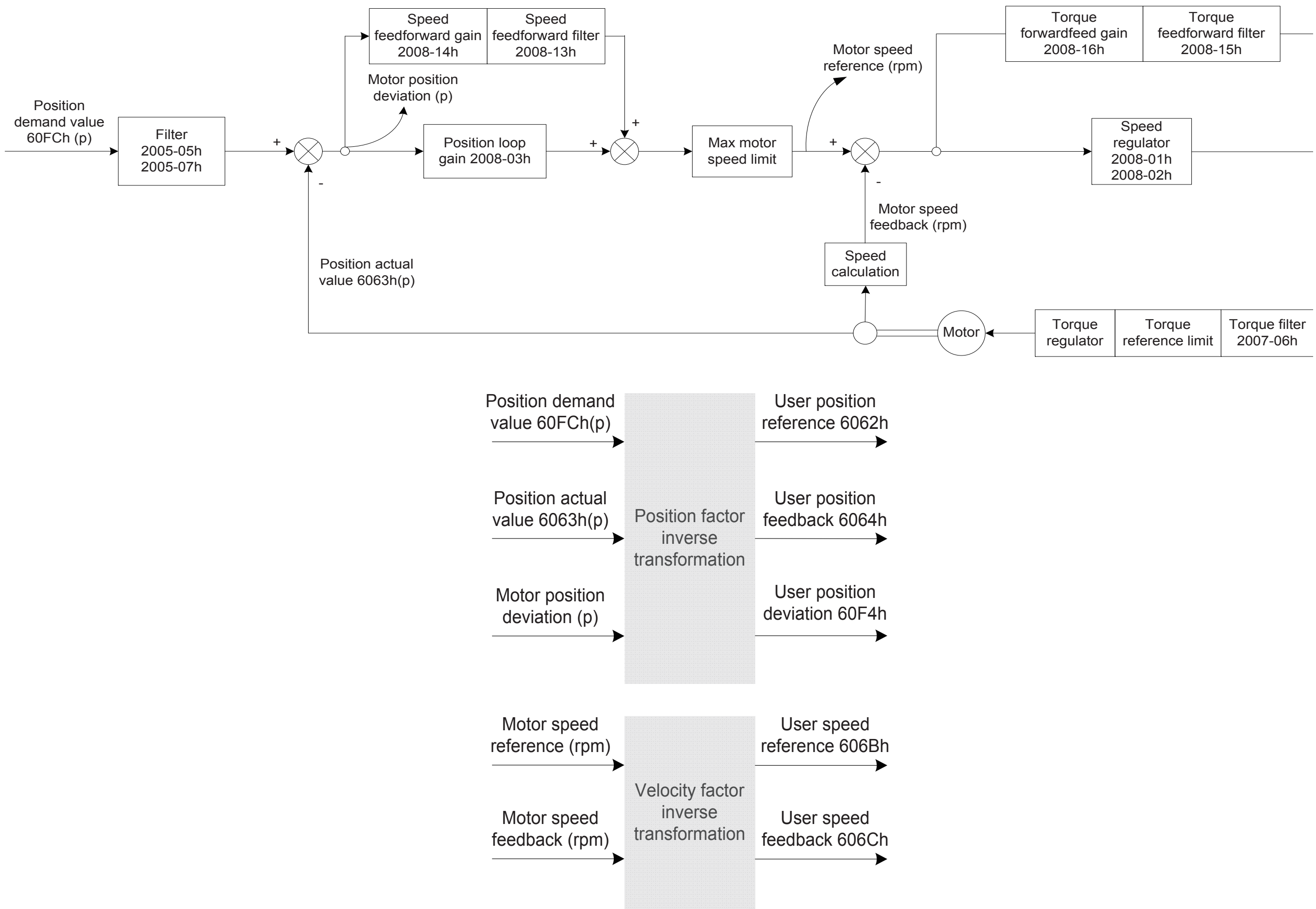
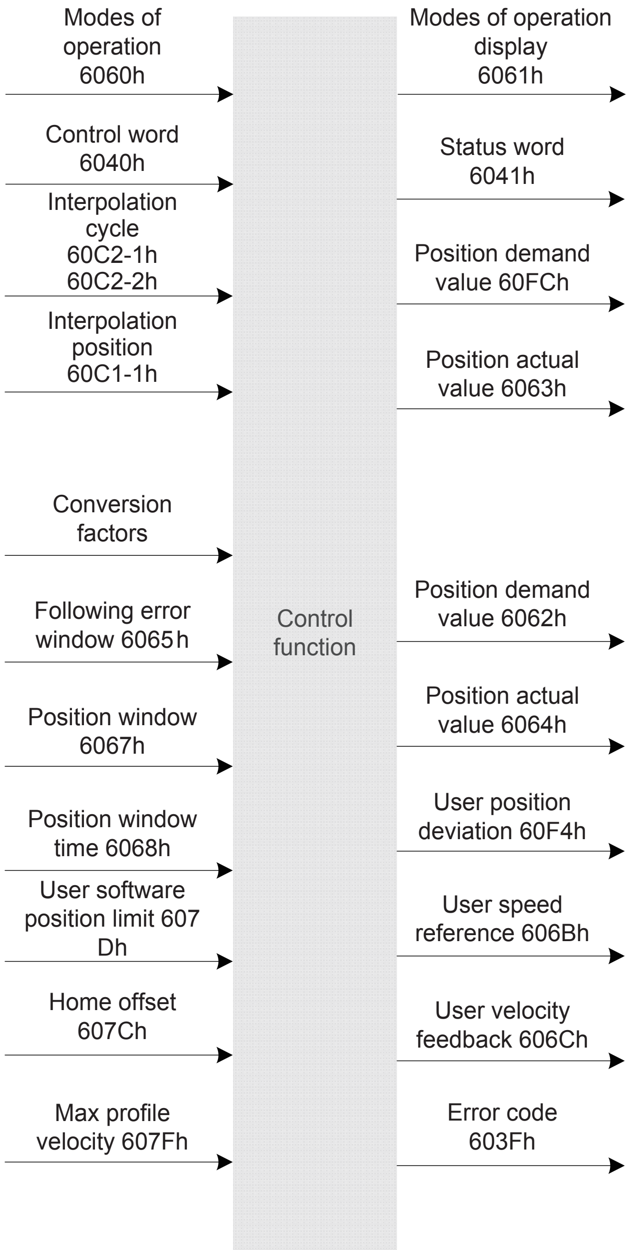
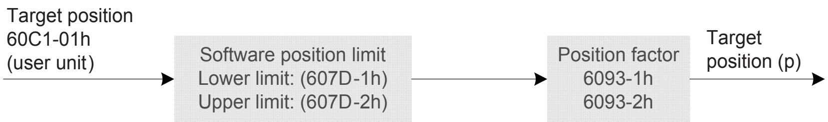


Figure 4-44 Input/output block diagram in interpolated position mode



By setting 0x200A-02h, you can check the absolute position limit of the user position reference and position feedback. By default, 200A-02h is 2, that is, after homing is complete and the reference zero position of mechanical operation is known, software position limit check is performed for the target position 60C1h and user position feedback 6064h. When the position reference exceeds the internal software position limit, bit11 of the status word 6041h is set to 1 and the drive runs by using the limit as the target position. After reaching the target position, the drive stops and provides a prompt. A reverse reference can make the drive exit the limit status and clear bit11 of 6041h. When external DI limit switch and internal software position limit are valid at the same time, the limit status is determined by the external DI limit switch.

Figure 4-45 Interpolation displacement 60C1h - internal software position limit



4.7.2 Relevant Object Setting

1) Positioning complete

Index 6067h	Name	Position Window				Data Structure	VAR	Data Type	Uint32
	Accessibility	RW	Mapping	YES	Relevant Mode pp/hm/ip	Data Range	0 to 4294967295	Factory Default	734 p

Sub-index: 00

When the position deviation 60F4h of the user unit is smaller than 6068h and time reaches this value, bit10 of 6041h is 1.

When either condition is not met, the position window is invalid.

Index 6068h	Name	Position Window Time				Data Structure	VAR	Data Type	Uint16
	Accessibility	RW	Mapping	YES	Relevant Mode pp/hm/ip	Data Range	0 to 65535	Factory Default	0 ms

Sub-index: 00

When the position deviation 60F4h of the user unit is smaller than 6068h and time reaches this value, bit10 of 6041h is 1.

When either condition is not met, the position window is invalid.

2) Detection for Following Error Window

Index 6065h	Name	Following Error Window				Data Structure	VAR	Data Type	Uint32
	Accessibility	RW	Mapping	YES	Relevant Mode pp/hm/ip	Data Range	0 to 4294967295	Factory Default	3145728 p

Sub-index: 00

When the position deviation is larger than this value, Er.B00 occurs and bit13 of the status word 6041h is set to 1.

Index	Sub-index	Name	Accessibility	Mapping	Data Type	Unit	Data Range	Factory Default
603Fh	00h	Error code	RO	TPDO	Uint16	-	0–65535	-
6040h	00h	Control word	RW	YES	Uint16	-	0–65535	0
6041h	00h	Status word	RO	TPDO	Uint16	-	0–65535	-
6060h	00h	Operation selection	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	User unit	-2^{31} to $(2^{31}-1)$	-
6063h	00h	Position actual internal value	RO	TPDO	Int32	Encoder unit	-2^{31} to $(2^{31}-1)$	-
6064h	00h	Position actual value	RO	TPDO	Int32	User unit	-2^{31} to $(2^{31}-1)$	-
6065h	00h	Following error window	RW	YES	Uint32	User unit	0 to $(2^{32}-1)$	3145728
6067h	00h	Position window	RW	YES	Uint32	User unit	0 to $(2^{32}-1)$	734

Index	Sub-index	Name	Accessibility	Mapping	Data Type	Unit	Data Range	Factory Default
6068h	00h	Position window time	RW	YES	Uint16	ms	0 to 65535	0
606Bh	00h	Velocity demand value	RO	TPDO	Int32	rpm	-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	User unit	-2^{31} to $(2^{31}-1)$	-2^{31}
	02h	Max Software Position Limit	RW	YES	Int32	User unit	-2^{31} to $(2^{31}-1)$	$2^{31}-1$
607Ch	00h	Home offset	RW	YES	Int32	User unit	-2^{31} to $(2^{31}-1)$	0
6098h	00h	Homing mode	RW	YES	Int8	-	0 to 35	0
6099h	01h	Speed during search for switch	RW	YES	Uint32	rpm	0 to $(2^{32}-1)$	100
	02h	Speed during search for zero	RW	YES	Int32	rpm	0 to $(2^{32}-1)$	10
609Ah	00h	Homing acceleration	RW	YES	Uint32	rpm/ms	0 to $(2^{32}-1)$	100
60C1h	01h	Interpolation target position	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	User 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

4.7.3 Control Commands in Interpolated Position Mode

The control word 6040h in interpolated position mode is described as follows:

Index 6040h	Name	Control Word				Data Structure	VAR	Data Type	Uint16
	Accessibility	RW	Mapping	YES	Relevant Mode All	Data Range	0 to 65535	Factory Default	-

It sets the control commands in homing mode.

Control Word 6040h				
Bit	bit7-15	bit5-bit6	Bit 4	bit0-bit3
Name	-	N/A	Enable IP mode	-
Value	See Table 4-2 Relationship between status switching and control commands.	-	-	See Table 4-2 Relationship between status switching and control commands.
Description	For details, see "6.5.3 Details of Parameters Defined by Sub-protocols" on Page 182.	-	0: Interrupt interpolation. 1: Enable interpolation. Bit4 must always be 1 during interpolation. Bit12 of 6041h can be used to determine whether the IP mode is activated.	For details, see "6.5.3 Details of Parameters Defined by Sub-protocols" on Page 182.

The status word 6041h in interpolated position mode is described as follows:

Index 6041h	Name	Status Word				Data Structure	VAR	Data Type	Uint16
	Accessibility	RO	Mapping	TPDO	Relevant Mode	All	Data Range	0 to 65535	Factory Default	-

It indicates the status of the servo drive in interpolated position mode.

State word 6041h							
Bit	bit15	bit14	bit13	bit12	bit11	bit10	bit0-bit9
Name	Homing completed	NA	Not used	IP mode activated	Software internal setting exceeding limit	Target reached	-
Value	-	-	-	-	-	-	For details, see Table 4-2. Table 4-2 Relationship between status switching and control commands
Description	0: Homing is not performed or homing is not complete. 1: Homing is complete and the reference point is found.	-	-	0: Interpolation is not activated. 1: Interpolation is activated.	0: The actual position value does not reach the software position limit. 1: The actual position value reaches the software position limit. *1	0: The target position is not reached. 1: The target position is reached. *2	For details, see "6.5.3 Details of Parameters Defined by Sub-protocols" on Page 182.

NOTE

Note 1: *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 Details of Parameters Defined by Sub-protocols" on Page 182.

Note 2: *2: When the position deviation is within the position window 6067h and the time reaches the value set by 6068h, the target position is reached. If either condition is not met, the target position is not reached.

4.7.4 Configuration Example

Functional Code	Object	Mapping Object	Input Content	Description
H18-34	1600h-00h	Number of RPDO1 mapping objects	2	
H18-35	1600h-01h	6040h-00h	60400010h	The first mapping parameter of RPDO1 is 6040-00h. The parameter is 16 bits long.
H18-37	1600h-02h	6060h-00h	60600008h	The second mapping parameter of RPDO1 is 6060-00h. The parameter is 8 bits long.
H19-02	1601h-00h	Number of RPDO2 mapping objects	1	
H19-03	1601h-01h	60C1h-01h	60C10020h	The first mapping parameter of RPDO2 is 60C1-00h. The parameter is 32 bits long.
H19-05	1601h-01h	-	0	-
H1A-24	1A00h-00h	Number of TPDO1 mapping objects	2	
H1A-25	1A00h-01h	6041h-00h	60410010h	The first mapping parameter of TPDO1 is 6041-00h. The parameter is 16 bits long.
H1A-27	1A00h-02h	6061h-00h	60610008h	The first mapping parameter of TPDO1 is 6061-00h. The parameter is 8 bits long.
H1A-41	1A01h-00h	Number of TPDO2 mapping objects	2	
H1A-42	1A01h-01h	6064h-00h	60640020h	The first mapping parameter of TPDO2 is 6064-00h. The parameter is 32 bits long.
H1A-44	1A01h-02h	606Ch-00h	606C0020h	The second mapping parameter of TPDO2 is 606C-00h. The parameter is 32 bits long.

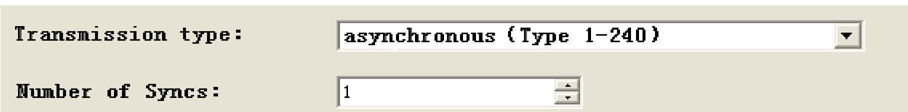
Example:

When 6060h = 0x07:

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

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

Figure 4-46 Configuration example of 60C1-01h



- Set the drive mode 6060h to 0x07 to make the drive run in interpolated position mode.
- Set the interpolation position 60C1-1h (only absolute position references are supported), interpolation time constant 60C2-1h, and interpolation time index 60C2-2h (the default value is 3 ms and can be modified to 10 ms). The synchronization cycle must be set to 1 to 20 ms.
- Set the control word 6040h to 0x1F from 0x0F so that the drive can run. An example of the specific configuration is as follows:

SN	Control Command 6040h	Status of 6041h	Description
1	0x06	0x0231	No fault → Ready
2	0x07	0x0233	Running → Wait to enable servo
3	0x0F	0x0637	The target position is reached.
4	0x0F	0x0A37	The target position is not reached and the position reference exceeds the limit.
5	0x0F	0x0E37	The target position is reached and the position reference exceeds the limit.
6	0x1F	0x1237	The IP mode is activated and the target position is not reached.
7	0x1F	0x1637	The IP mode is activated and the target position is reached.
8	0x1F	0x1A37	The IP mode is activated, the target position is not reached, and the position reference exceeds the limit.
9	0x1F	0x1E37	The IP mode is activated, the target position is reached, and the position reference exceeds the limit.

Monitoring parameters:

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

4.8 Profile Velocity Mode

In profile velocity mode, after the user sets the speed, acceleration, and deceleration, the servo drive can plan the velocity profile based on the setting and implement smooth transition between different velocity references.

4.8.1 Control Block Diagram

Figure 4-47 Control block diagram of the profile velocity mode

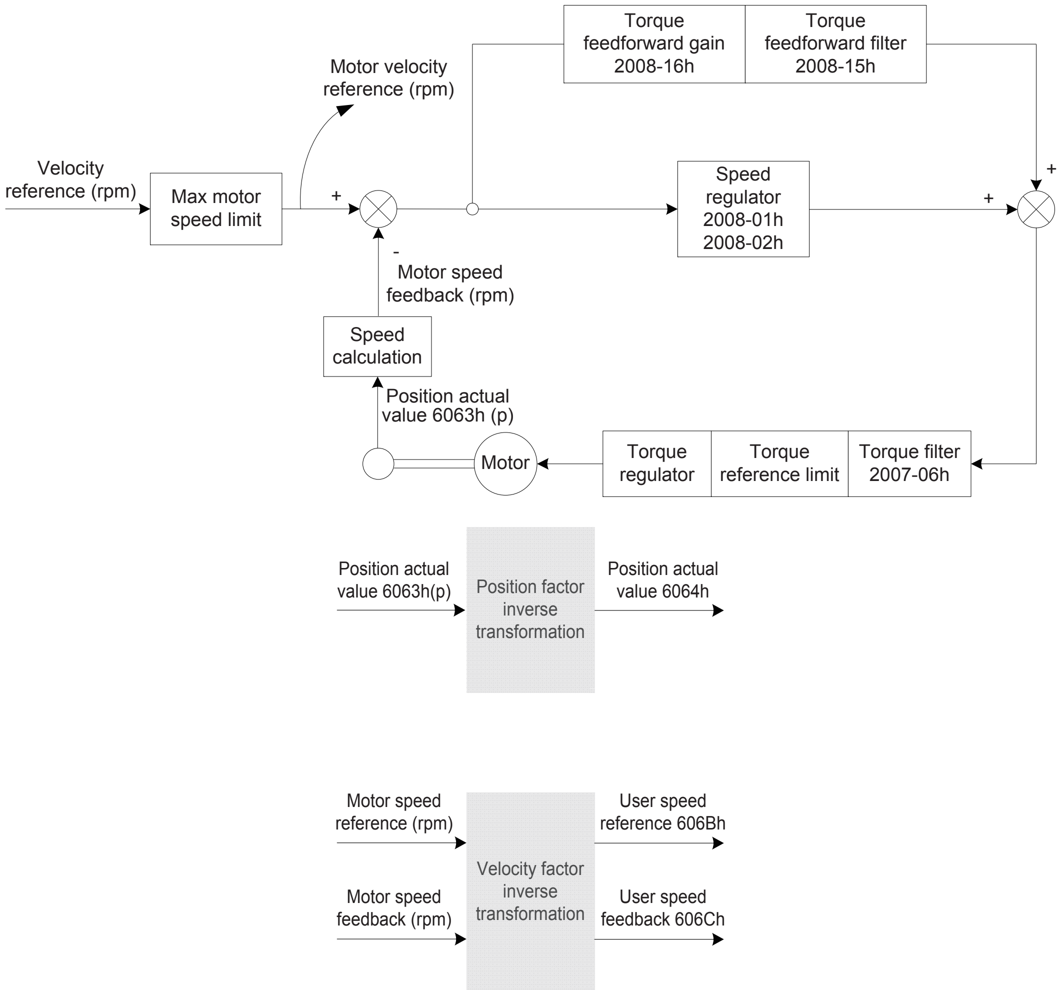
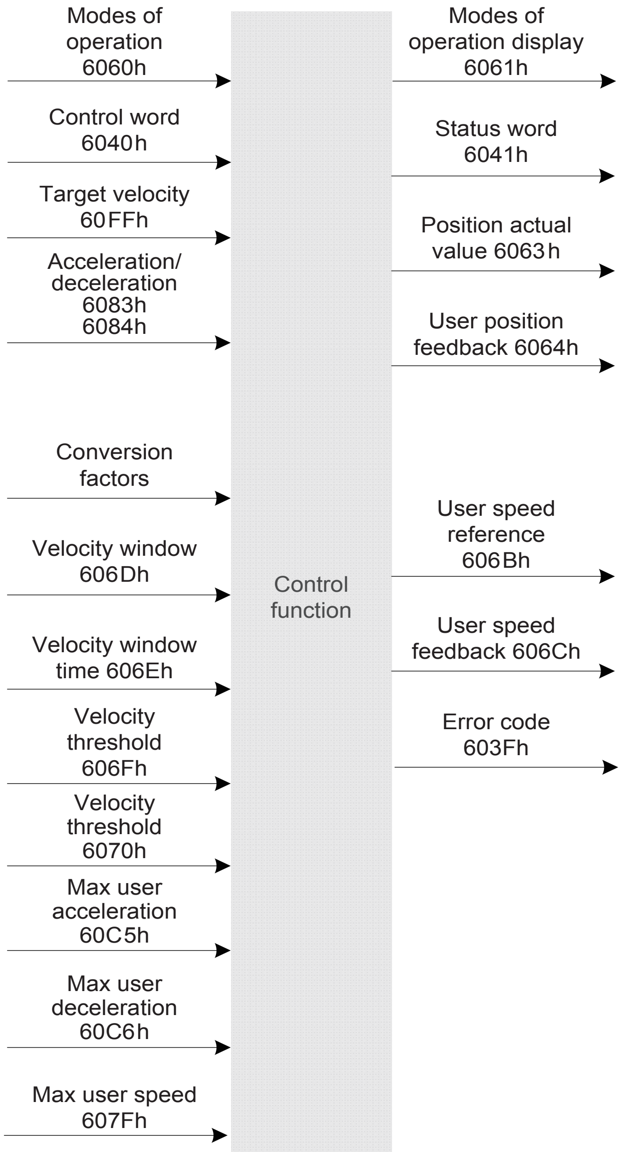


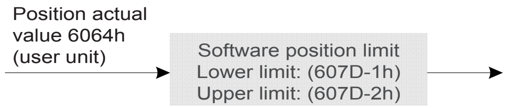
Figure 4-48 Input/output block diagram in profile velocity mode



Velocity profile planning involves the target velocity 60FFh (in user unit), profile acceleration 6083h (in user unit), and profile deceleration 6084h (in user unit). References of the host computer are entered in user units and are called references in the drive unit after they go through limiting and conversion. Figure 4-49, Figure 4-50, and Figure 4-51 show processing of the drive for the target velocity, profile acceleration, and profile deceleration.

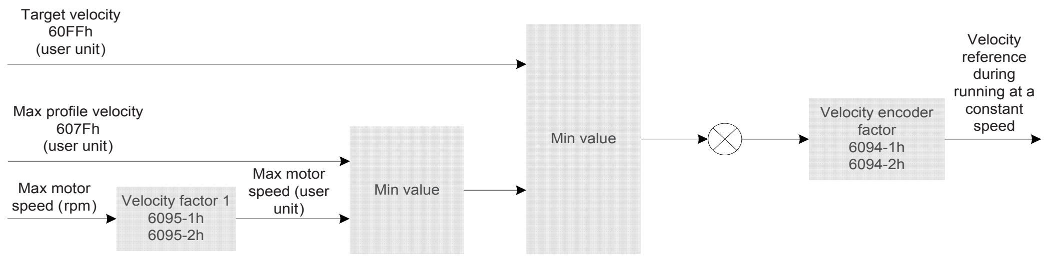
By setting 0x200A-02h, you can check the absolute position limit of the user position reference and position feedback. By default, 200A-02h is 2, that is, after homing is complete and the reference zero position of mechanical operation is known, software position limit check is performed for the user position feedback 6064h. When the user position feedback exceeds the software position limit, the motor stops and a limit fault occurs. When external DI limit switch and internal software position limit are valid at the same time, the limit status is determined by the external DI limit switch.

Figure 4-49 User position feedback 6064h - internal software position check



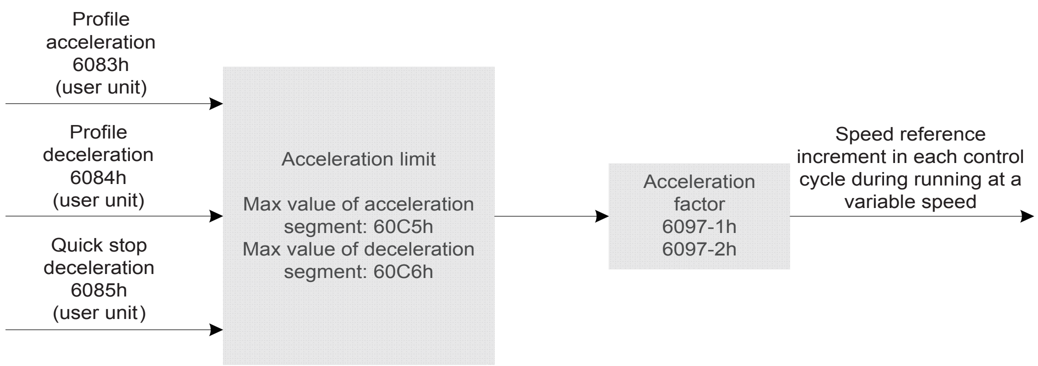
The target velocity 60FFh is used to set the maximum speed during running of the velocity reference. It cannot exceed the maximum velocity 607Fh set by the user and the maximum motor speed after conversion. Figure 4-50 shows the block diagram.

Figure 4-50 Target velocity 60FFh - velocity limit



Profile acceleration 6083h and profile deceleration 6084h are used to set acceleration and deceleration during running of the velocity reference. The values cannot exceed the maximum acceleration 60C5h and deceleration 60C6h set by the user. Figure 4-51 shows the block diagram.

Figure 4-51 Profile acceleration and deceleration limit



4 Motion Mode

4.8.2 Relevant Object Setting

1) Zero speed clamp

Index 2006h	Name	Speed Threshold for Zero Clamp				Data Structure	VAR	Data Type	Uint16
	Accessibility	RW	Mapping	YES	Relevant Mode pv	Data Range	0 to 6000	Factory Default	10 rpm

Sub-index: 10h

When the actual velocity is smaller than the value and the corresponding DI function 12 is enabled, the motor enters the position locked status.

2) Zero speed threshold

Index 606Fh	Name	Velocity Threshold				Data Structure	VAR	Data Type	Uint16
	Accessibility	RW	Mapping	YES	Relevant Mode pv	Data Range	0 to 65535	Factory Default	10 rpm

Sub-index: 00h

When the velocity feedback 606Ch of the user unit is smaller than 6070h and time reaches this value, bit12 of 6041h is 1.

When either condition is not met, the zero speed threshold is invalid.

Index 6070h	Name	Velocity Threshold Time				Data Structure	VAR	Data Type	Uint16
	Accessibility	RW	Mapping	YES	Relevant Mode pv	Data Range	0 to 65535	Factory Default	0 ms

Sub-index: 00h

When the velocity feedback 606Ch of the user unit is smaller than 6070h and time reaches this value, bit12 of 6041h is 1.

When either condition is not met, the zero speed threshold is invalid.

3) Velocity threshold

Index 606Dh	Name	Velocity Window				Data Structure	VAR	Data Type	Uint16
	Accessibility	RW	Mapping	YES	Relevant Mode pv	Data Range	0 to 65535	Factory Default	10 rpm

Sub-index: 00h

When the deviation of the velocity feedback 606Ch of the user unit from the target velocity 60FFh is smaller than 6070Eh and time reaches this value, bit10 of 6041h is 1. When either condition is not met, the velocity window is invalid.

Index 606Eh	Name	Velocity Window Time				Data Structure	VAR	Data Type	Uint16
	Accessibility	RW	Mapping	YES	Relevant Mode pv	Data Range	0 to 65535	Factory Default	0 ms

Sub-index: 00h

When the deviation of the velocity feedback 606Ch of the user unit from the target velocity 60FFh is smaller than 6070Eh and time reaches this value, bit10 of 6041h is 1. When either condition is not met, the velocity window is invalid.

Index	Sub-index	Name	Accessibility	Mapping	Data Type	Unit	Data Range	Factory Default
603Fh	00h	Error code	RO	TPDO	Uint16	-	0–65535	-
6040h	00h	Control word	RW	YES	Uint16	-	0–65535	0
6041h	00h	Status word	RO	TPDO	Uint16	-	0–65535	-
6060h	00h	Modes of operation	RW	YES	Int8	-	0 to 7	0
6061h	00h	Modes of operation display	RO	TPDO	Int8	-	0 to 7	-
6063h	00h	Position actual internal value	RO	TPDO	Int32	Encoder unit	-2^{31} to $(2^{31}-1)$	-
6064h	00h	Position actual value	RO	TPDO	Int32	User unit	-2^{31} to $(2^{31}-1)$	-
606Bh	00h	Velocity demand value	RO	TPDO	Int32	rpm	-2^{31} to $(2^{31}-1)$	-
606Ch	00h	Velocity actual value	RO	TPDO	Int32	rpm	-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 Window Time	RW	YES	Uint16	ms	0 to 65535	0
607Ch	00h	Home offset	RW	YES	Int32	User unit	-2^{31} to $(2^{31}-1)$	0
607Dh	01h	Min Software Position Limit	RW	YES	Int32	User unit	-2^{31} to $(2^{31}-1)$	-2^{31}
	02h	Max Software Position Limit	RW	YES	Int32	User unit	-2^{31} to $(2^{31}-1)$	$2^{31}-1$
6083h	00h	Profile acceleration	RW	YES	Uint32	rpm/ms	0 to $(2^{32}-1)$	100
6084h	00h	Profile deceleration	RW	YES	Uint32	rpm/ms	0 to $(2^{32}-1)$	100
6094h	01h	Numerator	RW	YES	Uint32	-	0 to $(2^{32}-1)$	1048576
	02h	Denominator	RW	YES	Uint32	-	1 to $(2^{32}-1)$	60
60C5h	00h	Max Profile Acceleration	RW	YES	Uint32	rpm/ms	0 to $(2^{32}-1)$	1000
60C6h	00h	Max Profile Deceleration	RW	YES	Uint32	rpm/ms	0 to $(2^{32}-1)$	1000
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

4.8.3 Control Commands in Profile Velocity Mode

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

Index 6040h	Name	Control Word				Data Structure	VAR	Data Type	Uint16
	Accessibility	RW	Mapping	YES	Relevant Mode	All	Data Range	0 to 65535	Factory Default	-

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

6040	Description
0x06	Servo ready
0x07	Ready, wait to enable servo
0x0F	The servo is enabled and runs according to the provided profile.

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

Index 6041h	Name	Status Word				Data Structure	VAR	Data Type	Uint16
	Accessibility	RO	Mapping	TPDO	Relevant Mode	All	Data Range	0 to 65535	Factory Default	-

It indicates the status of the servo drive in profile velocity mode.

State word 6041h							
Bit	bit15	bit14	bit13	bit12	bit11	bit10	bit0-bit9
Name	Homing completed	NA	Not used	Zero speed signal	Software internal setting exceeding limit	Target reached	-
Set value	-	-	-	-	-	-	For details, see Table 4-2. Table 4-2 Relationship between status switching and control commands
Description	0: Homing is not performed or homing is not complete. 1: Homing is complete and the reference point is found.	-	-	0: User velocity is not 0. 1: User velocity is 0. *1	0: The actual position value does not reach the software position limit. 1: The actual position value reaches the software position limit.*2	0: The target velocity is not reached. 1: The target velocity is reached. *3	For details, see "6.5.3 Details of Parameters Defined by Sub-protocols" on Page 182.

NOTE

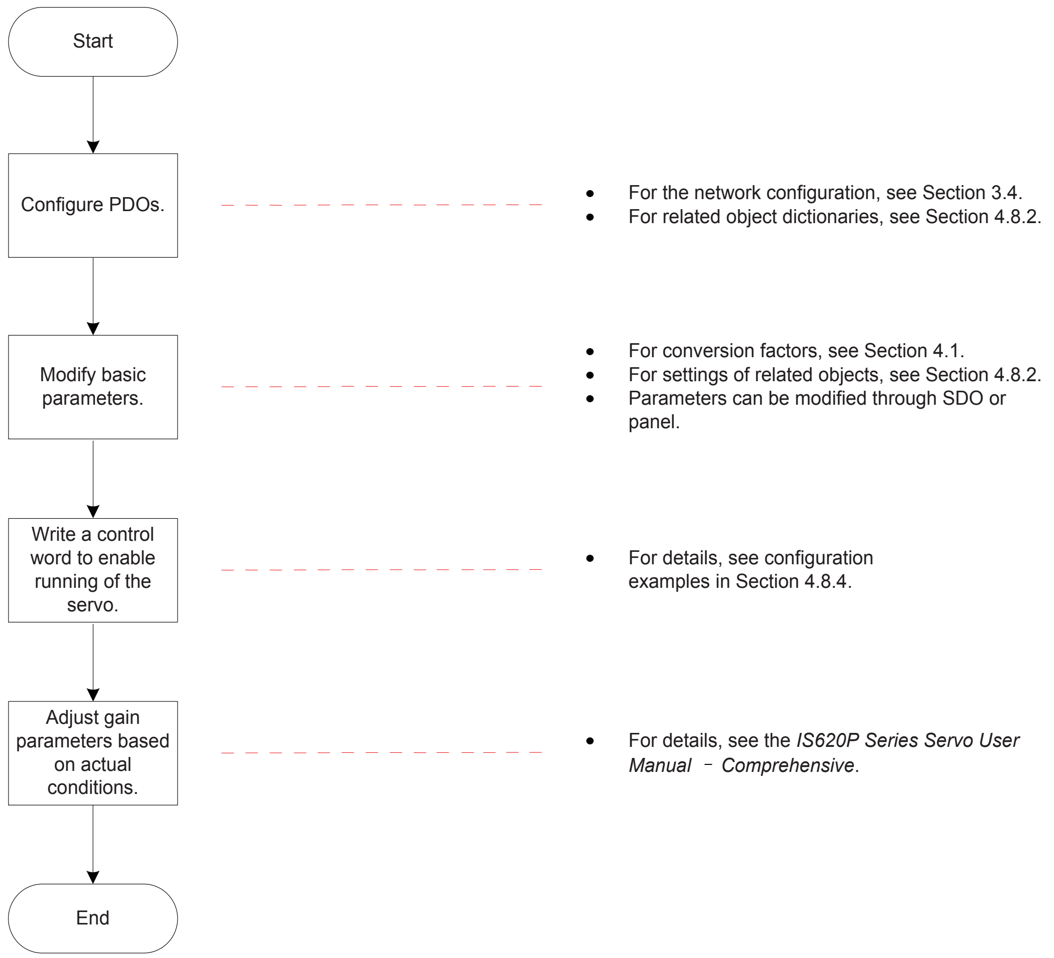
Note 1: *1 When the user velocity is within the velocity threshold (606Fh) and the time reaches the value set by 6070h, the user velocity is 0. When either condition is not met, the user velocity is considered not to be 0. This flag bit is valid only in profile velocity mode. This flag bit is unrelated to whether the servo drive is enabled.

Note 2. *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 Details of Parameters Defined by Sub-protocols" on Page 182.

Note 3: *3: When the target velocity is within the velocity window (606Dh) and the time reaches the value set by 606Eh, the target velocity is reached. If either condition is not met, the target velocity is not reached. This flag bit is valid only when the servo drive is enabled in profile velocity mode.

4.8.4 Configuration Example

Figure 4-52 Example of profile velocity mode configuration flowchart



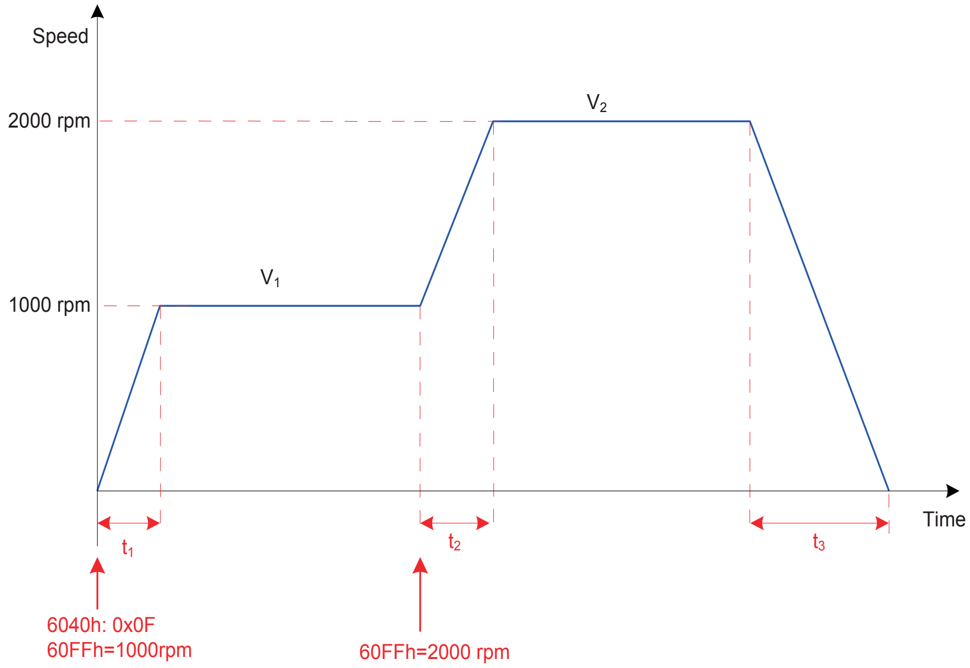
Functional Code	Object	Mapping Object	Input Content	Description
H18-34	1600h-00h	Number of RPDO1 mapping objects	2	
H18-35	1600h-01h	6040h-00h	60400010h	The first mapping parameter of RPDO1 is 6040-00h. The parameter is 16 bits long.
H18-37	1600h-02h	6060h-00h	60600008h	The second mapping parameter of RPDO1 is 6060-00h. The parameter is 8 bits long.
H19-02	1601h-00h	Number of RPDO2 mapping objects	1	
H19-03	1601h-01h	60FFh-00h	60FF0020h	The first mapping parameter of RPDO2 is 60FF-00h. The parameter is 32 bits long.
H19-05	1601h-01h	-	0	
H19-19	1602h-00h	Number of RPDO3 mapping objects	2	
H19-20	1602h-01h	6083h-00h	60830020h	The first mapping parameter of RPDO3 is 6083-00h. The parameter is 32 bits long.
H19-21	1602h-02h	6084h-00h	60840020h	The second mapping parameter of RPDO3 is 6084-00h. The parameter is 32 bits long.

Functional Code	Object	Mapping Object	Input Content	Description
H1A-24	1A00h-00h	Number of TPDO1 mapping objects	2	
H1A-25	1A00h-01h	6041h-00h	60410010h	The first mapping parameter of TPDO1 is 6041-00h. The parameter is 16 bits long.
H1A-27	1A00h-02h	6061h-00h	60610008h	The second mapping parameter of TPDO1 is 6061-00h. The parameter is 8 bits long.
H1A-41	1A01h-00h	Number of TPDO2 mapping objects	2	
H1A-42	1A01h-01h	6064h-00h	60640020h	The first mapping parameter of TPDO2 is 6064-00h. The parameter is 32 bits long.
H1A-44	1A01h-02h	606Ch-00h	606C0020h	The second mapping parameter of TPDO2 is 606C-00h. The parameter is 32 bits long.

- Set the drive mode 6060h to 0x03 to make the drive work in profile velocity mode.
- Set the target velocity ① 60FFh to 1000 rpm.
- Set profile acceleration ① 6083h to 100 rpm/ms.
- Set profile deceleration ① 6084h to 100 rpm/ms.
- Set the target velocity ② 60FFh to 1000 rpm.
- Set profile acceleration ② 6083h to 10 rpm/ms.
- Set profile deceleration ② 6084h to 10 rpm/ms.
- Set the control word 6040h and enable the drive to run. An example of the specific configuration is as follows:

SN	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 and 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 4-52 Motion profile of profile velocity



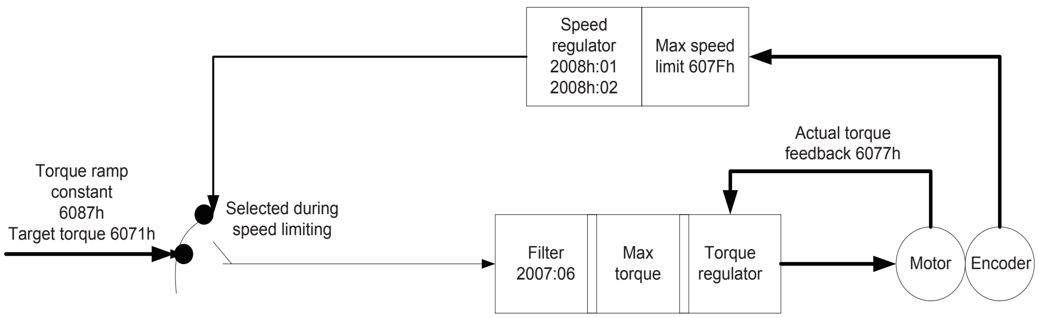
$$t_1 = \frac{V_1}{6083h} \text{ ms} \quad t_2 = \frac{V_2 - V_1}{6083h} \text{ ms} \quad t_3 = \frac{V_2}{6084h} \text{ ms}$$

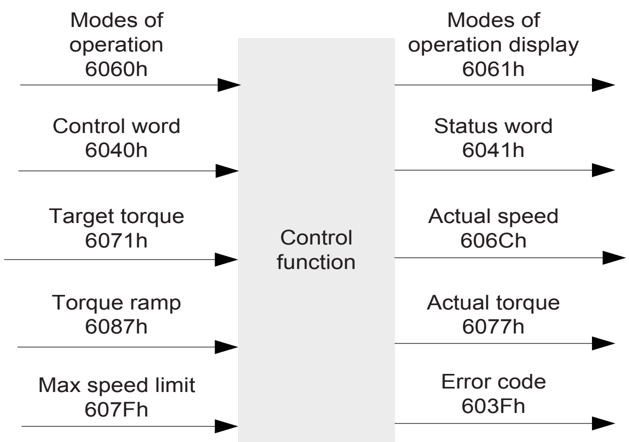
4.9 Profile Torque Mode

In this mode, the host computer sends the target torque (6071h) and torque ramp constant (6087h) to the servo drive. Torque control is performed by the servo drive. When the speed reaches the limit, the motor enters the speed adjustment status. However, the maximum output does not exceed the torque reference limit.

4.9.1 Control Block Diagram

Figure 4-53 Control block diagram of the profile torque mode





4.9.2 Relevant Object Setting

Control Word 6040h		
Bit	Name	Description
0		If bit0 to bit3 are 1, the servo drive starts running.
1		
2		
3		

State word 6041h		
Bit	Name	Description
10	Target reached	0: The target torque is not reached. 1: The target torque is reached.
12	Internal limit actice	0: Position feedback does not exceed the limit. 1: Position feedback exceeds the limit.
15	Home Find	0: Homing is not complete. 1: Homing is complete.

Index (hex)	Sub-index (hex)	Name	Accessibility	Size	Unit	Range	Default
603F	00	Error code	RO	UINT16	-	0-65535	0
6040	00	Control word	RW	UINT16	-	0-65535	0
6041	00	Status word	RO	UINT16	-	0-65535	0
6060	00	Operation Mode	RW	INT8	-	0-10	0
6061	00	Modes of operation display	RO	INT8	-	0-10	0
606C	00	Velocity actual value	RO	INT32	Reference unit: /s	-2^{31}-$(2^{31}$-1)	0
6071	00	Target torque	RW	INT16	0.1%	-5000 to 5000	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-$(2^{32}$-1)	0
6087	00	Torque ramp	RW	UINT32	0.1%/s	0-$(2^{32}$-1)	0
2007	06	Torque filter time	RW	UINT16	0.01 ms	0-65535	79
2008	01	Speed loop gain	RW	UINT16	0.1Hz	1-20000	250
	02	Speed loop integral time	RW	UINT16	0.01 ms	15-51200	3183

1) Torque reached signal setting

When the difference between the torque and the base 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 signal TOQREACH is invalid and bit10 of the status word 6041h bit10 is cleared.

Index (hex)	Sub-index (hex)	Name	Attribute	Size	Unit	Range	Default
2007	16	Base value for torque reached	RW	UINT16	0.1%	0-8000	0
2007	17	Valid value of torque reached	RW	UINT16	0.1%	0-8000	200
2007	18	Invalid value of torque reached	RW	UINT16	0.1%	0-8000	100

4.9.3 Speed Limit in Profile Torque Mode

The torque limit is selected based on 2007-12h.

<table>
<tr><th>Index (hex)</th><th>Sub-index (hex)</th><th>Name</th><th>Attribute</th><th>Size</th><th>Unit</th><th>Range</th><th>Default</th></tr>
<tr><td>2007</td><td>12</td><td>Speed limit source</td><td>RW</td><td>UINT16</td><td>1</td><td>0-4</td><td>0</td></tr>
<tr><td rowspan="2"></td><td colspan="3">Forward speed limit</td><td colspan="4">Reverse speed limit</td></tr>
<tr><td colspan="2">FunIn.36 ON</td><td>FunIn.36 OFF</td><td colspan="2">FunIn.36 ON</td><td colspan="2">FunIn.36 OFF</td></tr>
<tr><td>0</td><td colspan="3">2007-14h</td><td colspan="4">2007-15h</td></tr>
<tr><td>1</td><td colspan="3">min(2007-14h,AI)</td><td colspan="4">min(2007-15h,6AI)</td></tr>
<tr><td>2</td><td colspan="2">2007-15h</td><td>2007-14h</td><td colspan="2">2007-15h</td><td colspan="2">2007-14h</td></tr>
<tr><td rowspan="2">3</td><td colspan="3">If the initial value of 2007-12h is not 2, the minimum speed limit is (607F, 2007-14h).</td><td colspan="4">If the initial value of 2007-12h is not 2, the minimum speed limit is (607F, 2007-15h).</td></tr>
<tr><td colspan="3">If the initial value of 2007-12h is 2, the minimum speed limit is (607F, H07-17=2, corresponding to forward speed limit).</td><td colspan="4">If the initial value of 2007-12h is 2, the minimum speed limit is (607F, 2007-12h=2, corresponding to reverse speed limit).</td></tr>
</table>

Chapter 5 Troubleshooting

Chapter 5 Troubleshooting

When communication or the servo drive is abnormal, the IS620P servo drive sends an emergency packet to the network as a producer or sends an abort response packet when SDO transmission is abnormal. The following lists node errors and auxiliary information related to nodes.

5.1 CANopen Communication Fault Codes

Display	Fault Name	Reset	Error Code (603Fh)	Auxiliary Code (203Fh)
Er.200	Overcurrent 1	No	0x2311	0x02000200
Er.201	Overcurrent 2	No	0x2312	0x02010201
Er.210	Output short-circuit to ground	No	0x2330	0x02100210
Er.430	Control power undervoltage	No	0x3120	0x04300430
Er.420	Main circuit power cable phase loss	Yes	0x3130	0x04200420
Er.990	Input phase loss warning	Yes	0x3130	0x09900990
Er.400	Main circuit overvoltage	Yes	0x3210	0x04000400
Er.920	Braking resistor overload	Yes	0x3210	0x09200920
Er.410	Main circuit undervoltage	Yes	0x3220	0x04100410
Er.610	Drive overload	Yes	0x3230	0x06100610
Er.620	Motor overload	Yes	0x3230	0x06200620
Er.909	Motor overload warning	Yes	0x3230	0x09090909
Er.939	The motor power cables break.	Yes	0x3331	0x09390939
Er.650	Heatsink overheat	Yes	0x4210	0x06500650
Er.831	AI zero drift too large	Yes	0x5210	0x08310831
Er.834	AD sampling overvoltage	No	0x5210	0x08340834
Er.121	Invalid S-ON reference	Yes	0x5441	0x01210121
Er.900	DI emergency braking	Yes	0x5442	0x09000900
Er.950	Forward limit switch warning	Yes	0x5443	0x09500950
Er.952	Reverse limit switch warning	Yes	0x5444	0x09520952
Er.108	Parameter storage fault	No	0x5530	0x01080108
Er.101	Parameter abnormal	No	0x6320	0x01010101
Er.105	Internal program abnormal	No	0x6320	0x01050105
Er.111	H00/H01 group parameter abnormal	No	0x6320	0x01110111
Er.130	Same function allocated to different DIs	Yes	0x6320	0x01300130
Er.131	DO allocation exceeding limit	Yes	0x6320	0x01310131
Er.110	Setting error of frequency division pulse output	Yes	0x6320	0x01100110
Er.922	Resistance of external braking resistor too small	Yes	0x6320	0x09220922

Display	Fault Name	Reset	Error Code (603Fh)	Auxiliary Code (203Fh)
Er.941	Power-on required for parameter modification	Yes	0x6320	0x09410941
Er.b03	Electronic gear ratio setting exceeding limit	Yes	0x6320	0x0b030b03
Er.630	Motor rotor locked	Yes	0x7121	0x06300630
Er.120	Product model matching fault	No	0x7122	0x01200120
Er.136	Data check error or no parameter stored in the motor ROM	No	0x7305	0x01360136
Er.A33	Encoder data abnormal	No	0x7305	0x0A330A33
Er.A34	Encoder communication check abnormal	No	0x7305	0x0A340A34
Er.A35	Z signal lost	No	0x7305	0x0A350A35
Er.980	Encoder internal fault	Yes	0x7305	0x09800980
Er.740	Encoder interference	No	0x7305	0x07400740
Er.102	Programmable logic configuration fault	No	0x7500	0x01020102
Er.104	Programmable logic interruption fault	No	0x7500	0x01040104 0x01000104 0x0E940104
Er.942	Frequent parameter storage	Yes	0x7600	0x09420942
Er.500	Motor overspeed	Yes	0x8400	0x05000500
Er.b00	Too large position deviation	Yes	0x8611	0x0b000b00
Er.b02	Position deviation exceeding threshold in fully closed loop	Yes	0x8611	0x0b020b02
Er.208	FPGA system sampling timeout	No	0xFF00	0x02080208
Er.220	UVW phase sequence error	No	0xFF00	0x02200220
Er.207	D/Q shaft current overflow	Yes	0xFF00	0x02070207
Er.234	Runaway	No	0xFF00	0x02340234
Er.602	Angle auto-tuning failure	Yes	0xFF00	0x06020602
Er.510	Pulse output overspeed	Yes	0xFF00	0x05100510
Er.b01	Pulse input abnormal	Yes	0xFF00	0x0b010b01
Er.A40	Parameter auto-tuning failure	Yes	0xFF00	0x0A400A40
Er.601	Homing timeout	Yes	0xFF00	0x06010601
Er.996	CANopen network passive error	Yes	0x8120	0x09960996
Er.995	CANopen network disconnection recovery	Yes	0x8140	0x09950995
Er.d04	CANopen node protection or heartbeat timeout	Yes	0x8130	0x0d040d04
Er.d05	NMT steering initialization when the motor is enabled	No	0x8160	0x0d050d05
Er.d06	NMT steering stop when the motor is enabled	No	0x8170	0x0d060d06
Er.d07	CANopen network disconnection	Yes	0x8141	0x0d070d07
Er.d08	CANopen PDO transmission length error	Yes	0x8210	0x0d080d08

Display	Fault Name	Reset	Error Code (603Fh)	Auxiliary Code (203Fh)
Er.d09	Software position upper and 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 Troubleshooting Mode

For details on troubleshooting of IS620P series servo drive, see the *IS620P Series Servo User Manual - Comprehensive* or *IS620P Series Servo User Manual - Simplified*. This document describes communication troubleshooting only.

Displayed Fault	Name	Cause	Measure
Er.d04	Node protection or heartbeat timeout	The time configured by the consumer or the node protection time expires for the slave node.	■ Check whether all CAN nodes are online, check the CANopen configuration, or restore nodes or communication.
Er.d05	NMT steering initialization when the motor is enabled	NMT steering initialization received when the motor is enabled	■ Reset the NMT node. When the NMT is modified, disable the output stage.
Er.d06	NMT steering stop when the motor is enabled	When the motor is enabled, an NMT stop reference is received.	■ Reset the NMT node. When the NMT is modified, disable the output stage.
Er.d07	CANopen network disconnection	Too many errors	■ Check the CANopen network and reconnect the network.
Er.d08	PDO transmission length error	The length of content transmitted in a PDO is inconsistent with the mapping length during configuration.	■ Re-configure the PDO and reset nodes or communication.
Er.d09	Software position upper and lower limit setting error	The lower limit of software position is larger than the upper limit.	■ Set 0x607D correctly and ensure: 607D-1h ＜ 607D-2h
Er.d10	Home offset setting error	The home offset is set outside the software position lower/upper limit.	■ Set 607D and 607C correctly and ensure: 607C>(607D-1h) 607C<(607D-2h)
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 make sure that the synchronization cycle is correctly set. ■ Ensure that the synchronization cycle of the host computer is correctly set and is consistent with the parameter setting of 60C2h. ■ Check the cable connection between the slave node and the master node.

5.3 SDO Transmission Abort Code

Abort Code	Function Description
0503 0000	Trigger bits are not alternated.
0504 0000	Timeout occurs in the SDO protocol.
0504 0001	The client/server command word is invalid or unknown.
0504 0005	Memory overflow occurs.
0601 0000	Access to objects is not supported.
0601 0001	Attempt to read a write-only object.
0601 0002	Attempt to write a read-only object.
0602 0000	The object does not exist in the object dictionary.
0604 0041	The object cannot be mapped to the PDO.
0604 0042	The number and length of mapped objects exceed the PDO length.
0604 0043	General parameters are incompatible.
0604 0047	General device content is incompatible.
0606 0000	Accessing objects fails due to an hardware error.
0607 0010	The data type does not match and the service parameter length does not match.
0607 0012	The data type does not match and the service parameter is too long.
0607 0013	The data type does not match and the service parameter is too short.
0609 0011	The sub-index does not exist.
0609 0030	Invalid value for parameter.
0609 0031	The parameter value entered is too large.
0609 0032	The parameter value entered is too small.
0609 0036	The maximum value is smaller than the minimum value.
0800 0000	General error
0800 0020	Data cannot be transmitted or stored to the application.
0800 0021	Data cannot be transmitted or stored to the application due to local control.
0800 0022	Data cannot be transmitted or stored to the application due to the current device status.
0800 0023	An error occurs in the object dictionary or the object dictionary does not exist.
0800 0024	The value does not exist.

Memo NO. __________

Date / /

Chapter 6 Object Dictionary

Chapter 6 Object Dictionary

6.1 Object Classification

★ Definitions of terms

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

"Data type": See Table 6-1.

Table 6-1 Description of data types

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.

Table 6-2 Description of read/write types

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

"Object type": See Table 6-3.

Table 6-3 Description of object types

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

Index	Sub-index	Name	Accessibility	Mapping	Data Type	Unit	Data Range	Factory Default
1000h	-	Device type	RO	NO	Uint32	VAR	Uint 32	0x20192
1001h	-	Error register	RO	NO	Uint8	VAR	Uint 8	0x0
1003h	-	Predefined error field	RO	NO	Uint32	ARR	-	-
	1-4h	Error field	RW	NO	Uint32	-	Uint 32	0
1005h	-	Synchronization packet 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	IS620P Servo Drive
1009h	-	Hardware version	CONST	NO	String	VAR	String	V0.0
100Ah	-	Software version	CONST	NO	String	VAR	String	402.XX
100Ch	-	Node protection 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 sub-protocol 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 sub-protocol area	RW	NO	Uint32	-	-	1
1014h	-	Emergency packet COB-ID	RW	NO	Uint32	VAR	Uint 32	0x80_ Node_ID
1016h	-	Consumer heartbeat time	RW	NO	Uint32	ARR	-	-
	1-5h	Consumer heartbeat time	RW	NO	Uint32	-	Uint 32	0
1017h	-	Producer heartbeat time	RW	NO	Uint16	VAR	Uint 16	0

Index	Sub-index	Name	Accessibility	Mapping	Data Type	Unit	Data Range	Factory Default
1018h	-	Device object description	RO	NO	Related to individual	REC	-	-
	1h	Manufacturer ID	RO	NO	Uint32	-	Uint 32	0x3B9
	2h	Device code	RO	NO	Uint32	-	Uint 32	0xD0107
	3h	Device revision	RO	NO	Uint32	-	Uint 32	0x1920001
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	-	-
	1-8h	Mapping object of RPDO1	RW	NO	Uint32	-	Uint 32	-

Index	Sub-index	Name	Accessibility	Mapping	Data Type	Unit	Data Range	Factory Default
1601h		Mapping parameter of RPDO2	RW	NO	RPDO Mapping parameter	REC	-	-
	1-8h	Mapping object of RPDO2	RW	NO	Uint32	-	Uint 32	-
1602h		Mapping parameter of RPDO3	RW	NO	RPDO Mapping parameter	REC	-	-
	1-8h	Mapping object of RPDO3	RW	NO	Uint32	-	Uint 32	-
1603h		Mapping parameter of RPDO4	RW	NO	RPDO Mapping parameter	REC	-	-
	1-8h	Mapping object of RPDO4	RW	NO	Uint32	-	Uint 32	-
1800h		Communication parameter of TPDO1	RW	NO	Mapping 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	Disabled time	RW	NO	Uint16	-	Uint 16	0
	5h	Event timer	RW	NO	Uint16	-	Uint 16	0
1801h		Communication parameter of TPDO2	RW	NO	Mapping 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	Disabled time	RW	NO	Uint16	-	Uint 16	0
	5h	Event timer	RW	NO	Uint16	-	Uint 16	0
1802h		Communication parameter of TPDO3	RW	NO	Mapping 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	Disabled time	RW	NO	Uint16	-	Uint 16	0
	5h	Event timer	RW	NO	Uint16	-	Uint 16	0

Index	Sub-index	Name	Accessibility	Mapping	Data Type	Unit	Data Range	Factory Default
1803h		Communication parameter of TPDO4	RW	NO	Mapping 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	Disabled time	RW	NO	Uint16	-	Uint 16	0
	5h	Event timer	RW	NO	Uint16	-	Uint 16	0
1A00h		Mapping parameter of TPDO1	RW	NO	Mapping Mapping parameter	REC	-	-
	1-8h	Mapping object of TPDO1	RW	NO	Uint32	-	Uint 32	-
1A01h		Mapping parameter of TPDO2	RW	NO	Mapping Mapping parameter	REC	-	-
	1-8h	Mapping object of TPDO2	RW	NO	Uint32	-	Uint 32	-
1A02h		Mapping parameter of TPDO3	RW	NO	Mapping Mapping parameter	REC	-	-
	1h	Mapping object of TPDO3	RW	NO	Uint32	-	Uint 32	-
1A03h		Mapping parameter of TPDO4	RW	NO	Mapping Mapping parameter	REC	-	-
	1-8h	Mapping object of TPDO4	RW	NO	Uint32	-	Uint 32	-

6.3 Object Group 2000h

The object group 2000h is an object table defined by Inovance and is associated with functional codes of devices. All objects in the area support PDO mapping.

2000h Servo motor parameters

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

6 Object Dictionary

2001h Servo drive parameters

Index	Sub-index	Name	Accessibility	Mapping	Data Type	Unit	Data Range	Factory 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 No.	RW	YES	Uint16	-	0 to 65535	-

2002h Basic control parameters

Index	Sub-index	Name	Accessibility	Mapping	Data Type	Unit	Data Range	Factory Default
2002h	1h	Control mode selection	RW	YES	Uint16	-	0 to 6	1
	3h	Rotational direction selection	RW	YES	Uint16	-	0 to 1	0
	4h	Output pulse phase	RW	YES	Uint16	-	0–1	0
	6h	Selection of the mode for disabling the servo	RW	YES	Uint16	-	0–1	0
	7h	Selection of fault stop mode No.2	RW	YES	Uint16	-	0–1	0
	8h	Selection of limit stop mode	RW	YES	Uint16	-	0 to 2	1
	9h	Selection of fault stop mode No.1	RW	YES	Uint16	-	0	0
	0Ah	Delay from brake output ON to command receiving	RW	YES	Uint16	ms	0 to 500	250
	0Bh	Delay from brake output OFF to motor power-off in idle state	RW	YES	Uint16	ms	1 to 1000	150
	0Ch	Motor speed threshold at brake output OFF in the rotating status	RW	YES	Uint16	rpm	0 to 3000	30
	0Dh	Delay from motor power-off to brake output OFF in the rotating status	RW	YES	Uint16	ms	1 to 1000	500
	0Fh	LED warning display selection	RW	YES	Uint16	-	0–1	0
	13h	S-ON filter time constant	RW	YES	Uint16	ms	0 to 64	0
	16h	Allowable minimum resistance of braking resistor	RO	TPDO	Uint16	Ω	-	-
	17h	Power of built-in braking resistor	RO	TPDO	Uint16	W	-	-
	18h	Resistance of built-in braking resistor	RO	TPDO	Uint16	Ω	-	-
	19h	Resistor heat dissipation coefficient	RW	YES	Uint16	%	10 to 100	30
	1Ah	Braking resistor type	RW	YES	Uint16	-	0 to 3	0
	1Bh	Power of external braking resistor	RW	YES	Uint16	W	1 to 65535	-
	1Ch	Resistance of external braking 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 panel display function	RW	YES	Uint16	-	0 to 99	50
	29h	Enabling visual motor parameters	WO	NO	Uint16	-	-	-
	2Ah	Factory password	WO	NO	Uint16	-	-	-

2003h Terminal input parameters

Index	Sub-index	Name	Accessibility	Mapping	Data Type	Unit	Data Range	Factory Default
2003h	1h	Valid DI function allocation 1 at power-on	RW	YES	Uint16	-	0-0xFFFF	0
	2h	Valid DI function allocation 2 at power-on	RW	YES	Uint16	-	0-0xFFFF	0
	3h	DI1 terminal function selection	RW	YES	Uint16	-	0 to 37	14
	4h	DI1 terminal logic selection	RW	YES	Uint16	-	0 to 4	0
	5h	DI2 terminal function selection	RW	YES	Uint16	-	0 to 37	15
	6h	DI2 terminal logic selection	RW	YES	Uint16	-	0 to 4	0
	7h	DI3 terminal function selection	RW	YES	Uint16	-	0 to 37	13
	8h	DI3 terminal logic selection	RW	YES	Uint16	-	0 to 4	0
	9h	DI4 terminal function selection	RW	YES	Uint16	-	0 to 37	2
	0Ah	DI4 terminal logic selection	RW	YES	Uint16	-	0 to 4	0
	0Bh	DI5 terminal function selection	RW	YES	Uint16	-	0 to 37	1
	0Ch	DI5 terminal logic selection	RW	YES	Uint16	-	0 to 4	0
	0Dh	DI6 terminal function selection	RW	YES	Uint16	-	0 to 37	12
	0Eh	DI6 terminal logic selection	RW	YES	Uint16	-	0 to 4	0
	0Fh	DI7 terminal function selection	RW	YES	Uint16	-	0 to 37	3
	10h	DI7 terminal logic selection	RW	YES	Uint16	-	0 to 4	0
	11h	DI8 terminal function selection	RW	YES	Uint16	-	0 to 37	31
	12h	DI8 terminal logic selection	RW	YES	Uint16	-	0 to 4	0
	13h	DI9 terminal function selection	RW	YES	Uint16	-	0 to 37	0
	14h	DI9 terminal logic selection	RW	YES	Uint16	-	0 to 4	0
	23h	Valid DI function allocation 3 at power-on	RW	YES	Uint16	-	0-0xFFFF	0
	24h	Valid DI function allocation 4 at power-on	RW	YES	Uint16	-	0-0xFFFF	0
	33h	AI1 offset	RW	YES	Int16	mV	-5000 to 5000	0
	34h	AI1 input filter time constant	RW	YES	Uint16	ms	0 to 655.35	2.00
	36h	AI1 dead zone	RW	YES	Uint16	mV	0 to 1000.0	10.0
	37h	AI1 zero drift	RW	YES	Int16	mV	-500.0 to 500.0	0.0
	38h	AI2 offset	RW	YES	Int16	mV	-5000 to 5000	0
	39h	AI2 input filter time constant	RW	YES	Uint16	ms	0 to 655.35	2.00
	3Bh	AI2 dead zone	RW	YES	Uint16	mV	0 to 1000.0	10.0
	3Ch	AI2 zero drift	RW	YES	Int16	mV	-500.0 to 500.0	0.0
	51h	Speed corresponding to analog 10 V	RW	YES	Uint16	rpm	0 to 6000	3000
	52h	Torque corresponding to analog 10 V	RW	YES	Uint16	Times	1.00 to 8.00	1.00

2004h Output terminal parameters

Index	Sub-index	Name	Accessibility	Mapping	Data Type	Unit	Data Range	Factory Default
2004h	1h	DO1 terminal function selection	RW	YES	Uint16	-	0 to 19	1
	2h	DO1 terminal logic selection	RW	YES	Uint16	-	0–1	0
	3h	DO2 terminal function selection	RW	YES	Uint16	-	0 to 19	5
	4h	DO2 terminal logic selection	RW	YES	Uint16	-	0–1	0
	5h	DO3 terminal function selection	RW	YES	Uint16	-	0 to 19	3
	6h	DO3 terminal logic selection	RW	YES	Uint16	-	0–1	0
	7h	DO4 terminal function selection	RW	YES	Uint16	-	0 to 19	11
	8h	DO4 terminal logic selection	RW	YES	Uint16	-	0–1	0
	9h	DO5 terminal function selection	RW	YES	Uint16	-	0 to 19	16
	0Ah	DO5 terminal logic selection	RW	YES	Uint16	-	0–1	0
	17h	DO source selection	RW	YES	Uint16	-	0 to 31	0
	33h	AO1 signal selection	RW	YES	Uint16	-	0 to 9	0
	34h	AO1 bias voltage	RW	YES	Int16	mV	-10000 to 10000	5000
	35h	AO1 multiplying power	RW	YES	Int16	Times	-99.99 to 99.99	1.00
	36h	AO2 signal selection	RW	YES	Uint16	-	0 to 9	0
	37h	AO2 bias voltage	RW	YES	Int16	mV	-10000 to 10000	5000
	38h	AO2 multiplying power	RW	YES	Int16	Times	-99.99 to 99.99	1.00

2005h Position control parameters in

Index	Sub-index	Name	Accessibility	Mapping	Data Type	Unit	Data Range	Factory Default
2005h	1h	Position reference source	RW	YES	Uint16	-	0 to 2	0
	2h	Pulse reference input terminal selection	RW	YES	Uint16	-	0–1	0
	3h	Number of position references for each rotational round of motor	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 selection	RW	YES	Uint16	-	0 to 3	1
	15h	Output condition for positioning	RW	YES	Uint16	-	0 to 2	0
	16h	Positioning complete threshold	RW	YES	Uint16	Encoder unit	1 to 65535	734
	17h	Positioning approach threshold	RW	YES	Uint16	Encoder unit	1 to 65535	65535
	18h	Interruption fixed length	RW	YES	Uint16	-	0–1	0
	19h	Displacement of interruption fixed length	RW	YES	Uint32	Reference unit	0 to 1073741824	10000
	1Bh	Constant speed for interruption fixed length	RW	YES	Uint16	rpm	0 to 6000	200
	1Ch	Acceleration/Deceleration time of interruption fixed length	RW	YES	Uint16	ms	0 to 1000	10
	1Eh	Enabled signal for unlocking fixed length	RW	YES	Uint16	-	0–1	1
	1Fh	Homing control	RW	YES	Uint16	-	0 to 6	0
	20h	Homing mode	RW	YES	Uint16	-	0 to 13	0
	21h	Speed of searching for home switch signal (at a high speed)	RW	YES	Uint16	rpm	0 to 3000	100
	22h	Speed of searching for home switch signal (at a low speed)	RW	YES	Uint16	rpm	0 to 1000	10
	23h	Acceleration/Deceleration time during home searching	RW	YES	Uint16	ms	0 to 1000	1000
	24h	Duration limit of homing	RW	YES	Uint16	ms	0 to 65535	10000
	25h	Mechanical home offset	RW	YES	Uint32	Reference unit	-1073741824–1073741824	0
	27h	Servo pulse output source selection	RW	YES	Uint16	-	0 to 2	0
	28h	Condition for switching the electronic gear ratio	RW	YES	Uint16	-	0–1	0
	29h	Mechanical home offset and action after the limit is reached	RW	YES	Uint16	-	0 to 3	0
	2Ah	Selection of Z pulse output polarity	RW	YES	Uint16	-	0–1	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	-2147483648–2147483647	0
	31h	Position offset in absolute position linear mode (high 32 bits)	RW	YES	Int32	Encoder unit	-2147483648–2147483647	0
	33h	Mechanical gear ratio in absolute position rotation mode (numerator)	RW	YES	Uint16	1	1-65535	65535
	34h	Mechanical gear ratio in absolute position rotating mode (denominator)	RW	YES		1	1-65535	1
	35h	Number of pulses for one round of load rotation in absolute position rotation mode (low 32 bits)	RW	YES	Uint32	Encoder unit	0 to 4294967295	0

Index	Sub-index	Name	Accessibility	Mapping	Data Type	Unit	Data Range	Factory Default
2005h	37h	Number of pulses for one round of load rotation in absolute position rotation mode (high 32 bits)	RW	YES	Uint16	Encoder Unit	0 to 127	0
	39h	Stop zero speed threshold	RW	YES	Uint16	rpm	0 to 1000	2
	3Bh	Stop zero torque limit	RW	YES	Uint16	%	0 to 300.0	100.0%
	3Ch	Positioning complete window time	RW	YES	Uint16	ms	0 to 30000	1
	3Dh	Positioning complete hold 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	Accessibility	Mapping	Data Type	Unit	Data Range	Factory 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	Value set on keypad for speed reference	RW	YES	Int16	rpm	-6000 to 6000	200
	5h	Value set for jog speed	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	Maximum rotational 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 selection	RW	YES	Uint16	-	0–1	1
	10h	Speed threshold for zero speed clamp	RW	YES	Uint16	rpm	0 to 6000	10
	11h	Motor rotational speed threshold	RW	YES	Uint16	rpm	0 to 1000	20
	12h	Speed consistent signal threshold	RW	YES	Uint16	rpm	0 to 100	10
	13h	Speed reached signal threshold	RW	YES	Uint16	rpm	10 to 6000	1000
	14h	Zero speed output signal threshold	RW	YES	Uint16	rpm	1 to 6000	10

2007h Torque control parameters

Index	Sub-index	Name	Accessibility	Mapping	Data Type	Unit	Data Range	Factory 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 source	RW	YES	Uint16	-	0 to 3	0
	4h	Value set for torque reference on 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
	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	Internal positive torque limit	RW	YES	Uint16	%	0.0 to 300.0	300.0
	0Bh	Internal negative torque limit	RW	YES	Uint16	%	0.0 to 300.0	300.0
	0Ch	External positive torque limit	RW	YES	Uint16	%	0.0 to 300.0	300.0
	0Dh	External negative 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	Selection of 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	Base 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	Accessibility	Mapping	Data Type	Unit	Data Range	Factory 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	Second 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	Second position loop gain	RW	YES	Uint16	Hz	0.0 to 2000.0	64.0
	9h	Second gain mode setting	RW	YES	Uint16	-	0–1	1
	0Ah	Gain switching condition	RW	YES	Uint16	-	0 to 10	0
	0Bh	Gain switching delay	RW	YES	Uint16	ms	0.0 to 1000.0	5.0
	0Ch	Gain switching level	RW	YES	Uint16	Based on switching conditions	0 to 20000	50
	0Dh	Gain switching lag	RW	YES	Uint16	Based on switching conditions	0 to 20000	30
	0Eh	Position gain switching time	RW	YES	Uint16	ms	0.0 to 1000.0	3.0
	10h	Load rotation inertia ratio	RW	YES	Uint16	Times	0.00 to 120.00	1.00
	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
	17h	Speed feedback filter option	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 Automatic adjustment parameters

Index	Sub-index	Name	Accessibility	Mapping	Data Type	Unit	Data Range	Factory Default
2009h	1h	Selection of automatic adjustment mode	RW	YES	Uint16	-	0 to 2	0
	2h	Rigid level selection	RW	YES	Uint16	-	0 to 31	12
	3h	Mode selection of adaptive notch	RW	YES	Uint16	-	0 to 4	0
	4h	Online inertia auto-tuning mode	RW	YES	Uint16	-	0 to 3	0
	5h	Selection of low-frequency resonance suppression mode	RW	YES	Uint16	-	0–1	0
	6h	Selection of offline inertia auto-tuning mode	RW	YES	Uint16	-	0–1	0
	7h	Maximum speed for inertia auto-tuning	RW	YES	Uint16	rpm	100 to 1000	500
	8h	Time constant for acceleration to the maximum speed during inertia auto-tuning	RW	YES	Uint16	ms	20 to 800	125
	9h	Interval after an inertia auto-tuning	RW	YES	Uint16	ms	50 to 10000	800
	0Ah	Number of motor rotation rounds for an inertia auto-tuning	RO	TPDO	Uint16	r	0.00 to 2.00	-
	0Dh	Group 1 notch frequency	RW	YES	Uint16	Hz	50 to 4000	4000
	0Eh	Group 1 notch width level	RW	YES	Uint16	-	0 to 20	2
	0Fh	Group 1 notch depth level	RW	YES	Uint16	-	0 to 99	0
	10h	Group 2 notch frequency	RW	YES	Uint16	Hz	50 to 4000	4000
	11h	Group 2 notch width level	RW	YES	Uint16	-	0 to 20	2
	12h	Group 2 notch depth level	RW	YES	Uint16	-	0 to 99	0
	13h	Group 3 notch frequency	RW	YES	Uint16	Hz	50 to 4000	4000
	14h	Group 3 notch width level	RW	YES	Uint16	-	0 to 20	2
	15h	Group 3 notch depth level	RW	YES	Uint16	-	0 to 99	0
	16h	Group 4 notch frequency	RW	YES	Uint16	Hz	50 to 4000	0
	17h	Group 4 notch width level	RW	YES	Uint16	-	0 to 20	0
	18h	Group 4 notch depth level	RW	YES	Uint16	-	0 to 99	0
	19h	Obtained 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	Low-frequency resonance frequency	RW	YES	Uint16	Hz	1.0 to 100.0	100.0
	28h	Low-frequency resonance filter setting	RW	YES	Uint16	-	0 to 10	2

200Ah Fault and protection parameters

Index	Sub-index	Name	Accessibility	Mapping	Data Type	Unit	Data Range	Factory Default
200Ah	1h	Power input phase loss protection	RW	YES	Uint16	-	0 to 2	0
	2h	Setting of absolute position limit	RW	YES	Uint16	-	0 to 2	0
	4h	Enable power failure protection	RW	YES	Uint16	-	0–1	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	Maximum position pulse frequency	RW	YES	Uint16	kHz	100 to 4000	4000
	0Bh	Threshold for large position deviation	RW	YES	Uint32	Encoder unit	1 to 1073741824	3145728
	0Dh	Enable runaway protection function	RW	YES	Uint16	-	0–1	1
	11h	Position deviation threshold for low-frequency resonance suppression	RW	YES	Uint16	Encoder unit	1 to 1000	5
	12h	Selection of position setting unit	RW	YES	Uint16	-	0–1	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 terminal	RW	YES	Uint6	25 ns	0 to 255	30
	1Ah	Filter time constant of speed feedback display value	RW	YES	Uint16	ms	0 to 5000	50
	1Bh	Enable motor overload shielding	RW	YES	Uint16	-	0–1	0
	1Ch	Speed DO filter time constant	RW	YES	Uint16	ms	0 to 5000	10
	1Dh	Filter time constant of quadrature encoder	RW	YES	Uint16	25 ns	0 to 255	5
	1Eh	Linear encoder filter time	RW	YES	Uint16	25 ns	0 to 255	15
	1Fh	Filter time constant of high-speed pulse input pin	RW	YES	Uint16	25 ns	0 to 255	3
	21h	Time threshold for locked rotor over-temperature protection	RW	YES	Uint16	ms	10 to 65535	200
	22h	Locked rotor over-temperature protection	RW	YES	Uint16	-	0–1	1
	25h	Selection of encoder multi-round overflow fault	RW	YES	Uint16	-	0–1	0
	30h	Enable brake protection detection	RW	YES	Uint16	-	0–1	1
	31h	Gravity load detection value	RW	YES	Uint16	%	0 to 300.0	30.0

200Bh Display parameters

Index	Sub-index	Name	Accessibility	Mapping	Data Type	Unit	Data Range	Factory 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	Input signal (DI signal) monitoring	RO	TPDO	Uint16	-	-	-
	6h	Output signal (DO signal) monitoring	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	-	-
	16h	AI1 sampling voltage	RO	TPDO	Int16	V	-	-
	17h	AI2 sampling voltage	RO	TPDO	Int16	V	-	-
	19h	Valid value of phase current	RO	TPDO	Uint16	A	-	-
	1Bh	Bus voltage	RO	TPDO	Uint16	V	-	-
	1Ch	Module temperature	RO	TPDO	Uint16	℃	-	-
	22h	Fault record	RW	YES	Uint16	-	0 to 9	0
	23h	Fault code upon the displayed fault record	RO	TPDO	Uint16	-	-	-
	24h	Time stamp upon the displayed fault	RO	TPDO	Uint32	s	-	-
	26h	Motor speed upon the displayed fault	RO	TPDO	Int16	rpm	-	-
	27h	Motor phase U current upon the displayed fault	RO	TPDO	Int16	A	-	-
	28h	Motor phase V current upon the displayed fault	RO	TPDO	Int16	A	-	-
	29h	Bus voltage upon the displayed fault	RO	TPDO	Uint16	V	-	-

Index	Sub-index	Name	Accessibility	Mapping	Data Type	Unit	Data Range	Factory Default
200Bh	2Ah	Input terminal status upon the displayed fault	RO	TPDO	Uint16	-	-	-
	2Bh	Output terminal status upon the displayed fault	RO	TPDO	Uint16	-	-	-
	36h	Position deviation counter	RO	TPDO	Int32	Reference unit	-	-
	38h	Actual motor speed	RO	TPDO	Int32	rpm	-	-
	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 input position reference counter	RO	TPDO	Int32	Reference unit	-	-
	47h	Number of encoder rotation rounds in an absolute value	RO	TPDO	Uint16	r	-	0
	48h	Position of absolute encoder within one turn	RO	TPDO	Uint32	Encoder unit	-	0
	4Eh	Absolute encoder setting in an absolute value (low 32 bits)	RO	TPDO	Int32	Encoder unit	-	0
	50h	Absolute encoder position in an absolute value (high 32 bits)	RO	TPDO	Int32	Encoder unit	-	0
	52h	Position of rotating load at one round (low 32 bits)	RO	TPDO	Uint32	Encoder unit	-	0
	54h	Position of rotating load at one round (high 32 bits)	RO	TPDO	Uint32	Encoder unit	-	0
	56h	Position of rotating load at one round	RO	TPDO	Uint32	Reference unit	-	0

200Ch Communication parameters

Index	Sub-index	Name	Accessibility	Mapping	Data Type	Unit	Data Range	Factory Default
200Ch	1h	Servo axis address	RW	YES	Uint16	-	1 to 247	1
	3h	Serial port baud rate setting	RW	YES	Uint16	-	0 to 5	5
	4h	Modbus data format	RW	YES	Uint16	-	0 to 3	0
	5h	CANopen synchronization error too large Threshold	RW	YES	Uint16	-	0 to 5	0
	9h	CAN communication rate setting	RW	YES	Uint16	-	0 to 7	5
	0Ah	Communication VDI	RW	YES	Uint16	-	0–1	0
	0Bh	Default VDI value after power-on	RW	YES	Uint16	-	0 to 65535	0
	0Ch	Communication VDO	RW	YES	Uint16	-	0–1	0
	0Dh	Default level when VDO function is set to 0	RW	YES	Uint16	-	0 to 65535	0
	0Eh	Update functional codes to EEPROM during Modbus communication	RW	YES	Uint16	-	0–1	1
	0Fh	Modbus error code	RO	TPDO	Uint16	1	0 to 65535	-
	1Ah	Modbus reference 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 functional parameters

Index	Sub-index	Name	Accessibility	Mapping	Data Type	Unit	Data Range	Factory Default
200Dh	1h	Software reset	RW	YES	Uint16	-	0–1	0
	2h	Fault reset	RW	YES	Uint16	-	0–1	0
	6h	Emergency stop	RW	YES	Uint16	-	0–1	0
	0Bh	Analog automatic adjustment	RW	YES	Uint16	-	0–1	0
	12h	Enable DIDO	RW	YES	Uint16	-	0 to 3	0
	13h	Provide DI	RW	YES	Uint16	-	0-0x01FF	0x01FF
	14h	Provide DO	RW	YES	Uint16	-	0-0x001F	0

200Fh Fully closed-loop parameters

Index	Sub-index	Name	Accessibility	Mapping	Data Type	Unit	Data Range	Factory Default
200Fh	1h	Encoder feedback mode	RW	YES	Uint16	-	0 to 2	0
	2h	Usage of external encoder	RW	YES	Uint16	-	0–1	0
	5h	External encoder pulses per one round of the motor	RW	YES	Uint32	External encoder unit	0 to 1073741824	10000
	9h	Fully closed-loop position deviation threshold	RW	YES	Uint32	External encoder unit	0 to 1073741824	10000
	0Bh	Fully closed-loop position deviation clearing setting	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	Fully closed-loop position deviation counter	RO	TPDO	Uint32	External encoder unit	-1073741824–1073741824	0
	13h	Internal encoder feedback value	RO	TPDO	Uint32	Internal encoder unit	-1073741824–1073741824	0
	15h	External encoder feedback value	RO	TPDO	Uint32	External encoder unit	-1073741824–1073741824	0

2017h VDI/VDO function

Index	Sub-index	Name	Accessibility	Mapping	Data Type	Unit	Data Range	Factory Default
2017h	1h	VDI1 terminal function selection	RW	YES	Uint16	-	0 to 37	0
	2h	VDI1 terminal logic selection	RW	YES	Uint16	-	0–1	0
	3h	VDI2 terminal function selection	RW	YES	Uint16	-	0 to 37	0
	4h	VDI2 terminal logic selection	RW	YES	Uint16	-	0–1	0
	5h	VDI3 terminal function selection	RW	YES	Uint16	-	0 to 37	0
	6h	VDI3 terminal logic selection	RW	YES	Uint16	-	0–1	0
	7h	VDI4 terminal function selection	RW	YES	Uint16	-	0 to 37	0
	8h	VDI4 terminal logic selection	RW	YES	Uint16	-	0–1	0
	9h	VDI5 terminal function selection	RW	YES	Uint16	-	0 to 37	0
	0Ah	VDI5 terminal logic selection	RW	YES	Uint16	-	0–1	0
	0Bh	VDI6 terminal function selection	RW	YES	Uint16	-	0 to 37	0
	0Ch	VDI6 terminal logic selection	RW	YES	Uint16	-	0–1	0
	0Dh	VDI7 terminal function selection	RW	YES	Uint16	-	0 to 37	0
	0Eh	VDI7 terminal logic selection	RW	YES	Uint16	-	0–1	0
	0Fh	VDI8 terminal function selection	RW	YES	Uint16	-	0 to 37	0
	10h	VDI8 terminal logic selection	RW	YES	Uint16	-	0–1	0
	11h	VDI9 terminal function selection	RW	YES	Uint16	-	0 to 37	0

Index	Sub-index	Name	Accessibility	Mapping	Data Type	Unit	Data Range	Factory Default
2017h	12h	VDI9 terminal function selection	RW	YES	Uint16	-	0–1	0
	13h	VDI10 terminal function selection	RW	YES	Uint16	-	0 to 37	0
	14h	VDI10 terminal logic selection	RW	YES	Uint16	-	0–1	0
	15h	VDI11 terminal function selection	RW	YES	Uint16	-	0 to 37	0
	16h	VDI11 terminal logic selection	RW	YES	Uint16	-	0–1	0
	17h	VDI12 terminal function selection	RW	YES	Uint16	-	0 to 37	0
	18h	VDI12 terminal logic selection	RW	YES	Uint16	-	0–1	0
	19h	VDI13 terminal function selection	RW	YES	Uint16	-	0 to 37	0
	1Ah	VDI13 terminal logic selection	RW	YES	Uint16	-	0–1	0
	1Bh	VDI14 terminal function selection	RW	YES	Uint16	-	0 to 37	0
	1Ch	VDI14 terminal logic selection	RW	YES	Uint16	-	0–1	0
	1Dh	VDI15 terminal function selection	RW	YES	Uint16	-	0 to 37	0
	1Eh	VDI15 terminal logic selection	RW	YES	Uint16	-	0–1	0
	1Fh	VDI16 terminal function selection	RW	YES	Uint16	-	0 to 37	0
	20h	VDI16 terminal logic selection	RW	YES	Uint16	-	0–1	0
	21h	VDO virtual level	RO	TPDO	Uint16	-	-	-
	22h	VDO1 terminal function selection	RW	YES	Uint16	-	0 to 19	0
	23h	VDO1 terminal logic selection	RW	YES	Uint16	-	0–1	0
	24h	VDO2 terminal function selection	RW	YES	Uint16	-	0 to 19	0
	25h	VDO2 terminal logic selection	RW	YES	Uint16	-	0–1	0
	26h	VDO3 terminal function selection	RW	YES	Uint16	-	0 to 19	0

Index	Sub-index	Name	Accessibility	Mapping	Data Type	Unit	Data Range	Factory Default
2017h	27h	VDO3 terminal logic selection	RW	YES	Uint16	-	0–1	0
	28h	VDO4 terminal function selection	RW	YES	Uint16	-	0 to 19	0
	29h	VDO4 terminal logic selection	RW	YES	Uint16	-	0–1	0
	2Ah	VDO5 terminal function selection	RW	YES	Uint16	-	0 to 19	0
	2Bh	VDO5 terminal logic selection	RW	YES	Uint16	-	0–1	0
	2Ch	VDO6 terminal function selection	RW	YES	Uint16	-	0 to 19	0
	2Dh	VDO6 terminal logic selection	RW	YES	Uint16	-	0–1	0
	2Eh	VDO7 terminal function selection	RW	YES	Uint16	-	0 to 19	0
	2Fh	VDO7 terminal logic selection	RW	YES	Uint16	-	0–1	0
	30h	VDO8 terminal function selection	RW	YES	Uint16	-	0 to 19	0
	31h	VDO8 terminal logic selection	RW	YES	Uint16	-	0–1	0
	32h	VDO9 terminal function selection	RW	YES	Uint16	-	0 to 19	0
	33h	VDO9 terminal logic selection	RW	YES	Uint16	-	0–1	0
	34h	VDO10 terminal function selection	RW	YES	Uint16	-	0 to 19	0
	35h	VDO10 terminal logic selection	RW	YES	Uint16	-	0–1	0
	36h	VDO11 terminal function selection	RW	YES	Uint16	-	0 to 19	0
	37h	VDO11 terminal logic selection	RW	YES	Uint16	-	0–1	0
	38h	VDO12 terminal function selection	RW	YES	Uint16	-	0 to 19	0
	39h	VDO12 terminal logic selection	RW	YES	Uint16	-	0–1	0
	3Ah	VDO13 terminal function selection	RW	YES	Uint16	-	0 to 19	0
	3Bh	VDO13 terminal logic selection	RW	YES	Uint16	-	0–1	0
	3Ch	VDO14 terminal function selection	RW	YES	Uint16	-	0 to 19	0
	3Dh	VDO14 terminal logic selection	RW	YES	Uint16	-	0–1	0
	3Eh	VDO15 terminal function selection	RW	YES	Uint16	-	0 to 19	0
	3Fh	VDO15 terminal logic selection	RW	YES	Uint16	-	0–1	0
	40h	VDO16 terminal function selection	RW	YES	Uint16	-	0 to 19	0
	41h	VDO16 terminal logic selection	RW	YES	Uint16	-	0–1	0

203Fh Inovance drive fault codes

Index	Sub-index	Name	Accessibility	Mapping	Data Type	Unit	Data Range	Factory Default
203Fh		Inovance drive fault code	RO	TPDO	Uint32	-	-	-

6.4 Object Group 6000h

The object group 6000h includes objects related to the supported sub-protocol DSP 402.

Index	Sub-index	Name	Accessibility	Mapping	Data Type	Unit	Data Range	Factory 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	NO	Int16	-	0 to 7	2
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	User 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	User unit	-2^{31} to $(2^{31}-1)$	-
6065h	-	Following error window	RW	YES	Uint32	User unit	0 to $(2^{32}-1)$	3145728 p
6067h	-	Position window	RW	YES	Uint32	User 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	rpm	-2^{31} to $(2^{31}-1)$	-
606Ch	-	Velocity actual value	RO	TPDO	Int32	rpm	-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 Window Time	RW	YES	Uint16	ms	0 to 65535	0
6071h	-	Target torque	RW	RPDO	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	User unit	-2^{31} to $(2^{31}-1)$	0
607Ch	-	Home offset	RW	YES	Int32	User unit	-2^{31} to $(2^{31}-1)$	0
607Dh		Software position limit						
	1h	Min position limit	RW	YES	Int32	User unit	-2^{31} to $(2^{31}-1)$	-2^{31}
	2h	Max position limit	RW	YES	Int32	User unit	-2^{31} to $(2^{31}-1)$	$(2^{31}-1)$
607Eh	-	Polarity	RW	Y	Uint8	-		0
607Fh	-	Max profile velocity	RW	YES	Uint32	User unit	0 to $(2^{32}-1)$	6000
6081h	-	Profile velocity	RW	YES	Uint32	User unit	0 to $(2^{32}-1)$	100
6083h	-	Profile acceleration	RW	YES	Uint32	User unit	0 to $(2^{32}-1)$	100

Index	Sub-index	Name	Accessibility	Mapping	Data Type	Unit	Data Range	Factory Default
6084h	-	Profile deceleration	RW	YES	Uint32	User unit	0 to (2^{32}-1)	100
6085h	-	Quick stop deceleration	RW	YES	Uint32	rpm/ms	0 to (2^{32}-1)	100
6086h	-	Motion profile type	RW	YES	Int16	-	0	0
6087h	-	Torque slope	RW	RPDO	UINT32	0.1%/s	0 to (2^{32}-1)	0
6093h		Position factor						
	1h	Numerator	RW	Y	Uint32	-	0 to (2^{32}-1)	1
	2h	Feed constant	RW	Y	Uint32	-	1 to (2^{32}-1)	1
6094h		Velocity encoder factor						
	1h	Numerator	RW	Y	Uint32	-	0 to (2^{32}-1)	1048576
	2h	Denominator	RW	Y	Uint32	-	1 to (2^{32}-1)	60
6095h		Velocity factor 1						
	1h	Numerator	RW	Y	Uint32	-	0 to (2^{32}-1)	1
	2h	Denominator	RW	Y	Uint32	-	1 to (2^{32}-1)	1
6097h		Acceleration factor						
	1h	Numerator	RW	Y	Uint32		0 to (2^{32}-1)	1048576000
	2h	Denominator	RW	Y	Uint32		1 to (2^{32}-1)	60
6098h	-	Homing mode	RW	YES	Int8	-	0 to 35	0
6099h		Homing speed						
	1h	Speed during search for switch	RW	YES	Uint32	rpm	0 to (2^{32}-1)	100
	2h	Speed during search for zero signal	RW	YES	Uint32	rpm	0 to (2^{32}-1)	10
609Ah		Homing acceleration	RW	YES	Uint32	rpm/ms	0 to (2^{32}-1)	100
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 profile acceleration	RW	YES	Uint32	rpm/ms	0 to (2^{32}-1)	1000
60C6h	-	Max profile deceleration	RW	YES	Uint32	rpm/ms	0 to (2^{32}-1)	1000
60F4h	-	Following error actual value	RO	TPDO	Int32	User unit	-2^{31} to (2^{31}-1)	-
60FCh	-	Position demand value	RO	TPDO	Int32	Encoder unit	-2^{31} to (2^{31}-1)	-
60FDh	-	DI status	RO	TPDO	Uint32	-	0 to (2^{32}-1)	-

Index	Sub-index	Name	Accessibility	Mapping	Data Type	Unit	Data Range	Factory Default
60FEh		Digital output						
	1h	DO status	RW	TPDO	Uint32	-	0 to $(2^{32}-1)$	0
60FFh	-	Target velocity	RW	YES	Int32	rpm	-2^{31} to $(2^{31}-1)$	0
6502h	-	Supported drive modes	RO	TPDO	Uint32	-	0 to $(2^{32}-1)$	0x0000006D

6.5 Details of Object Dictionary

6.5.1 Details of Communication Parameters

Index 1000h	Name	Device Type					Data Structure	VAR	Data Type	Uint32
	Accessibility	RO	Mapping	NO	Relevant Mode	-	Data Range	Uint 32	Factory Default	0x20192

The device type parameter is used to describe the sub-protocol or application specification of the used device.

Index 1001h	Name	Error Register					Data Structure	VAR	Data Type	Uint8
	Accessibility	RO	Mapping	NO	Relevant Mode	-	Data Range	Uint 8	Factory Default	0x0

Information about error types is included in bits, as listed in the table below:

Bit	Meaning	Bit	Meaning
0	Conventional	4	Communication
1	Current	5	Sub-protocol
2	Voltage	6	Reserved
3	Temperature	7	Defined by the manufacturer

When an error occurs, the bit corresponding to the error is bit 1. So long as an error occurs, bit 0 must be set to 1.

Index 1003h	Name	Pre-defined Error Field					Data Structure	ARR	Data Type	Uint32
	Accessibility	RO	Mapping	NO	Relevant Mode	-	Data Range	-	Factory Default	-

Sub-index 00h	Name	Number of Errors					Data Structure	-	Data Type	Uint8
	Accessibility	RW	Mapping	NO	Relevant Mode	-	Data Range	0 to 4	Factory Default	0

Only 0 can be entered. In this case, all error records are cleared.

Sub-index 1-4h	Name	Standard Error Field					Data Structure	-	Data Type	Uint32
	Accessibility	RW	Mapping	NO	Relevant Mode	-	Data Range	Uint 32	Factory Default	0

When the sub-index is 0, the data is unreadable.

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

31 16	15 0
Error code of the manufacturer	Standard error code

MSB LSB

6

Object Dictionary

Index 1005h	Name	COB-ID (COB-ID SYNC Message)				Data Structure	VAR	Data Type	Uint32
	Accessibility	RW	Mapping	NO	Relevant Mode	-	Data Range	Uint 32	Factory Default	0x80

Only 0x80h and 0x40000080h can be set.
When 0x80h is set, the synchronization generator does not work.
When 0x40000080h is set, the synchronization generator is activated.
Before the synchronization generator is activated, the synchronization cycle (1006h) must be set to a value rather than 0.

Index 1006h	Name	Communication Cycle Period				Data Structure	VAR	Data Type	Uint32
	Accessibility	RW	Mapping	NO	Relevant Mode	-	Data Range	Uint 32	Factory Default	0

The object dictionary is provided for the synchronization generator only and its unit is us.

Index 1008h	Name	Manufacturer Device Name				Data Structure	VAR	Data Type	String
	Accessibility	CONST	Mapping	NO	Relevant Mode	-	Data Range	String	Factory Default	IS620P Servo Drive

Index 1009h	Name	Manufacturer Hardware Version				Data Structure	VAR	Data Type	String
	Accessibility	CONST	Mapping	NO	Relevant Mode	-	Data Range	String	Factory Default	V0.0

Index 100Ah	Name	Manufacturer Software Version				Data Structure	VAR	Data Type	String
	Accessibility	CONST	Mapping	NO	Relevant Mode	-	Data Range	String	Factory Default	402.XX

In 402.XX:
YY: CANopen software update record number

Index 100Ch	Name	Guard Time				Data Structure	VAR	Data Type	Uint16
	Accessibility	RW	Mapping	NO	Relevant Mode	-	Data Range	Uint 16	Factory Default	0

Unit: ms

Index 100Dh	Name	Life Time Factor				Data Structure	VAR	Data Type	Uint8
	Accessibility	RW	Mapping	NO	Relevant Mode	-	Data Range	Uint 8	Factory Default	0

When the life time factor is used, it must be greater than 1.

Index 1010h	Name	Store Parameters				Data Structure	ARR	Data Type	Uint32
	Accessibility	RW	Mapping	NO	Relevant Mode	-	Data Range	Uint 8	Factory Default	0

Storing parameters means to storing current values of parameters to the EEPROM. When the EEPROM is loaded (the device is powered on again, a node is reset, or communication is reset) next time, the stored values are loaded.

To store parameters, specify the sub-index of the storage area and write "save" based on ASCII code. If other values are written, storing parameters fails.

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 are not automatically stored or stored based on commands.
1	Parameters are saved based on commands and are not automatically saved.
2	Parameters are automatically stored and are not stored based on commands.
3	Parameters are automatically stored or are stored based on commands.

The value returned after a sub-index is read indicates the mode in which the sub-index saves parameters.

The IS620P servo drive saves parameters based on commands and does not automatically save parameters. The value 1 is returned after a sub-index is read and saved.

Sub-index 00h	Name	Highest Sub-index Supported				Data Structure	-	Data Type	Uint8
	Accessibility	RO	Mapping	NO	Relevant Mode	-	Data Range	4	Factory Default	4

Sub-index 01h	Name	Save All Parameters				Data Structure	-	Data Type	Uint32
	Accessibility	RW	Mapping	NO	Relevant Mode	-	Data Range	-	Factory Default	1

Save all parameters in the object dictionary list.

Sub-index 02h	Name	Save Communication Parameters				Data Structure	-	Data Type	Uint32
	Accessibility	RW	Mapping	NO	Relevant Mode	-	Data Range	-	Factory Default	1

Save parameters of the object group 1000h.

6

Object Dictionary

Sub-index 03h	Name	Save Application Parameters				Data Structure	-	Data Type	Uint32
	Accessibility	RW	Mapping	NO	Relevant Mode -	Data Range	-	Factory Default	1

Save parameters of the object group 6000h.

Sub-index 04h	Name	Save Manufacturer Defined Parameters				Data Structure	-	Data Type	Uint32
	Accessibility	RW	Mapping	NO	Relevant Mode -	Data Range	-	Factory Default	1

Save parameters of the object group 2000h.

Index 1011h	Name	Restore Default Parameters				Data Structure	ARR	Data Type	Uint32
	Accessibility	RW	Mapping	NO	Relevant Mode -	Data Range	-	Factory Default	-

Restoring default parameters means to restore default parameters to the EEPROM. However, the operation does not take effect immediately. When the EEPROM is loaded (the device is powered on again, a node is reset, or communication is reset) next time, default values (factory defaults) are loaded.

To restore default parameters, specify the sub-index of the restoration area and write "load" based on ASCII code. If other values are written, restoring default parameters fails. 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 sub-index is read indicates the mode in which the sub-index restores default parameters.

The IS620P servo drive can restore default parameters. The value 1 is returned after a non-zero sub-index is read and saved.

Sub-index 00h	Name	Highest Sub-index Supported				Data Structure	-	Data Type	Uint8
	Accessibility	RO	Mapping	NO	Relevant Mode -	Data Range	4	Factory Default	4

Sub-index 01h	Name	Restore All Default Parameters				Data Structure	-	Data Type	Uint32
	Accessibility	RW	Mapping	NO	Relevant Mode -	Data Range	-	Factory Default	1

Restore all default parameters in the object dictionary list.

Sub-index 02h	Name	Restore default parameters of communication objects (Restore Communication Default Parameters)					Data Structure	-	Data Type	Uint32
	Accessibility	RW	Mapping	NO	Relevant Mode	-	Data Range	-	Factory Default	1
Restore default parameters of the object group 1000h.										

Sub-index 03h	Name	Restore default parameters of objects in the sub-protocol area (Restore Application Default Parameters)					Data Structure	-	Data Type	Uint32
	Accessibility	RW	Mapping	NO	Relevant Mode	-	Data Range	-	Factory Default	1
Restore default parameters of the object group 6000h.										

Sub-index 04h	Name	Restore Manufacturer Defined Default Parameters					Data Structure	-	Data Type	Uint32
	Accessibility	RW	Mapping	NO	Relevant Mode	-	Data Range	-	Factory 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
	Accessibility	RW	Mapping	NO	Relevant Mode	-	Data Range	Uint 32	Factory Default	0x80+ Node_ID

The highest bit indicates whether to disable the emergency packet of the device. Only the data "0x80+Node_ID" can be written for the bit to enable the emergency packet of the device.

If the data "0x80000080+Node_ID" is written, the emergency packet is disabled.

When the emergency packet takes effect, its COB-ID must be consistent with the object.

Index 1016h	Name	Consumer Heartbeat Time					Data Structure	ARR	Data Type	Uint32
	Accessibility	RW	Mapping	NO	Relevant Mode	-	Data Range	-	Factory Default	-

Parameters include the address of the monitored node and actual consumer time, which must be longer than the heartbeat producer time (unit: ms) of the corresponding node. Two different consumer time cannot be set for one node.

The parameters are described as followed:

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

MSB LSB

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

The IS620P servo drive can restore default parameters. The value 1 is returned after a non-zero sub-index is read and saved.

Sub-index 00h	Name	Highest Sub-index Supported				Data Structure	-	Data Type	Uint8
	Accessibility	RO	Mapping	NO	Relevant Mode	-	Data Range	5	Factory Default	5

Sub-index 1-5h	Name	Consumer Heartbeat Time				Data Structure	-	Data Type	Uint32
	Accessibility	RW	Mapping	NO	Relevant Mode	-	Data Range	Uint 32	Factory Default	0

Index 1017h	Name	Producer Heartbeat Time				Data Structure	VAR	Data Type	Uint16
	Accessibility	RW	Mapping	NO	Relevant Mode	-	Data Range	Uint 16	Factory Default	0

The unit is ms.

Index 1018h	Name	Identity Object				Data Structure	REC	Data Type	Related to individual
	Accessibility	RO	Mapping	NO	Relevant Mode	-	Data Range	-	Factory Default	-

Sub-index 00h	Name	Highest Sub-index Supported				Data Structure	-	Data Type	Uint8
	Accessibility	RO	Mapping	NO	Relevant Mode	-	Data Range	3	Factory Default	3

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

Unique ID allocated by the CiA organization.

Sub-index 02h	Name	Product Code				Data Structure	-	Data Type	Uint32
	Accessibility	RO	Mapping	NO	Relevant Mode	-	Data Range	Uint 32	Factory Default	0xD0107

The product series and models of Inovance device codes and electronic labels are associated. The mapping between them is as follows:

31 16	15 0
Product Series	Product Model

MLB LSB

6

Object Dictionary

Sub-index 3h	Name	Revision Number				Data Structure	-	Data Type	Uint32
	Accessibility	RO	Mapping	NO	Relevant Mode -	Data Range	Uint 32	Factory Default	0x1920001

The parameter is associated with the software version (100Ah). Its meaning is as follows:

31 16	15 0
Primary revision	Secondary revision

MLB LSB

Main revisions are based on the number 0x192. Each time when the code is updated, the next revision number accumulates upward.

Index 1029h	Name	Error Behavior					Data Structure	ARR	Data Type	Uint8
	Accessibility	RW	Mapping	NO	Relevant Mode	-	Data Range	-	Factory Default	-

Status control over automatic steering required by the NMT during CANopen communication when errors of different types occur. Based on different values, different status of NMT steering is provided.

Value	Meaning
0	The current operating status is switched to the pre-operation status.
1	The current status is maintained.
2	Switch to the stop status.
Other	Reserved

The IS620P servo drive only supports automatic NMT conversion in the case of communication failure.

Sub-index 00h	Name	Highest Sub-index Supported					Data Structure	-	Data Type	Uint8
	Accessibility	RO	Mapping	NO	Relevant Mode	-	Data Range	1	Factory Default	1

Sub-index 01h	Name	Communication Error					Data Structure	-	Data Type	Uint8
	Accessibility	RW	Mapping	NO	Relevant Mode	-	Data Range	Uint 8	Factory Default	0

Communication errors include: NMT error control timeout, PDO length error, and bus separation.

Index 1200h	Name	SDO Server Parameter					Data Structure	REC	Data Type	SDO parameter
	Accessibility	RO	Mapping	NO	Relevant Mode	-	Data Range	-	Factory Default	-

The default SDO always exists and is a read-only constant.

6

Object Dictionary

Sub-index 00h	Name	Highest Sub-index Supported				Data Structure	-	Data Type	Uint8
	Accessibility	RO	Mapping	NO	Relevant Mode	-	Data Range	2	Factory Default	2

Sub-index 01h	Name	COB-ID (COB-ID Client → Server(rx))				Data Structure	-	Data Type	Uint32
	Accessibility	RO	Mapping	NO	Relevant Mode	-	Data Range	Uint 32	Factory Default	0x600 + Node_ID

Sub-index 02h	Name	COB-ID (COB-ID Server → Client(tx))				Data Structure	-	Data Type	Uint32
	Accessibility	RO	Mapping	NO	Relevant Mode	-	Data Range	Uint 32	Factory Default	0x580 + Node_ID

Index 1400H to 1403h	Name	RPDO Communication Parameter				Data Structure	REC	Data Type	PDO parameter
	Accessibility	RW	Mapping	NO	Relevant Mode	-	Data Range	-	Factory Default	-

Sub-index 00h	Name	Highest Sub-index Supported				Data Structure	-	Data Type	Uint8
	Accessibility	RO	Mapping	NO	Relevant Mode	-	Data Range	2 to 6	Factory Default	2

Sub-index 01h	Name	COB-ID Used by RPDO				Data Structure	-	Data Type	Uint32
	Accessibility	RW	Mapping	NO	Relevant Mode	-	Data Range	Uint 32	Factory Default	See below.

Only the highest bit can be changed. When the highest bit is 0, the PDO is valid; when the highest bit 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	Transmission Type				Data Structure	-	Data Type	Uint8
	Accessibility	RW	Mapping	NO	Relevant Mode -	Data Range	Uint 8	Factory Default	255

The parameter value can be modified only when the PDO is invalid.

Different values indicate different PDO transmission types. See the table below:

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

Index 1600h to 1603h	Name	RPDO Mapping Parameter					Data Structure	REC	Data Type	Mapping parameter of RPDO
	Accessibility	RW	Mapping	NO	Relevant Mode	-	Data Range	-	Factory Default	-

The 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 instead of bits is supported.

Sub-index 00h	Name	Number of Mapped Application Objects in PDO					Data Structure	-	Data Type	Uint8
	Accessibility	RW	Mapping	NO	Relevant Mode	-	Data Range	0 to 8	Factory Default	-

When 0 is written, the mapping objects of other sub-indexes are cleared.

Sub-index 1-8h	Name	Application Object					Data Structure	-	Data Type	Uint32
	Accessibility	RW	Mapping	NO	Relevant Mode	-	Data Range	Uint 32	Factory Default	-

The indexes and sub-indexes of mapping objects must exist in the object dictionary list. The attribute of mapping objects is readable and the objects can be mapped.

Sub-indexes are written in the following format:

31 16	15 8	7 0
Index	Sub-index	Object Length

MLB LSB

Default mapping content of an RPDO

1) RPDO1:

Sub-index	Value	Meaning
0	1	One object is mapped.
1	0x60400010	Command word

2) RPD02:

Sub-index	Value	Meaning
0	2	Two objects are mapped.
1	0x60400010	Command word
2	0x60600008	Running mode selection

3) RPD03:

Sub-index	Value	Meaning
0	2	Two objects are mapped.
1	0x60400010	Command word
2	0x607A0020	Target position

4) RPD04:

Sub-index	Value	Meaning
0	2	Two objects are mapped.
1	0x60400010	Command word
2	0x60FF0020	Target velocity

Index 1800h to 1803h	Name	TPDO Communication Parameter					Data Structure	REC	Data Type	Communication parameter of PDO
	Accessibility	RW	Mapping	NO	Relevant Mode	-	Data Range	-	Factory Default	-

Sub-index 00h	Name	Highest Sub-index Supported					Data Structure	-	Data Type	Uint8
	Accessibility	RO	Mapping	NO	Relevant Mode	-	Data Range	2 to 6	Factory Default	5

Sub-index 01h	Name	COB-ID Used by TPDO					Data Structure	-	Data Type	Uint32
	Accessibility	RW	Mapping	NO	Relevant Mode	-	Data Range	Uint 32	Factory Default	See below.

Only the highest bit and the second highest bit can be modified.

When the highest bit is 0, the PDO is valid; when the highest bit is 1, the PDO is invalid.

The second highest bit indicates whether to support a remote frame in triggering the PDO. Because the IS620P servo drive does not support the function, the bit is meaningless. It is recommended that the bit is set to 1, which indicates that a remote frame is not allowed to trigger the PDO.

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	Transmission Type				Data Structure	-	Data Type	Uint8
	Accessibility	RW	Mapping	NO	Relevant Mode	-	Data Range	Uint 8	Factory Default	255

The parameter value can be modified only when the PDO is invalid. Different values indicate different PDO transmission types. See the table 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
	Accessibility	RW	Mapping	NO	Relevant Mode	-	Data Range	Uint 16	Factory Default	0

The parameter value can be modified only when the PDO is invalid.

The unit is 100 us. When the parameter is set to 0, the inhabit time is invalid.

Sub-index 05h	Name	Event Timer					Data Structure	-	Data Type	Uint16
	Accessibility	RW	Mapping	NO	Relevant Mode	-	Data Range	Uint 16	Factory Default	0

The parameter value can be modified only when the PDO is invalid.

The unit is 1 ms. When the parameter is set to 0, the event timer is invalid.

Index 1A00h-1A03h	Name	TPDO Mapping					Data Structure	REC	Data Type	PDO mapping parameter
	Accessibility	RW	Mapping	NO	Relevant Mode	-	Data Range	-	Factory Default	-

The 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 instead of bits is supported.

Sub-index 00h	Name	Number of Mapped Application Objects in TPDO					Data Structure	-	Data Type	Uint8
	Accessibility	RW	Mapping	NO	Relevant Mode	-	Data Range	0 to 8	Factory Default	-

When 0 is written, the mapping objects of other sub-indexes are cleared.

Sub-index 1-8h	Name	Application Object					Data Structure	-	Data Type	Uint32
	Accessibility	RW	Mapping	NO	Relevant Mode	-	Data Range	Uint 32	Factory Default	-

The indexes and sub-indexes of mapping objects must exist in the object dictionary list. The attribute of mapping objects is readable and the objects can be mapped.

Sub-indexes are written in the following format:

31 16	15 8	7 0
Index	Sub-index	Object Length

MLB LSB

Default mapping content of an TPDO:

1） TPD01：

Sub-index	Value	Meaning
0	1	One object is mapped.
1	0x60410010	Status word

2） TPD02：

Sub-index	Value	Meaning
0	2	Two objects are mapped.
1	0x60410010	Status word
2	0x60610008	Current running mode

3） TPD03：

Sub-index	Value	Meaning
0	2	Two objects are mapped.
1	0x60410010	Status word
2	0x60640020	Current position

4） TPD04：

Sub-index	Value	Meaning
0	2	Two objects are mapped.
1	0x60410010	Status word
2	0x606C0020	Current speed

6.5.2 Details of Parameters Defined by the Manufacturer

For parameters that are the same as functions of the IS620P servo drive, see the *IS620P Series Servo Design and Maintenance User Manual*. This section lists only parameters whose functions are changed.

Index 2005h	Name	Position Control Parameters					Data Structure	ARR	Data Type	Uint16
	Accessibility	-	Mapping	-	Relevant Mode	-	Data Range	OD data range	Factory Default	OD default value
It sets position control parameters.										

Sub-index 3Eh	Name	Position window unit set					Data Structure	-	Data Type	Uint16
	Accessibility	RW	Mapping	YES	Relevant Mode	All	Data Range	0 to 1	Factory Default	0
It sets the unit of the position window in 6067h.										

Value	Unit
0	Encoder unit
1	Reference unit

Index 2007h	Name	Torque Control Parameters					Data Structure	ARR	Data Type	Uint16
	Accessibility	-	Mapping	-	Relevant Mode	-	Data Range	OD data range	Factory Default	OD default value
It sets torque control parameters.										

Sub-index 12h	Name	Speed limit source					Data Structure	-	Data Type	Uint16
	Accessibility	RW	Mapping	YES	Relevant Mode	All	Data Range	0 to 3	Factory Default	0
It sets the unit of the position window in 6067h.										

Value	Selection of speed limit source
0	Internal speed limit
1	V-Lmt is used as input of external speed limit.
2	V-SEL is used to select speed limit 1 or speed limit 2.
3	607F is used for the speed limit.

Index 200Ah	Name	Fault and Protection Parameters				Data Structure	ARR	Data Type	Uint16
	Accessibility	-	Mapping	-	Relevant Mode	-	Data Range	OD Data Range	Factory Default	OD Default Value

It sets the fault and protection parameters.

Sub-index 02h	Name	Absolute Position Limit Set				Data Structure	-	Data Type	Uint16
	Accessibility	RW	Mapping	YES	Relevant Mode	All	Data Range	0 to 2	Factory Default	0

It sets the conditions for enabling absolute position limit.

Value	Communication rate
0	Disable
1	Enable
2	Enable software position limit after homing

Index 200Ch	Name	Communication Parameters				Data Structure	ARR	Data Type	Uint16
	Accessibility	-	Mapping	-	Relevant Mode	-	Data Range	OD Data Range	Factory Default	OD Default Value

It sets communication parameters.

Sub-index 01h	Name	Axis Address				Data Structure	-	Data Type	Uint16
	Accessibility	RW	Mapping	NO	Relevant Mode	All	Data Range	1 to 127	Factory Default	1

It sets the axis address of the servo drive.

When multiple servo drives are connected for networking, each drive can have only one unique address; otherwise, communication becomes abnormal or communication fails.

Sub-index 09h	Name	CAN Communication Baud Rate				Data Structure	-	Data Type	Uint16
	Accessibility	RW	Mapping	NO	Relevant Mode	All	Data Range	0 to 7	Factory Default	5

It sets the communication rate between the servo drive and the host computer during CAN communication.

The communication rate set in the servo drive must be the same as that in the host computer. Otherwise, communication fails.

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

An appropriate communication rate should be set according to actual use conditions (communication distance and communication data amount).

Sub-index 0Eh	Name	Update Function Code Values Written via Communication to EEPROM				Data Structure	-	Data Type	Uint16
	Accessibility	RW	Mapping	YES	Relevant Mode	All	Data Range	0 to 3	Factory Default	0

It sets whether parameters modified in communication are stored in the EEPROM.

If parameters need to be stored in the EEPROM, 200C-0EH must be set before parameters are modified.

Value	Name	Description
0	Not store	-
1	Store parameters in 2000h.	Parameters in 2000h refer to functional codes of the IS620P servo drive. When 200C-0Eh is set to 1, parameters modified in RS232/485 communication can also be stored in the EEPROM.
2	Store parameters in 6000h.	Parameters in 6000h refer to object dictionaries in the DSP402 area.
3	Store all parameters.	Parameters in 2000h and 6000h are stored in the EEPROM.

6.5.3 Details of Parameters Defined by Sub-protocols

Index 603Fh	Name	Error Code				Data Structure	VAR	Data Type	Uint16
	Accessibility	RO	Mapping	TPDO	Relevant Mode	All	Data Range	0 to 65535	Factory Default	-

When an error described in the DSP402 sub-protocol occurs in the servo drive, 603Fh is the same as the description in DSP402. For details, see Section 5.1.

When an error specified by the user occurs in 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.

The value of 203Fh is in hexadecimal; the high 16 bits indicate the manufacturer internal fault code, and the low 16 bits indicate the manufacturer external fault code.

Index 6040h	Name	Control Word				Data Structure	VAR	Data Type	Uint16
	Accessibility	RW	Mapping	YES	Relevant Mode	All	Data Range	0 to 65535	Factory Default	0

It controls the state machine of the servo drive.

bit	Name	Description
0	Servo ready	0: Disabled 1: Enabled
1	Switch on	0: Disabled 1: Enabled
2	Quick stop	0: Enabled 1: Disabled
3	Running	0: Disabled 1: Enabled
4 to 6		Related to drive modes.
7	Fault reset	Fault reset is implemented for faults and warnings that can be reset. The rising edge of bit7 is valid. If bit7 is 1, other control commands are invalid.
8	Halt	Not supported
9 to 10	NA	Reserved
11 to 15	Defined by the manufacturer	Reserved

◆ Note:

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

■ The meanings of bit0 to bit3 and bit7 are the same in each mode of the servo drive. The servo drive switches to the preset status according to the CiA402 state machine only when control words are sent in sequence. Each command corresponds to one status.

■ The meanings of bit4 to bit6 vary according to the drive modes. For details, see control commands in different modes.

Index 6041h	Name	Status Word				Data Structure	VAR	Data Type	Uint16
	Accessibility	RO	Mapping	TPDO	Relevant Mode	All	Data Range	0 to 65535	Factory Default	-

It indicates the state of the servo drive.

bit	Name	Description
0	No fault	-
1	Wait to enable servo	-
2	Running	-
3	Fault	-
4	Switch on	-
5	Quick stop	-
6	Servo ready	-
7	Warning	-
8	Defined by the manufacturer	Reserved
9	Remote control	0: In a mode other than CANopen mode, some IS620P standard software functions can be used. 1: CANopen remote control mode
10	Target reached	0: The target position is not reached. 1: The target position is reached.
11	Software internal position limit	0: The position reference or feedback does not reach the software internal position limit. 1: When the position reference or feedback reaches the internal position limit, the servo drive runs by using the position limit as the target position in a position mode and stops after the motor reaches the limit. If a reverse displacement reference is entered, the motor exits the position limit status and the bit is cleared.
12 to 13		Related to drive modes.
14	NA	Reserved
15	Homing completed	0: Homing is not performed or complete. 1: Homing is complete and the reference point is found.

◆ Note:

■ All bits in the control word work together to show the current status of the servo drive. One bit is meaningless if it is set separately.

■ The meanings of bit0 to bit9 are the same in each mode of the servo drive. After control commands in 6040h are sent in sequence, the servo drive shows a certain status.

■ The meanings of bit12 to bit13 vary according to the drive modes. For details, see control commands in different modes.

■ The meanings of bit10, bit11, and bit15 are the same in each mode of the servo drive and indicate the status after a control mode is implemented.

Index 605Ah	Name	Quick Stop Option Code				Data Structure	VAR	Data Type	Int16
	Accessibility	RW	Mapping	NO	Relevant Mode	All	Data Range	0 to 7	Factory Default	2

Set the quick stop mode.

Value	Stop Mode
0	Free stop. The free running status is maintained.
1	Ramp stop based on the deceleration set in 6084h (hm: 609Ah). After stop, the free running status is maintained.
2	Ramp stop based on the deceleration set in 6085h. After stop, the free running status is maintained.
3	Torque stop for emergency stop set in 2007-10h. After stop, the free running status is maintained.
4	NA
5	Ramp stop based on the deceleration set in 6084h (hm: 609Ah). After stop, the position locked status is maintained.
6	Ramp stop based on the deceleration set in 6085h. After stop, the position locked status is maintained.
7	Torque stop for emergency stop set in 2007-10h. After stop, the position locked status is maintained.

Index 6060h	Name	Modes of Operation				Data Structure	VAR	Data Type	Int8
	Accessibility	RW	Mapping	YES	Relevant Mode	All	Data Range	0 to 7	Factory Default	0

Select modes of operation:

bit	Description	Remarks
0	NA	Reserved
1	Profile position (PP) mode	For parameter settings, see "4.5 Profile Position Mode" on Page <?>.
2	NA	Reserved
3	Profile velocity (PV) mode	For parameter settings, see "4.8 Profile Velocity Mode" on Page <?>.
4	Profile torque (PT) mode	For parameter settings, see "4.9 Profile Torque Mode" on Page <?>.
5	NA	Reserved
6	Homing mode	For parameter settings, see "4.6 Homing Mode" on Page <?>.
7	IP mode	For parameter settings, see "4.7 Interpolated Position Mode" on Page <?>.

■ If an unsupported operation mode is selected through an SDO, a SDO error is returned.

■ If an unsupported drive mode is selected through a PDO, the change of the drive mode is invalid.

Index 6061h	Name	Modes of Operation Display					Data Structure	VAR	Data Type	Int8
	Accessibility	RO	Mapping	TPDO	Relevant Mode	All	Data Range	0 to 7	Factory Default	-

Display the actual operation mode:

bit	Description	Description
0	NA	Reserved
1	Profile position (PP) mode	For parameter settings, see "4.5 Profile Position Mode" on Page <?>.
2	NA	Reserved
3	Profile velocity (PV) mode	For parameter settings, see "4.8 Profile Velocity Mode" on Page <?>.
4	Profile torque (PT) mode	For parameter settings, see "4.9 Profile Torque Mode" on Page <?>.
5	NA	Reserved
6	Homing mode	For parameter settings, see "4.6 Homing Mode" on Page <?>.
7	IP mode	For parameter settings, see "4.7 Interpolated Position Mode" on Page <?>.

Index 6062h	Name	Position Demand Value					Data Structure	VAR	Data Type	Int32
	Accessibility	RO	Mapping	TPDO	Relevant Mode	pp/hm/ip	Data Range	-2^{31} to $(2^{31}-1)$	Factory Default	-

It indicates the real-time position reference (user unit).

Index 6063h	Name	Position Actual Value					Data Structure	VAR	Data Type	Int32
	Accessibility	RO	Mapping	TPDO	Relevant Mode	All	Data Range	-2^{31} to $(2^{31}-1)$	Factory Default	-

It indicates the absolute position feedback in real time.

Index 6064h	Name	Position Actual Value					Data Structure	VAR	Data Type	Int32
	Accessibility	RO	Mapping	TPDO	Relevant Mode	All	Data Range	-2^{31} to $(2^{31}-1)$	Factory Default	-

It indicates the absolute position in real time, in reference unit.

Position Actual Value (6064h) x Position Factor (6093h) = Position Actual Value (6063h)

Index 6065h	Name	Following Error Window				Data Structure	VAR	Data Type	Uint32
	Accessibility	RW	Mapping	YES	Relevant Mode	pp/hm/ip	Data Range	0 to (2^{32}-1)	Factory Default	3145728 p

It sets the position deviation threshold (user unit).

■ When the difference value between Position Demand Value (6062h) and Position Actual Value (6064h) exceeds ±6065h, Er.B00 (excessive position deviation) occurs.

■ When 6065h is set to 0xFFFFFFFF, the servo drive does not detect whether the position deviation is excessive. Use this setting with caution.

Index 6067h	Name	Position Window				Data Structure	VAR	Data Type	Uint32
	Accessibility	RW	Mapping	YES	Relevant Mode	pp/hm/ip	Data Range	0 to (2^{32}-1)	Factory Default	734 p

It sets the position window.

If the difference value between 6062h and 6064h is within ±6067h and the time reaches 6068h, the position is considered to be reached and bit10 of the status word 6041h is set to 1 in profile position mode.

This flag bit is valid only when the S-ON signal is valid in profile position mode.

Index 6068h	Name	Position Window Time				Data Structure	VAR	Data Type	Uint16
	Accessibility	RW	Mapping	YES	Relevant Mode	pp/hm/ip	Data Range	0 to 65535	Factory Default	0 ms

It sets the Position Window Time.

If the difference value between 6062h and 6064h is within ±6067h, and the time reaches 6068h, the position is considered to be reached and bit10 of the status word 6041h is set to 1 in profile position mode.

This flag bit is valid only when the S-ON signal is valid in profile position control mode.

Index 606Bh	Name	Velocity Demand Value				Data Structure	VAR	Data Type	Int32
	Accessibility	RO	Mapping	TPDO	Relevant Mode	All	Data Range	-2^{31} to (2^{31}-1)	Factory Default	-

It indicates the actual velocity reference.

In a position mode, 606Bh indicates the velocity reference corresponding to the position regulator.

In a velocity mode, 606Bh indicates the input reference of the speed regulator.

Index 606Ch	Name	Velocity Actual Value				Data Structure	VAR	Data Type	Int32
	Accessibility	RO	Mapping	TPDO	Relevant Mode	All	Data Range	-2^{31} to (2^{31}-1)	Factory Default	-

It indicates the velocity actual value.

Index 606Dh	Name	Velocity Window				Data Structure	VAR	Data Type	Uint16
	Accessibility	RW	Mapping	YES	Relevant Mode	pv	Data Range	0 to 65535	Factory Default	10 rpm

It sets the velocity threshold.

If the difference value between 60FFh and 606Ch is within ±606Dh and the time reaches 606Eh, the position is considered to be reached and bit10 of the status word 6041h is set to 1 in profile velocity mode.

This flag bit is valid only when the servo drive is enabled in profile velocity mode.

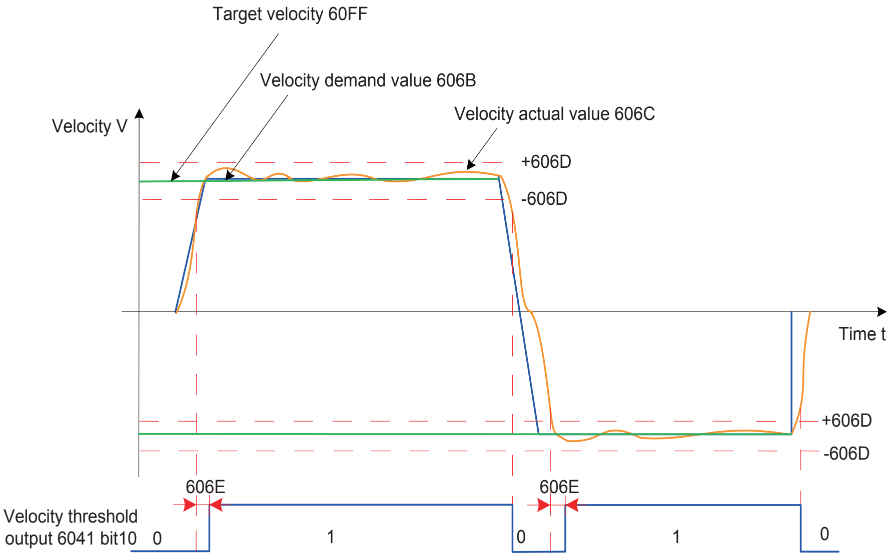
Index 606Eh	Name	Velocity Window Time				Data Structure	VAR	Data Type	Uint16
	Accessibility	RW	Mapping	YES	Relevant Mode	pv	Data Range	0 to 65535	Factory Default	0 ms

It sets the Velocity Window Time.

If the difference value between 60FFh and 606Ch is within ±606Dh and the time reaches 606Eh, the position is considered to be reached and bit10 of the status word 6041h is set to 1 in profile velocity mode.

This flag bit is valid only when the servo drive is enabled in profile velocity mode.

Figure 6-1 Velocity window



Index 606Fh	Name	Velocity Threshold				Data Structure	VAR	Data Type	Uint16
	Accessibility	RW	Mapping	YES	Relevant Mode	pv	Data Range	0 to 65535	Factory Default	10 rpm

It sets the threshold for determining whether the velocity is 0.

When 606Ch is within 606Fh and the time reaches the value set by 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 valid only in profile velocity mode.

This flag bit is unrelated to whether the servo drive is enabled.

Index 6070h	Name	Velocity Threshold Time				Data Structure	VAR	Data Type	Uint16
	Accessibility	RW	Mapping	YES	Relevant Mode	pv	Data Range	0 to 65535	Factory Default	0 ms

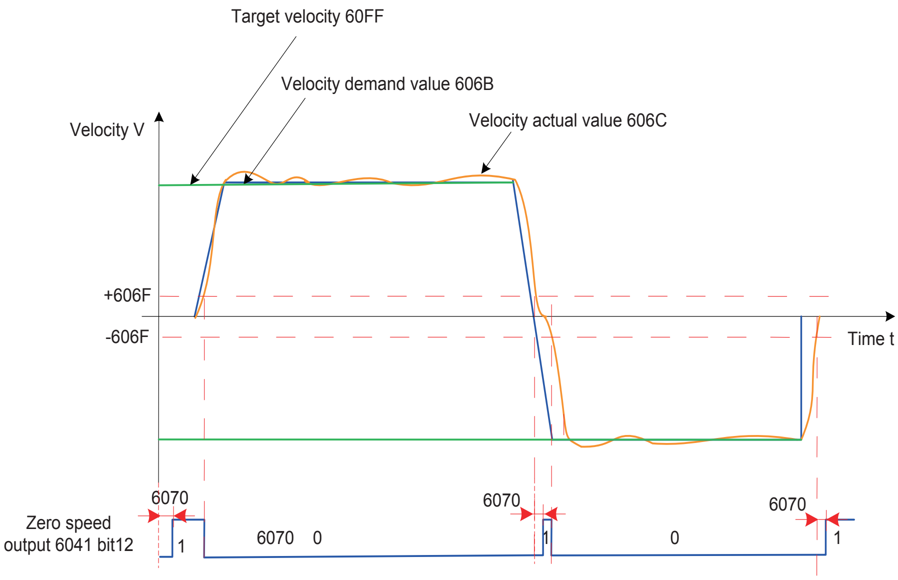
It sets the time window for determining whether the velocity is 0.

When 606Ch is within 606Fh and the time reaches the value set by 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 valid only in profile velocity mode.

This flag bit is unrelated to whether the servo drive is enabled.

Figure 6-2 Velocity threshold



Index 6071h	Name	Target Torque				Data Structure	VAR	Data Type	INTER16
	Accessibility	RW	Mapping	YES	Relevant Mode	PT CST	Data Range	0xEC78 to 0x1388 (Unit: 0.1%)	Factory Default	0x0000

It sets the target torque in 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
	Accessibility	RO	Mapping	TPDO	Relevant Mode	All	Data Range	- (Unit: 0.1%)	Factory Default	-

It displays the internal actual torque of the servo drive.

The value 100% corresponds to the rated motor torque.

Index 607Ah	Name	Target Position				Data Structure	VAR	Data Type	Int32
	Accessibility	RW	Mapping	YES	Relevant Mode	pp	Data Range	-2^{31} to $(2^{31}-1)$	Factory Default	0

It sets the target position of the servo drive in profile position mode.

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

Index 607Ch	Name	Home Offset				Data Structure	VAR	Data Type	Int32
	Accessibility	RW	Mapping	YES	Relevant Mode	All	Data Range	-2^{31} to $(2^{31}-1)$	Factory Default	0

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

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

■ The home offset has the following effect:

After homing is complete: position actual value 6064h = 607Ch.

■ If 607Ch is outside 607Dh (Software Absolute Limit), Er.D10 occurs (home offset setting error).

Index 607Dh	Name	Software Position Limit				Data Structure	ARR	Data Type	Int32
	Accessibility	RW	Mapping	YES	Relevant Mode	All	Data Range	OD Data Range	Factory Default	OD Default Value

It sets the minimum and maximum software absolute position limits.

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

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

■ This parameter is used to judge the absolute position. When the homing operation is not performed, this parameter is invalid.

■ The conditions of software position limit are set in the functional code H0A-01 (object dictionary 0x200A-02h).

0: No absolute software position limit

1: Valid absolute software position limit

2: Valid absolute software position limit after homing The absolute software position takes effect in the following conditions: The homing operation is complete and bit15 of the status word 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) occurs.

■ When the position reference or position feedback reaches the internal position limit, the servo drive runs by using the position limit as the target position in a position mode, stops after the motor reaches the limit, and prompts a limit fault. If a reverse displacement reference is entered, the motor exits the position limit status and the bit is cleared.

■ When the external DI limit switch and internal software position limit are valid at the same time, the limit status is determined by the external DI limit switch.

Sub-index 00h	Name	Number of Entries					Data Structure	-	Data Type	Uint8
	Accessibility	RO	Mapping	NO	Relevant Mode	-	Data Range	2	Factory Default	2

Sub-index 01h	Name	Min Software Position Limit					Data Structure	-	Data Type	Int32
	Accessibility	RW	Mapping	YES	Relevant Mode	-	Data Range	-2^{31} to $(2^{31}-1)$	Factory Default	-2^{31}p

It sets the minimum software position limit, relative to the mechanical zero.

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

Sub-index 02h	Name	Max Software Position Limit					Data Structure	-	Data Type	Int32
	Accessibility	RW	Mapping	YES	Relevant Mode	-	Data Range	-2^{31} to $(2^{31}-1)$	Factory Default	$(2^{31}-1)$ p

It sets 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
	Accessibility	RW	Mapping	YES	Data Range	OD Data Range	Factory Default	0

Set the polarity of position or velocity references.

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Position reference polarity	Speed reference polarity	Torque reference feature	NA	NA	NA	NA	NA

When Bit7 is 1, it indicates the position reference×(-1) reverses the motor in standard position mode or interpolated position mode.

When Bit6 is 1, it indicates the speed reference (60FFh)×(-1) reverses the motor in velocity mode.

When Bit5 is 1, it indicates the torque reference (6071h)×(-1) reverses the motor in torque mode.

NA: not defined

Index 607Fh	Name	Max Profile Velocity					Data Structure	VAR	Data Type	Uint32
	Accessibility	RW	Mapping	YES	Relevant Mode	All	Data Range	0 to $(2^{32}-1)$	Factory Default	6000 rpm

It sets the maximum running speed.

The set value takes effect when the velocity reference of the slave node changes.

Index 6081h	Name	Profile Velocity					Data Structure	VAR	Data Type	Uint32
	Accessibility	RW	Mapping	YES	Relevant Mode	pp	Data Range	0 to (2^{32}-1)	Factory Default	100 rpm

It sets the constant running speed of the displacement reference in profile position mode.

The set value takes effect after the slave node receives the displacement reference.

Index 6083h	Name	Profile Acceleration					Data Structure	VAR	Data Type	Uint32
	Accessibility	RW	Mapping	YES	Relevant Mode	pp/pv	Data Range	0 to (2^{32}-1)	Factory Default	100 rpm/ms

It sets the acceleration of the displacement reference in profile position mode.

The set value takes effect after the slave node receives the displacement reference.

■ By default, the object dictionary means that the acceleration (in the unit of rpm/ms) of the motor from 0 rpm to 1000 rpm. It is calculated in the following formula:

$$6083h(rpm/ms) = \frac{1000rpm}{Xms}$$

X: time used by the motor to decelerate from 0 rpm to 1000 rpm. For example, 10 rpm/ms means that X equals 100 ms, that is, the motor accelerates to 1000 rpm within 100 ms.

■ If the parameter is set 0, the value is forcibly changed into 1.

Index 6084h	Name	Profile Deceleration					Data Structure	VAR	Data Type	Uint32
	Accessibility	RW	Mapping	YES	Relevant Mode	pp/pv	Data Range	0 to (2^{32}-1)	Factory Default	100 rpm/ms

It sets the deceleration of the displacement reference in profile position mode.

The set value takes effect after the slave node receives the displacement reference.

■ By default, the object dictionary means that the deceleration (in the unit of rpm/ms) of the motor from 1000 rpm to 0 rpm. It is calculated in the following formula:

$$6084h(rpm/ms) = \frac{1000rpm}{Xms}$$

X: time used by the motor to decelerate from 1000 rpm to 0 rpm. For example, 100 rpm/ms means that X equals 10 ms, that is, the motor decelerates from 1000 rpm to 0 within 10 ms.

■ If the parameter is set 0, the value is forcibly changed into 1.

Index 6085h	Name	Quick Stop Deceleration				Data Structure	VAR	Data Type	Uint32
	Accessibility	RW	Mapping	YES	Relevant Mode	All	Data Range	0 to (2^{32}-1)	Factory Default	100 rpm/ms

It sets the deceleration when the quick stop command (6040h is set to 0x0002) and stop mode (605Ah is set to 2 or 5) are valid.

■ By default, the object dictionary means that the deceleration (in the unit of rpm/ms) of the motor from 1000 rpm to 0 rpm. It is calculated in the following formula:

$$6085h(rpm/ms) = \frac{1000rpm}{Xms}$$

X: time used by the motor to decelerate from 1000 rpm to 0 rpm. For example, 100 rpm/ms means that X equals 10 ms, that is, the motor decelerates from 1000 rpm to 0 within 10 ms.

■ If the parameter is set 0, the value is forcibly changed into 1.

Index 6086h	Name	Motion Profile Type				Data Structure	VAR	Data Type	Int16
	Accessibility	RW	Mapping	YES	Relevant Mode	pp/pv	Data Range	0	Factory Default	0

It sets the motion profile type of a motor position reference or speed reference.

0: Linear

Index 6087h	Name	Torque Slope				Data Structure	VAR	Data Type	UNSIGNED32
	Accessibility	RW	Mapping	YES	Relevant Mode	PT CST	Data Range	0x00000000 to 0xFFFFFFFF (Unit: 0.1%/s)	Factory Default	0xFFFFFFFF

It sets the acceleration of the torque reference in profile torque mode, that is, torque increment per second.

In profile torque or cyclic synchronous torque mode, if 605A (Quick stop option code) is set to 1, 2, 5 or 6 or 605D (Halt option code) is set to 1 or 2, the servo drive decelerates and stops according to the setting of 6087h.

If the value exceeds the torque reference limit, the limit is forcibly used.

If the parameter is set 0, the value is forcibly changed into 1.

Index 6098h	Name	Homing Method				Data Structure	VAR	Data Type	Int8
	Accessibility	RW	Mapping	YES	Relevant Mode	hm	Data Range	0 to 35	Factory Default	0

It selects the homing method.

Value	Description
1	Reverse homing is performed and the deceleration point is the reverse limit switch. The home is the motor Z signal. The falling edge of the reverse limit must be reached before the motor Z signal.
2	Forward homing is performed and the deceleration point is the forward limit switch. The home is the motor Z signal. The falling edge of the forward limit must be reached before the motor Z signal.
3	Forward homing is performed and the deceleration point is the home switch. The home is the motor Z signal. The falling edge of the forward limit on one side must be reached before the motor Z signal.
4	Reverse homing is performed and the deceleration point is the home switch. The home is the motor Z signal. The rising edge of the home limit on one side must be reached before the motor Z signal.
5	Reverse homing is performed and the deceleration point is the home switch. The home is the motor Z signal. The falling edge of the forward limit on one side must be reached before the motor Z signal.
6	Forward homing is performed and the deceleration point is the home switch. The home is the motor Z signal. The rising edge of the home limit on one side must be reached before the motor Z signal.
7	Forward homing is performed and the deceleration point is the home switch. The home is the motor Z signal. The falling edge of the forward limit on one side must be reached before the motor Z signal.
8	Forward homing is performed and the deceleration point is the home switch. The home is the motor Z signal. The rising edge of the home limit on one side must be reached before the motor Z signal.
9	Forward homing is performed and the deceleration point is the home switch. The home is the motor Z signal. The rising edge of the home limit on another side must be reached before the motor Z signal.
10	Forward homing is performed and the deceleration point is the home switch. The home is the motor Z signal. The falling edge of the home limit on another side must be reached before the motor Z signal.
11	Reverse homing is performed and the deceleration point is the home switch. The home is the motor Z signal. The falling edge of the home limit on one side must be reached before the motor Z signal.
12	Reverse homing is performed and the deceleration point is the home switch. The home is the motor Z signal. The rising edge of the home limit on one side must be reached before the motor Z signal.
13	Reverse homing is performed and the deceleration point is the home switch. The home is the motor Z signal on anther side of the home switch. The rising edge of the home limit on another side must be reached before the motor Z signal.
14	Reverse homing is performed and the deceleration point is the home switch. The home is the motor Z signal on anther side of the home switch. The falling edge of the home limit another side must be reached before the motor Z signal.
17 to 32	Similar to 1 to 14. However, the deceleration point overlaps with the home.
33	Reverse homing is performed and the home is the motor Z signal.
34	Forward homing is performed and the home is the motor Z signal.
35	The current position is used as the home.

When 6098h is set to 15, 16, 31 or 32, the parameter is meaningless and the servo drive does not perform any homing operation.

Index 6099h	Name	Homing Speeds					Data Structure	ARR	Data Type	Uint32
	Accessibility	RW	Mapping	YES	Relevant Mode	All	Data Range	OD Data Range	Factory Default	OD Default Value

It sets the two speeds used in homing mode:

■ Speed during search for switch

■ Speed during search for zero

Sub-index 00h	Name	Number of Entries					Data Structure	-	Data Type	Uint8
	Accessibility	RO	Mapping	NO	Relevant Mode	-	Data Range	2	Factory Default	2

Sub-index 01h	Name	Speed During Search for Switch					Data Structure	-	Data Type	Uint32
	Accessibility	RW	Mapping	YES	Relevant Mode	-	Data Range	0 to (2^{32}-1)	Factory Default	100 rpm

It sets the speed during search for the deceleration point signal. The speed can be set to a large value to prevent homing timeout due to long homing time.

Sub-index 02h	Name	Speed During Search for Zero					Data Structure	-	Data Type	Int32
	Accessibility	RW	Mapping	YES	Relevant Mode	-	Data Range	0 to (2^{32}-1)	Factory Default	10 rpm

It sets the speed (user speed unit) during search for the home signal. The parameter can be set to a small value to prevent overshoot during high-speed stop and large deviation of the stop position from the preset mechanical home.

Index 609Ah	Name	Homing Acceleration					Data Structure	VAR	Data Type	Uint32
	Accessibility	RW	Mapping	YES	Relevant Mode	hm	Data Range	0 to (2^{32}-1)	Factory Default	100 rpm/ms

It sets the acceleration during the homing operation.

The setting value take effect after homing is enabled.

■ By default, the object dictionary means that the acceleration (in the unit of rpm/ms) of the motor from 0 rpm to 1000 rpm. It is calculated in the following formula:

$$609Ah(rpm/ms) = \frac{1000rpm}{Xms}$$

X: time used by the motor to decelerate from 0 rpm to 1000 rpm. For example, 10 rpm/ms means that X equals 100 ms, that is, the motor accelerates to 1000 rpm within 100 ms.

Index 60C1h	Name	Interpolation Data Record					Data Structure	ARR	Data Type	Int32
	Accessibility	RW	Mapping	YES	Relevant Mode	ip	Data Range	OD Data Range	Factory Default	OD Default Value

It sets the displacement reference in interpolated position mode.

Sub-index 00h	Name	Number of Entries				Data Structure	-	Data Type	Uint8
	Accessibility	RO	Mapping	NO	Relevant Mode	-	Data Range	1	Factory Default	1

Sub-index 01h	Name	First Interpolation Point				Data Structure	-	Data Type	Int32
	Accessibility	RW	Mapping	YES	Relevant Mode	-	Data Range	-2^{31} to $(2^{31}-1)$	Factory Default	0

Interpolation displacement is an absolute displacement reference.

When the interpolated position mode is used, 60C1-1h mus be set to a synchronous PDO and the transmission type is set to 1.

Every time when the synchronization cycle is reached, the host computer sends a displacement reference to the slave node.

Index 60C2h	Name	Interpolation Time Period				Data Structure	ARR	Data Type	Uint8
	Accessibility	RW	Mapping	YES	Relevant Mode	ip	Data Range	OD Data Range	Factory Default	OD Default Value

It sets the interpolation period in interpolated position mode.

The IS620P servo drive supports the synchronization cycle in the range 1 ms to 20 ms. When a synchronization cycle beyond the range is set, the synchronization cycle is set to a limited value.

The synchronization period must be set when the servo drive stops running. If the servo driving is running, the setting does not take effect.

Sub-index 00h	Name	Number of Entries				Data Structure	-	Data Type	Uint8
	Accessibility	RO	Mapping	NO	Relevant Mode	-	Data Range	2	Factory Default	2

Sub-index 01h	Name	Interpolation Time Units				Data Structure	-	Data Type	Uint8
	Accessibility	RW	Mapping	YES	Relevant Mode	-	Data Range	1 to 20	Factory Default	1

It set the interpolation time units.

Sub-index 02h	Name	Interpolation Time Index				Data Structure	-	Data Type	Int8
	Accessibility	RO	Mapping	TPDO	Relevant Mode	-	Data Range	-3	Factory Default	-3

It sets the interpolation time index.

-3 indicates the time unit is ms. Therefore, the actual interpolation period (ms) is 60C2-01h.

6

Object Dictionary

Index 60C5h	Name	Max Profile Acceleration				Data Structure	VAR	Data Type	Uint32
	Accessibility	RW	Mapping	YES	Relevant Mode	All	Data Range	0 to (2^{32}-1)	Factory Default	1000 rpm/ms

It sets the maximum allowed deceleration in profile position mode, profile velocity mode, or homing mode.

The set value takes effect when the motor runs in accelerated mode next time.

■ By default, the object dictionary means that the maximum acceleration (in the unit of rpm/ms) of the motor from 0 rpm to 1000 rpm. It is calculated in the following formula:

$$60C5h(rpm/ms) = \frac{1000rpm}{Xms}$$

X: time used by the motor to decelerate from 0 rpm to 1000 rpm. For example, 1000 rpm/ms means that X equals 1 ms, that is, the motor accelerates to 1000 rpm within 1 ms.

■ If the parameter is set 0, the value is forcibly changed into 1.

Index 60C6h	Name	Max Profile Deceleration				Data Structure	VAR	Data Type	Uint32
	Accessibility	RW	Mapping	YES	Relevant Mode	All	Data Range	0 to (2^{32}-1)	Factory Default	1000 rpm/ms

It sets the maximum allowed acceleration in profile position mode, profile velocity mode, or homing mode.

The set value takes effect when the motor runs in decelerated mode next time.

■ By default, the object dictionary means that the maximum deceleration (in the unit of rpm/ms) of the motor from 1000 rpm to 0 rpm. It is calculated in the following formula:

$$60C6h(rpm/ms) = \frac{1000rpm}{Xms}$$

X: time used by the motor to decelerate from 10000 rpm to 0 rpm. For example, 1000 rpm/ms means that X equals 1 ms, that is, the motor decelerates from 1000 rpm to 0 within 1 ms.

■ If the parameter is set 0, the value is forcibly changed into 1.

Index 60F4h	Name	Following Error Actual Value				Data Structure	VAR	Data Type	Int32
	Accessibility	RO	Mapping	TPDO	Relevant Mode	pp/hm/ip	Data Range	-2^{31} to (2^{31}-1)	Factory Default	-

It indicates the real-time position deviation (in user position unit).

Index 60FCh	Name	Position Demand Value				Data Structure	VAR	Data Type	Int32
	Accessibility	RO	Mapping	TPDO	Relevant Mode	pp/hm/ip	Data Range	-2^{31} to (2^{31}-1)	Factory Default	-

It indicates the real-time position reference of the motor.

Position Demand Value (6062h) x Position Factor (6093h) = Position Demand Value (60FCh)

Index 60FDh	Name	Digital Input				Data Structure	VAR	Data Type	Uint32
	Accessibility	RO	Mapping	TPDO	Relevant Mode	All	Data Range	0 to (2^{32}-1)	Factory Default	-

It indicates whether the current DI terminal 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
Defined by the manufacturer (Not defined)	Reserved	Not defined	Home switch	Positive limit switch	Negative limit switch

Index 60FEh	Name	Digital Output				Data Structure	ARR	Data Type	Uint32
	Accessibility	RO	Mapping	TPDO	Relevant Mode	All	Data Range	OD Data Range	Factory Default	OD Default Value

It indicates whether the current DO terminal logic of the servo drive.

Sub-index 00h	Name	Number of Entries				Data Structure	-	Data Type	Uint8
	Accessibility	RO	Mapping	NO	Relevant Mode	-	Data Range	1	Factory Default	1

Sub-index 01h	Name	Physical Outputs				Data Structure	-	Data Type	Uint32
	Accessibility	RO	Mapping	TPDO	Relevant Mode	-	Data Range	0 to (2^{32}-1)	Factory Default	0

It indicates whether the current DO terminal 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
Defined by the manufacturer (Not defined)	Reserved	Brake output

Index 60FFh	Name	Target Velocity				Data Structure	VAR	Data Type	Int32
	Accessibility	RW	Mapping	YES	Relevant Mode	pv	Data Range	-2^{31} to (2^{31}-1)	Factory Default	0 rpm

It sets the user velocity in position profile mode.

Index 6502h	Name	Supported Drive Modes				Data Structure	VAR	Data Type	Uint32
	Accessibility	RO	Mapping	TPDO	Relevant Mode	-	Data Range	0 to 4294967295	Factory Default	0x0000006D

It indicates the supported drive modes.

bit	Description	Support 0: Not supported 1: Supported
0	Profile position (PP) mode	1
1	vl	0
2	Profile velocity (PV) mode	1
3	tq	1
4	NA	0
5	Homing (HM) mode	1
6	Interpolated position (IP) mode	1
7 to 15	NA	Reserved
16 to 31	Defined by the manufacturer	Reserved

If the CANopen device supports the object dictionary 6502h, you can learn the drive modes supported by the servo drive from 6502h.

Chapter 7 Application Cases

Chapter 7 Application Cases

This chapter describes specific operations based on position modes. For details, see "4.8 Profile Velocity Mode" on Page 122.

In a position mode, objects that are used as PDOs are allocated as follows:

Table 7-1 PDO allocation

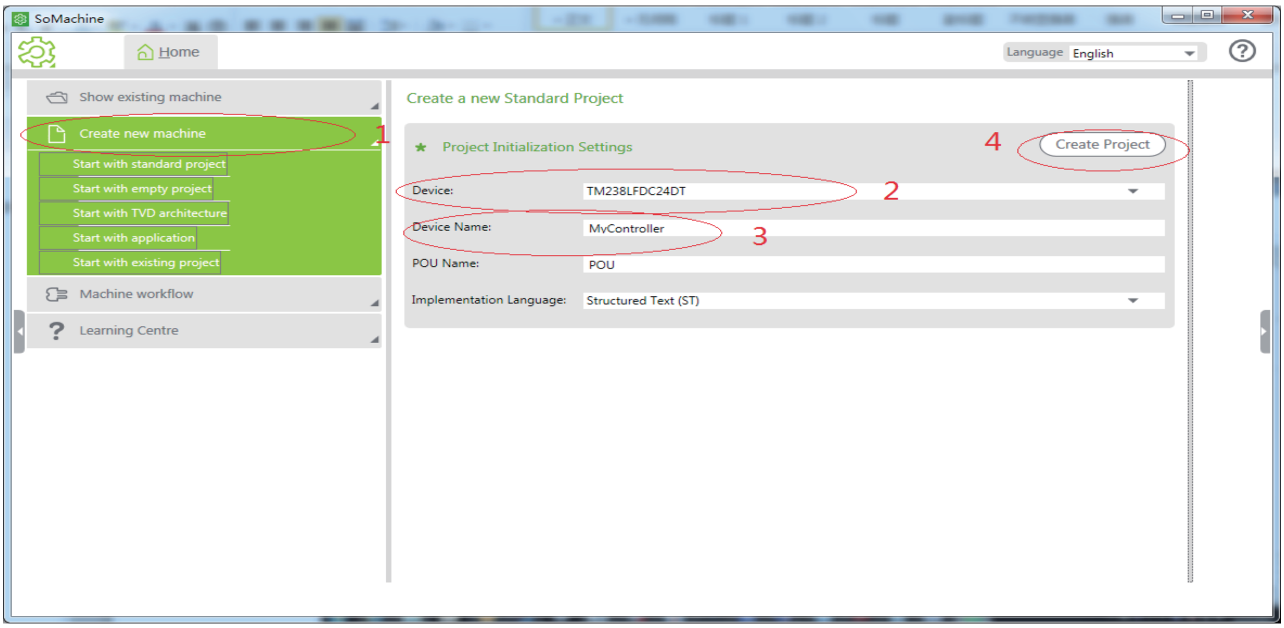
PDO	Object	Meaning	Bit Length
RPDO1	6040h-00h	Command word	Uint16
	6060h-00h	Modes of operation	Int8
RPDO2	6081h-00h	Speed reference	Uint32
	607Ah-00h	Position reference	Int32
TPDO1	6041h-00h	Status word	Uint16
	6061h-00h	Mode feedback	Int8
TPDO2	606Ch-00h	Speed feedback	Int32
	6064h-00h	Position feedback	Int32
TPDO3	200Bh-19h	Phase current feedback	Uint16

In an SDO, write 6083h (acceleration), 6084h (deceleration), and 605Ah (emergency stop mode).

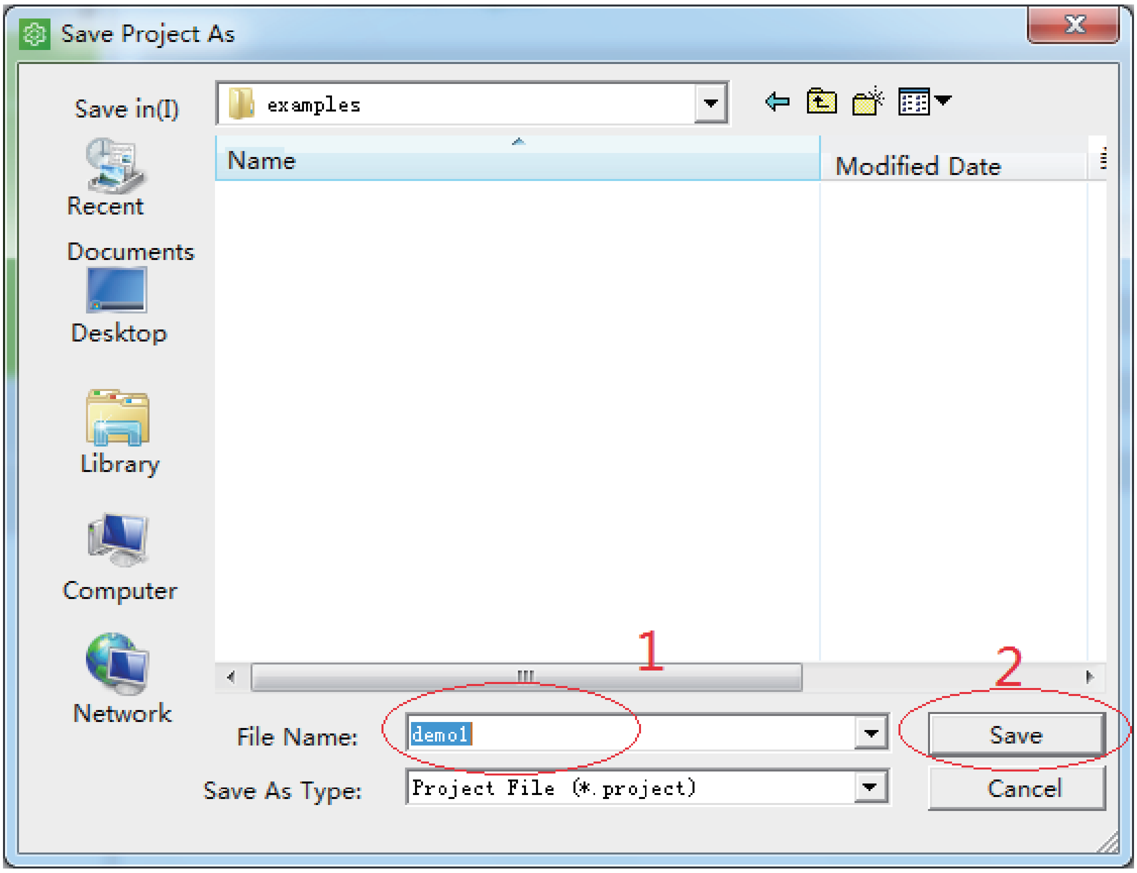
7.1 Connecting IS620P Servo Drive to Schneider 3S Master Node

SoMachine is Schneider 3S series master node background software. This section describes how to connect the IS620P servo drive to Schneider M238 master node.

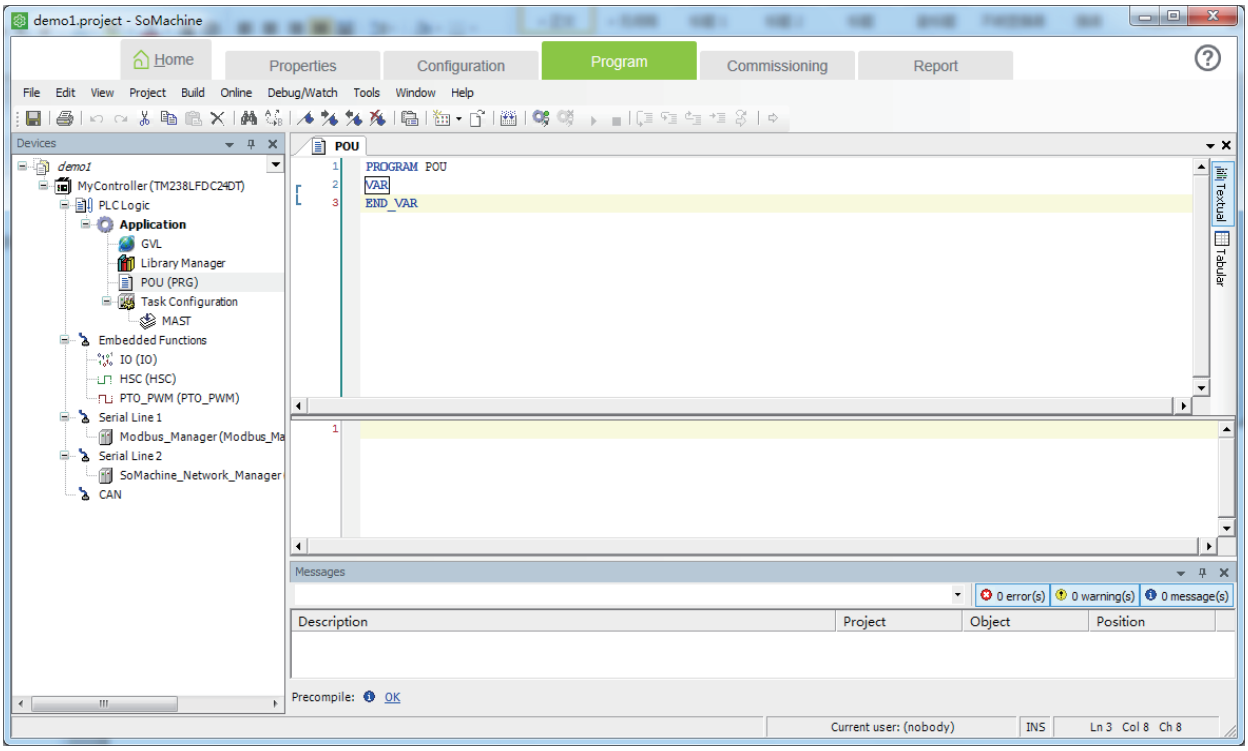
1) Start SoMachine and click **Create new machine** based on a standard project. Select a master device, for example, TM238LFDC24DT, modify the device name, and click **Create Project**, as shown in the figure below.



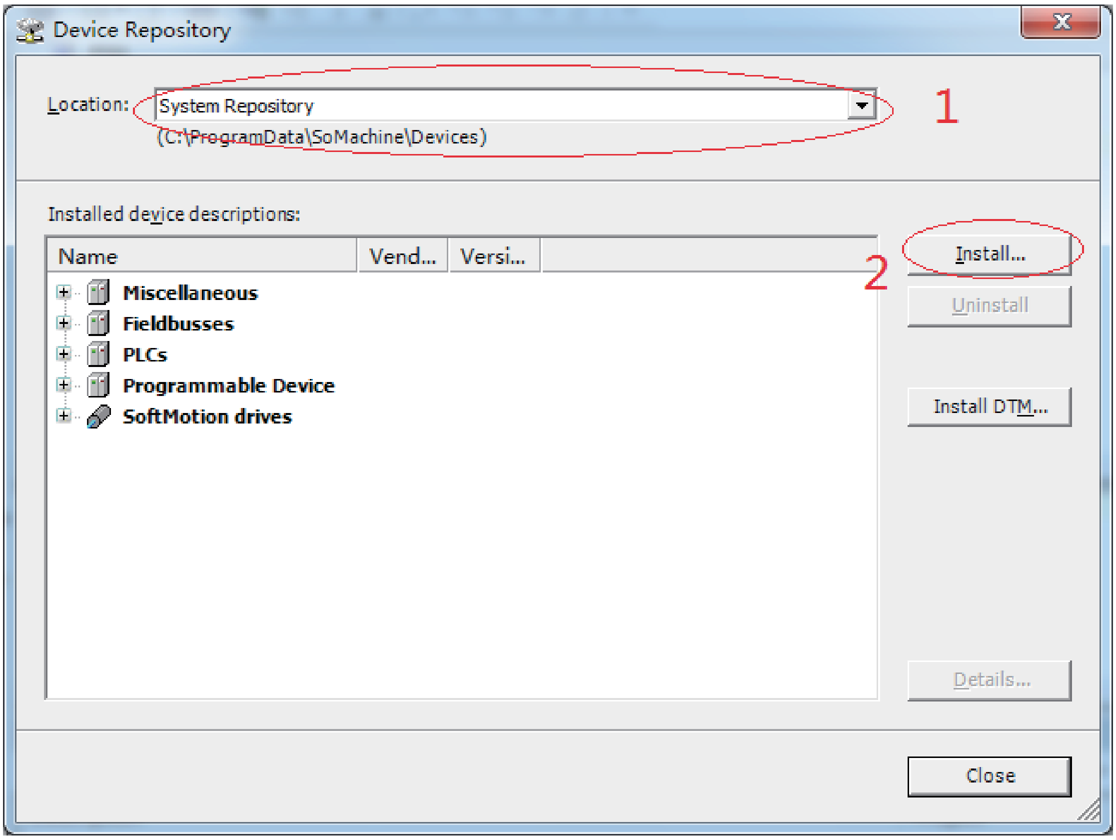
2) In the dialog box that is displayed, enter a proper file name and click **Save**.



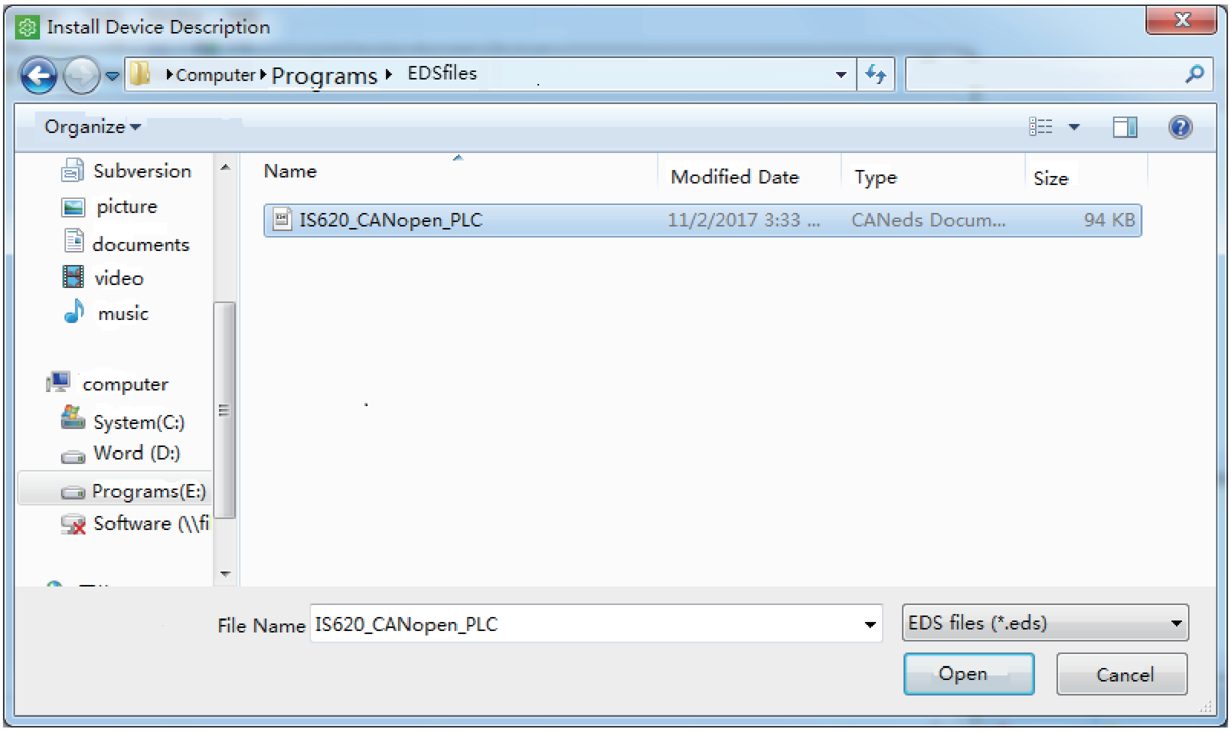
3) The following window is displayed:



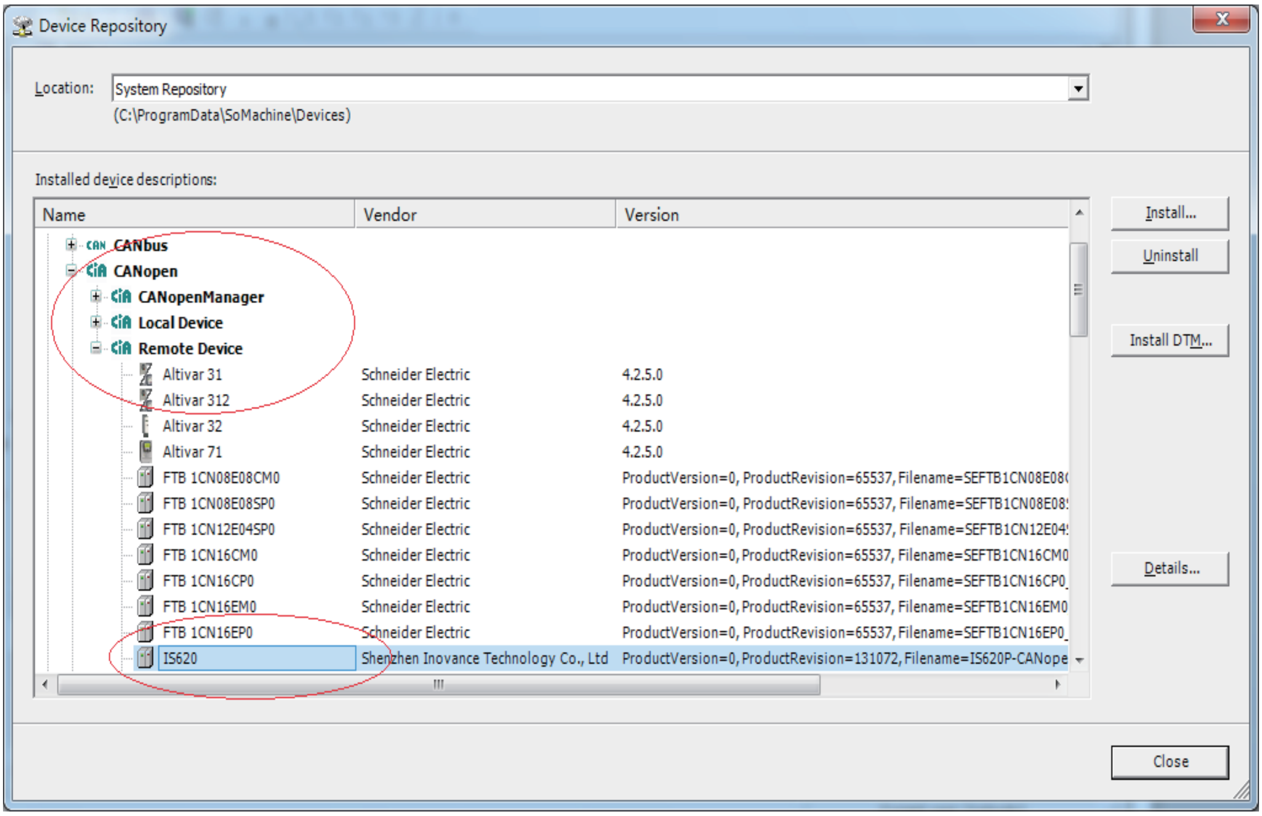
4) Choose **Tools** > **Device Repository** in the toolbar. The **Device Repository** dialog box is displayed. (If the EDS file is imported, steps 4 to 6 can be omitted.)



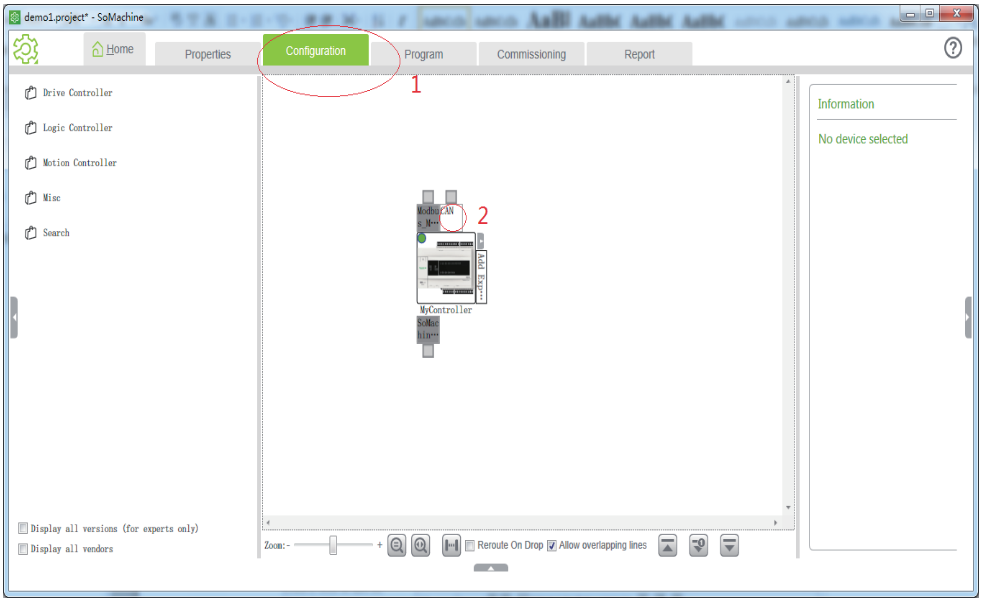
5) In the location bar, select **System Repository**, click **Install**, and select the storage location of the target EDS file.



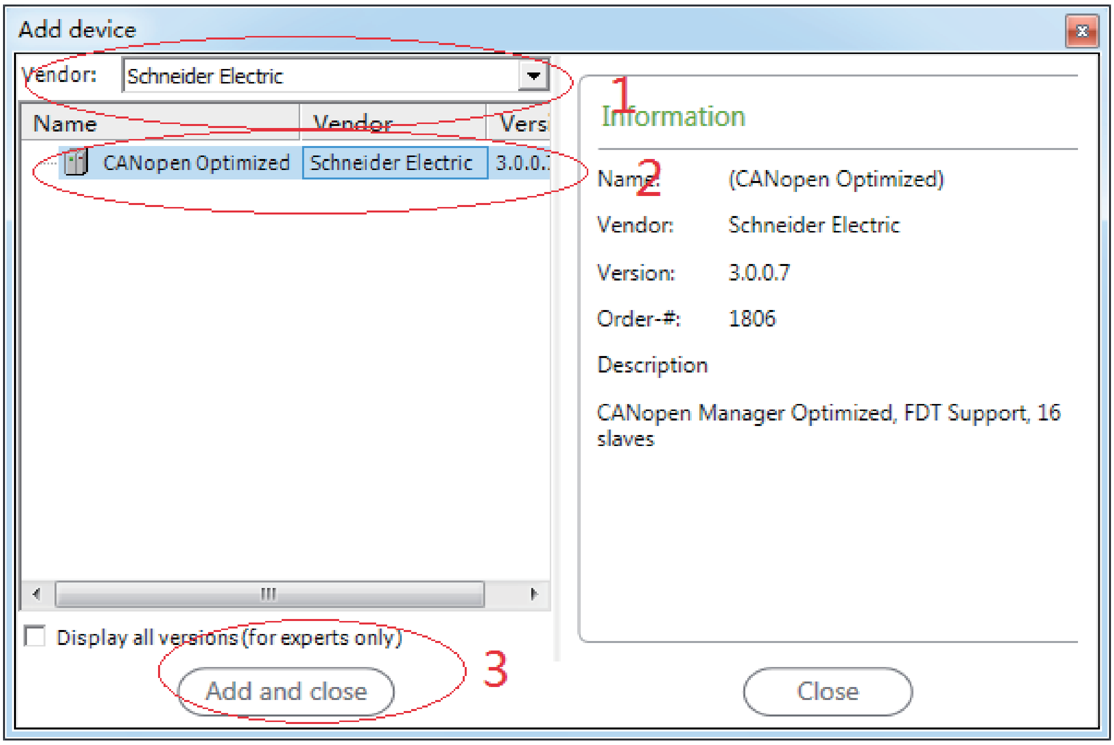
6) Click **Open**. The EDS file of the IS620P servo drive is imported into SoMachine. In the **Device Repository** dialog box, you can choose **Field Bus** > **CANopen** > **Remote Device** to view devices.



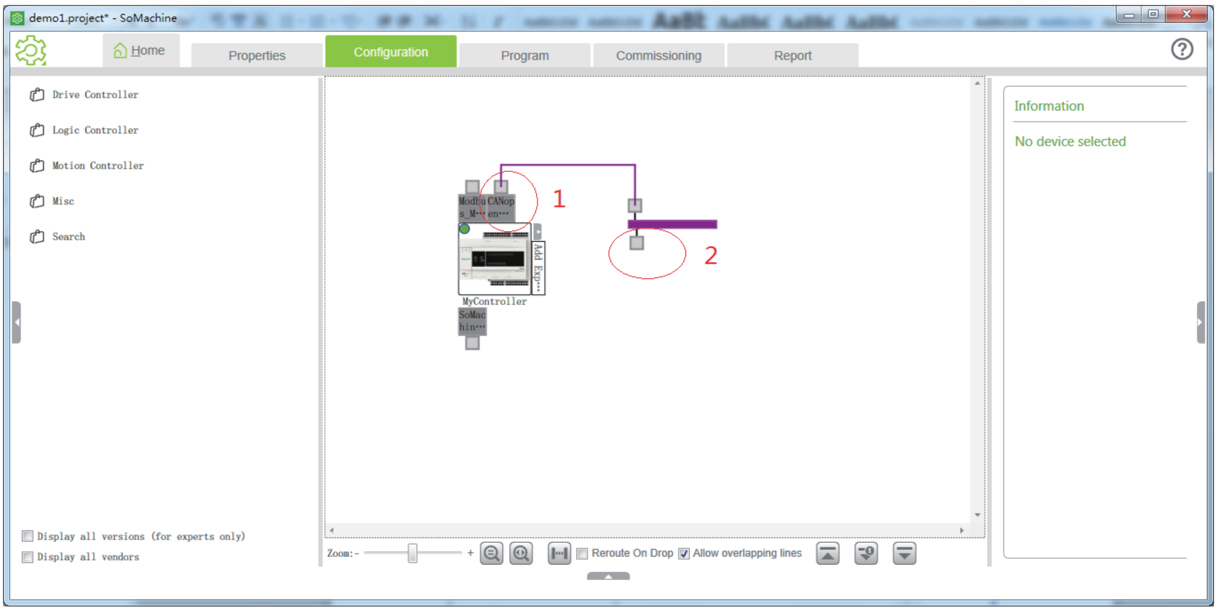
7) Close the **Device Repository** dialog box. Click **Configuration** in the window. Only the M238 master node can be seen. Click **CAN** of the master node.



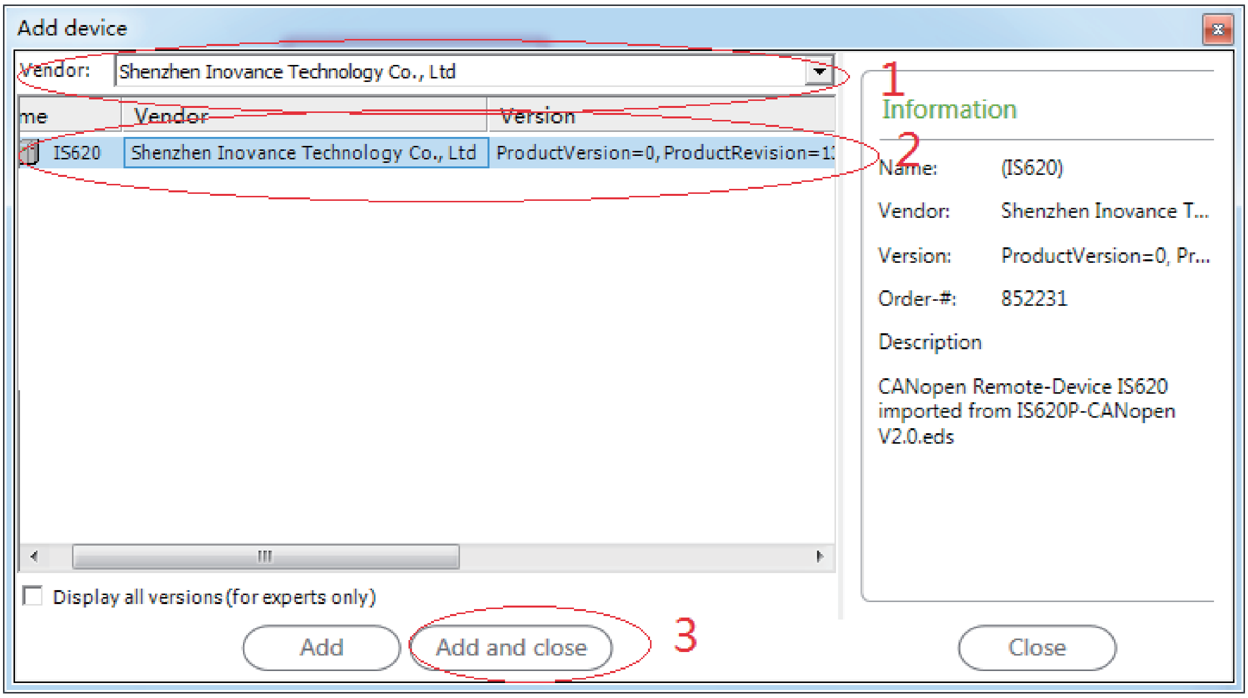
8) The **Add device** dialog box is displayed. Add a CANopen gateway, select **Schneider Electric** for **Supplier**, select **CANopen Optimized**, and click **Add and close**.



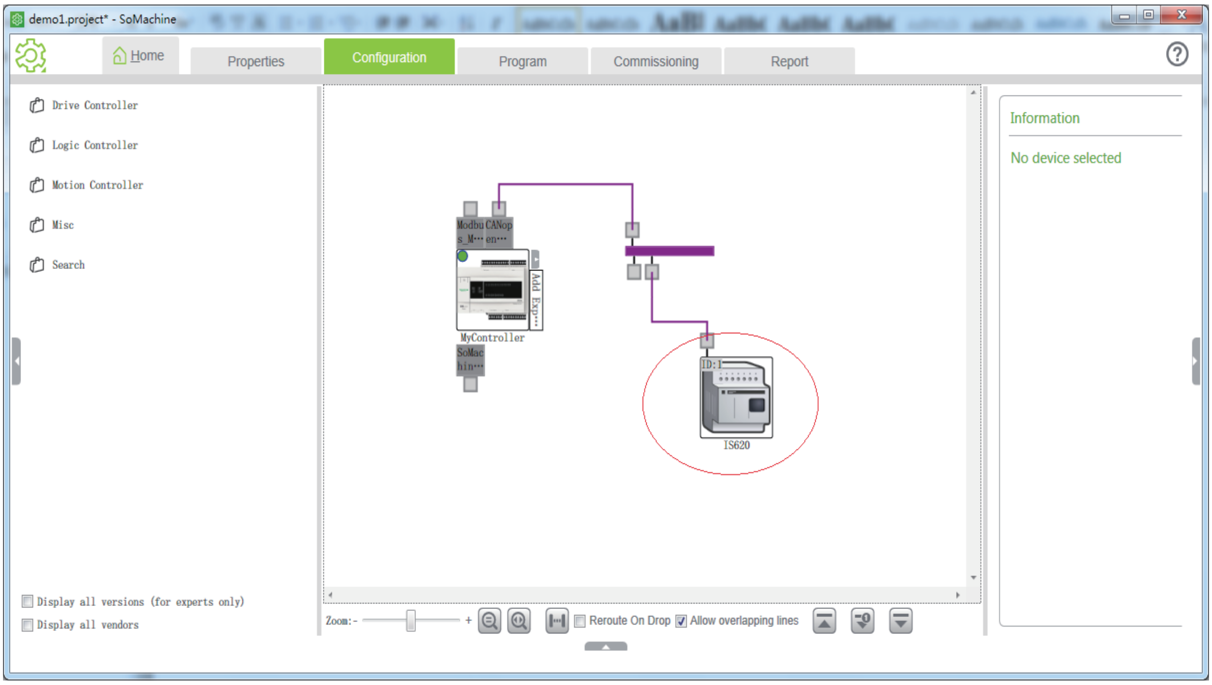
9) The CANopen gateway can be seen in the window. Click **2** in the figure.



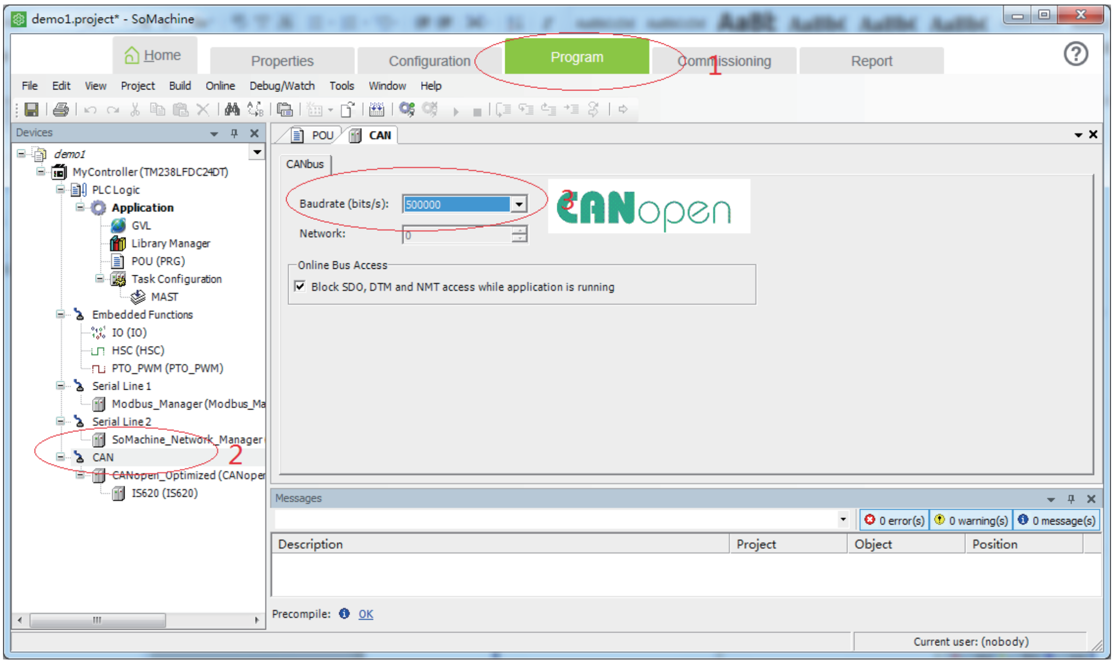
10) In the **Add Device** dialog box that is displayed, select **Inovance** for **Supplier**, select **IS620P Servo Drive** for the device, and click **Add and close**.



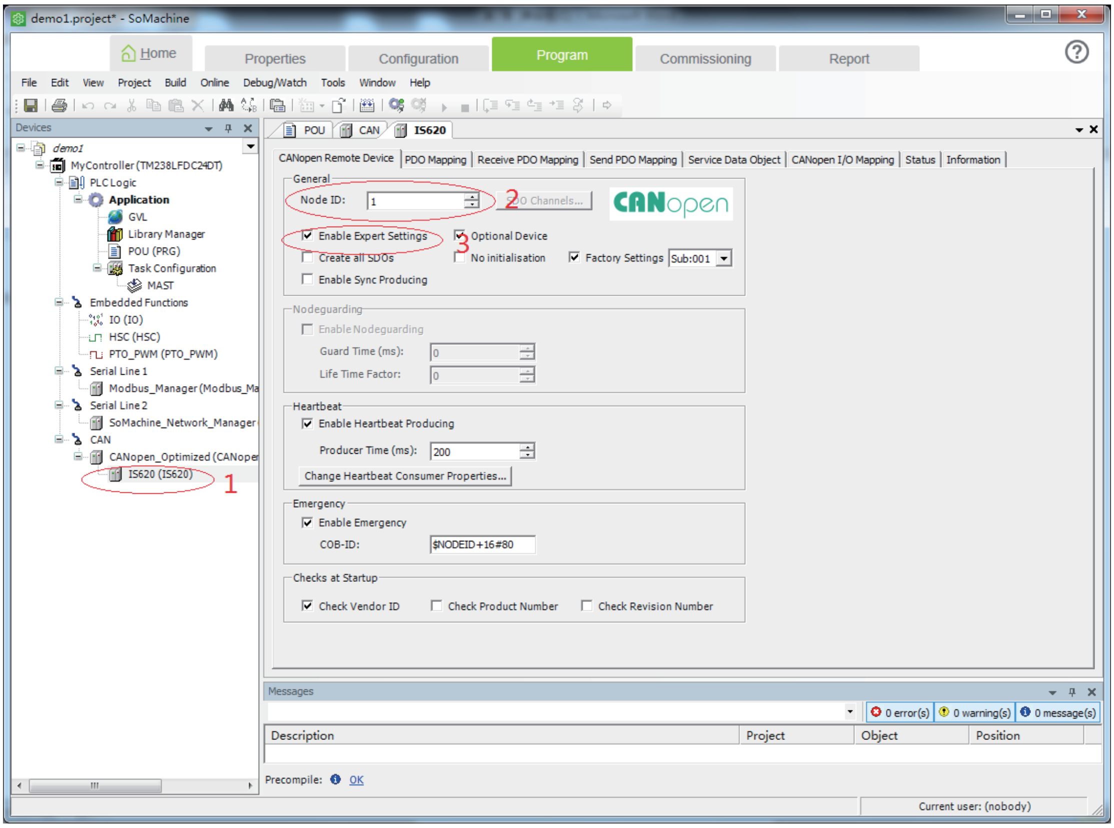
11) You can see that the IS620P drive is added.



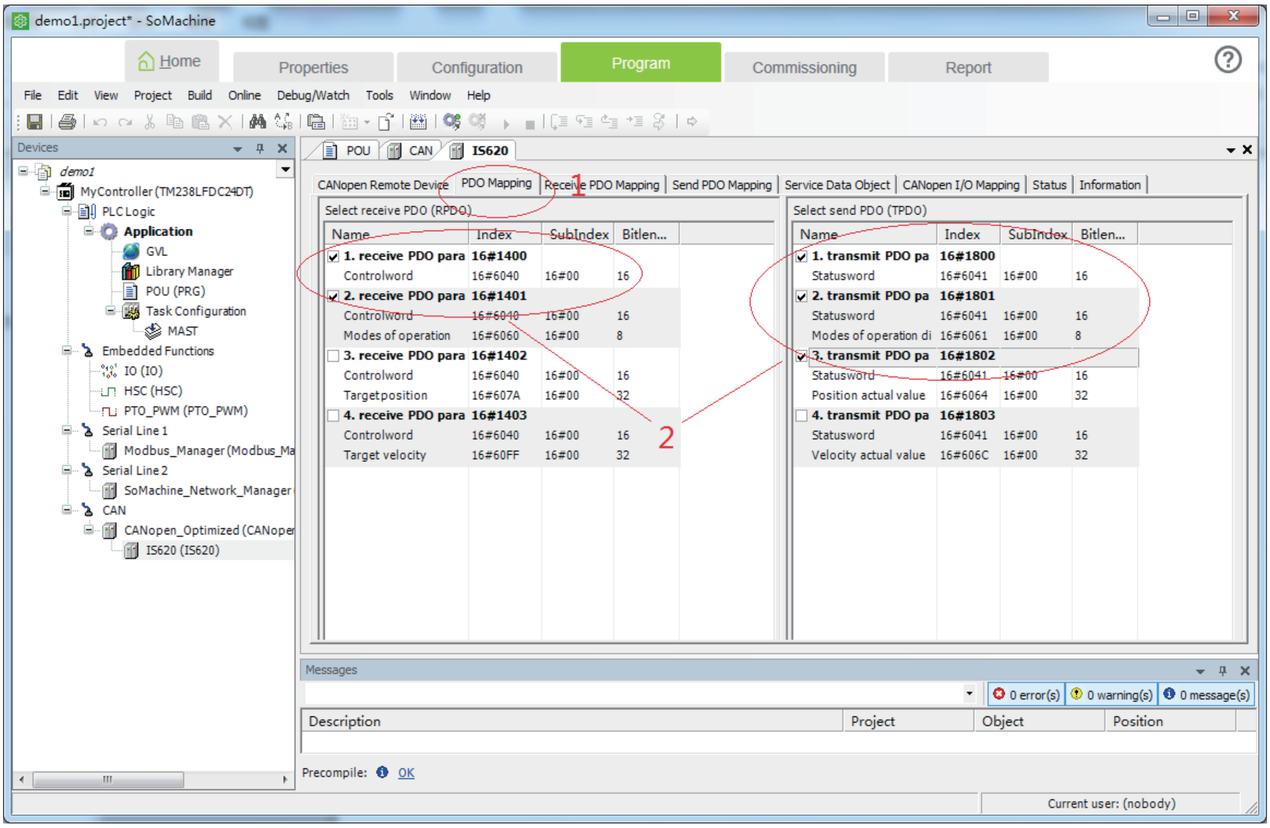
12) Click **Program** in the window, double-click **CAN** on the left, and select an appropriate baud rate, for example, 500Kbps.



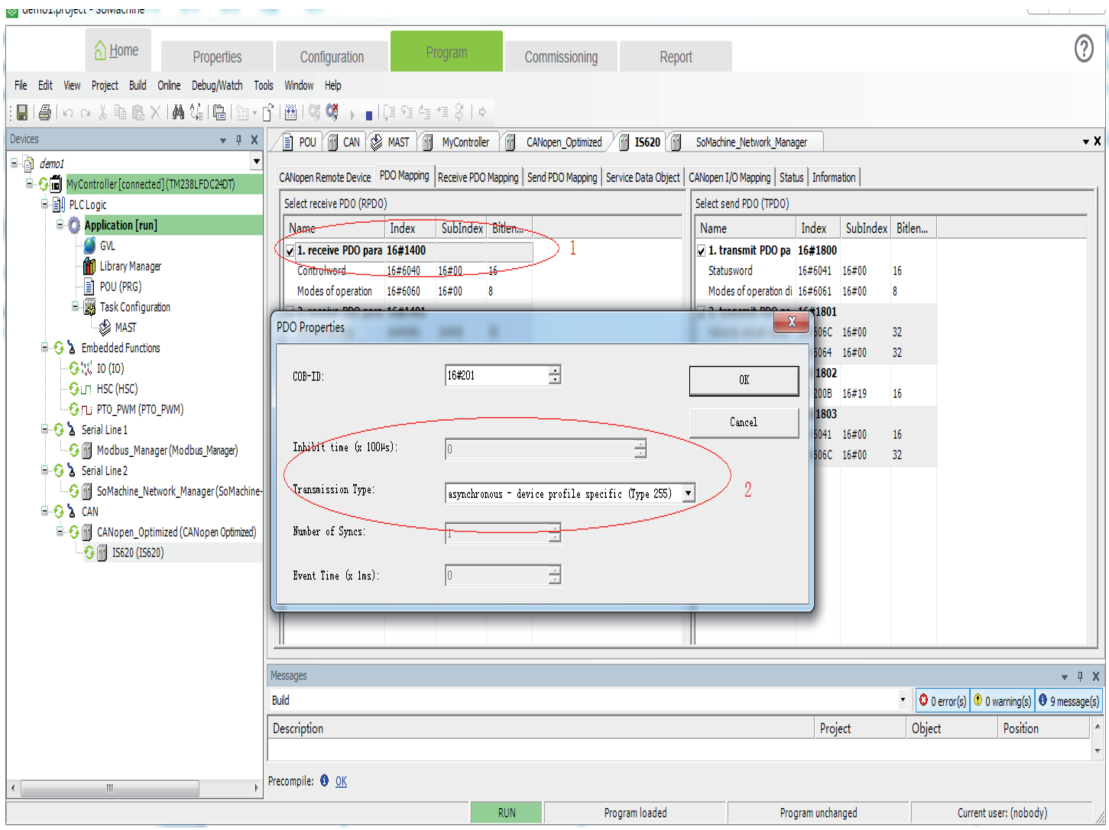
13) Double-click **IS620P_Servo_Driver** on the left. The node ID can be modified. Select **Enable Expert Settings**.



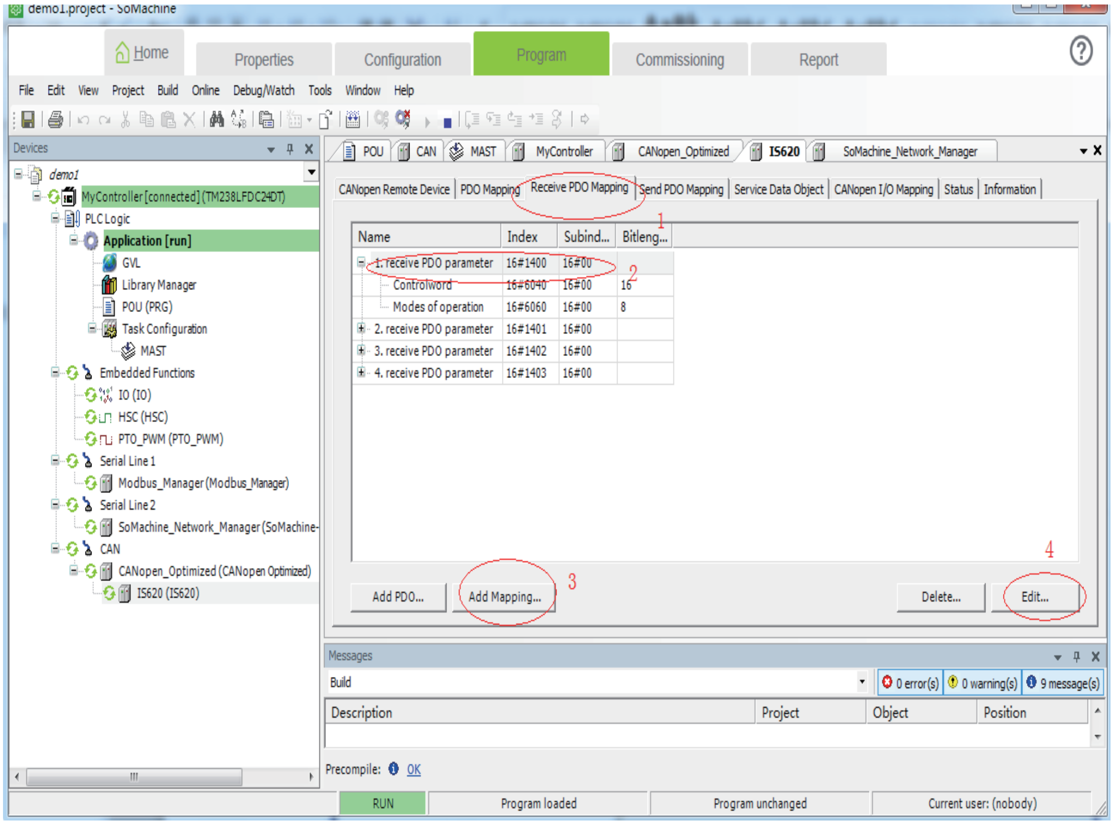
14) Click **PDO Mapping** and select two RPDOs and three TPDOs.



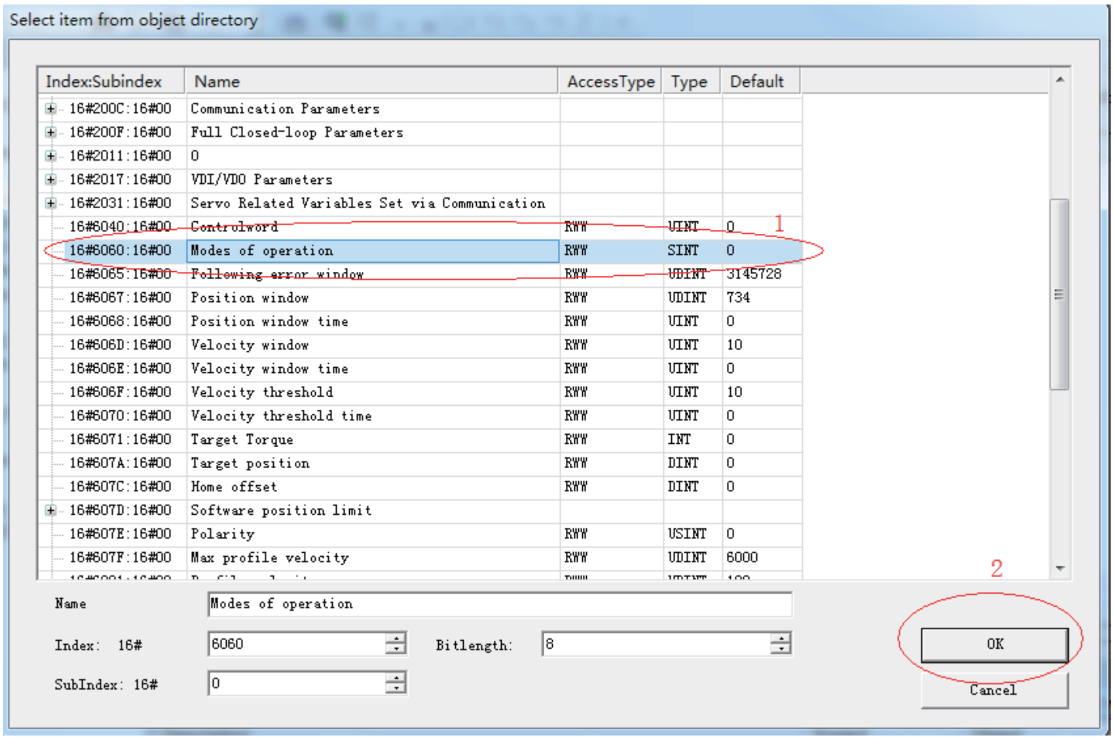
15) Double-click **RPDO1**. The **PDO Properties** dialog box is displayed. Modify **Transmission Type** to **Type 255**. Perform the same operation for other PDOs.



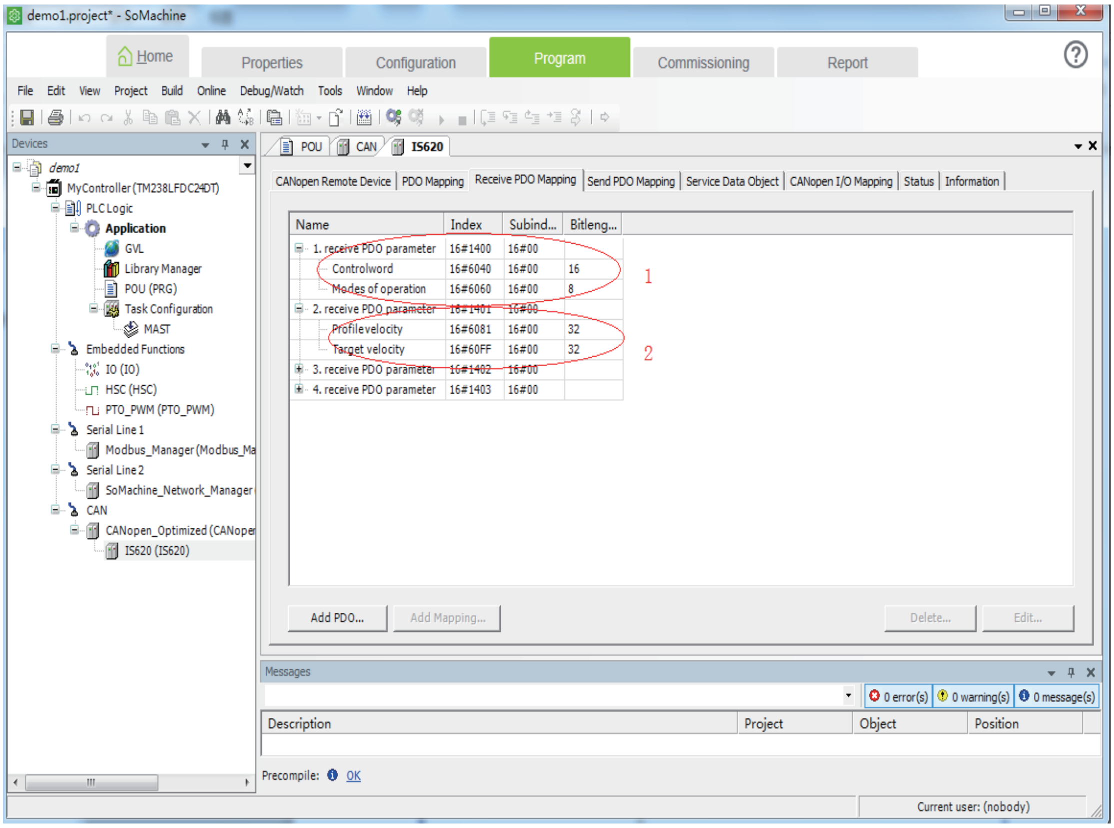
16) Select **Receive PDO Mapping** and click **receive PDO parameter**. Click **Add Mapping** or select a mapping and click **Edit**.



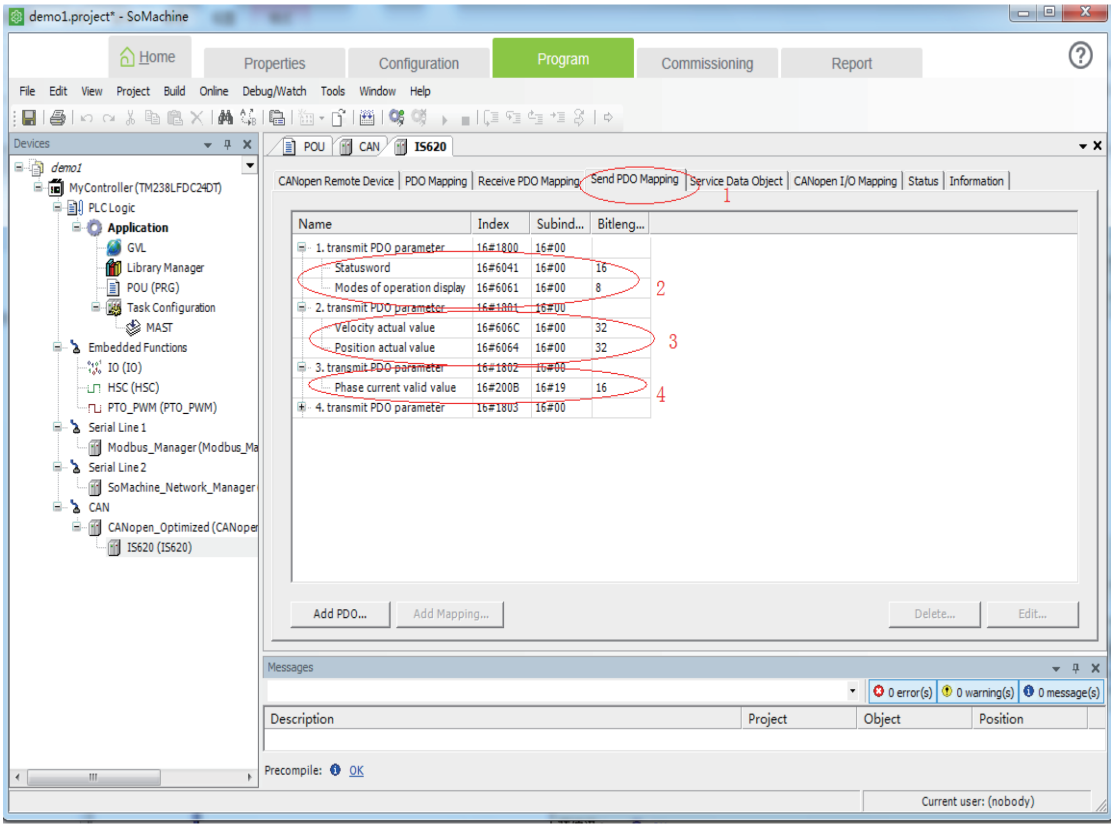
17) The **Add** dialog box is displayed. Select appropriate mapping objects based on Table 7-1.



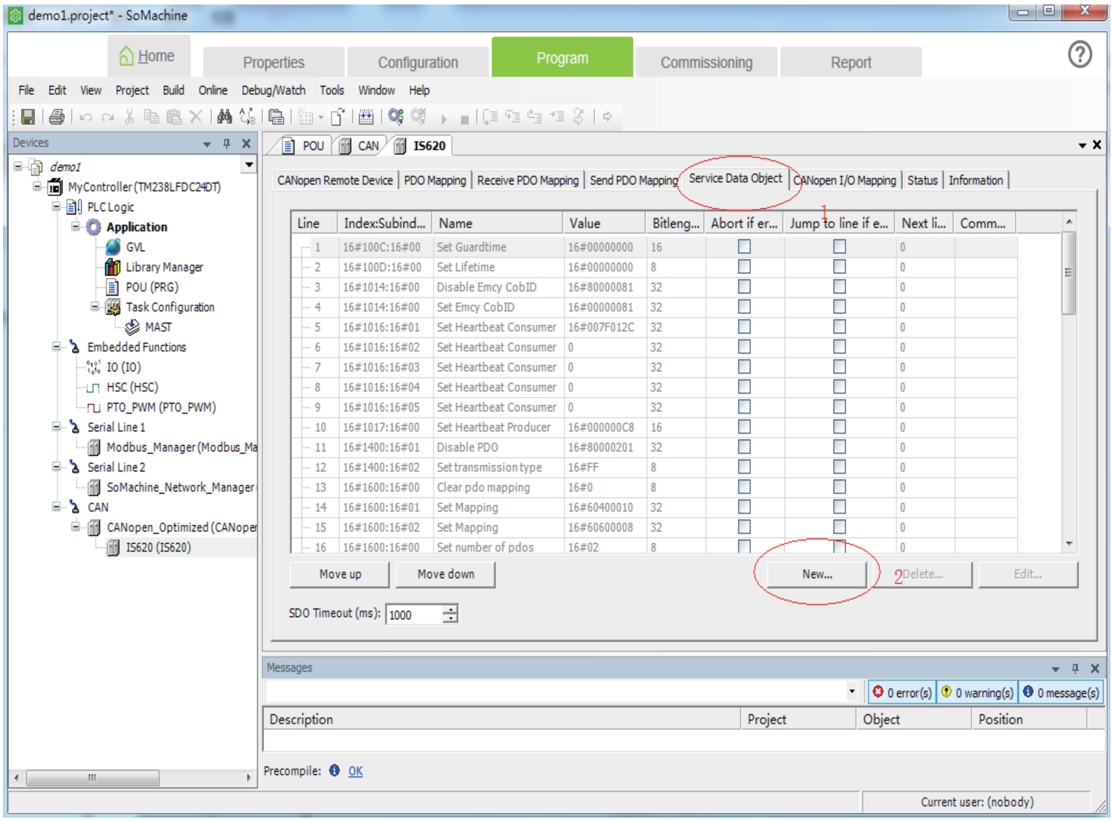
18) After mapping objects are added, the RPDO mapping is as follows:



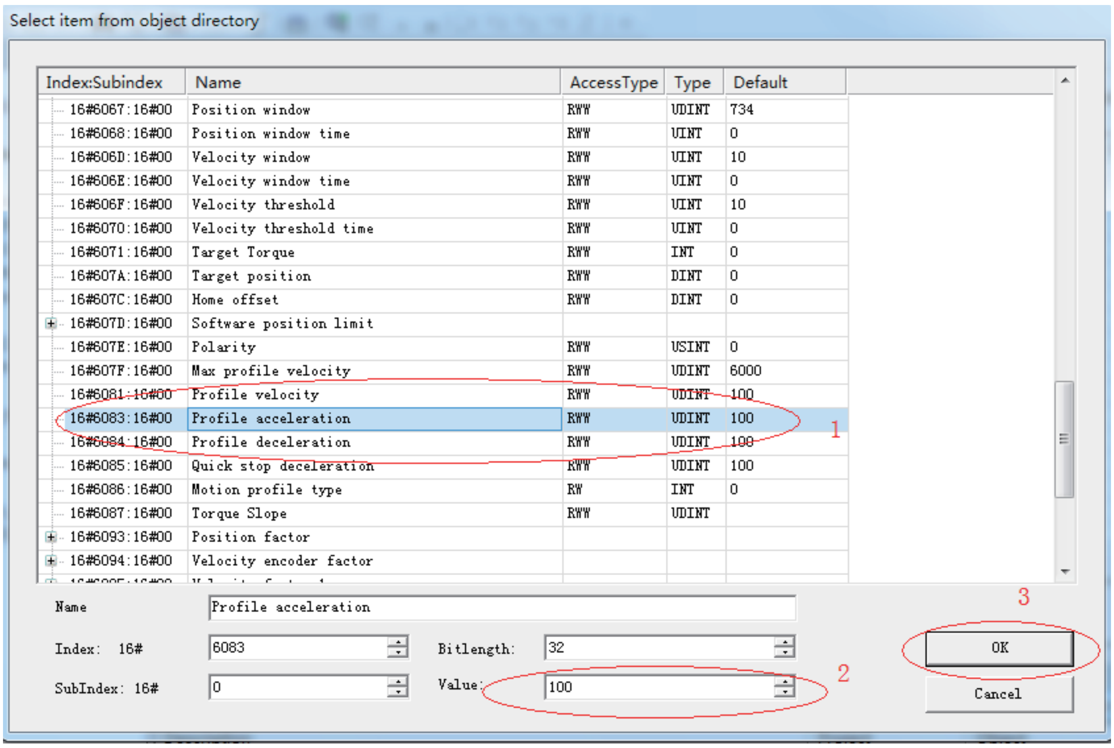
19) Similarly, click **Send PDO Mapping**. Configure the PDO mapping based on Table 7-1.



20) Click **Service Data Object** and click **New** to add a required SDO. (Optional) (If default values are used, steps 20 to 22 can be omitted.)

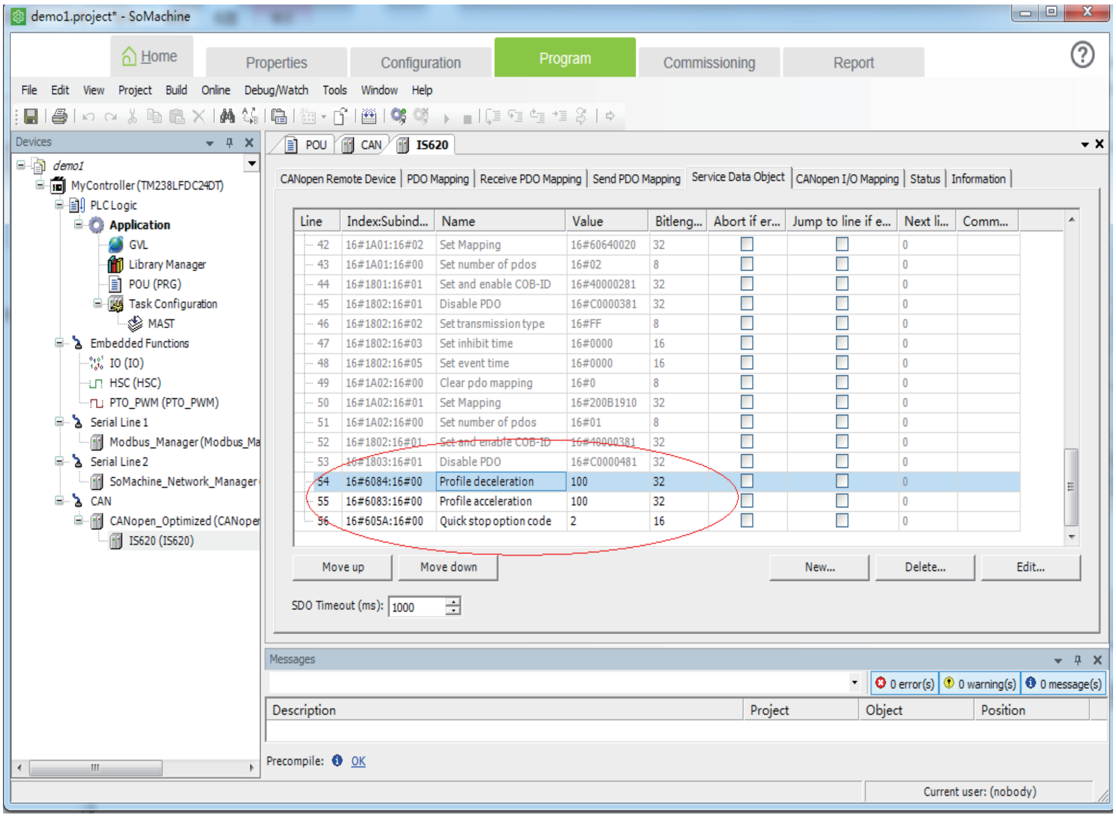


21) Select an SDO from the list, modify its value, and click **OK**. (optional)

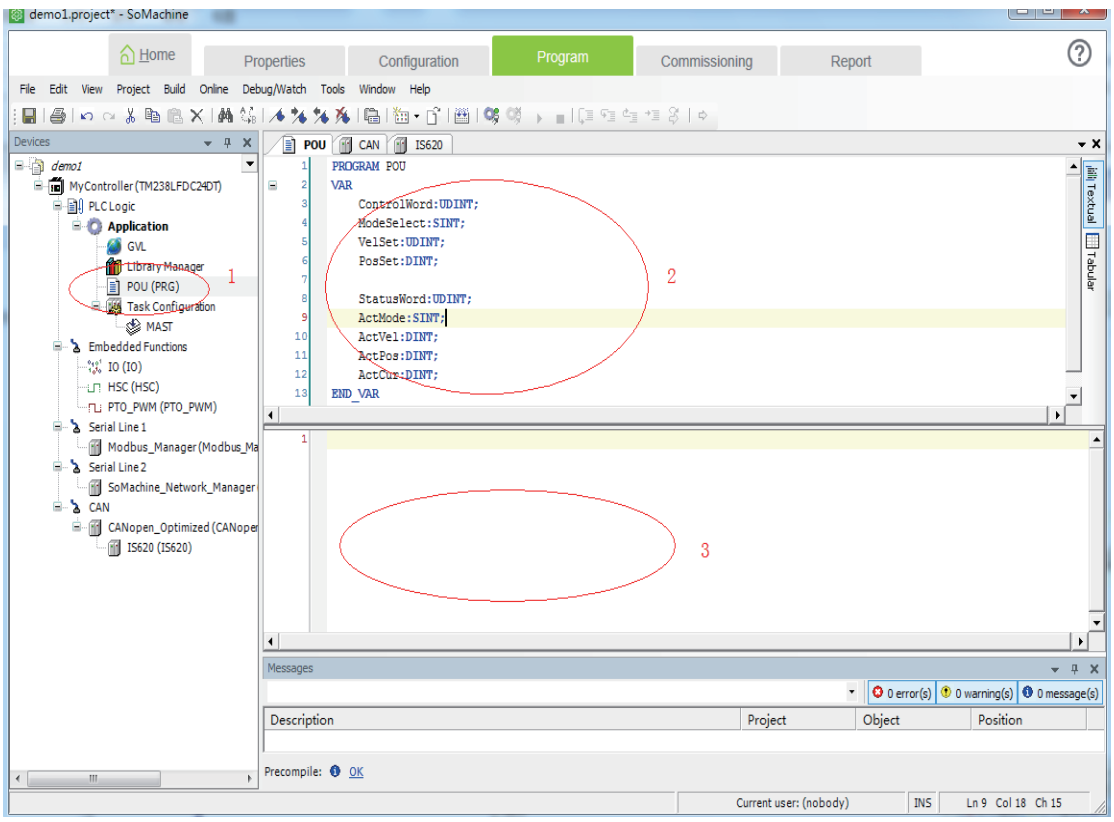


7 Application Cases

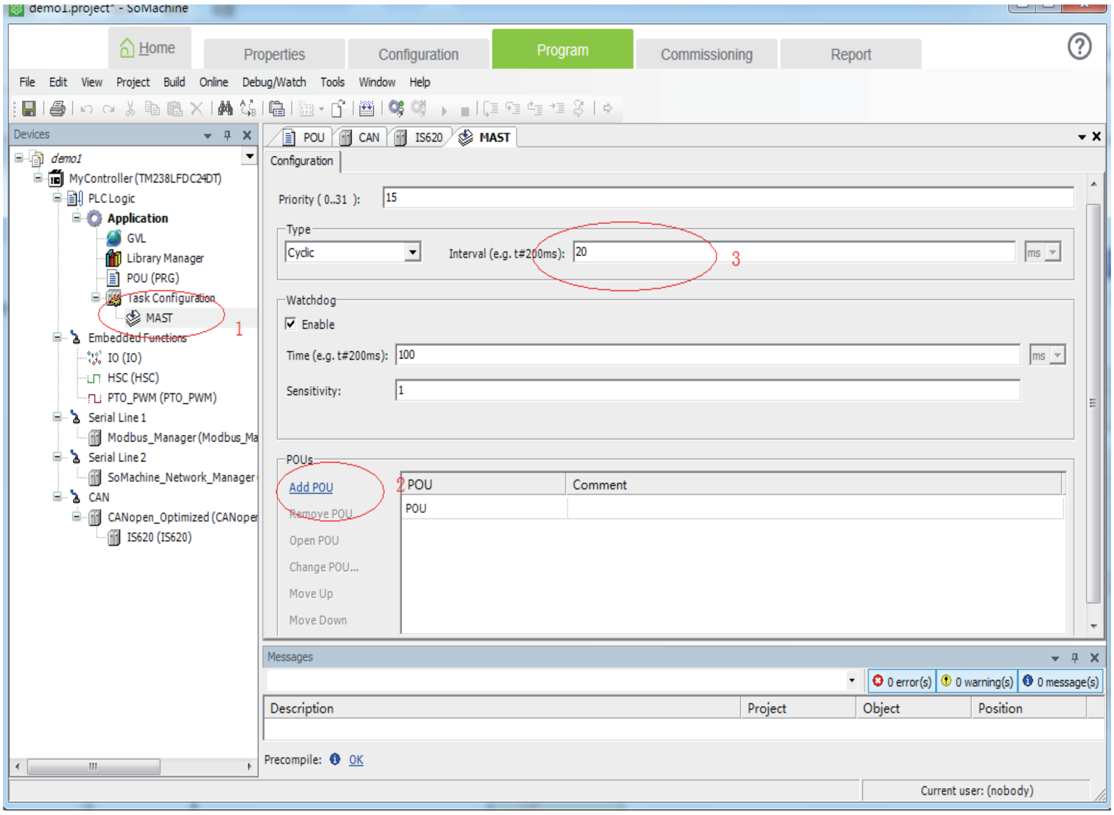
22) Added SDOs are as follows (optional):



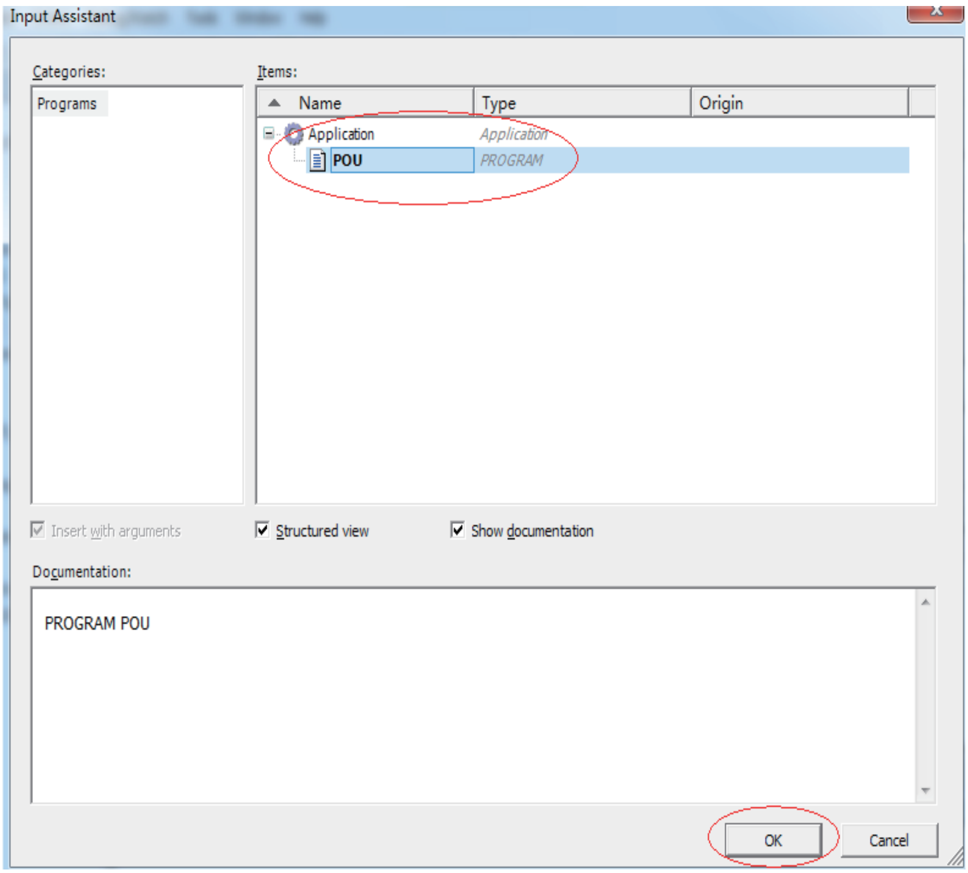
23) Double-click **POU** on the left. Add variable definitions in **2** and add PLC program logic in **3**. Click **Edit** or press **F11**. If no error occurs, go to the next step.



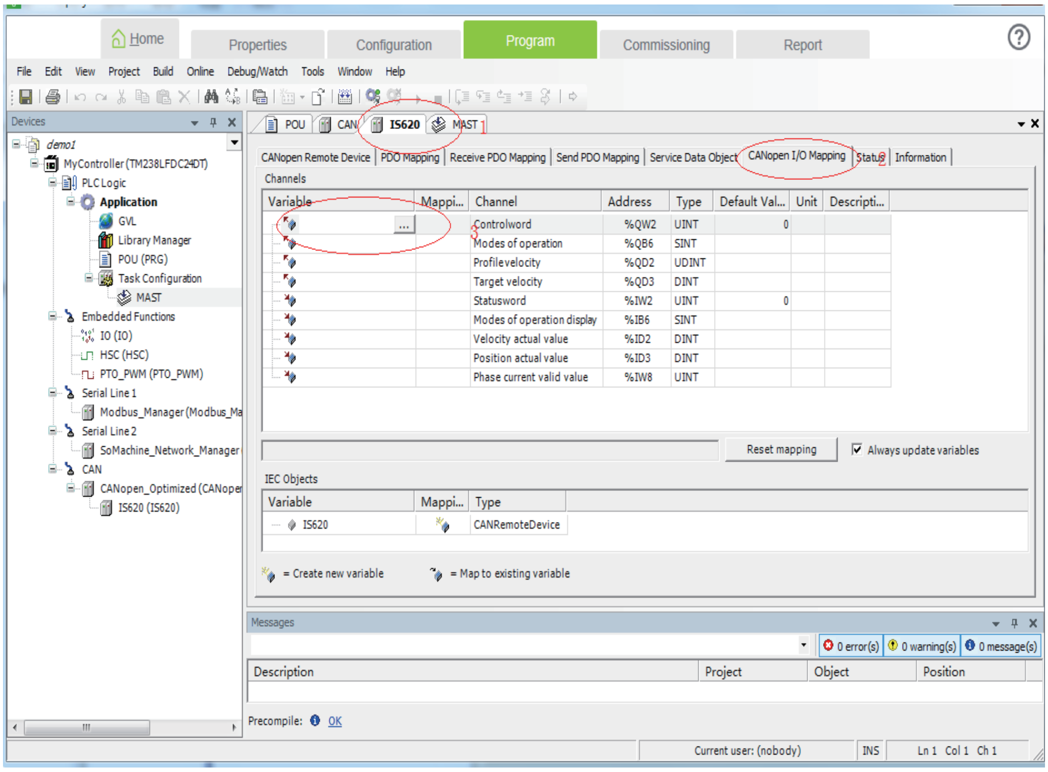
24) Double-click **MAST**, click **Add POU**, and set the program circulation interval.



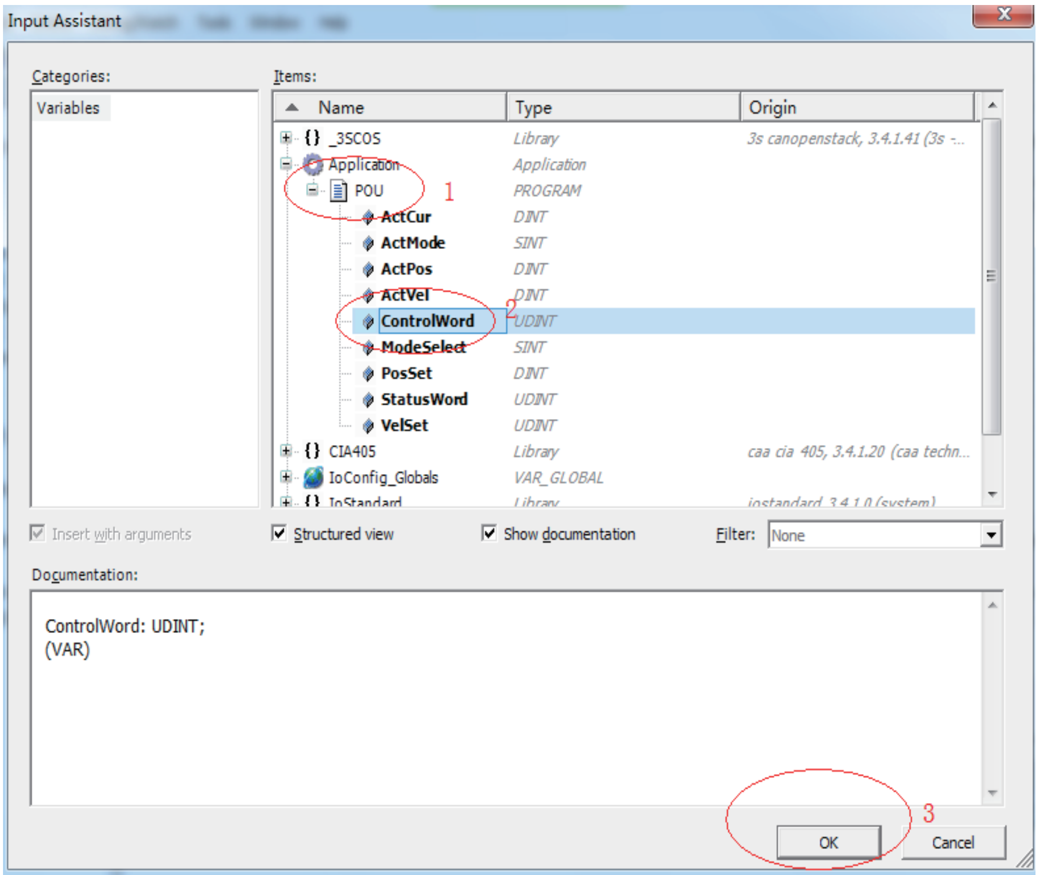
25) Select the added POU in the following dialog box.



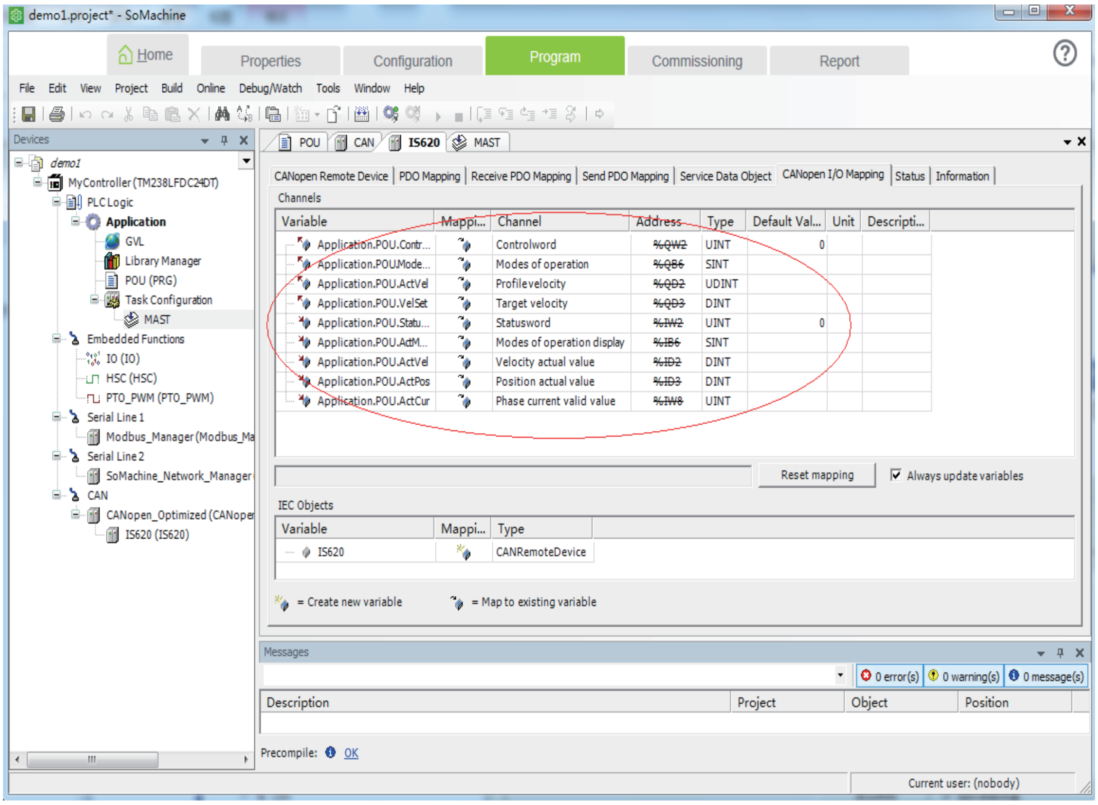
26) Select **CANopen I/O Mapping** of **IS620P_Servo_driver**. In Variables, double-click and then click the **…** button.



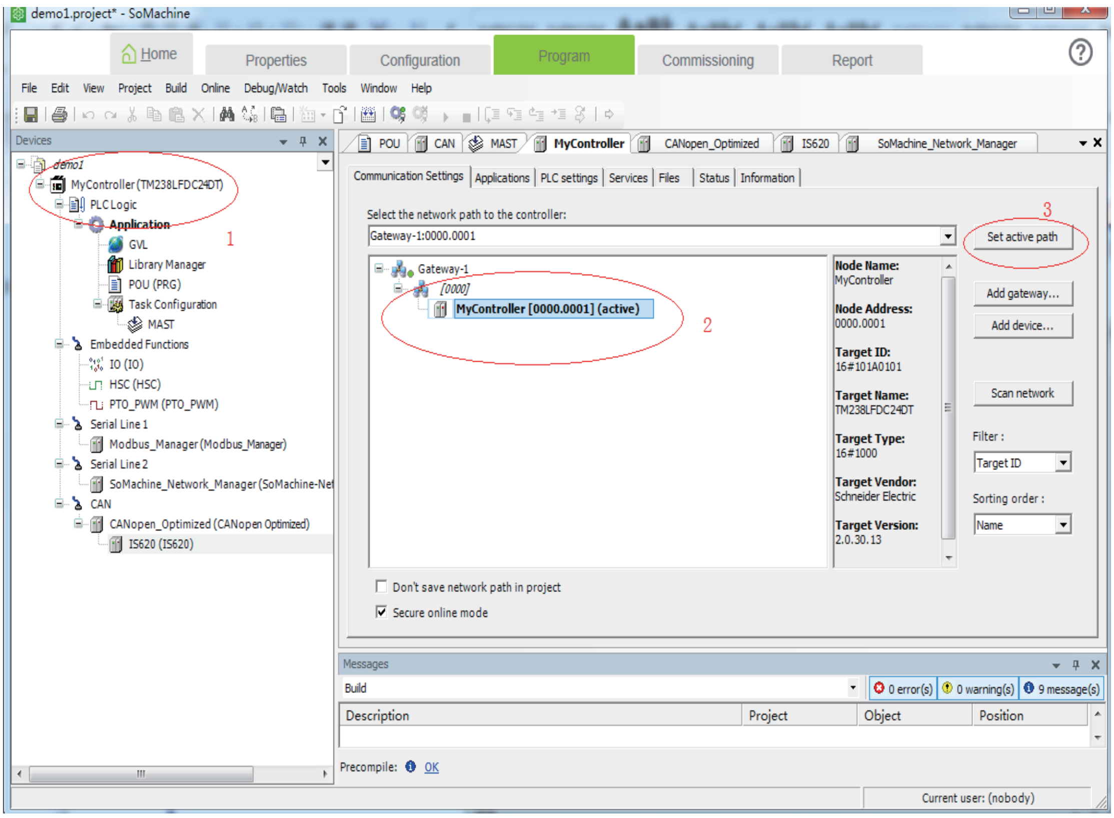
27) Select variables defined by PLC as follows:



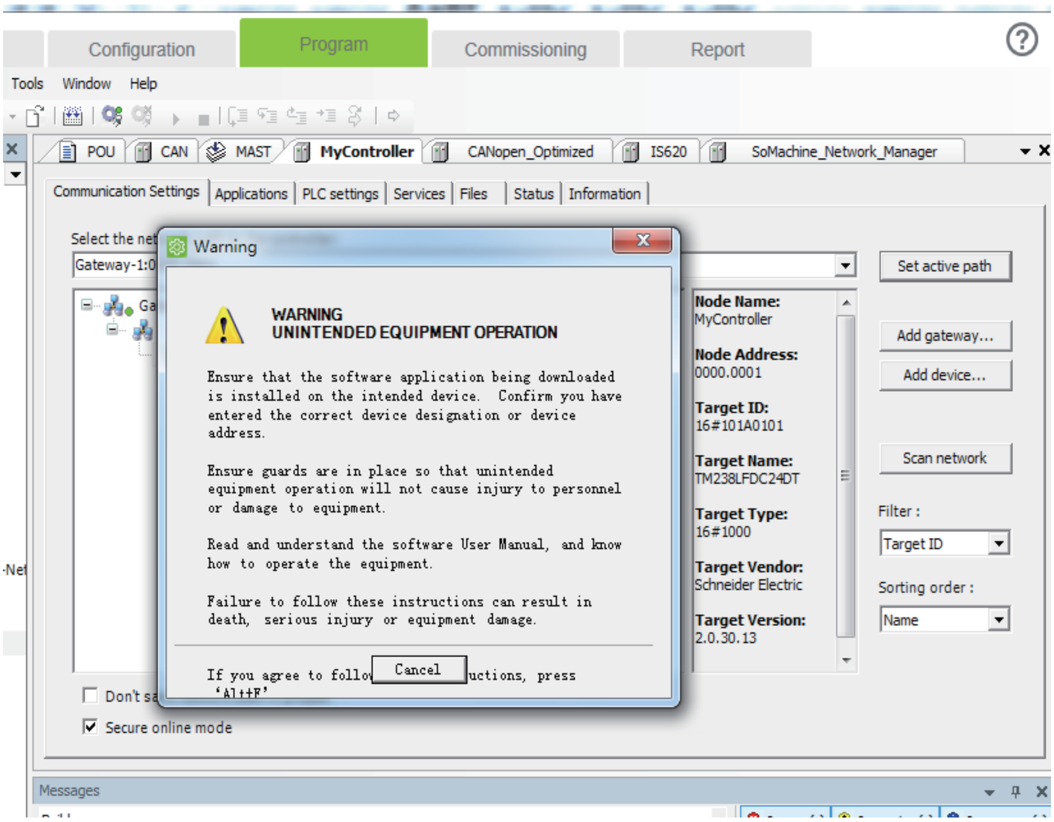
28) Add other variables by using similar methods. The completed mapping is as follows:



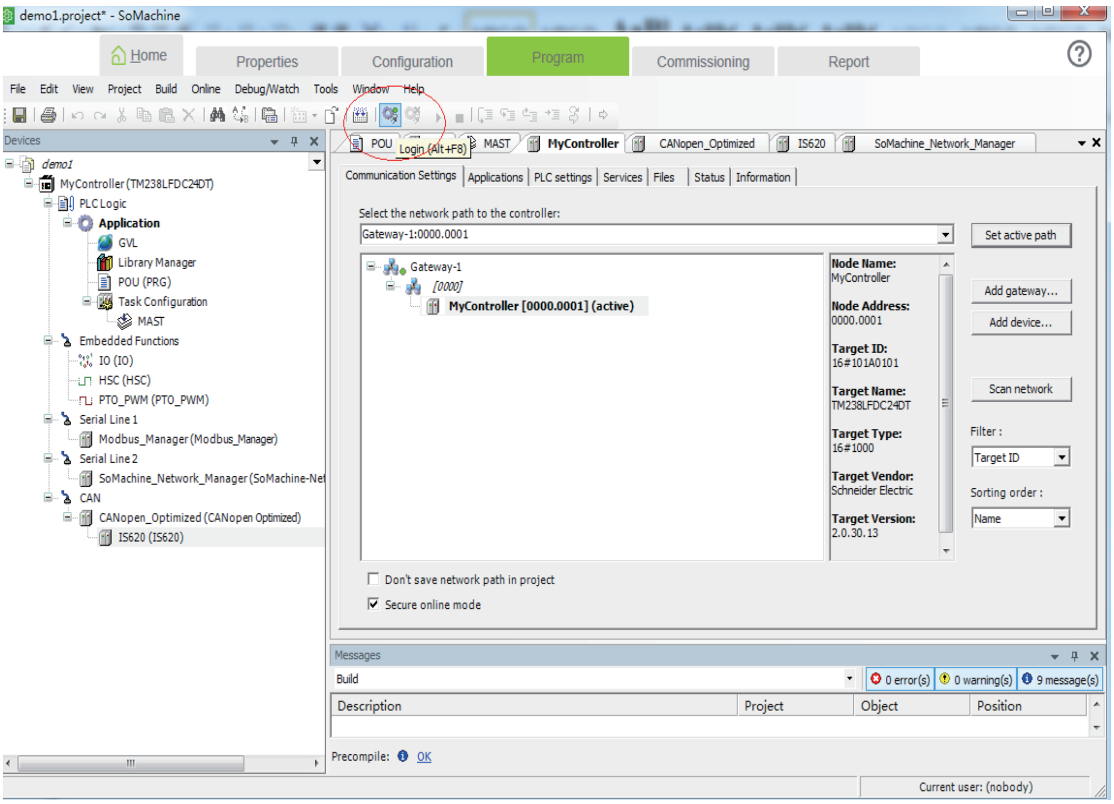
29) Double-click the master node name on the left, select **MyController**, and click **Set active path** on the right.



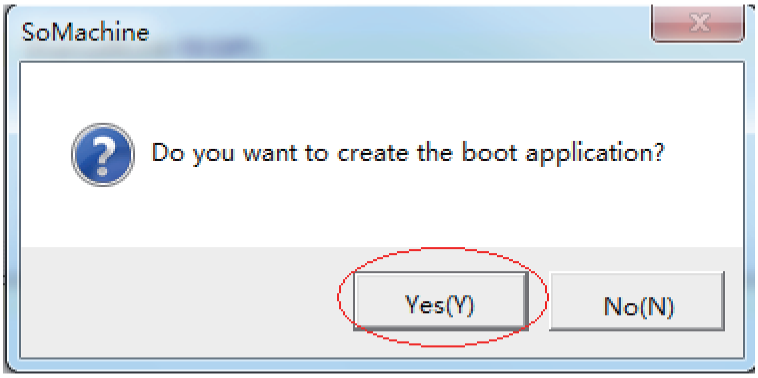
30) The following warning is displayed. Press **Alt** + **F** based on the prompt.



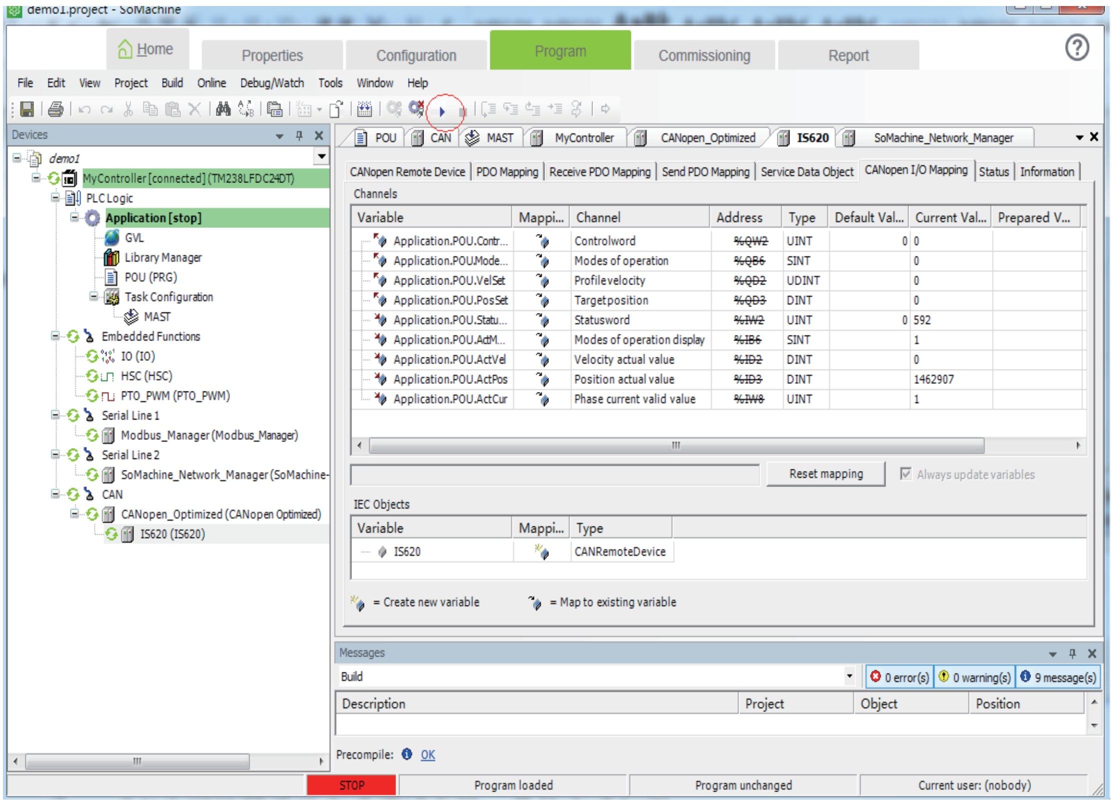
31) Click the icon marked in a circle in the figure, or choose **Online** > **Login** in the toolbar, or press **Alt** + **F8**.



32) In the dialog box that is displayed, click **Yes**.

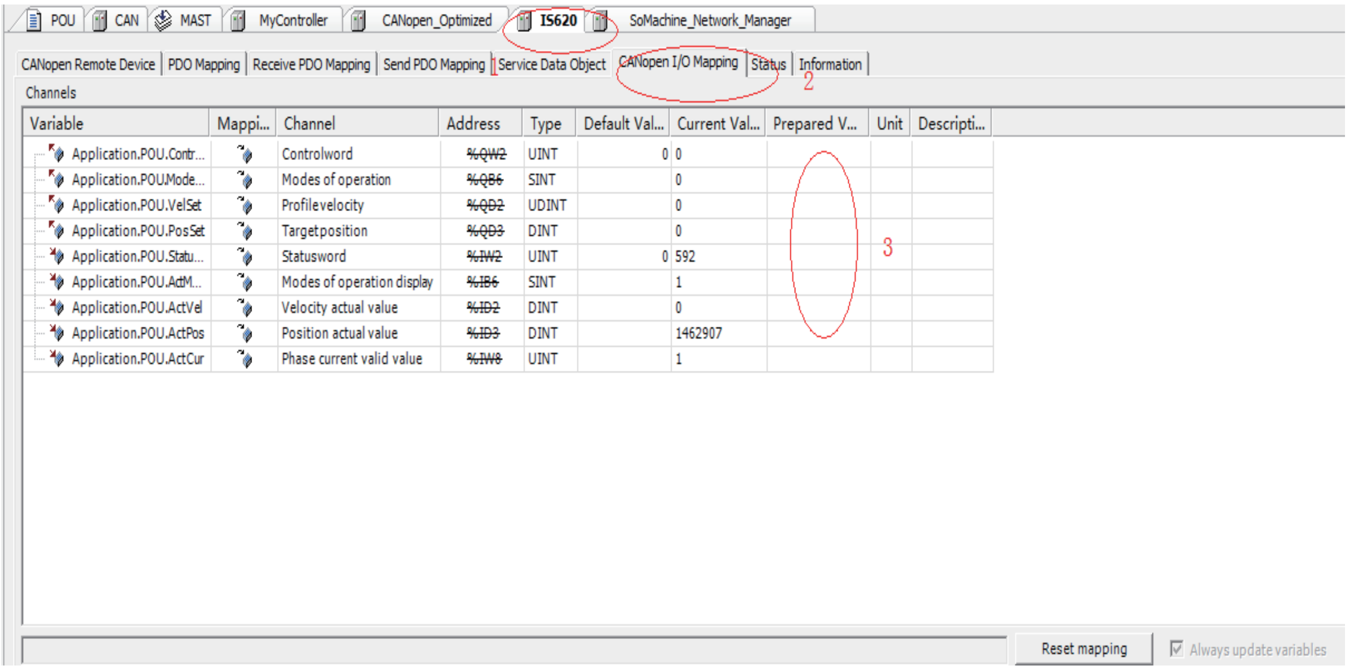


33) Wait until the application is downloaded. Click the small triangle marked in the circle, or choose **Online** > **Start** in the toolbar, or press **F5** to start the PLC program compiled by the user. The motor runs in the mode specified by the user.



34) You can also commission the motor manually. The details are as follows:

Select **CANopen I/O Mapping** under **IS620P_Servo_driver**. In the **Prepared Value** column, enter a required value for a variable and choose **Debug/Watch** > **Force Values** in the toolbar or press **F7** to forcibly change the output variable information.



35) Set 6060h to **1**, 6081h to **100**, and 607Ah to **10485760** (10 rounds) and set 6040h to **6**, **7**, **47(0x2f)**, and **63(0x3f)** in turn. The motor starts running.

NOTE

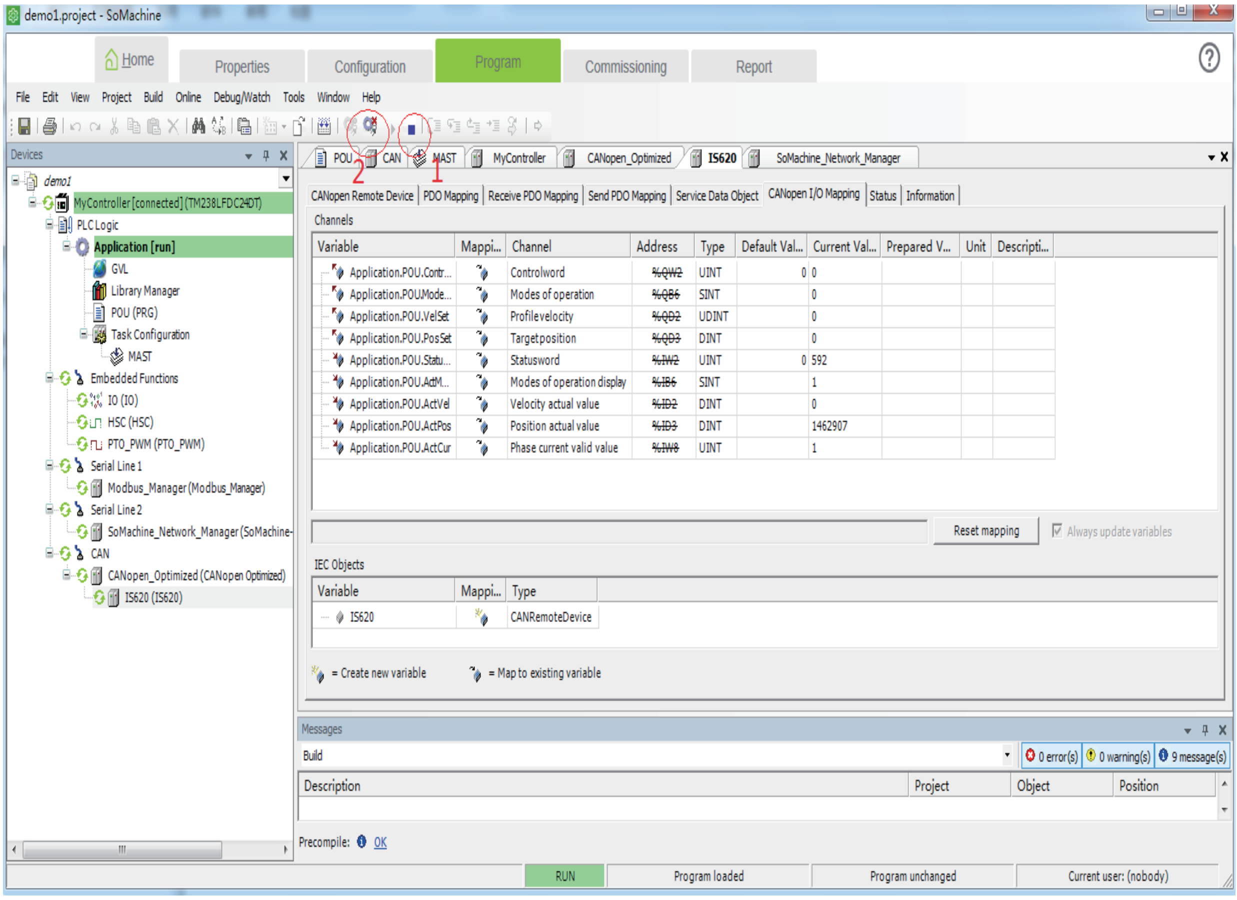
For the same variable, each time when a value is written, the "Force Values" reference is executed. You can enter values for different variables and execute the "Force Values" reference once.

When a new position or velocity reference is required, write the new reference and set 6040h to **47(0x2f)** and **63(0x3f)** in turn. The motor runs to the position according to the new reference no matter whether execution of the previous reference is complete.

To stop the motor, set 6040h to **0**.

Do not enter values forcibly. In the toolbar, choose **Debug/Watch** > **Release Values** or press **Alt** + **F7**. Variables are no longer entered and follow the logic of the PLC program.

36) Execute **1** marked in the figure, or choose **Online** > **Stop** in the toolbar or press **Shift** + **F8** to stop the PLC program. Click 2 in the figure, or choose **Online** > **Exit** in the toolbar, or press **Ctrl** + **F8** to exit the online function of the routine.



7.2 Connecting IS620P Servo Drive to Beckoff CANopen Master Node

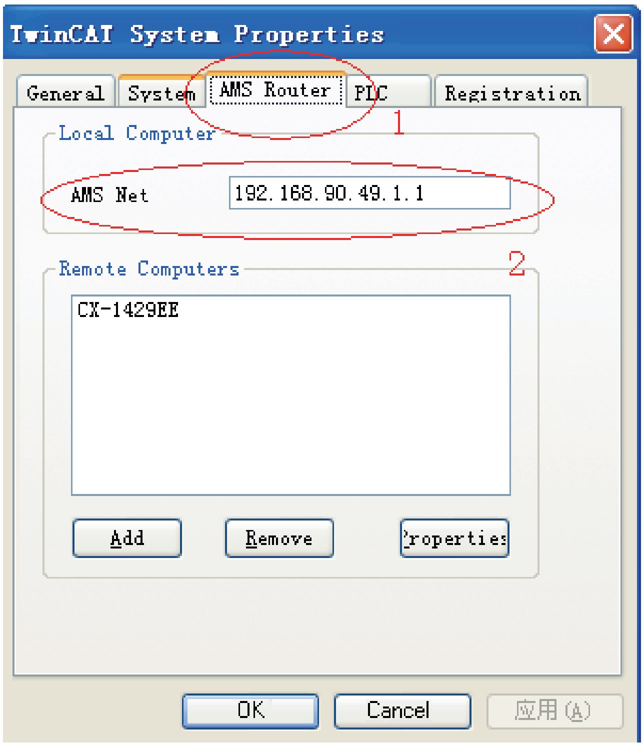
Similarly, in a position mode, allocate PDOs by following Table 7-2.

1) Configuring PDO mapping is complex on a Beckoff master node. Therefore, before connecting the network, manually configure the PDO mapping. Based on the following table and the appendix, change the mapping by modifying functional codes. Table 7-2 lists modified functional codes.

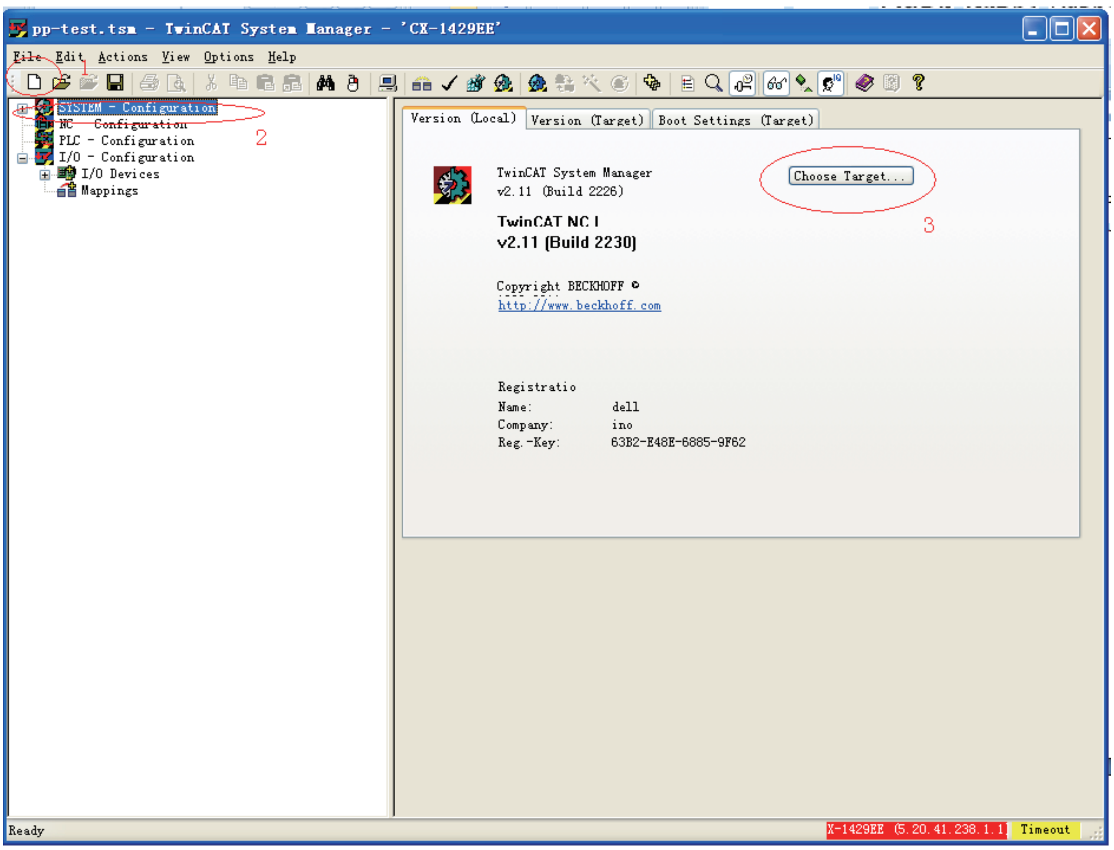
Table 7-2 Examples of PDO mapping for a Beckoff master node

Functional Code	Object	Mapping Object	Input Content
H18-34	1600h-00h	Number of RPDO1 mapping objects	2
H18-35	1600h-01h	6040h-00h	60400010h
H18-37	1600h-02h	6060h-00h	60600008h
H19-02	1601h-00h	Number of RPDO2 mapping objects	2
H19-03	1601h-01h	6081h-00h	60810020h
H19-05	1601h-02h	607Ah-00h	607A0020h
H1A-24	1A00h-00h	Number of TPDO1 mapping objects	2
H1A-25	1A00h-01h	6041h-00h	60410010h
H1A-27	1A00h-02h	6061h-00h	60610008h
H1A-41	1A01h-00h	Number of TPDO2 mapping objects	2
H1A-42	1A01h-01h	606Ch-00h	606C0020h
H1A-44	1A01h-02h	6064h-00h	60640020h
H1B-10	1A02h-00h	Number of TPDO3 mapping objects	1
H1B-11	1A02h-01h	200Bh-19h	200B1910h
H1B-13	1A02h-02h	-	0

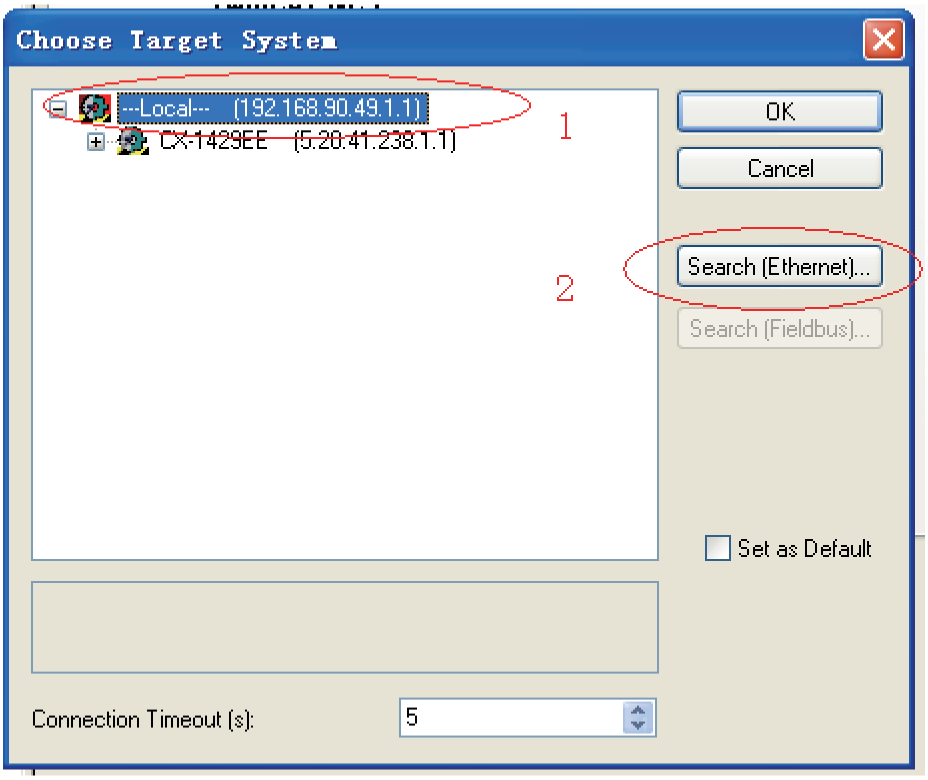
2) Connect Beckoff CX9020, as a master node, to the CANopen module of EL6751 and perform the test. Ensure that the IP address of CX9020 is in the same network segment as the IP address of the PC and the first four bytes of AMS Net (**Properties** > **AMS Router** > **AMS Net**) of Beckoff TwinCAT software are the same as the IP address of the PC.



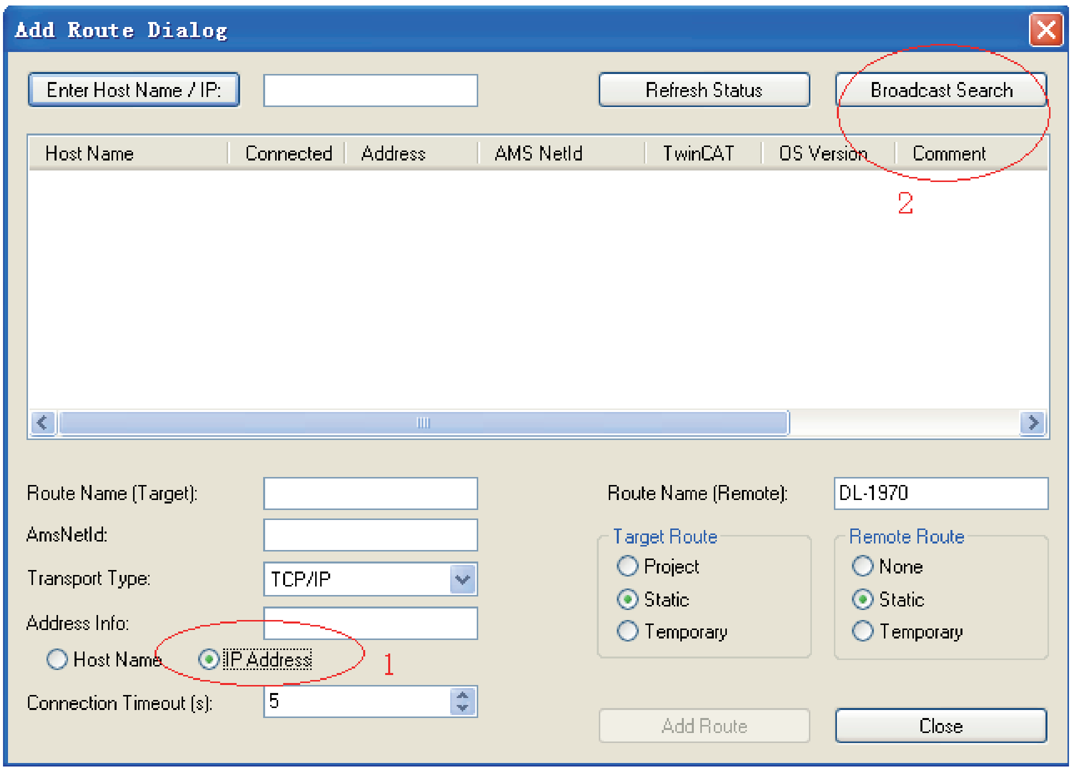
3) Open TwinCAT System Manager and create an empty project. Click **SYSTEM - Configuration** on the left and click **Choose Target** on the right.



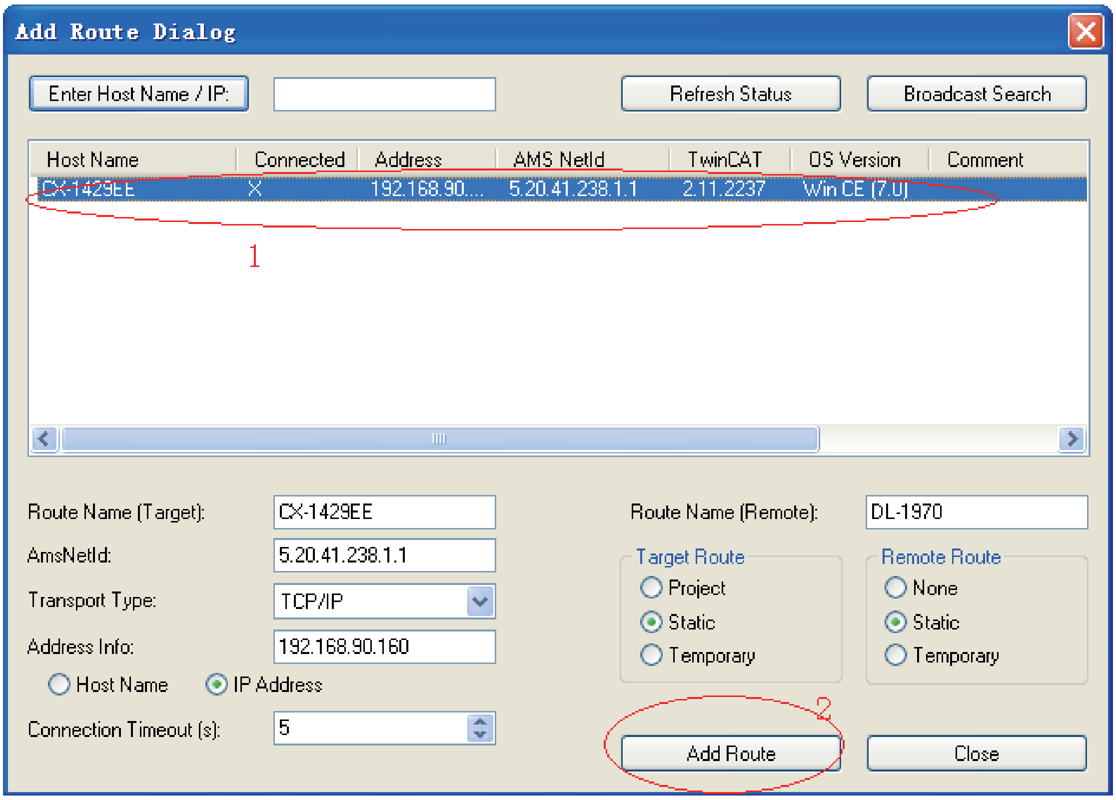
4) In the dialog box that is displayed, select **…local…** and click **Search (Ethernet)**.



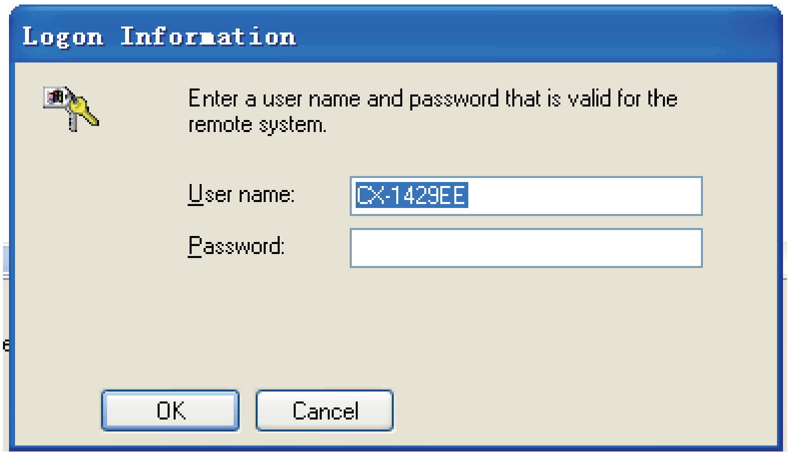
5) In the dialog box that is displayed, select **IP Address** in **1** marked in the figure and click **Broadcast Search**.



6) The master node is searched out. Select the master node and click **Add Route**.



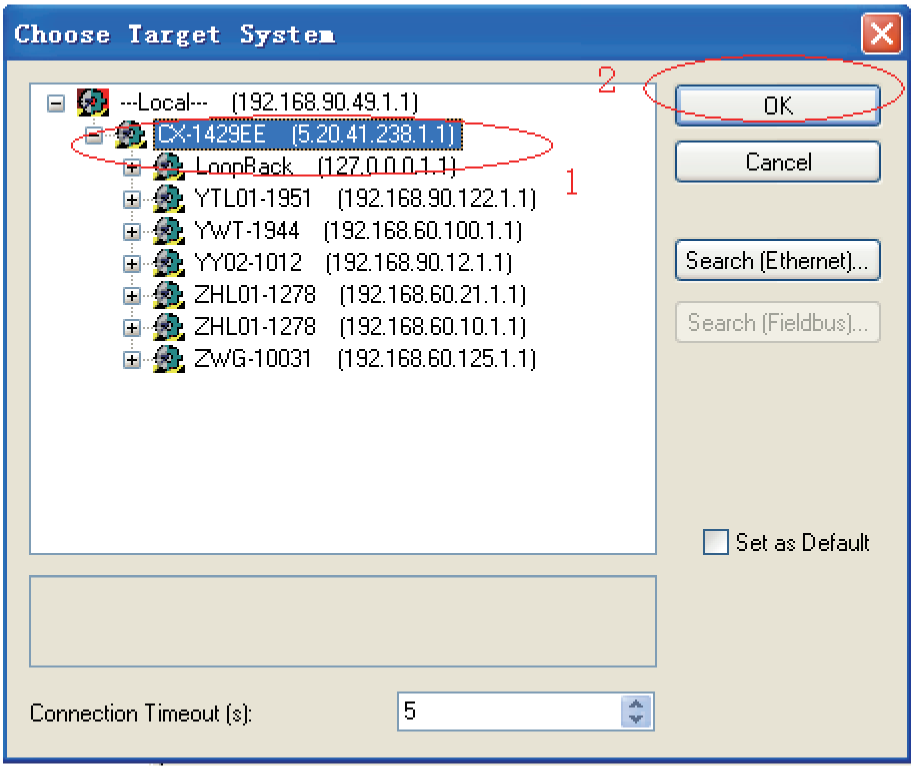
7) In the dialog box that is displayed, the account name is the same as the **Host Name** and the password is empty. Click **OK**.



7

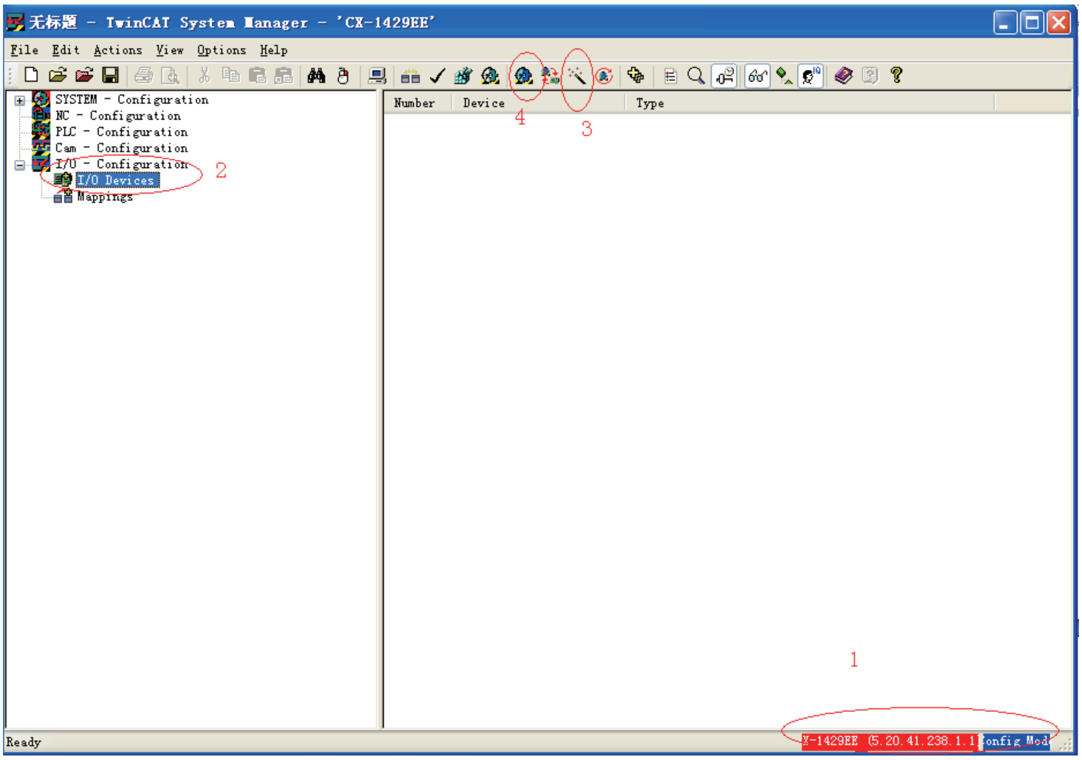
Application Cases

8) Click **Close** in Step 6. Click **+** in the **Choose Target System** dialog box, select the master node, and click **OK**.

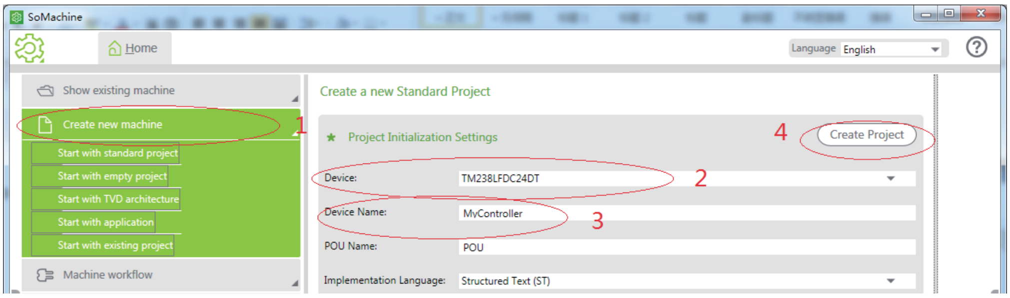


9) The master node link marked in red can be seen in the lower right corner of the window and is in the configuration status marked in blue. If the master node is in the operating status marked in green, click **4** in the figure to switch to the configuration mode and perform the next step.

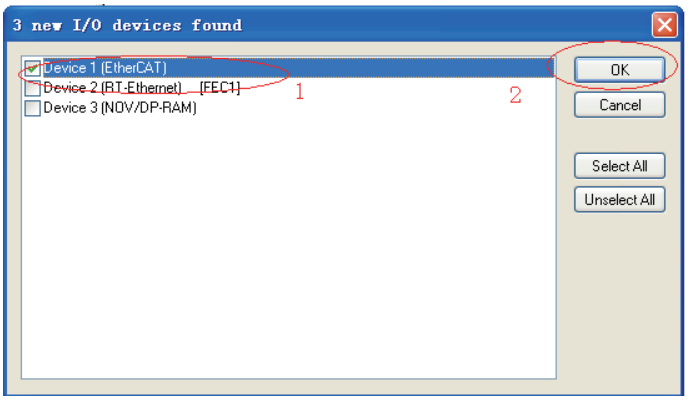
Select **I/O Devices** on the left and click **3** or right-click **I/O Devices** and choose **Scan Devices**.



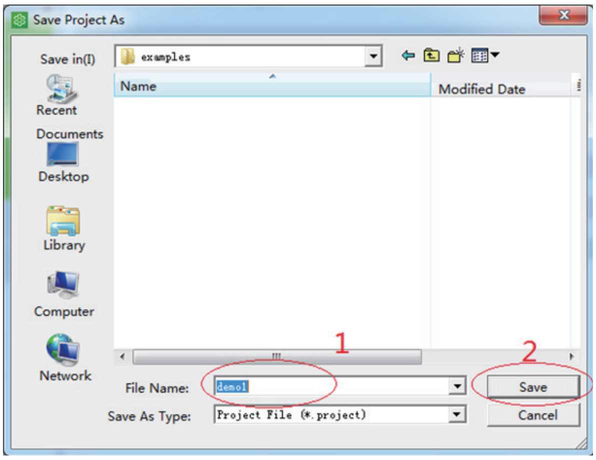
10) In the warning dialog box that is displayed, click **OK**.



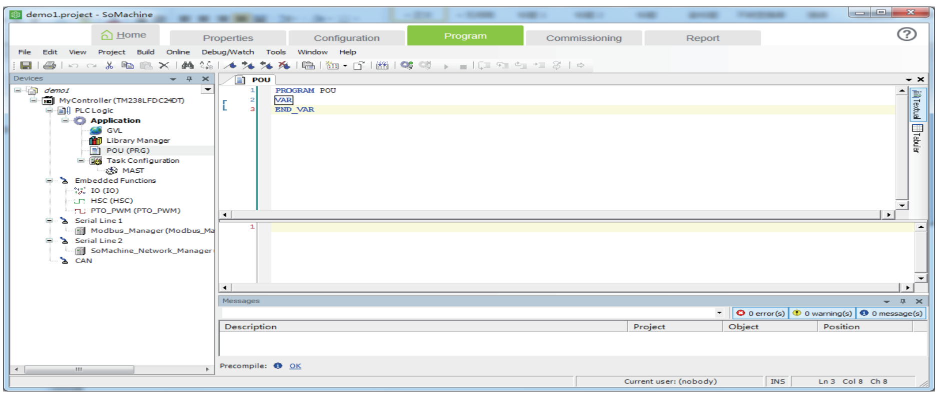
11) In the dialog box that is displayed, select the device of EtherCAT and click **OK**.



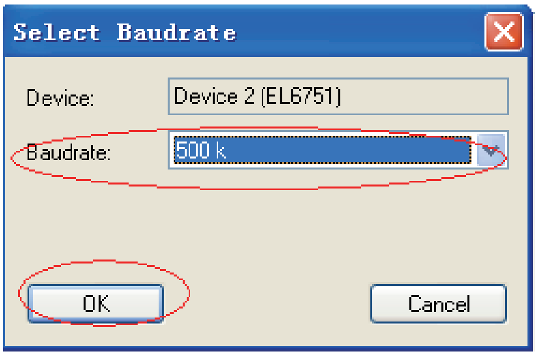
12) In the dialog box that is displayed, click **Yes**.



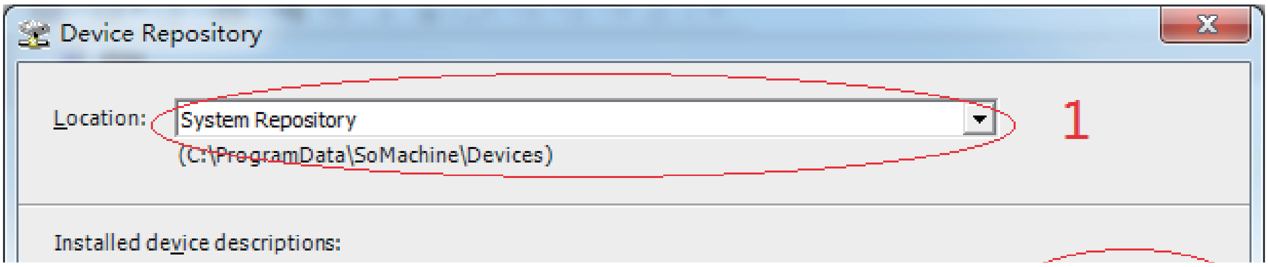
13) In the dialog box that is displayed, click **Yes**.



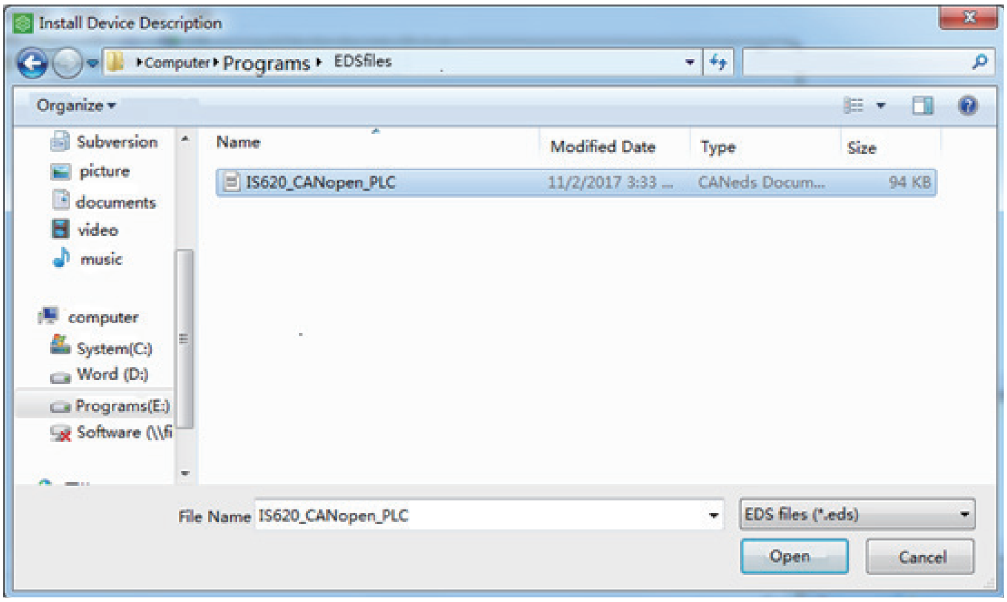
14) Select a value for **Baud rate** (the default value is **500 Kbps**) and click **OK**. The master node starts to search for the device. You need to wait.



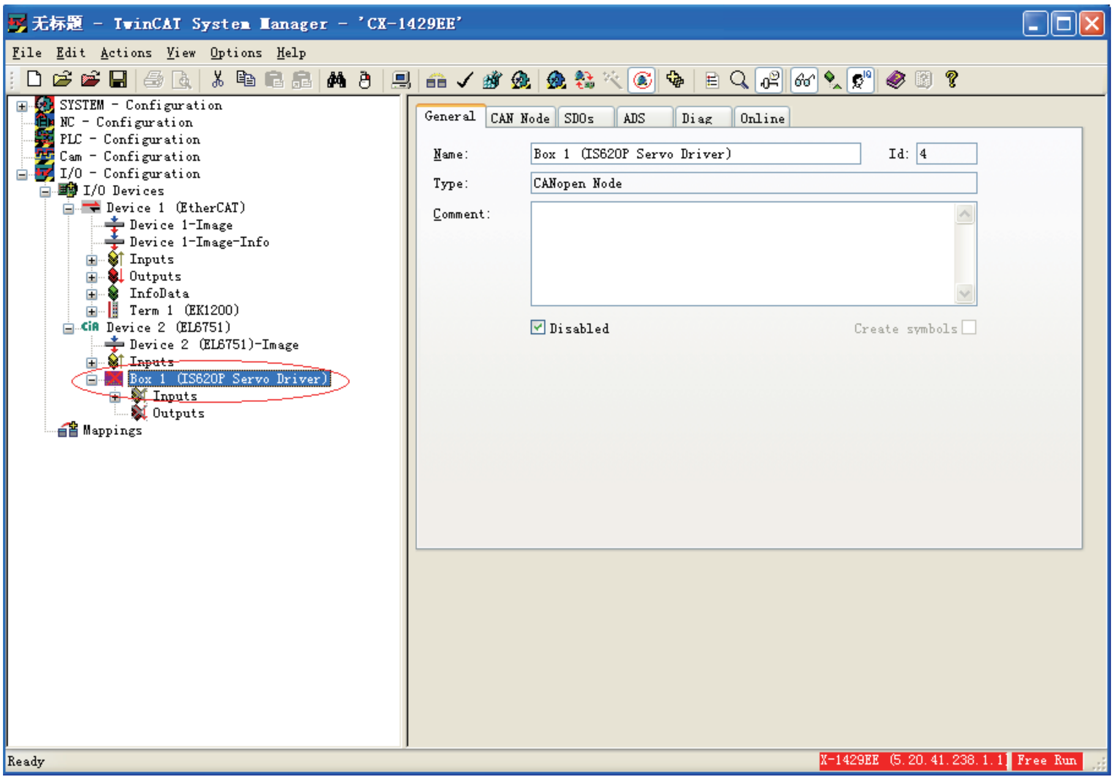
15) After the search is complete, a warning dialog box is displayed. Click **OK**.



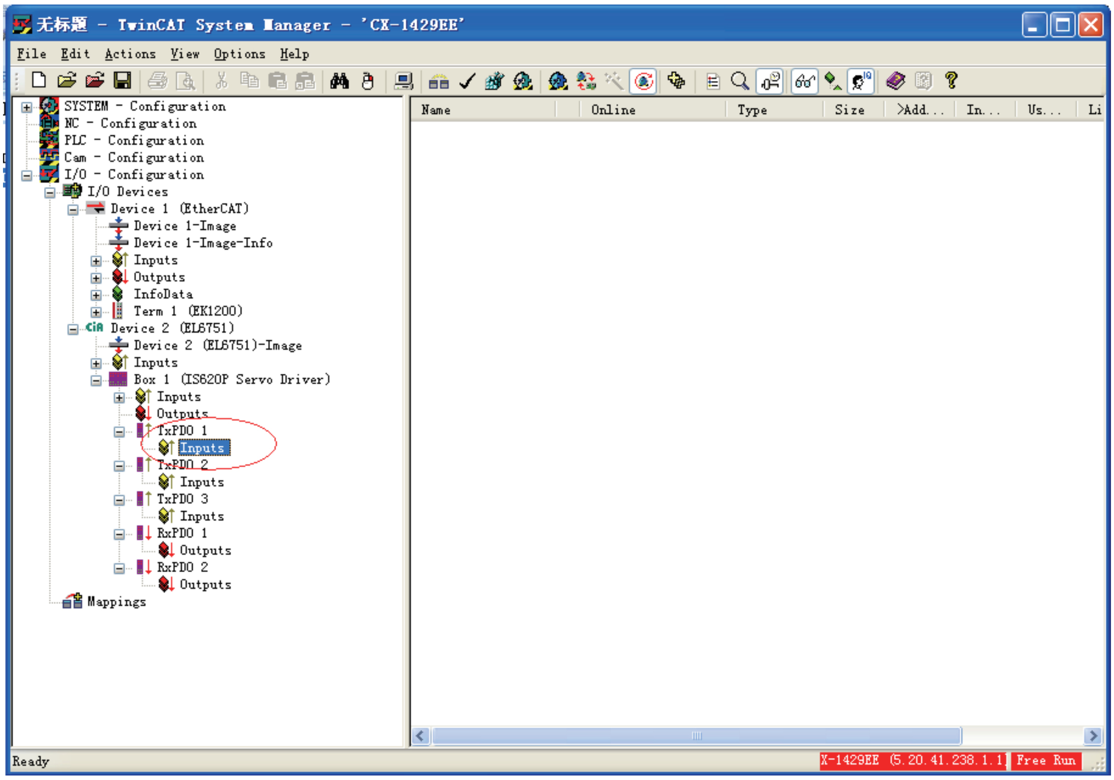
16) In the dialog box that is displayed, click **Yes**.



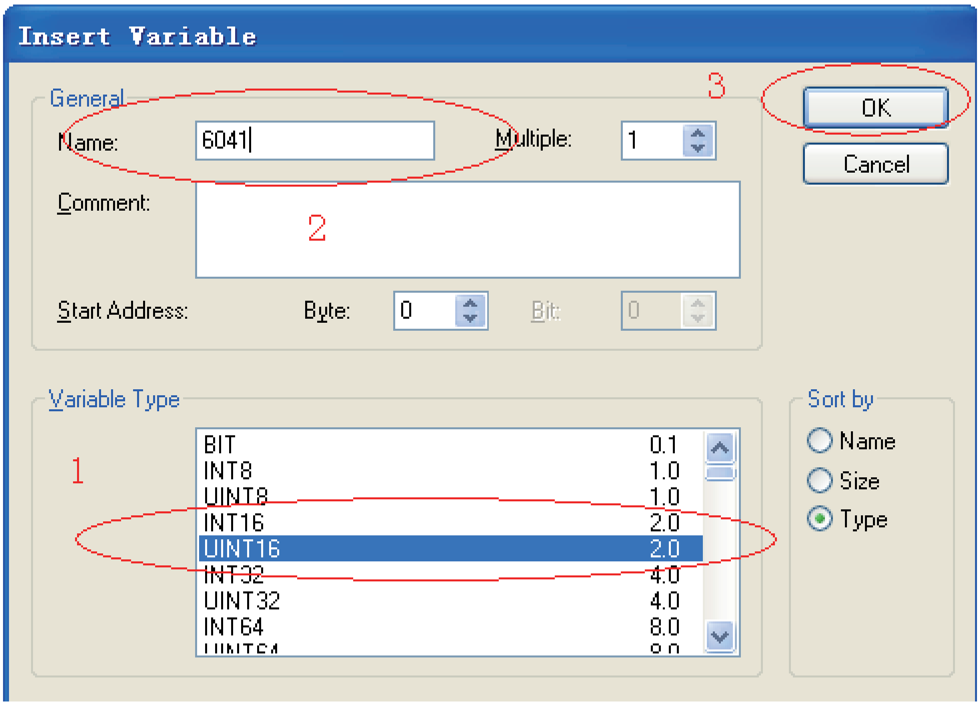
17) A box of the IS620P servo drive can be seen on the left. Select the box, right-click, choose **Insert Variables**, insert three TPDOs and two RPDOs, and select **Disabled** on the right.



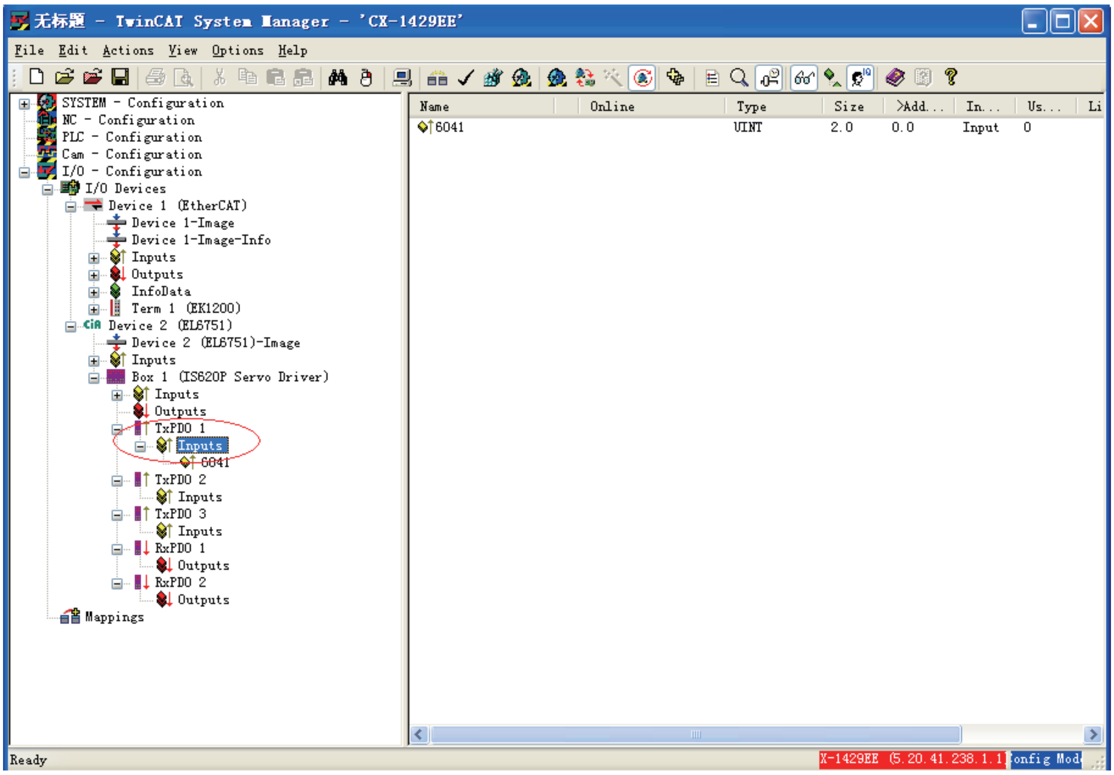
18) The following figure shows the effect after the previous operation is complete. Choose **TPDO1** > **Inputs**, right-click, and choose **Insert Variable**.



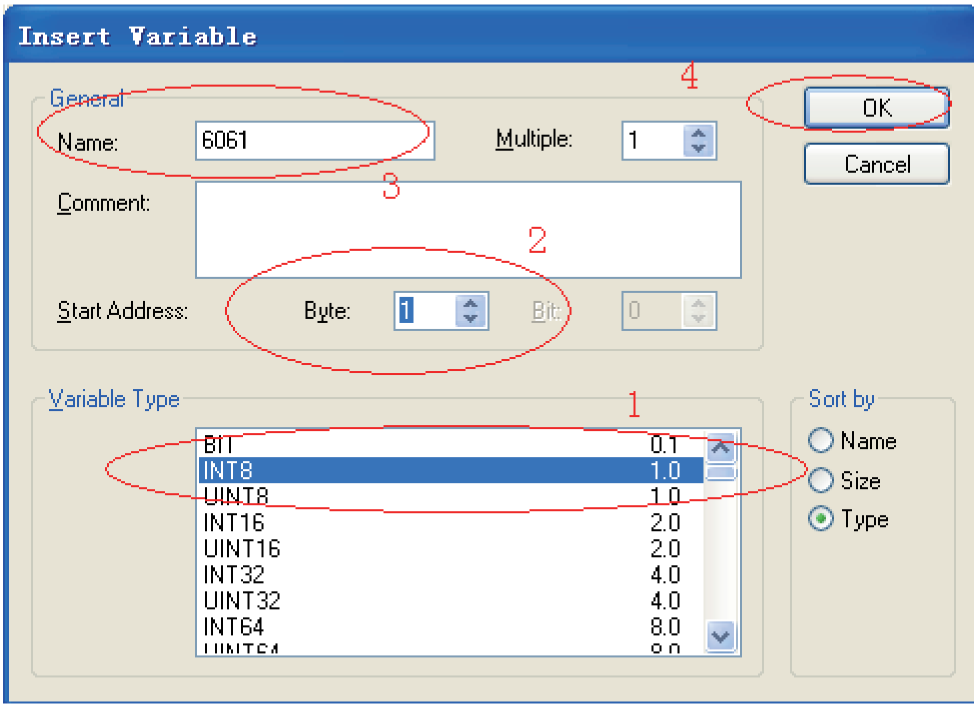
19) Based on Table B-1, map different variables by using PDOs. TPDO1 maps 6041h-00 and 6061h-00. First insert the first variable 6041h, select **UINT16** for **Variable Type**, enter a proper name, and click **OK**.



20) 6041h is added to TPDO1. Select **Inputs** again, right-click, choose **Insert Variable**, and insert the second variable.

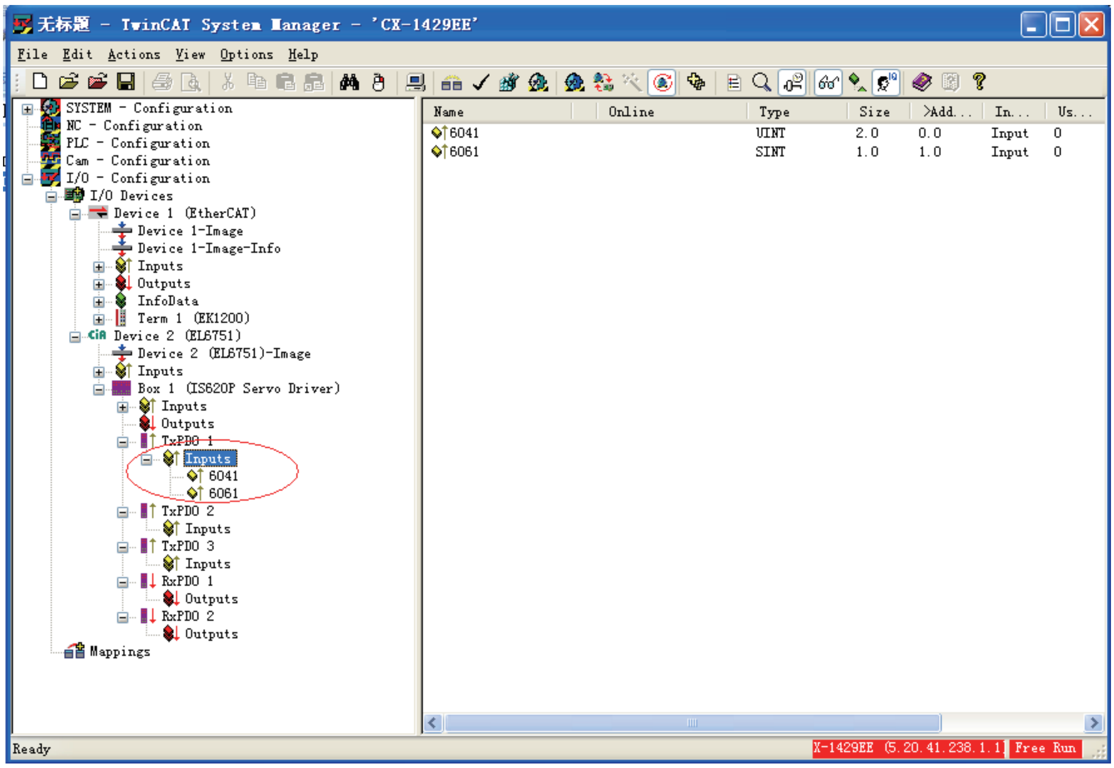


21) For the inserted variable 6061, select **INT8** (the object dictionary can be queried) for **Variable Type**, enter a large value for **Byte** of **Start Address** to prevent 6061h from being inserted in front of 6041h, enter a proper name, and click **OK**.

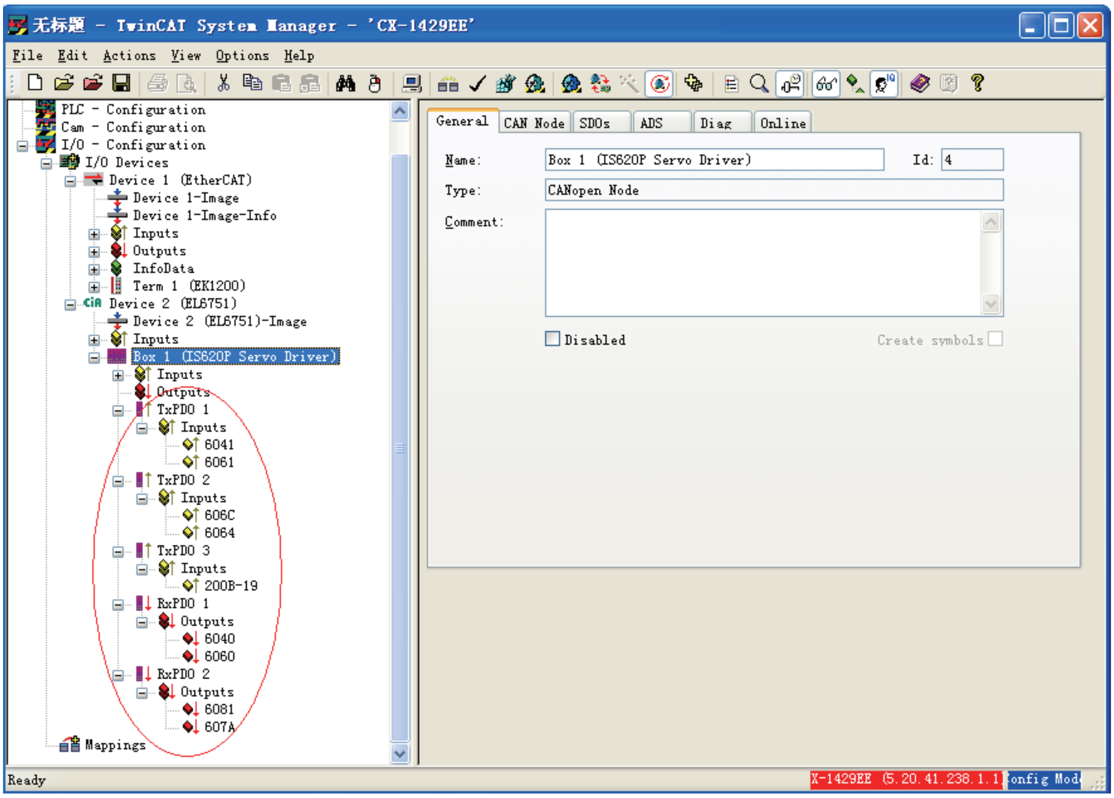


22) You can see that two objects are added to TPDO1. Note that the sequence of the two variables must be the same as that in Table B-1. Otherwise, the second variable must be deleted and inserted again and a large value must be entered in **2** marked in the figure in Step 21.

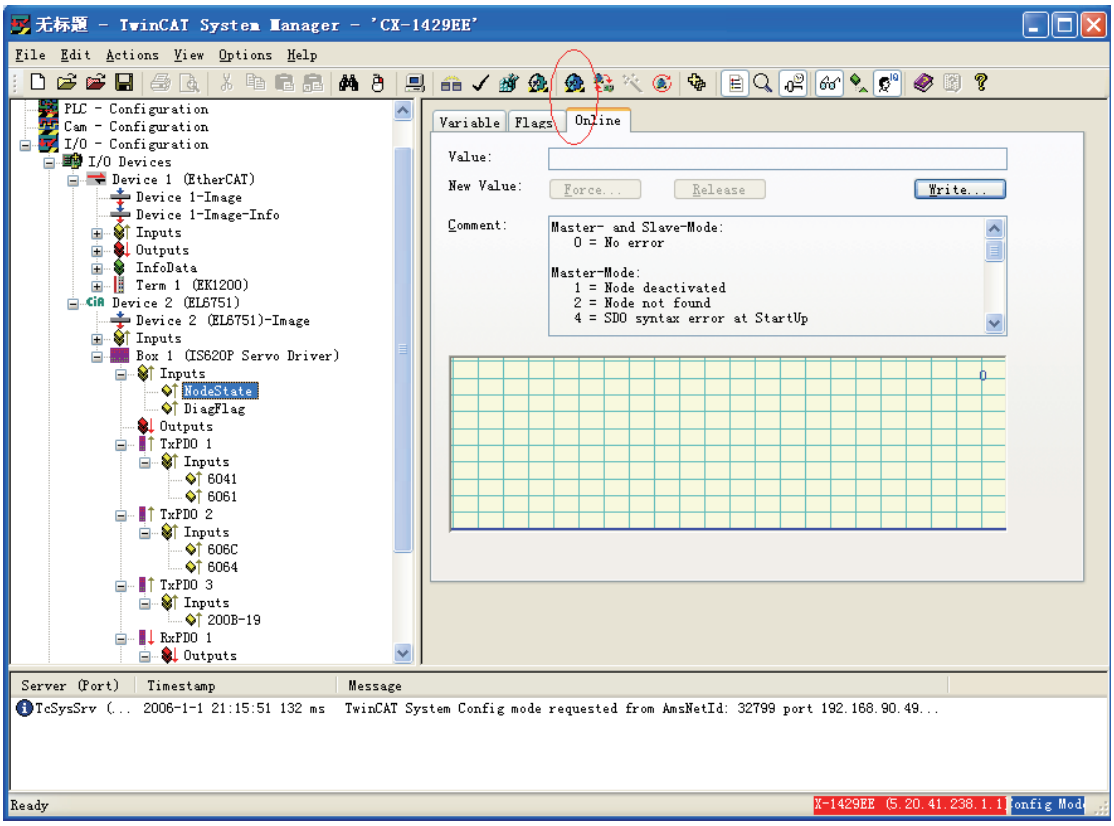
After making sure that the variable sequence is correct, choose **TPDO1** > **Inputs**, right-click, and choose **Recalc Address** to allocate addresses. This step must be performed. Otherwise, address chaos occurs.



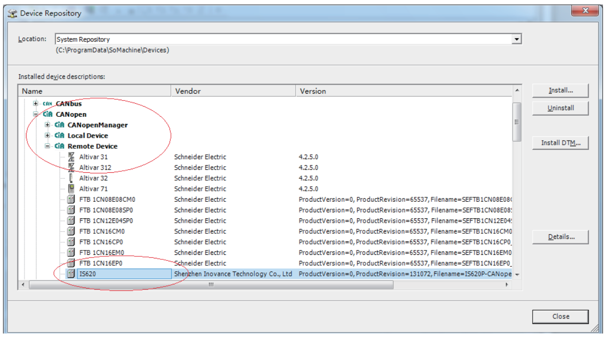
23) Repeat steps 18 to 22 for other PDOs. Add corresponding mapping variables based on Table B-1. After variables are added, the following window is displayed:



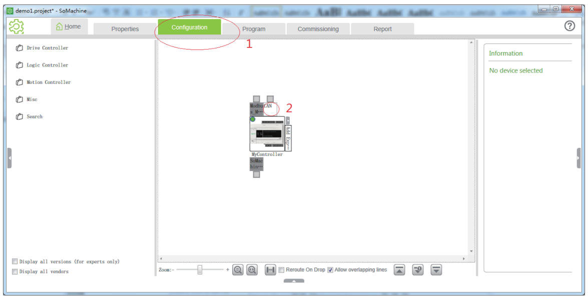
24) Click the icon marked in a red circle in the figure or press **Shift** + **F4**.



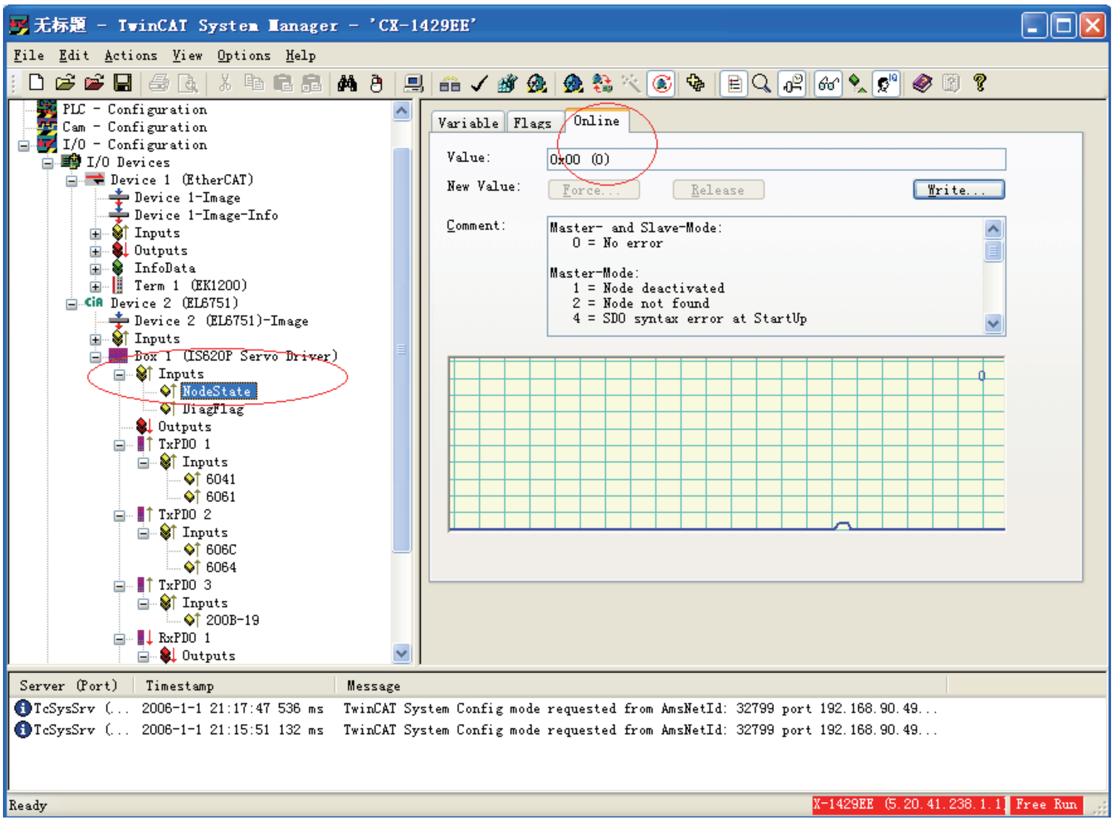
25) In the dialog box that is displayed, click **Yes**.



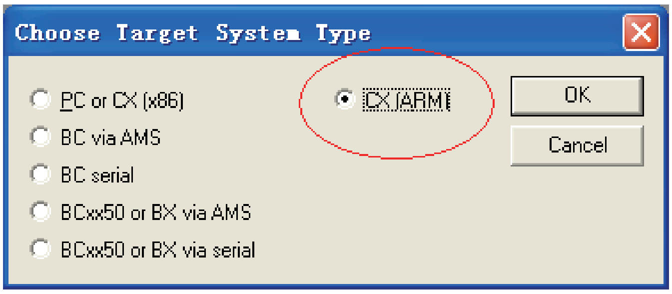
26) In the dialog box that is displayed, click **Yes**.



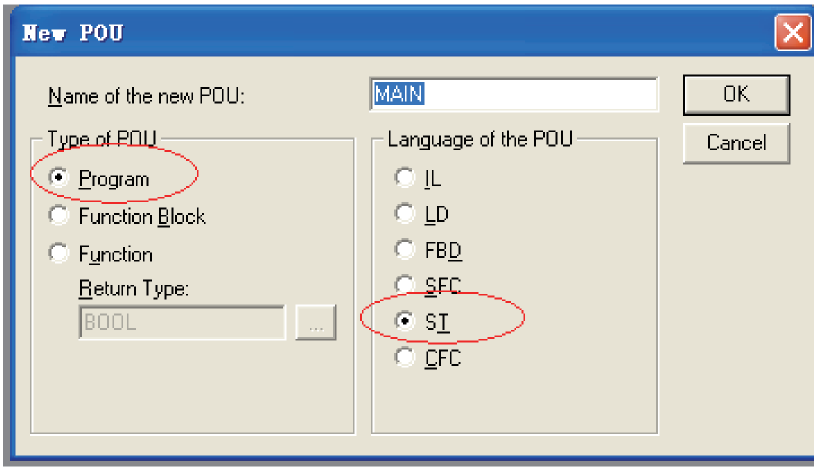
27) Select the box of IS620P, choose **Inputs** > **NodeState**. You can see that **Value** is **0** when the node status is **Online**, indicating that no fault occurs in the node.



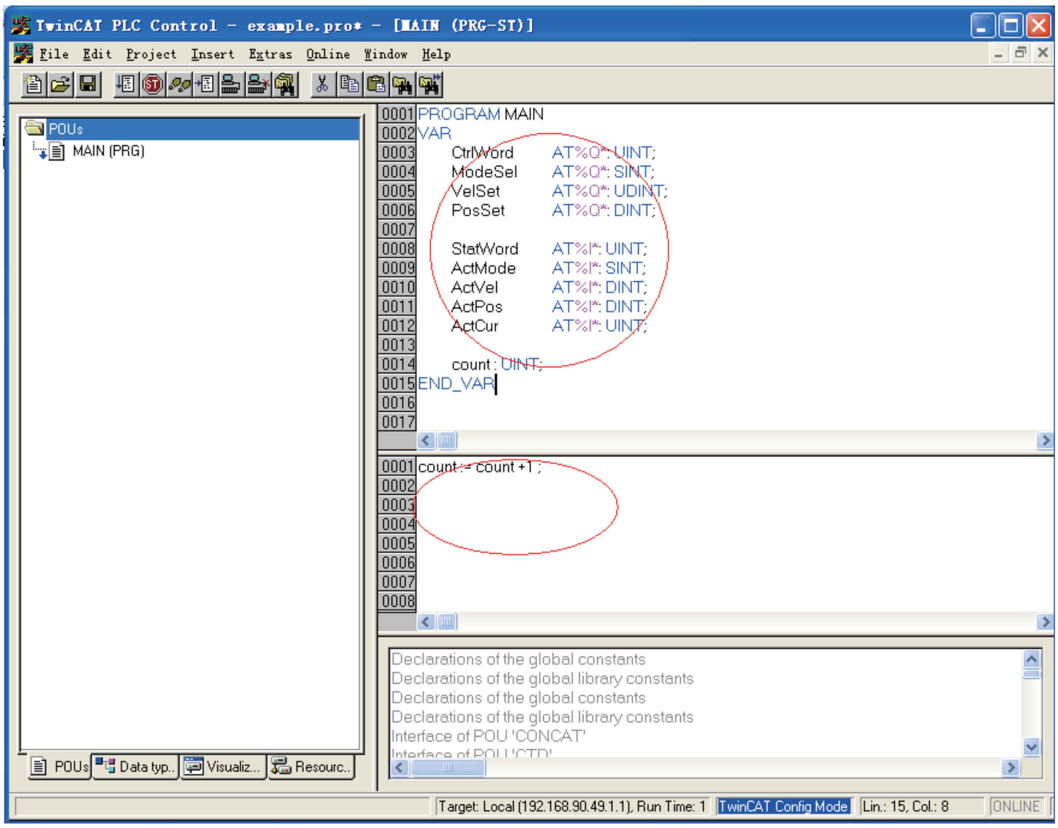
28) Open the TwinCAT PLC Control software and create a project. In the dialog box that is displayed, select **CX**.



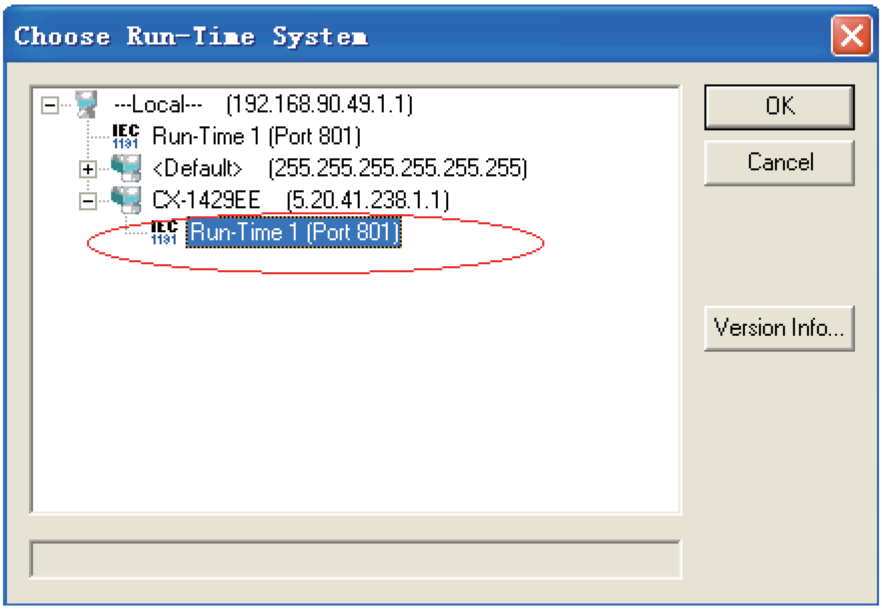
29) In the dialog box that is displayed, select the following options:



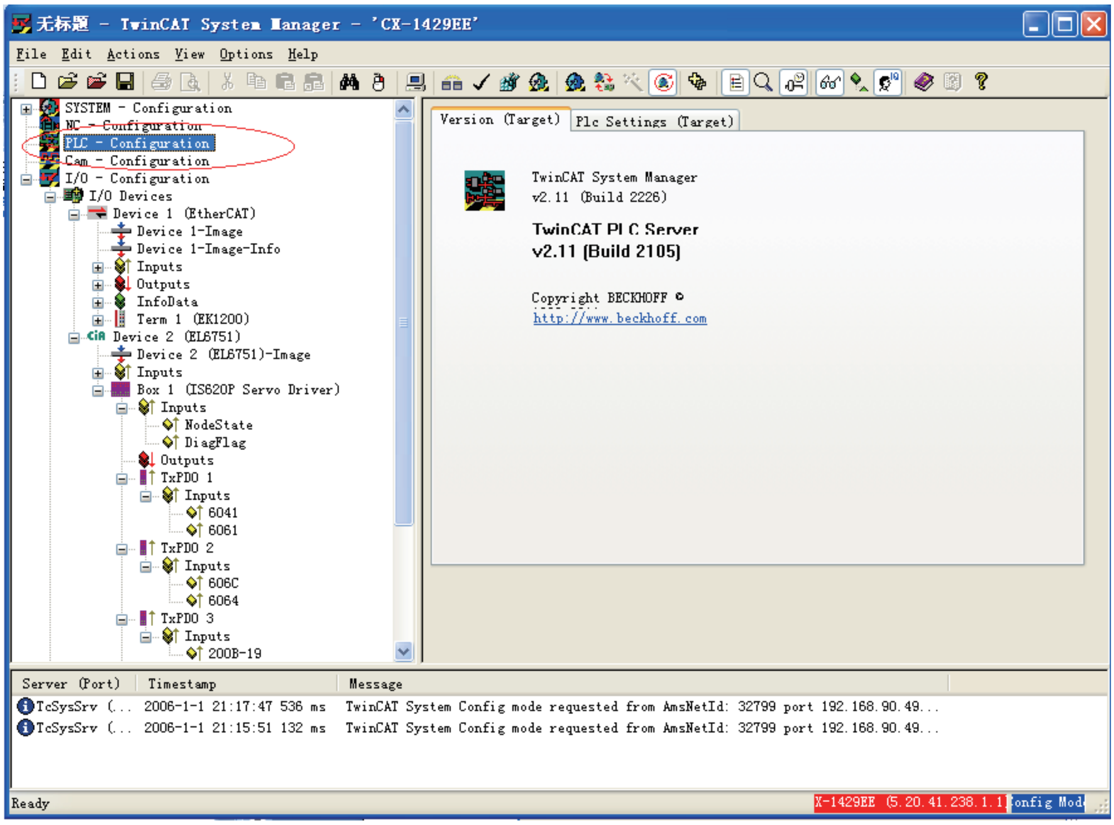
30) Enter corresponding variable definitions and PLC logic.



31) In the toolbar, choose **Online** > **Choose Run-time System**. In the dialog box that is displayed, select a port for the master node and click **OK**.



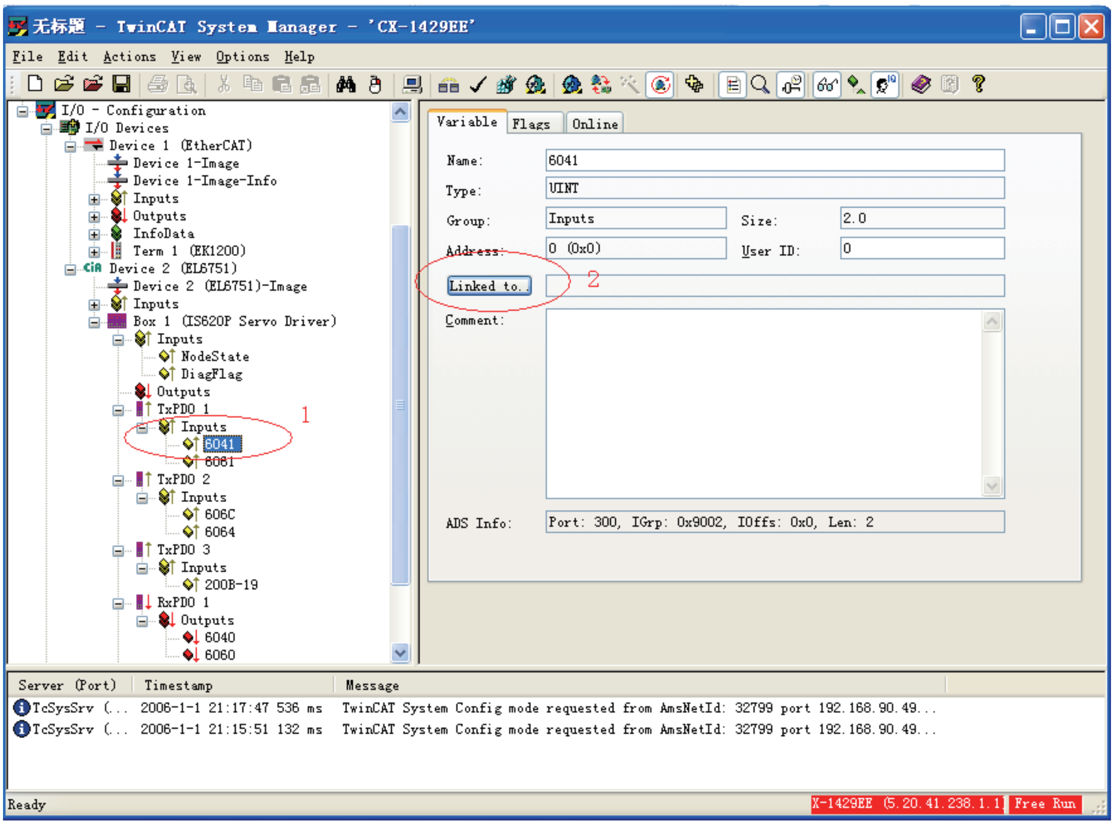
32) In TwinCAT System Manager, select **PLC - Configuration**, right-click and choose **Append PLC Project**, and select the created PLC program (.tpy).



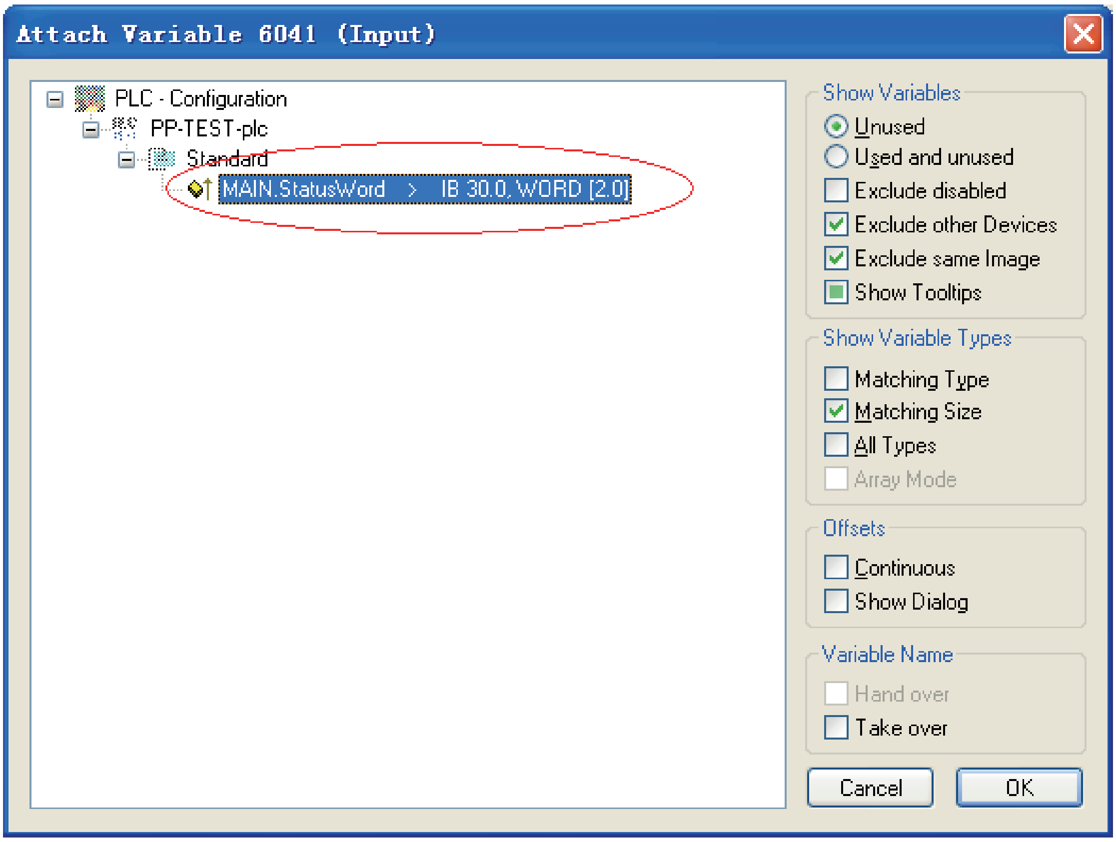
7

Application Cases

33) After the PLC program is added, select a PDO variable and click **Linked to** or directly double-click the variable to link the variable to the PLC program.

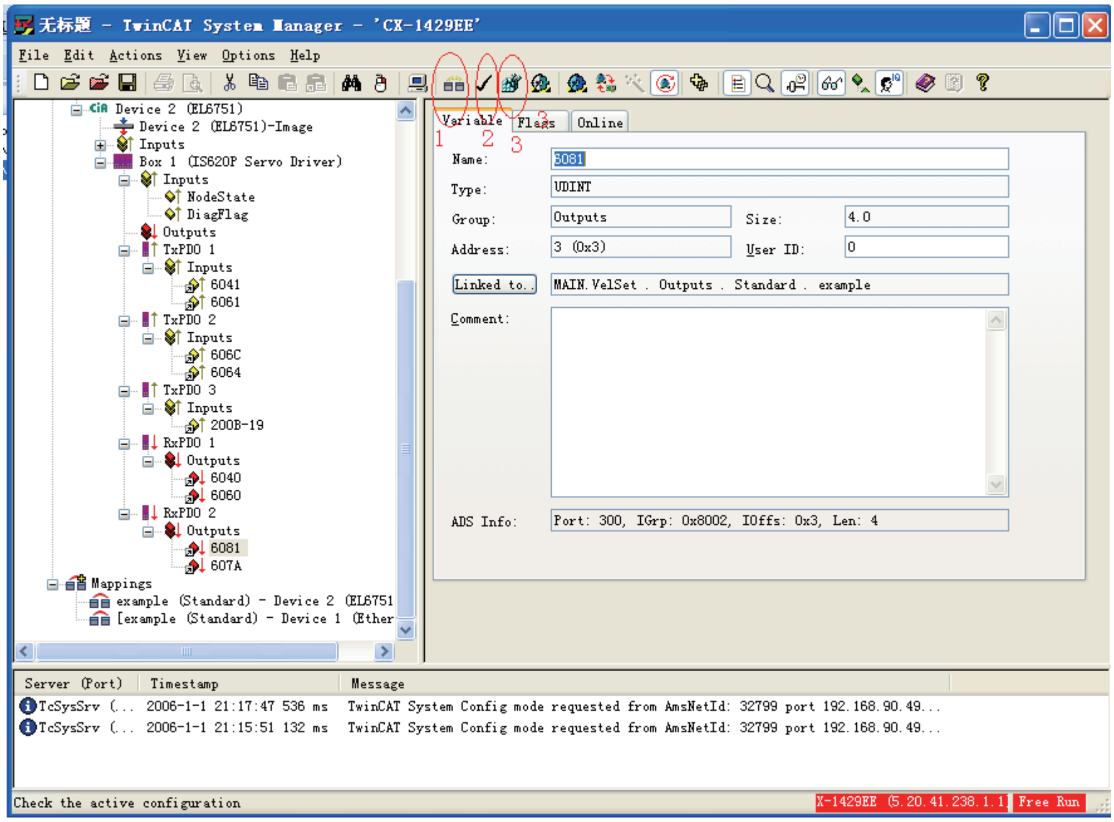


34) Select the corresponding PLC variable and click **OK**.

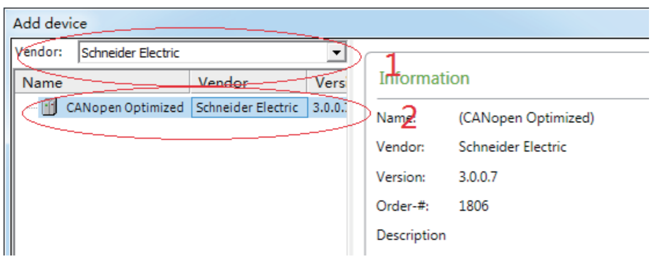


35) After the variable is linked, a small arrow pointing to the upper right part appears in the lower left part of the variable name. As shown in the following figure, the name of the variable not linked is displayed on the left and the name of the linked variable is displayed on the right.

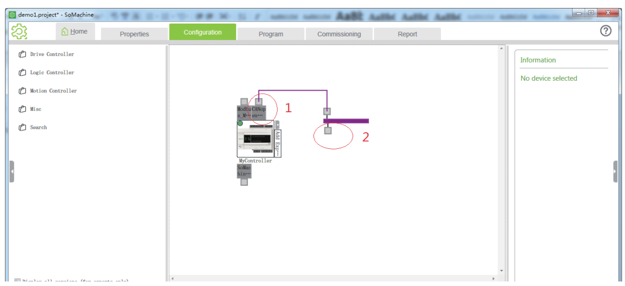
36) Click **Generate mapping**, **Check Configuration**, and **Activate Configuration** in turn, as shown in **1**, **2**, and **3** in the figure.



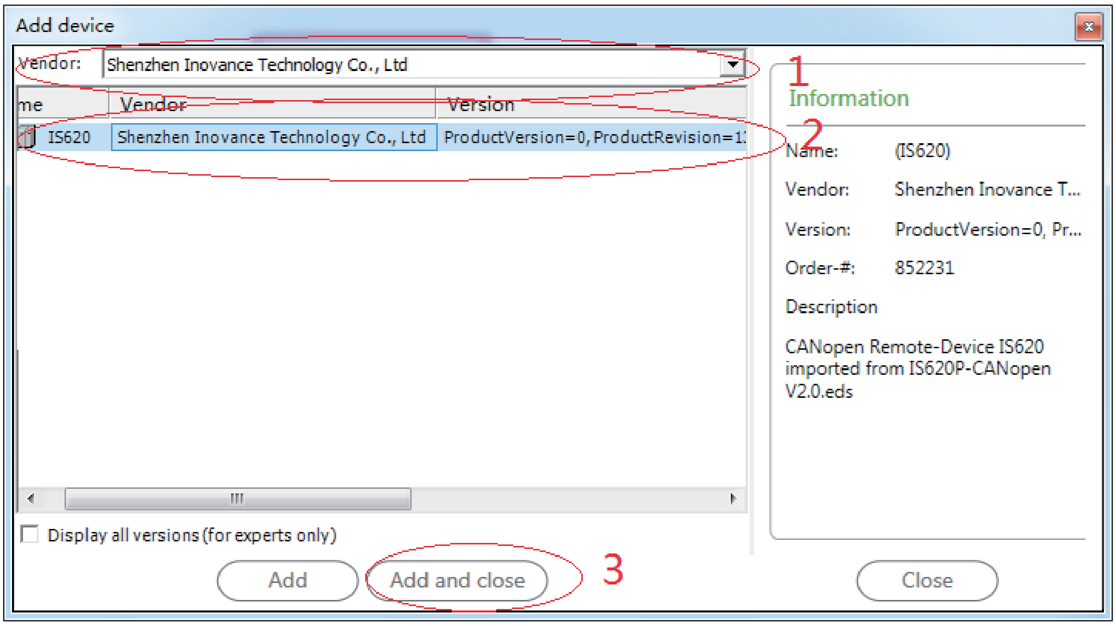
37) In the dialog box that is displayed. Click **OK**.



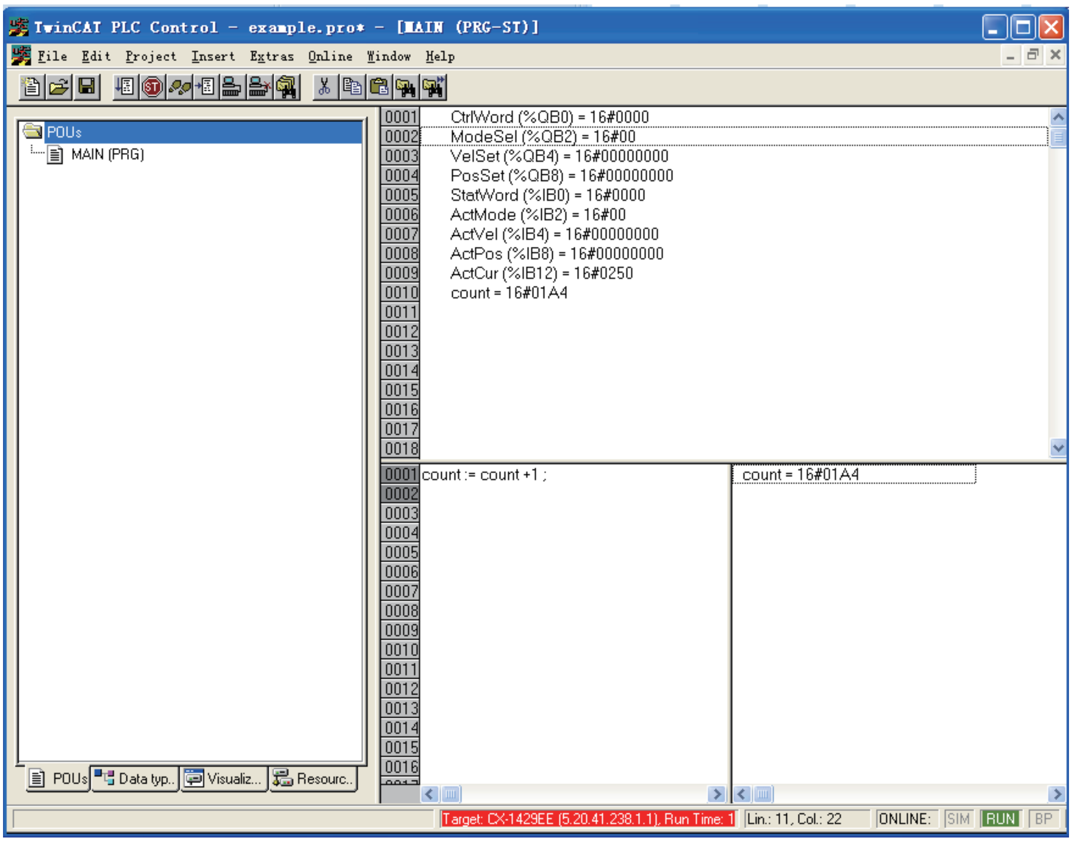
38) In the dialog box that is displayed. Click **OK**.



39) Open the project created using the TwinCAT PLC Control software and choose **Online** > **Login** or press **F11**. The following dialog box is displayed, click **Yes**.

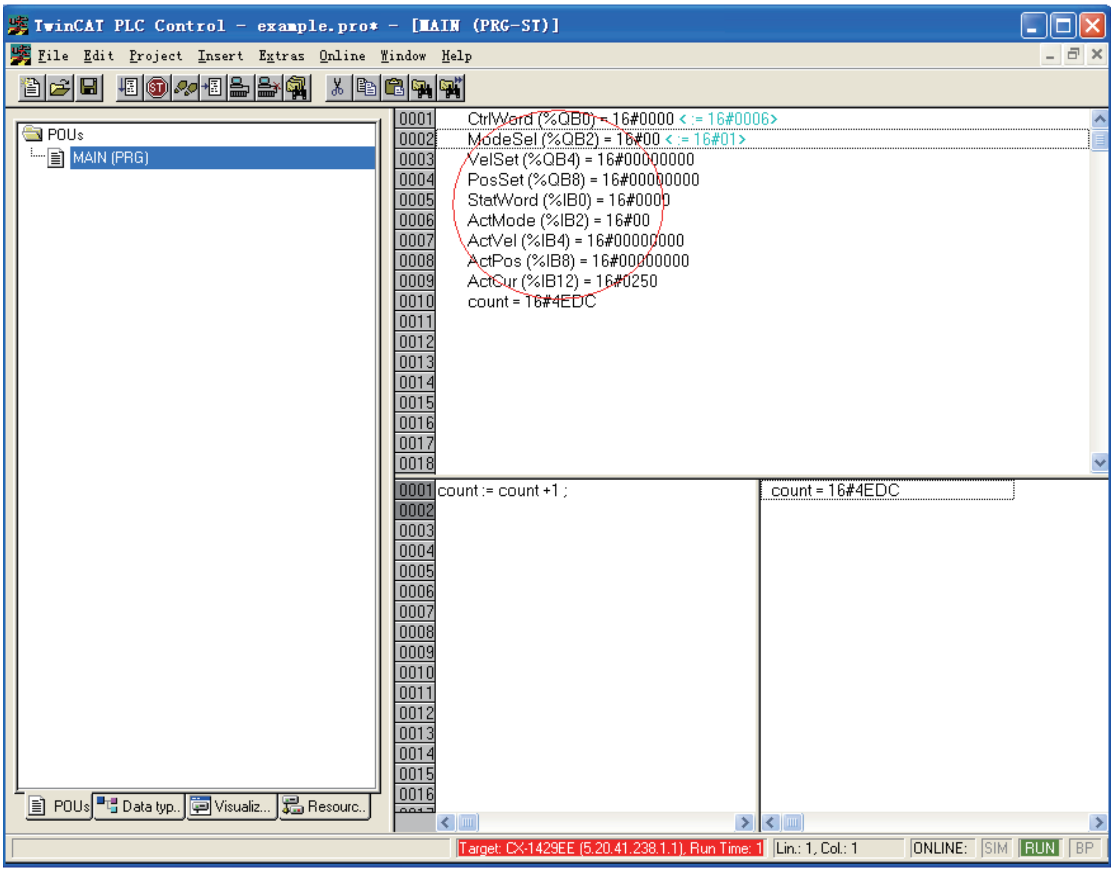


40) Choose **Online** > **Run** or press **F5** to run the PLC program.

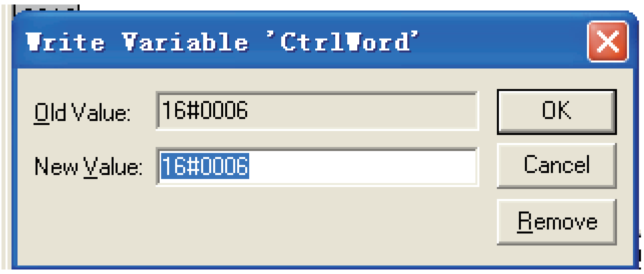


41) Perform forced write-in commissioning manually. The method is similar to that for a Schneider master node.

Double-click a variable defined in the circle marked in the figure and enter a value.



42) Click **OK**.



A new value included in square brackets appears behind the original variable. Choose **Online** > **Force Values** or press **F7** and write the value in a forced way.

Set 6060h to **1**, 6081h to **100**, and 607Ah to **10485760** (10 rounds) and set 6040h to **6**, **7**, **47(0x2f)**, and **63(0x3f)** in turn. The motor starts running.

NOTE

For the same variable, each time when a value is written, the "Force Values" reference is executed. You can enter values for different variables and execute the "Force Values" reference once.

When a new position or velocity reference is required, write the new reference and set 6040h to **47(0x2f)** and **63(0x3f)** in turn. The motor runs to the position according to the new reference no matter whether execution of the previous reference is complete.

To stop the motor, set 6040h to **0**.

Do not enter values forcibly. In the toolbar, choose **Online** > **Release Force** or press **Shift** + **F7**. Variables are no longer entered and follow the logic of the PLC program.

43) In the toolbar, choose **Online** > **Stop** to stop execution of the PLC program. Choose **Online** > **Logout** to continue to edit the PLC program or exit.

Memo NO. __________

Date / /

Chapter 8 Appendixes

Appendix A Mapping between PDO Mapping Objects and Functional Codes

For some master nodes where it is complex to configure the PDO mapping of slave nodes, you can directly modify the PDO mapping through functional codes. A mapping sub-index includes the number of mapping objects of the PDO. Other indexes are mapping objects. The format of mapping objects is as follows:

31 16	15 8	7 0
Index	Sub-index	Object Length
MLB		LSB

NOTE

The total length of an PDO mapping object cannot exceed 64 bits. After modifying the PDO mapping through functional codes, power on the device again.

Object	Name	Functional Code	Type
1400h-01h	COB-ID of RPDO1	H18-20	Uint32
1400h-02h	Transmission type of RPDO1	H18-22	Uint8
1401h-01h	COB-ID of RPDO2	H18-23	Uint32
1401h-02h	Transmission type of RPDO2	H18-25	Uint8
1402h-01h	COB-ID of RPDO3	H18-26	Uint32
1402h-02h	Transmission type of RPDO3	H18-28	Uint8
1403h-01h	COB-ID of RPDO4	H18-29	Uint32
1403h-02h	Transmission type of RPDO4	H18-31	Uint8
1600h-00h	Number of valid mapping objects of RPDO1	H18-34	Uint8
1600h-01h	Mapping object 1 of RPDO1	H18-35	Uint32
1600h-02h	Mapping object 2 of RPDO1	H18-37	Uint32
1600h-03h	Mapping object 3 of RPDO1	H18-39	Uint32
1600h-04h	Mapping object 4 of RPDO1	H18-41	Uint32
1600h-05h	Mapping object 5 of RPDO1	H18-43	Uint32
1600h-06h	Mapping object 6 of RPDO1	H18-45	Uint32
1600h-07h	Mapping object 7 of RPDO1	H18-47	Uint32
1600h-08h	Mapping object 8 of RPDO1	H19-00	Uint32
1601h-00h	Number of valid mapping objects of RPDO2	H19-02	Uint8
1601h-01h	Mapping object 1 of RPDO2	H19-03	Uint32
1601h-02h	Mapping object 2 of RPDO2	H19-05	Uint32
1601h-03h	Mapping object 3 of RPDO2	H19-07	Uint32

Object	Name	Functional Code	Type
1601h-04h	Mapping object 4 of RPDO2	H19-09	Uint32
1601h-05h	Mapping object 5 of RPDO2	H19-11	Uint32
1601h-06h	Mapping object 6 of RPDO2	H19-13	Uint32
1601h-07h	Mapping object 7 of RPDO2	H19-15	Uint32
1601h-08h	Mapping object 8 of RPDO2	H19-17	Uint32
1602h-00h	Number of valid mapping objects of RPDO3	H19-19	Uint8
1602h-01h	Mapping object 1 of RPDO3	H19-20	Uint32
1602h-02h	Mapping object 2 of RPDO3	H19-22	Uint32
1602h-03h	Mapping object 3 of RPDO3	H19-24	Uint32
1602h-04h	Mapping object 4 of RPDO3	H19-26	Uint32
1602h-05h	Mapping object 5 of RPDO3	H19-28	Uint32
1602h-06h	Mapping object 6 of RPDO3	H19-30	Uint32
1602h-07h	Mapping object 7 of RPDO3	H19-32	Uint32
1602h-08h	Mapping object 8 of RPDO3	H19-34	Uint32
1603h-00h	Number of valid mapping objects of RPDO4	H19-36	Uint8
1603h-01h	Mapping object 1 of RPDO4	H19-37	Uint32
1603h-02h	Mapping object 2 of RPDO4	H19-39	Uint32
1603h-03h	Mapping object 3 of RPDO4	H19-41	Uint32
1603h-04h	Mapping object 4 of RPDO4	H19-43	Uint32
1603h-05h	Mapping object 5 of RPDO4	H19-45	Uint32
1603h-06h	Mapping object 6 of RPDO4	H19-47	Uint32
1603h-07h	Mapping object 7 of RPDO4	H1A-00	Uint32
1603h-08h	Mapping object 8 of RPDO4	H1A-02	Uint32
1800h-01h	COB-ID of TPDO1	H1A-04	Uint32
1800h-02h	Transmission type of TPDO1	H1A-06	Uint8
1800h-03h	Inhibit time of TPDO1	H1A-07	Uint16
1800h-04h	Event timer of TPDO1	H1A-08	Uint16
1801h-01h	COB-ID of TPDO2	H1A-09	Uint32
1801h-02h	Transmission type of TPDO2	H1A-11	Uint8
1801h-03h	Inhibit time of TPDO2	H1A-12	Uint16
1801h-04h	Event timer of TPDO2	H1A-13	Uint16
1802h-01h	COB-ID of TPDO3	H1A-14	Uint32
1802h-02h	Transmission type of TPDO3	H1A-16	Uint8

Object	Name	Functional Code	Type
1802h-03h	Inhibit time of TPDO3	H1A-17	Uint16
1802h-04h	Event timer of TPDO3	H1A-18	Uint16
1803h-01h	COB-ID of TPDO4	H1A-19	Uint32
1803h-02h	Transmission type of TPDO4	H1A-21	Uint8
1803h-03h	Inhibit time of TPDO4	H1A-22	Uint16
1803h-04h	Event timer of TPDO4	H1A-23	Uint16
1A00h-00h	Number of valid mapping objects of TPDO1	H1A-24	Uint8
1A00h-01h	Mapping object 1 of TPDO1	H1A-25	Uint32
1A00h-02h	Mapping object 2 of TPDO1	H1A-27	Uint32
1A00h-03h	Mapping object 3 of TPDO1	H1A-29	Uint32
1A00h-04h	Mapping object 4 of TPDO1	H1A-31	Uint32
1A00h-05h	Mapping object 5 of TPDO1	H1A-33	Uint32
1A00h-06h	Mapping object 6 of TPDO1	H1A-35	Uint32
1A00h-07h	Mapping object 7 of TPDO1	H1A-37	Uint32
1A00h-08h	Mapping object 8 of TPDO1	H1A-39	Uint32
1A01h-00h	Number of valid mapping objects of TPDO2	H1A-41	Uint8
1A01h-01h	Mapping object 1 of TPDO2	H1A-42	Uint32
1A01h-02h	Mapping object 2 of TPDO2	H1A-44	Uint32
1A01h-03h	Mapping object 3 of TPDO2	H1A-46	Uint32
1A01h-04h	Mapping object 4 of TPDO2	H1B-00	Uint32
1A01h-05h	Mapping object 5 of TPDO2	H1B-02	Uint32
1A01h-06h	Mapping object 6 of TPDO2	H1B-04	Uint32
1A01h-07h	Mapping object 7 of TPDO2	H1B-06	Uint32
1A01h-08h	Mapping object 8 of TPDO2	H1B-08	Uint32
1A02h-00h	Number of valid mapping objects of TPDO3	H1B-10	Uint8
1A02h-01h	Mapping object 1 of TPDO3	H1B-11	Uint32
1A02h-02h	Mapping object 2 of TPDO3	H1B-13	Uint32
1A02h-03h	Mapping object 3 of TPDO3	H1B-15	Uint32
1A02h-04h	Mapping object 4 of TPDO3	H1B-17	Uint32
1A02h-05h	Mapping object 5 of TPDO3	H1B-19	Uint32
1A02h-06h	Mapping object 6 of TPDO3	H1B-21	Uint32
1A02h-07h	Mapping object 7 of TPDO3	H1B-23	Uint32
1A02h-08h	Mapping object 8 of TPDO3	H1B-25	Uint32

Object	Name	Functional Code	Type
1A03h-00h	Number of valid mapping objects of TPDO4	H1B-27	Uint8
1A03h-01h	Mapping object 1 of TPDO4	H1B-28	Uint32
1A03h-02h	Mapping object 2 of TPDO4	H1B-30	Uint32
1A03h-03h	Mapping object 3 of TPDO4	H1B-32	Uint32
1A03h-04h	Mapping object 4 of TPDO4	H1B-34	Uint32
1A03h-05h	Mapping object 5 of TPDO4	H1B-36	Uint32
1A03h-06h	Mapping object 6 of TPDO4	H1B-38	Uint32
1A03h-07h	Mapping object 7 of TPDO4	H1B-40	Uint32
1A03h-08h	Mapping object 8 of TPDO4	H1B-42	Uint32

Revision History

Date	Version	Change Description
April 2017	A00	Issued the first version.
October 2018	A01	Update LOGO.

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
Service Hotline: 400-777-1260
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
Service Hotline: 400-777-1260
http: //www.inovance.com