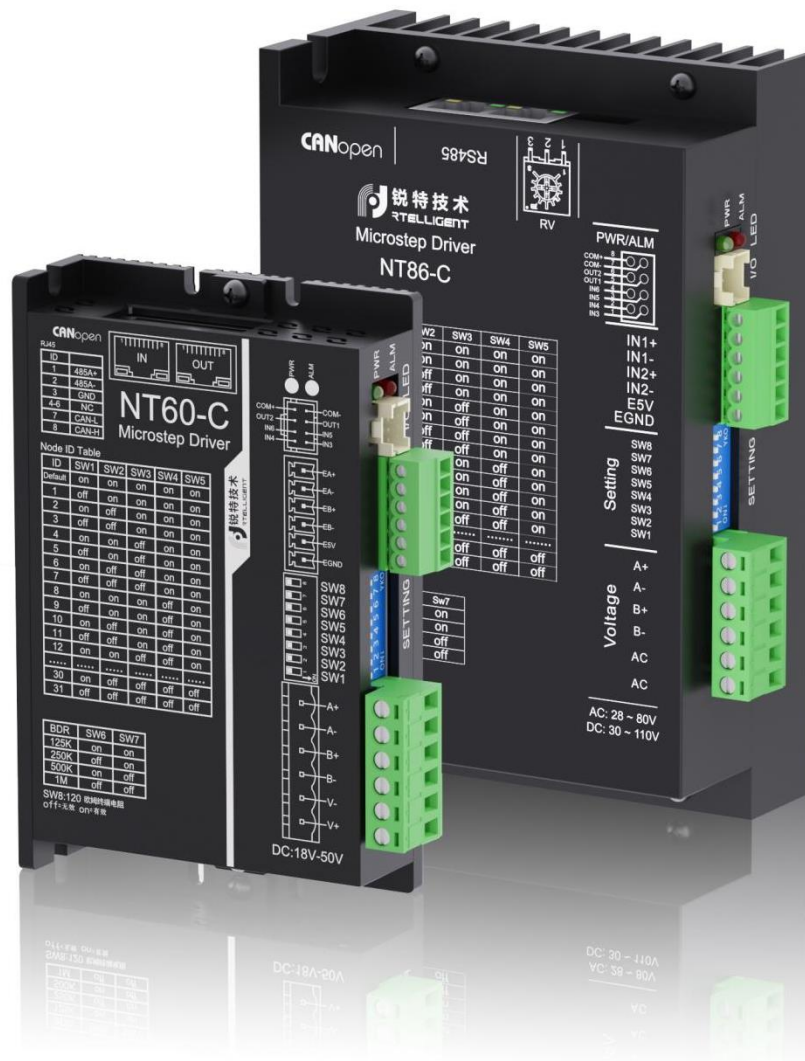


NT - CANpen Series Stepper Driver

User Manual



Shenzhen Rtelligent Technology Co., Ltd

Revision History

Date	Version	Description
2024.09.06	V2.1	Version 2 product update.
2024.09.13	V2.2	Chapter 1: Modify the signal definition of communication port.

1. Driver Description

1.1. Product Introduction

NT60-C and NT86-C are high-performance CAN bus controlled stepper motor drivers that integrate intelligent motion controller functions and built-in S-shaped acceleration and deceleration commands. Support the position mode (PP), speed mode (PV) and homing mode (HM) in CiA402 protocol. The standardized protocol makes the whole control system more stable and reliable. At the same time, the field wiring can be simplified to effectively avoid the instruction pulse interference problem existing in traditional pulse drivers.

1.1.1. Features

- ◆ Programmable small-size stepper motor driver
- ◆ Operating voltage:
NT60-C: 18~50VDC
NT86-C: 30~110VDC or 24~80VAC
- ◆ Control: CiA301, CiA402 protocol
- ◆ Communication: CAN
- ◆ Maximum phase current output:
NT60-C: 5A/phase (Sine peak)
NT86-C: 7A/phase (Sine peak)
- ◆ Digital IO port:
6-way photoelectric isolated digital signal input: IN1 and IN2 are 5V differential inputs, which can also be connected as 5V single-ended inputs; IN3 to IN6 are 24V single-ended inputs, with a common anode connection method.
2-way photoelectric isolated digital signal output, maximum withstand voltage 30V, maximum input or output current 100mA, common cathode connection.

1.2. Safety Precautions

- ◆ The transportation, installation, use, or maintenance of this product must be carried out by personnel with professional qualifications and familiarity with the above operations.
- ◆ In order to minimize potential safety hazards, you should comply with all local and national safety regulations when using this device. Different regions have different safety regulations, and you should ensure that the installation and use of the device comply with the regulations in your area.
- ◆ System errors can also cause damage to equipment or personal injury. We do not guarantee that this product is suitable for your specific application, and we cannot assume responsibility for the reliability of your system design.
- ◆ Please read all relevant documents before installation and use. Incorrect use may cause equipment damage or personal injury. Please strictly follow the relevant technical requirements during installation. Please make sure to confirm the grounding of all devices in the system. Non grounded systems cannot guarantee electrical safety.
- ◆ Some components inside the product may be damaged due to external static electricity. Operators should ensure that they have no static electricity before coming into contact with the product and avoid touching objects that are prone to static electricity (such as chemical fibers, plastic films, etc.).
- ◆ If your device is placed in the control cabinet, please close the outer cover or cabinet door during operation, otherwise it may cause equipment damage or personal injury.
- ◆ It is strictly prohibited to hot plug cables during system operation, as the arc generated by hot plugging may pose a hazard to both operators and equipment.
- ◆ After turning off the power, please wait at least 3 seconds before touching the product or removing the wiring. Capacitive devices may still store hazardous electrical energy after a power outage and require a certain amount of time to release. To ensure safety, a multimeter can be used to measure before coming into contact with the product.
- ◆ Please follow the important safety tips provided in this manual, including clear warning symbols for potential safety hazards. Before installation, operation, and maintenance, you should read and familiarize yourself with these instructions.

Note: The purpose of this section is to inform users of necessary safety precautions and reduce risks that may endanger personal and equipment safety. Misestimation of the importance of safety precautions may result in serious losses or render equipment unusable.

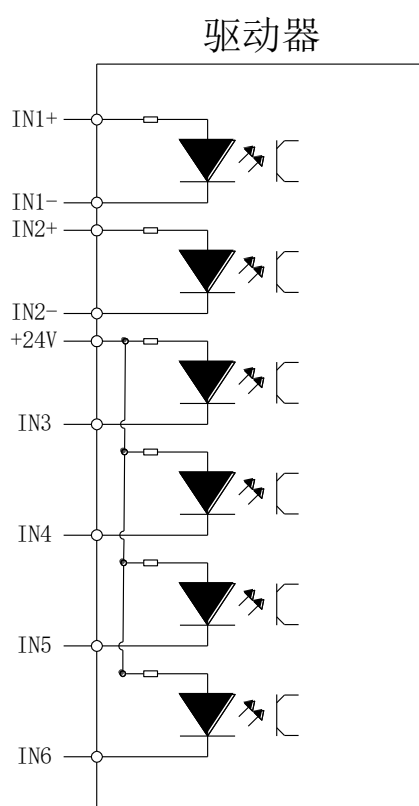
1.3. Digital Input and Output Ports

The NT series CANopen stepper driver has 6 digital input ports and 2 digital output ports. The digital input and output ports can be freely configured with various functions according to their own application needs.

1.3.1. Digital Input Port

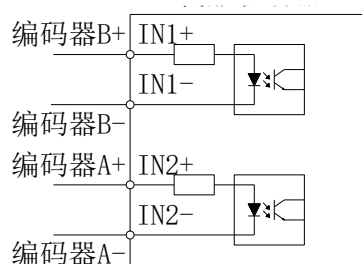
◆ **Note:** IN1+/IN1- and IN2+/IN2- are 5V input terminals, do not directly connect input signals above this voltage, otherwise the driver will be damaged.

The diagram of the input port is shown below, and the user can wire the system according to the diagram.



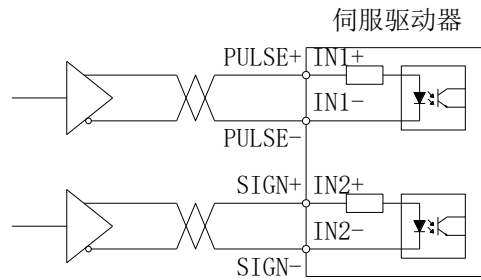
1. IN1+/IN1-, IN2+/IN2- differential input terminals

(1) Closed-loop control: the external motor encoder constitutes a closed-loop system

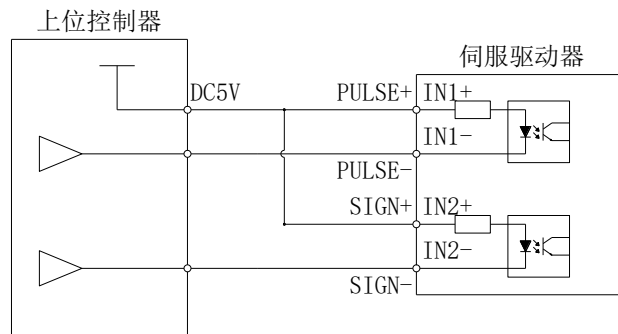


(2) Pulse control: external pulse + direction difference signal

A. 5V differential input

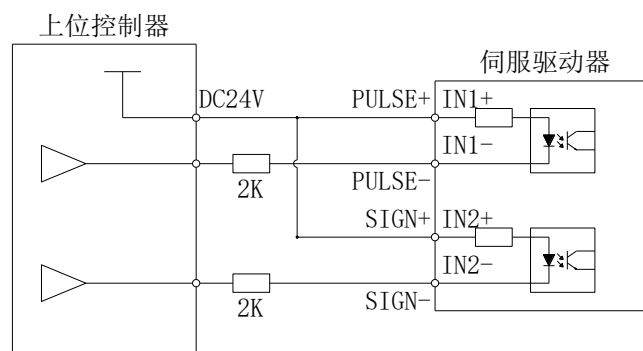


B. 5V single-ended input

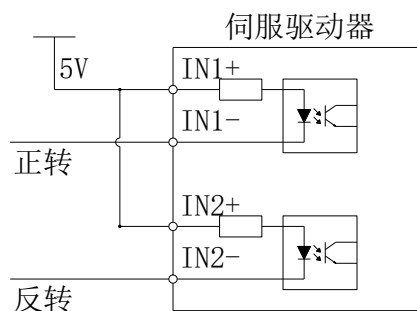


C. 24V single-ended input

◆ Note: When using 24V input, string the 2K current limiting resistance externally, otherwise the driver will be damaged.



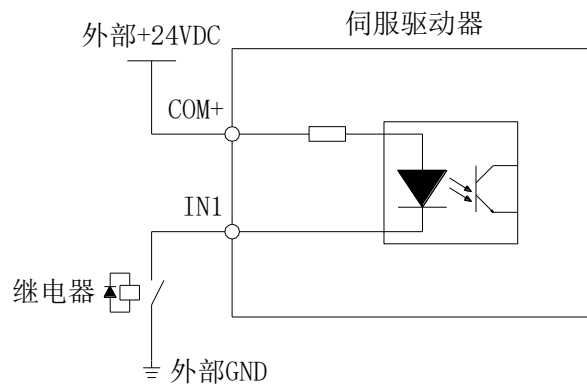
(3) Ordinary IO control: using single-ended connection method, external universal input signal, such as jog forward/reverse signal:



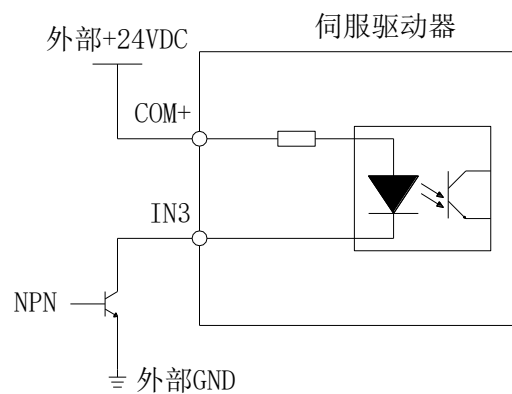
2. IN3~IN6 single-ended input terminals

Taking IN3 as an example, IN3~IN6 have the same interface circuit..

(1) When the upper device is the relay output

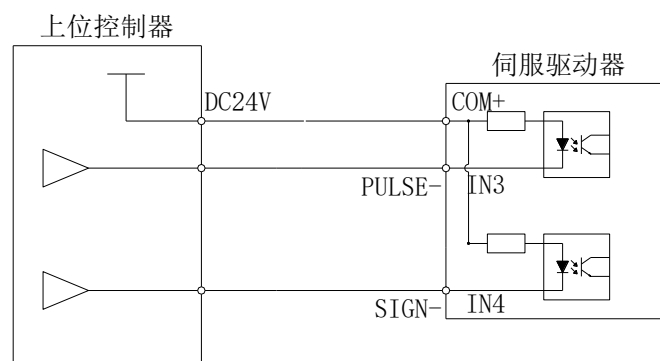


(2) When the upper device is the open collector output



◆ Note: PNP input is not supported.

(3) Pulse and direction signal input using IN3 and IN4 terminals



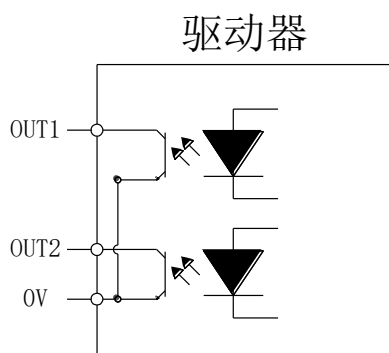
◆ If conditions permit, please prioritize using IN1 and IN2 as input terminals for pulse-direction signal.

1.3.2. Digital Output Port

The NT series CANopen stepper driver contains two photoelectric isolation output signals.

- ◆ OUT1 has an output current capacity of 100mA.
- ◆ OUT2 has an output current capacity of 100mA.

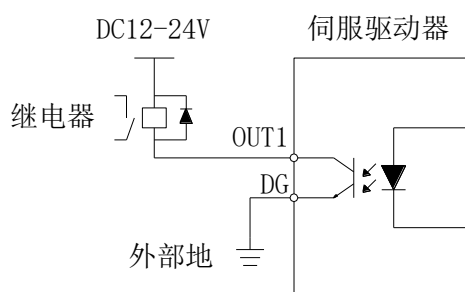
By default, all the digital output ports are normally open. You can use NTConfigurator or RTConfigurator debugging software to change the polarity of the output ports. The diagram of the output port is shown below, and the user can wire the system according to the diagram.



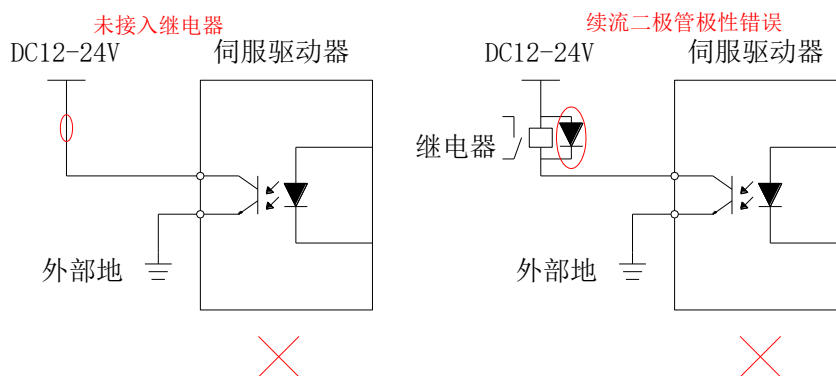
Taking OUT1 as an example, OUT1~OUT2 have the same interface circuit.

1. When the upper device is the relay input

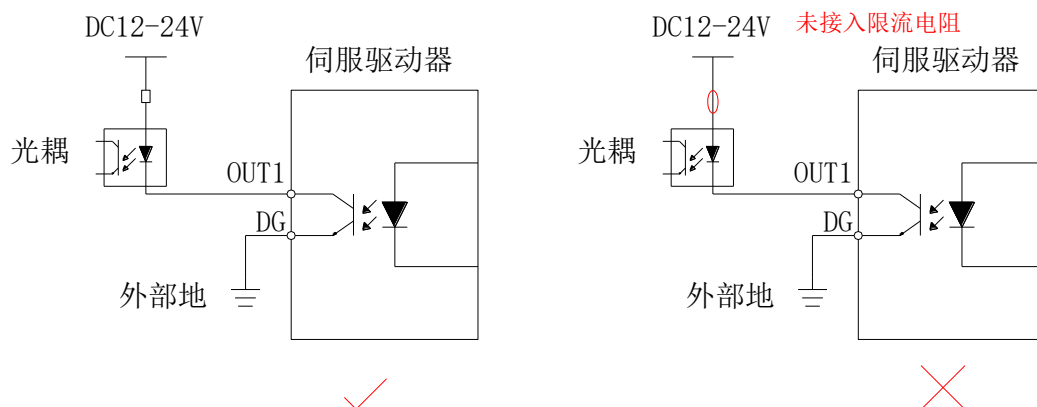
Correct wiring diagram:



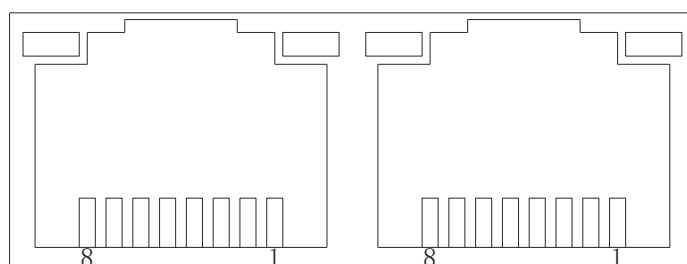
Error wiring diagram:



2. When the upper device is the optocoupler input



1.4. CAN Communication Terminal



Signal name		Pin	Description
Communication signal	RS485+	1	RS485 communication port
	RS485-	2	
	GND	3	GND signal
	-	4	-
	-	5	
	-	6	
	CAN_L	7	CAN communication port
	CAN_H	8	

1.5. CAN Communication Baud Rate Setting

Baud rate	SW6	SW7
125Kbps	ON	ON
250Kbps	OFF	ON
500Kbps	ON	OFF
1Mbps	OFF	OFF

1.6. CAN Communication ID Setting

In the same network, each slave has a unique address.

Slave ID	SW1	SW2	SW3	SW4	SW5
0x2150 object settings	ON	ON	ON	ON	ON
1	OFF	ON	ON	ON	ON
2	ON	OFF	ON	ON	ON
3	OFF	OFF	ON	ON	ON
4	ON	ON	OFF	ON	ON
.....					
30	ON	OFF	OFF	OFF	OFF
31	OFF	OFF	OFF	OFF	OFF

ON = 0, OFF = 1

Slave address = $SW1 + SW2*2 + SW3*4 + SW4*8 + SW5*16$

Calculation example:

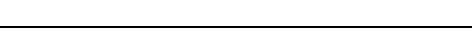
If: SW1 = OFF, SW2 = ON, SW3 = ON, SW4 = ON, SW5 = ON

So: Slave address = $1 + 0*2 + 0*4 + 0*8 + 0*16 = 1$

1.7. CAN Communication Terminal Matching Resistance

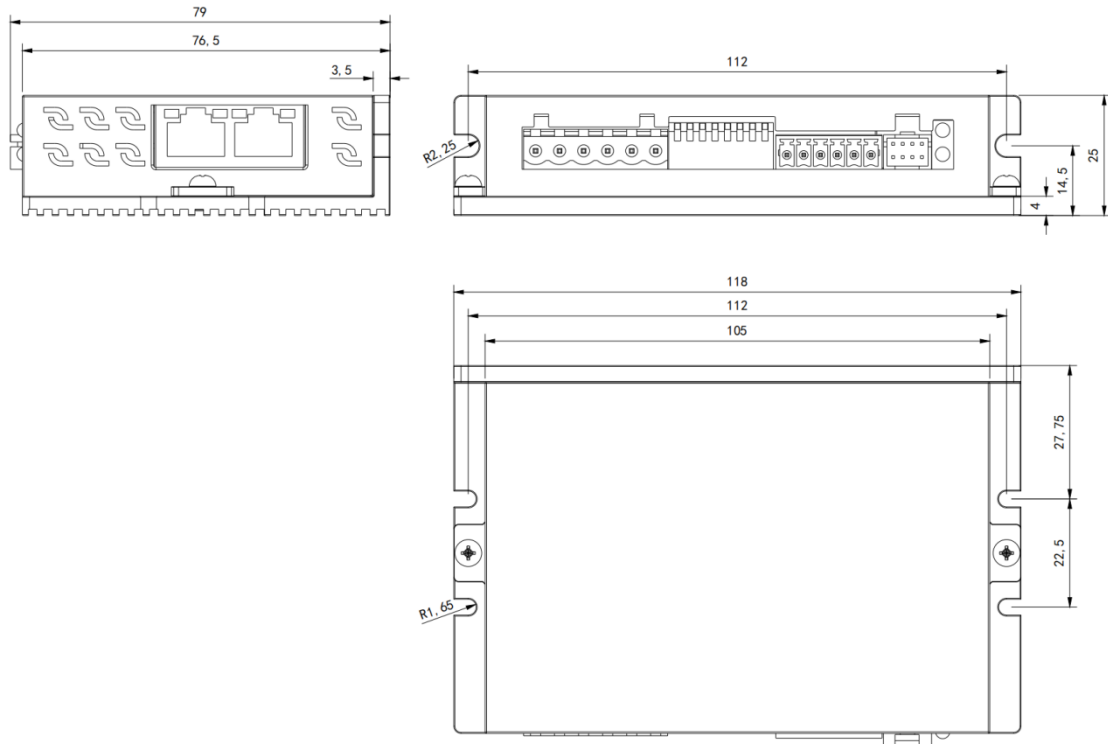
120Ω terminal resistance	SW8
Invalid	OFF
Effective	ON

1.8. Alarm Code

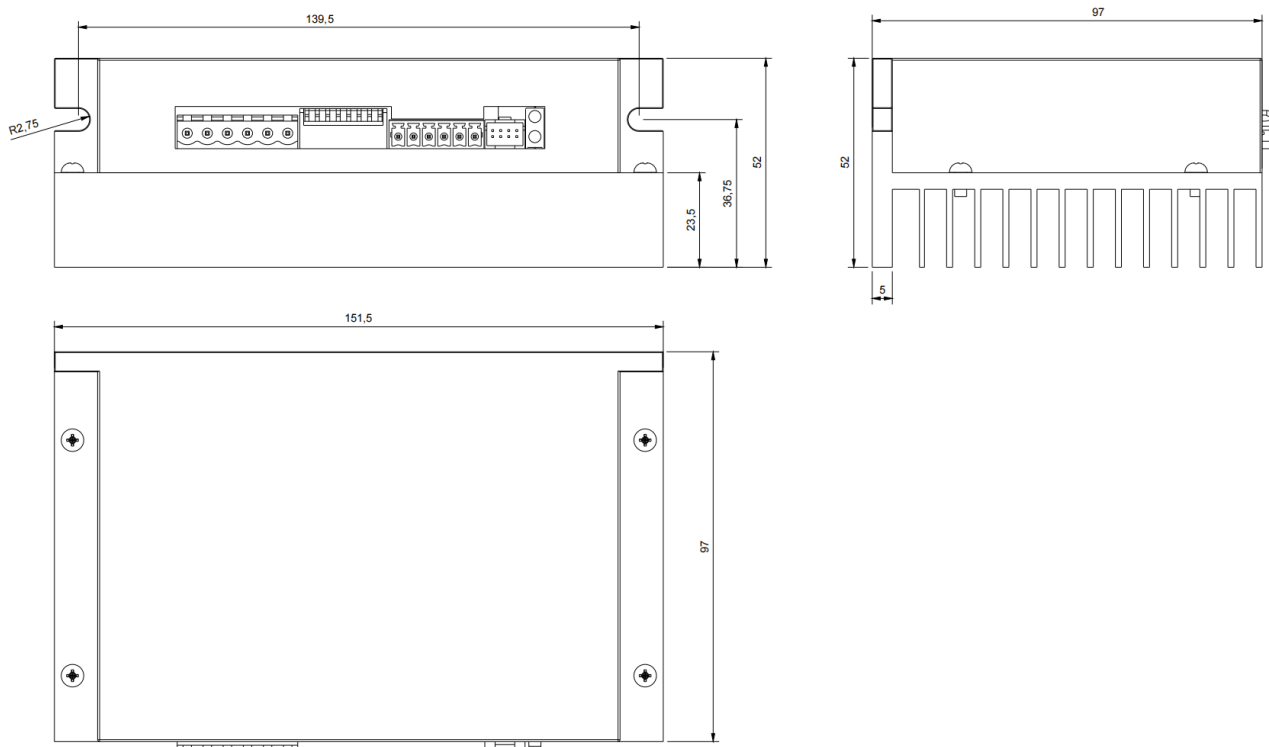
LED status		Driver status
	The green light is on	Driver is not enabled
	Flashing green light	Driver works
	1 green, 1 red	Driver overcurrent
	1 green, 2 red	Driver input power overvoltage
	1 green, 3 red	Driver internal voltage error
	1 green, 4 red	Encoder out-of-tolerance alarm
	1 green, 5 red	Encoder error
	1 green, 6 red	Parameter check error
	1 green, 7 red	Motor phase loss alarm
	1 green, 8 red	Command pulse direction is abnormal
	1 green, 9 red	CAN receives a cache overflow error
	1 green, 10 red	CAN sends a cache overflow error
	1 green, 11 red	Other error

1.9. Mechanical Size

1.9.1. NT60-C Mechanical Size



1.9.2. NT86-C Mechanical Size



1.10. Accessories

1.10.1. X1 Universal IO Signal Line

All 8 signal ports are drawn out with a shielded cable for easy customer wiring.

Model	Length (Unit: meter)	Picture
CNT60-250	0.25	
CNT60-500	0.5	
CNT60-750	0.75	
CNT60-1000	1	

1.10.2. CAN Extension Line

Network cable conforming to the CAT 6 standard.

Model	Length (Unit: meter)	Picture
CRJ45-250	0.25	
CRJ45-500	0.5	
CRJ45-750	0.75	
CRJ45-1000	1	

1.11. Anti-interference Countermeasures of Electrical Wiring

To suppress interference, please take the following measures:

- ◆ The length of the command input cable should be less than 3m, and the encoder cable should be less than 20m.
- ◆ Use thick wires as much as possible for grounding wiring. (Above 2.0mm²)
- ◆ Please use a noise filter to place radio frequency interference. When using in a civil environment where the power supply interference noise is strong, please install a noise filter on the input side of the power cord.
- ◆ In order to place the malfunction caused by electromagnetic interference, the following treatment methods can be adopted:
 - ① Install the upper computer device and the noise filter near the servo driver as far as possible.
 - ② Install surge suppressors on the coils of relays, screw tubes, and electromagnetic contactors.
 - ③ When wiring, please lay the strong current line separately from the weak current line, and keep an interval of more than 30cm. Do not put them in the same pipe or bundle them together.
 - ④ Do not share power supply with electric welders, electrical discharge machining equipment, etc. When there is a high-frequency generator nearby, install a noise filter on the input side of the power cord.

2. Communication Network Configuration

2.1. CANopen Protocol Overview

CANopen is an application layer protocol of a network transmission system based on CAN serial bus, following the ISO/OSI standard model. Different devices in the network exchange data with each other through object dictionaries or objects. The master node can obtain or modify data in the object dictionary lists of other nodes through process data objects (PDO) or service data objects (SDO).

2.1.1. Object Dictionary

The object dictionary is the most important part of the device specification. It is an ordered collection of a set of parameters and variables, including all parameters of device description and device network status. A set of objects that can be accessed in an orderly and predefined way through the network.

The CANOpen protocol uses an object dictionary with a 16-bit index and an 8-bit sub-index. The structure of the object dictionary is shown in the following table.

Index	Object
0000h	Unused
0001h~001Fh	Static data types (Standard data types, such as Boolean, Integer16)
0020h~003Fh	Complex data types (Predefined structures composed of simple types such as PDOCommPar, SDOParmeter)
0040h~005Fh	Complex data types 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	Reserve
1000h~1FFFh	Communication sub-protocol area (Such as device type, error register, number of supported PDOs)
2000h~5FFFh	Manufacturer-specific sub-protocol area (Such as function code mapping)
6000h~9FFFh	Standard equipment sub-protocol area (Such as DSP-402 protocol)
A000h~FFFFh	Reserve

2.1.2. Communication Object

1. CANopen specifications that the NT series CAN bus stepper drivers follow:

- ◆ Follow CAN 2.0A standard
- ◆ Follow CANopen standard protocol DS 301 V4.02
- ◆ Follow CANopen standard protocol DSP 402 V2.01

2. CANopen communication services supported by NT series CAN bus stepper drivers:

- ◆ Network Management Object (NMT): NMT Slave service
- ◆ Device monitoring: heartbeat message
- ◆ Process Data Object (PDO): Up to 4 TxPDO and 4 RXPDO can be configured. PDO transmission type: support event trigger, event trigger, synchronous cycle, synchronous acyclic
- ◆ Service Data Object (SDO)
- ◆ Emergency message (EMCY)

2.1.3. Communication Object Identifier

The communication object identifier (COB-ID) specifies the priority of the object and the identification of the communication object in the communication process. The COB-ID corresponds to the 11-bit frame ID of CAN2.0A. The 11-bit COB-ID consists of two parts, a 4-bit object function code and a 7-bit node address, as follows:

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

Each CANopen communication object has a default COB-ID, which can be read through SDO, and some can be modified through SDO. Object list:

Communication object	Function code	Node address	COB-ID	Object dictionary index
NMT network management	0000b	0	0x000	-
Synchronization object	0001b	0	0x080	1005h/1006h
Emergency message object	0001b	1~127	80h + Node ID	1014h
TPDO1	0011b	1~127	180h + Node ID	1800h
RPDO1	0100b	1~127	200h + Node ID	1400h
TPDO2	0101b	1~127	280h + Node ID	1801h
RPDO2	0110b	1~127	300h + Node ID	1401h
TPDO3	0111b	1~127	380h + Node ID	1802h

RPDO3	1000b	1~127	400h + Node ID	1402h
TPDO4	1001b	1~127	480h + Node ID	1802h
RPDO4	1010b	1~127	500h + Node ID	1402h
T_SDO	1011b	1~127	580h + Node ID	1200h
R_SDO	1100b	1~127	600h + Node ID	1200h
网络管理错误控制	1110b	1~127	700h + Node ID	1016h/1017h

For example:

The COB-ID of slave station No. 3 TPDO1 is $180h + 3 = 183h$

2.2. Communication Settings

In order to enable the NT series CAN bus stepper driver to accurately connect to the CANopen field bus network, it is necessary to set the relevant parameters of the NT series stepper driver.

CAN communication baud rate:

Baud rate	SW6	SW7
125Kbps	ON	ON
250Kbps	OFF	ON
500Kbps	ON	OFF
1Mbps	OFF	OFF

CAN node address:

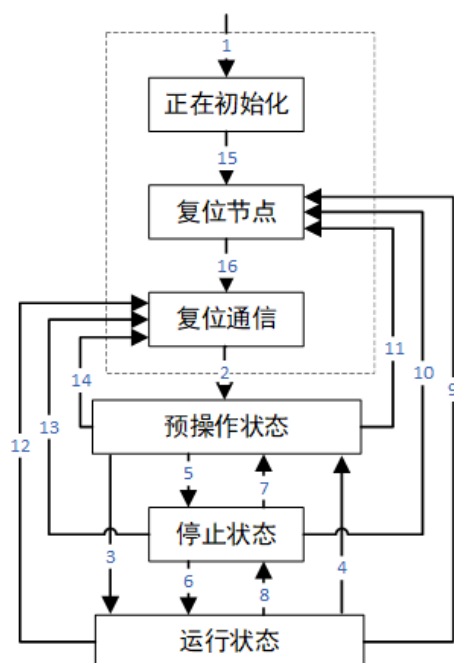
Slave ID	SW1	SW2	SW3	SW4	SW5
0x2150 object settings	ON	ON	ON	ON	ON
1	OFF	ON	ON	ON	ON
2	ON	OFF	ON	ON	ON
3	OFF	OFF	ON	ON	ON
4	ON	ON	OFF	ON	ON
.....					
30	ON	OFF	OFF	OFF	OFF
31	OFF	OFF	OFF	OFF	OFF
0x2150 object settings	ON	ON	ON	ON	ON

2.3. Network Management (NMT)

The Network Management System (NMT) is responsible for initializing, starting and stopping the network and the devices in the network and is a master/slave system. there is one and only one NMT host in the CANopen network that can configure the CANopen network including itself.

2.3.1. NMT Services

CANopen performs the corresponding work according to the state machine specified in the protocol. Part of it is automatically implemented internally by the driver and part of it must be implemented by the NMT host sending NMT messages. The NMT state machine is shown below.



In the above figure, except for the transitions in (1), (15), and (16), all of them are implemented by NMT messages, and only the NMT host can send NMT control messages, and the message data format is shown as follows.

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

The COB-ID of the NMT message is fixed at "0x000"

The data area of an NMT message consists of two bytes.

- ◆ The first byte is the command word, indicating the control role of the communication frame
- ◆ The second byte is the CANopen node address, when it is 0 it is a broadcast message and is valid for all slave devices in the network

The NMT message command is as follows:

Command word	Description
0x01	Start remote node
0x02	Stop remote nodes
0x80	Enter the pre-operation state
0x81	Reset node
0x82	Reset communication

The device automatically enters an initialization state upon power-up, including being initialized, resetting the node, and resetting the communication. Ongoing initialization loads the parameters of each module. And the reset node restores the object dictionary manufacturer definition area and sub-protocol area to the last saved value. Reset communication restores the communication parameters in the object dictionary to the last saved value.

At the end of initialization, the device will send a Boot-up message and automatically enter the pre-operation state, which is the primary configuration node state.

After completing the configuration, the node needs the NMT host to send an NMY message to enter the operational state. The operational state is the state when CANopen is working normally and each module should be working normally.

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

The CANopen services supported in various NMT states are shown below:

Services	Pre-operation	Preoperational	Stop
Process Data Objects (PDO)	NO	YES	NO
Service Data Object (SDO)	YES	YES	NO
Synchronous objects (SYNC)	YES	YES	NO
Emergency Message (EMCY)	YES	YES	NO
Network Management (NMT)	YES	YES	YES
Error control	YES	YES	YES

2.3.2. NMT Error Control

NMT error control is mainly used to detect if a device in the network is online and the state the device is in, including node protection, lifetime protection and heartbeat.

- ◆ Does not allow simultaneous use of node/lifetime protection and heartbeat
- ◆ The node/life protection and heartbeat should not be set too short to avoid increasing the network load

1. Heartbeat

The heartbeat mode uses the "producer-consumer" model, where the CANopen device sends heartbeat messages in ms according to the period set by the producer heartbeat interval object 0x1017. The node with the consumer heartbeat function in the network monitors this producer according to the consumer time set in object 0x1016 and considers the node to be faulty once the producer heartbeat of the corresponding node is not received within the consumer heartbeat time.

After configuring the producer heartbeat interval 0x1017, the node heartbeat function activates and starts generating heartbeat messages. After configuring a valid sub-index of consumer heartbeat 0x1016, monitoring starts upon receiving a frame of heartbeat from the corresponding node.

The host sends a heartbeat telegram at its producer time. the slave monitoring the host does not receive the heartbeat telegram within 0x1016 sub-index time. the host is considered offline. $0x1016 \text{ some sub-index time} \geq \text{producer time} \times 1.8$. otherwise it is easy to falsely report that the host is offline.

The slave sends a heartbeat telegram every 0x1017 time. the host (or other slave) that monitors the slave. if the heartbeat telegram is not received within the consumer time. the slave is considered dropped. $0x1017 \times 1.8 \leq \text{the consumer time of the host (or other slave) that monitors the slave}$. otherwise. it is easy to falsely report that the slave is dropped.

The number of heartbeat messages is shown below, the data segment is one byte long, the highest bit is fixed to "0" and the other bits indicate the status of the CANopen device:

COB-ID	RTR	Data
0x700 + Node ID	0	Bit7: 0 Bit6~Bit0: 4 - Stop state 5 - Operational status 127 - Pre-operation status

2. Node/Life Protection

NT series CAN bus stepper driver does not support node/life protection mode at this time

2.4. Service Data Object (SDO)

The SDO is used to access the object dictionary of a device, the visitor is called a Client and the CANopen device to which the object dictionary is accessed and which provides the requested service is called a Server. A request from a client must have an answer from the server.

SDO transfers are divided into object data transfers of no more than 4 bytes and higher than 4 bytes. The accelerated SDO transfer method is used for not higher than 4 bytes, and the segmented transfer or block transfer method is used for higher than 4 bytes.

The SDO transmission message consists of a COB-ID and a data segment. the COB-IDs of the T_SDO (0x580+Node ID) and R_SDO (0x600+Node ID) messages do not match. The data segments are arranged in small-end mode, i.e., low bits first and high bits second. All SDO message data segments must be 8 bytes (although not all data bytes are necessarily meaningful). the SDO transmission message format is listed in the following table:

COB-ID	Data							
	0	1	2	3	4	5	7	8
0x580+Node ID 0x600+Node ID	Command code	Index		Sub- index	Data			

Among them, the command code indicates the transmission type and transmission data length of the segment of SDO, the index and sub-index are the position of the object in the list, and the data is the value of the object.

2.4.1. SDO Accelerated Write Transmission Messages

For reads and writes of no more than 4 bytes, accelerated SDO transmission is adopted. According to the inconsistency of the reading and writing mode and the length of the content data, the transmission messages are different. The speed up SDO write message is as follows:

Device		COB-ID	Data							
			0	1	2	3	4	5	6	7
Client		0x600 + Node ID	0x23	Index		Sub-index	Data			
			0x27				Data			-
			0x2B				Data		-	
			0x2F				Data	-		
Server	Normal	0x580 + Node	0x60	Index		Sub-index	-			
	Abnormal	ID	0x80				Termination Code			

◆ Note: "-" means data are available but not considered, it is recommended to write 0

For example:

The station number of the slave station is 4, the operating speed value 60FFh-00 in the speed mode is written with SDO, and the written value is 1000, that is, 0x3E8. The message sent by the master station is as follows (all data are in hexadecimal):

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

If the write is normal, the driver will return the following message:

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

If the written data type does not match, the error code 0x06070010 will be returned, and the message will be as follows:

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

2.4.2. SDO Accelerated Read Transmission Messages

When the SDO read operation is not higher than 4 bytes of object data, the acceleration mode is adopted, and the message is as follows:

Device		COB-ID	Data							
			0	1	2	3	4	5	6	7
Client		0x600 + Node ID	0x40	Index		Sub-index	-			
Server	Normal	0x580 + Node ID	0x43	Index		Sub-index	Data			
			0x47				Data			-
			0x4B				-			
			0x4F				-			
	Abnormal	0x80h				Termination code				

For example:

The station number of the slave station is 4, use SDO to read the parameter control mode P01.00, which is the object 0x2003-01, and the master station sends the message as follows (all data are in hexadecimal):

COB-ID	Data							
	0	1	2	3	4	5	6	7
604	40	03	20	01	00	00	00	00

The default control mode is 3, and the return message under normal circumstances is as follows:

COB-ID	Data							
	0	1	2	3	4	5	6	7
584	4B	03	20	01	03	00	00	00

If the written command word does not match, an invalid command word error will be returned, the error code is 0x05040001, and the message is as follows:

COB-ID	Data							
	0	1	2	3	4	5	6	7
584	80	03	20	01	01	00	04	05

2.5. Process Data Objects (PDO)

2.5.1. PDO Transfer Framework

Process Data Object (PDO) is used to transfer data in real time and is the main data transfer method in CANopen. PDO uses the producer/consumer mode, the PDO length can be less than 8 bytes and the transfer speed is fast. PDO data transfer can be tried in a one-to-one or one-to-many way.

2.5.2. PDO Object

According to the difference between receiving and transmitting, PDO can be divided into RPDO and TPDO. PDO is determined by the communication parameters and mapping parameters together to determine the final transmission method and content. NT series stepper drivers use 4 RPDO and 4 TPDO to implement the PDO transmission, the related objects are listed as follows.

Name		COB-ID	Communication object	Mapping object
RPDO	1	0x200 + Node ID	0x1400	0x1600
	2	0x300 + Node ID	0x1401	0x1601
	3	0x400 + Node ID	0x1402	0x1602
	4	0x500 + Node ID	0x1403	0x1603
TPDO	1	0x180 + Node ID	0x1800	0x1A00
	2	0x280 + Node ID	0x1801	0x1A01
	3	0x380 + Node ID	0x1802	0x1A02
	4	0x480 + Node ID	0x1803	0x1A03

2.5.3. PDO Communication Parameters

1. CAN identifier for PDO

The CAN identifier of the PDO is the COB-ID of the PDO, which contains control bits and identification data, and determines the bus priority of the PDO. On the sub-index 0x01 of the COB-ID bit and communication parameters (RPDO: 1400h~1403h, TPDO: 1800h~1803h), the highest bit determines whether the PDO is valid.

For example:

For a node with a station number of 4, when TPDO3 is in an invalid state, its COB-ID should be 0x80000384, and writing 0x384 to the COB-ID indicates that the PDO is activated.

2. Transport type of the PDO

The transmission type bit of the PDO and the sub-index 0x02 of the communication parameters (RPDO: 1400h~1403h, TPDO: 1800h~1803h) determine which transmission mode the PDO follows. The different values of the communication parameters represent different transmission types, and define the method of triggering TPDO transmission or processing the received RPDO. The specific corresponding relationship is as follows:

Communication type value	Synchronize		Synchronous
	Cyclic	Non-cyclic	
0	-	√	-
1~240	√	-	-
241~53	Reserve		
254~255	-	-	√

- ◆ When the transmission type of TPDO is 0, if the mapping data is changed and a synchronization frame is received, the change TPDO is sent.
- ◆ When the transmission type of TPDO is 1~240, the TPDO is sent when the corresponding number of synchronous frames is received.
- ◆ When the transmission type of TPDO is 254 or 255, the TPDO is sent if the mapping data changes or the event counter arrives.
- ◆ When the transmission type of RPDO is 0~240, as long as a synchronization frame is received, the latest data of the RPDO will be updated to the application.
- ◆ When the transmission type of RPDO is 254 or 255, the received data is directly updated to the application.

3. Prohibition time

The prohibition time is set for TPDO, which is stored in the sub-index 0x03 of the communication parameter (0x1800~0x1803) to prevent the PDO with a lower priority in the CAN network from being continuously occupied. The unit of changing the parameter is 100us. After setting the value, the transmission interval of the same TPDO will not be less than the time corresponding to the changed parameter.

For example:

The prohibition time of TPDO2 is 300, then the transmission interval of TPDO will not be less than 30ms.

4. Event timer

For the TPDO of asynchronous transmission (transmission type is 254 or 255), define the event timer, which is located on the sub-index 0x05 of the communication parameter (0x1800~0x1803). The event timer can also be regarded as a trigger event, which will also trigger the corresponding TPDO transmission. If other events such as data changes occur during the timer running period, TPDO will also be triggered, and the event counter will be reset immediately.

2.5.4. PDO Mapping Parameters

The PDO mapping parameter contains a pointer to the process data corresponding to the PDO that the PDO needs to send or receive, including the index, sub-index, and the length of the mapping object. The data length of each PDO can be up to 8 bytes, and one or more objects can be mapped at the same time. Among them, the sub-index 0 records the number of objects specifically mapped by the PDO, and the sub-indexes 1 to 8 are the mapping content. The contents of the mapping parameters are defined as follows.

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

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

Object length	Bit length
08h	8-bit
10h	16-bit
20h	32-bit

For example:

Indicates that the mapping parameter of the 16-bit command word 6040h-00 is 60400010h

2.6. Synchronous objects (SYNC)

A synchronization object (SYNC) is a special mechanism that controls the tuning and synchronization between multiple nodes transmitting and receiving for synchronous transmission of PDO.

2.6.1. Synchronization Generator

NT series stepper drivers are not only synchronous consumers, but also try synchronous producers. The objects that support synchronization are the synchronization object COB-ID (1005h) and the synchronization cycle period (1006h).

The second highest bit of the synchronization object COB-ID determines whether to activate the synchronization generator.

The synchronization cycle period is only for the synchronization generator, and the unit is us, indicating the interval when the node generates the synchronization object.

2.6.2. Synchronous Object Transfer Framework

Similar to the transmission of PDO, the transmission of synchronization objects follows a producer—consumer model, where a synchronization producer sends a synchronization frame and all other nodes in the CAN network can receive this synchronization frame as consumers and without feedback. Only one active synchronization generator is allowed in the same CAN network.

The transmission of synchronous PDO is closely linked to synchronous frames:

- ◆ For synchronous RPDO, as long as the PDO is received, the received PDO will be updated to the application at the next SYNC.
- ◆ For synchronous TPDO, it is divided into synchronous cycle and synchronous acyclic. The synchronization cycle is a PDO with a PDO transmission type of 1 to 240. As long as the SYNC specified by the transmission type is reached, the TPDO needs to be sent regardless of whether the data changes. Synchronous acyclic means that the PDO transmission type is 0, and the content of the PDO mapping object changes, and it is sent at the next SYNC.

For example:

The transmission type of RPDO1 is 0, the transmission type of RPDO2 is 5, the transmission type of TPDO1 is 0, and the transmission type of TPDO2 is 20. Then, as long as RPDO1 and RPDO2 receive PDO, they will update the latest PDO data to the corresponding application at the next SYNC; while the mapping data of TPDO1 only changes, it will send TPDO1 at the next SYNC, and when TPDO2 has experienced 20 SYNC in total. Regardless of whether the data is changed, PDO will be sent.

2.7. Emergency Object Service (EMCY)

When a CANopen node has an error, the node sends an emergency message frame according to the standardized mechanism. The emergency message follows the producer-consumer model, where other nodes in the CAN network can choose to handle the fault after the node fault is sent. The NT series stepper drivers only act as an emergency message producer and does not handle other node emergency messages.

When a node fails, the error register and predefined error field need to be updated regardless of whether the emergency object is activated or not. The contents of the emergency message are as follows:

COB-ID	Data							
	0	1	2	3	4	5	6	7
0x80+Node ID	Error code		Error register	Reserve	Auxiliary byte			

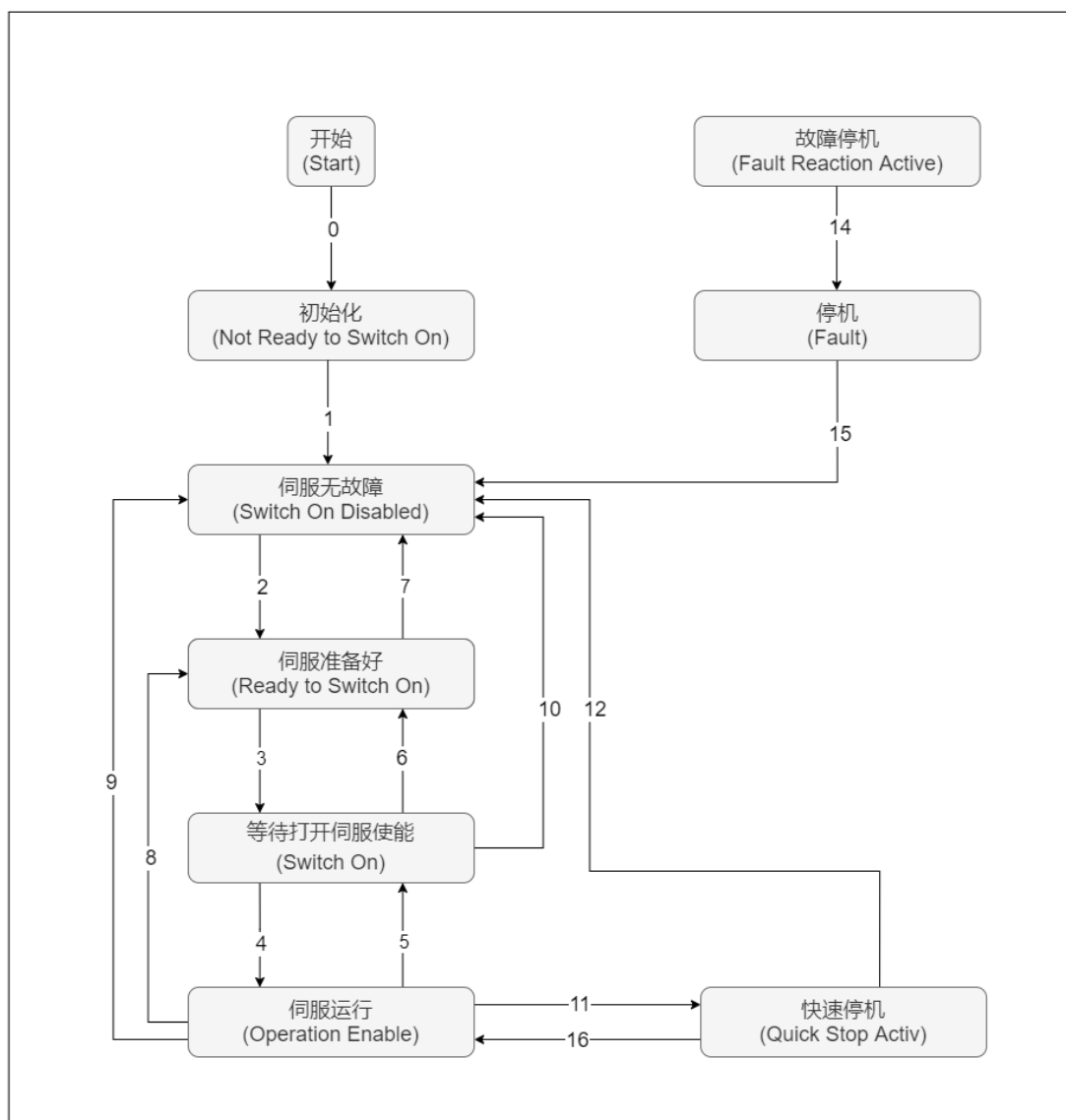
The error register is always consistent with 1001h:

- ◆ When the communication is abnormal, the error code is consistent with the requirements of DS301, and the auxiliary byte is zero when the communication is abnormal.
- ◆ When the driver has an error described in the DSP402 sub-protocol, the error code is consistent with the DS402 requirement and corresponds to the object 603Fh, and the auxiliary byte is an additional description.
- ◆ When a user-specified exception occurs in the driver, the error code is 0xFF00, and the auxiliary byte displays the user-specified error code

3. Operation

3.1. Status Setting

The driver must be guided in accordance with the process specified in the standard CiA402 protocol for the drive to operate in the specified state.



Initialization	Driver initialization and internal self-check have been completed The parameters of the driver cannot be set, nor can the driver function be executed.
Servo without failure	The servo driver has no fault or the error has been eliminated Driver parameters can be set.
Servo ready	Servo driver is ready Driver parameters can be set.
Waiting to turn on servo enable	The servo driver is waiting to turn on the servo enable Driver parameters can be set.
Servo operation	The driver is operating normally, a certain servo operation mode has been enabled, the motor has been energized, and when the command is not 0, the motor will rotate The drive parameter attribute can be set as "operational change", otherwise it cannot be set.

Quick stop	The quick stop function is activated, and the driver is executing the quick stop function The driver parameter attribute can be set as "operational change", otherwise it cannot be set
Fault shutdown	The driver has failed and is in the process of shutdown. The driver parameter attribute can be set as "operational change", otherwise it cannot be set
Faults	When the fault stop is completed, all driver functions are prohibited, and the driver parameters are allowed to be changed in order to eliminate the fault.

Control commands and status switching:

CiA402 state switching		Control word 6040h	Bit0 to Bit9 of status word 6041h
0	Power on → Initialization	Natural transition, no control commands required	0x0000
1	Initialization → No servo failure	Natural transition, no control commands required	0x0250
2	Servo is fault-free → Servo is ready	0x0006	0x0231
3	Servo ready → Wait to turn on servo enable	0x0007	0x0233
4	Wait to turn on servo enable → Servo operation	0x000F	0x0237
5	Servo operation → Wait to turn on servo enable	0x007	0x0233
6	Wait to turn on servo enable → Servo ready	0x006	0x0231
7	Servo ready → No servo failure	0x0000	0x0250
8	Servo running → Servo ready	0x0006	0x0231
9	Servo operation → No servo failure	0x0000	0x0250
10	Wait to turn on servo enable → No servo failure	0x0000	0x0250
11	Servo operation → Quick stop	0x0002	0x217
12	Quick stop → servo-free	Natural transition after shutdown is complete, no control commands required	0x0250
13	→ Failure to stop	If the servo driver fails in any state other than "fault", it will automatically switch to the fault stop state without command control.	0x021F
14	Failure to stop → Failure	Natural transition after the fault stop is done, no control commands required	0x0218
15	Fault → Servo without fault	0x80	0x0250
16	Quick stop → servo operation	When the shutdown is complete, send 0x0F	0x0237

3.1.1. Control Word 6040h

Index	Name	Control word					Data structure	VAR	Data type	Uint16
6040h	Range	0~65535	Factory setting	0	Accessibility	RW	Related models	ALL	Map	RPDO

Set control commands:

Bit	Name	Description
0	Switch on	0: invalid, 1: valid
1	Enable voltage	0: invalid, 1: valid
2	Quick stop	0: invalid, 1: valid
3	Enable operation	0: invalid, 1: valid
4~6	Operation mode specific	Related to servo operation mode
7	Fault reset	For resettable faults and warnings, perform fault reset function The rising edge of Bit7 is valid; Bit7 remains at 1, other control commands are invalid
8	Halt	Please consult the object dictionary 605Dh for the pause method in each mode
9	Operation mode specific	Related to each servo operation mode
10	Reverse	Undefined
11~15	Manufacturer-specific	Manufacturer customization

◆ Note:

- (1) Each Bit of the control word is meaningless when assigned individually, and must be used with other bits that do not form part of a control command.
- (2) Bit0 to Bit3 and Bit7 have the same meaning in each servo mode, and commands must be sent in sequence to direct the servo driver into the expected state according to the CiA402 state machine switching process, with each command corresponding to a defined state.
- (3) Bit4 to Bit6 are related to each servo mode, please check the control commands in different modes
- (4) Bit9 undefined function.

3.1.2. Status Word 6041h

Index	Name	Status word					Data structure	VAR	Data type	Uint16
6041h	Range	0~65535	Factory setting	0	Accessibility	RO	Related models	ALL	Map	TPDO

Reflect the current running status of the servo driver:

Bit	Name	Description
0	Ready to switch on	0: invalid, 1: valid
1	Switch on	0: invalid, 1: valid
2	Operation enabled	0: invalid, 1: valid
3	Fault	0: invalid, 1: valid
4	Voltage enabled	0: invalid, 1: valid
5	Quick stop	0: invalid, 1: valid
6	Switch on disabled	0: invalid, 1: valid
7	Warning	0: invalid, 1: valid
8	Manufacturer specific	undefined
9	Remote	0: invalid, 1: valid (control word in effect)
10	Target reach	0: invalid, 1: valid
11	Internal limit active	0: invalid, 1: valid
12~13	Operation limit active	Related to each servo operation mode
14	Manufacturer specific	undefined
15	Home find	0: invalid, 1: valid

Displayed value (Binary value)	Description
xxxx xxxx x0xx 0000	未准备好(Not ready to switch on)
xxxx xxxx x1xx 0000	启动失效(Switch on disabled)
xxxx xxxx x01x 0001	准备好(Ready to switch on)
xxxx xxxx x01x 0011	启动(Switch on)
xxxx xxxx x01x 0111	操作使能(Operation enabled)
xxxx xxxx x00x 0111	快速停机有效(Quick stop active)
xxxx xxxx x0xx 1111	故障反应有效(Fault reaction active)
xxxx xxxx x0xx 1000	故障(Fault)

◆ Note:

- (1) Bit0 to Bit9 have the same meaning in each servo mode. After the control word 6040h sends the command in sequence, the servo feeds back a determined state.
- (2) Bit12 to Bit13 are related to each servo mode (please check the control commands in different modes).
- (3) Bit10, Bit11, Bit15 have the same meaning in each servo mode, feedback the state of the servo after executing a certain servo mode.

3.2. Servo Mode Setting

3.2.1. Introduction to Servo Mode

Index	Name	Supports servo operation mode					Data structure	VAR	Data type	Uint32
6052h	Range	-	Factory setting	0x29	Accessibility	RO	Related models	-	Map	NO

Reflect the servo operation mode supported by the driver:

Bit	Description	Support or not (0: not support, 1: support)
0	Profile Position Mode (PP)	1
1	Variable frequency speed control mode (VL)	0
2	Profile Velocity Mode (PV)	1
3	Profile Torque Mode (PT)	0
4	NA	0
5	Homing mode (HM)	1
6	Interpolated Position (IP)	0
7	Cyclic Synchronous Position Mode (CSP)	0
8	Cyclic Synchronous Velocity mode (CSV)	0
9	Cyclic Synchronous Torque Mode (CST)	0
10~31	NA	0

Index	Name	Mode selection					Data structure	VAR	Data type	Int16
6060h	Range	0~10	Factory setting	8	Accessibility	RW	Related models	ALL	Map	RPDO

Select the servo operation mode:

Set value	Servo mode
0/2/5	NA
1	Profile Position Mode (PP)
3	Profile Velocity Mode (PV)
4	Profile Torque Mode (PT)
6	Homing mode (HM)
7	Interpolated Position (IP)
8	Cyclic Synchronous Position Mode (CSP)
9	Cyclic Synchronous Velocity mode (CSV)
10	Cyclic Synchronous Torque Mode (CST)

Index	Name	Operation mode display					Data structure	VAR	Data type	Int16
6061h	Range	0~10	Factory setting	0	Accessibility	RO	Related models	ALL	Map	TPDO

Select the current operating mode of the servo driver:

Set value	Servo mode
0/2/5	NA
1	Profile Position Mode (PP)
3	Profile Velocity Mode (PV)
4	Profile Torque Mode (PT)
6	Homing mode (HM)
7	Interpolated Position (IP)
8	Cyclic Synchronous Position Mode (CSP)
9	Cyclic Synchronous Velocity mode (CSV)
10	Cyclic Synchronous Torque Mode (CST)

3.3. Profile Position Mode (PP)

Profile position mode is mainly used for point-to-point positioning applications. In this mode, the upper computer gives the target position (absolute or relative), velocity, acceleration and deceleration of the position curve, and the trajectory generator inside the servo will generate the target position curve command according to the settings, and the drive completes the position, velocity and torque control internally.

3.3.1. Related Objects

Control word 6040h		
Bit	Name	Description
0	伺服准备好(Switch On)	When the value of Bit0 to Bit3 is 1, the motor is enabled
1	接通主回路电(Enable Voltage)	
2	快速停机(Quick Stop)	
3	伺服运行(Enable Operation)	
4	新目标位置(New Set-Point)	This is the rising edge from 0 to 1 indicates the pre-triggered new target position 607Ah, profile velocity 6081h, acceleration 6083h, deceleration 6084h given
5	立即更新(Change Set Immediately)	0: Not immediately updated 1: Update immediately
6	绝对位置/相 BitBit 置(Absolute/Relative)	0: The target position is an absolute position command 1: The target position is a relative position command
7	复位故障(Reset Fault)	0: no effect 1: Reset driver failure
8	暂停 Halt	0: Servo is set according to Bit0~Bit3 1: Servo pauses according to 605Dh setting

Status word 6041h		
Bit	Name	Description
10	目标到达 Target Reached	0: target position not reached 1: target position reached
11	软件内部位置超限 Internal Limit Active	0: Neither the position command nor the position feedback exceeds the limit 1: Position command or position feedback overrun
12	目标位置更新 Set-Point Acknowledge	0: Slave not following command 1: Slave follow command
13	跟随错误 Following Error	0: No excessive position deviation fault 1: Excessive position deviation fault occurs
15	原点回零完成 Home Find	0: Home not completed 1: Home completed

Index (Hex)	Sub-index (Hex)	Name	Data type	Setting range	Accessibility	Default value
6040	00	Control word	Uint16	0~65535	RW	0
6041	00	Status word	Uint16	-	RO	0
6060	00	Operating mode	Int16	0~10	RW	1
6061	00	Mode display	Int16	-	RO	-
6062	00	Position command (Unit: command unit)	Int32	-	RO	-
6064	00	Position feedback (Unit: encoder unit)	Int32	-	RO	-
607A	00	Target position (Unit: command unit)	Int32	$-2^{31} \sim 2^{31}-1$	RW	0
606C	00	Actual speed (Unit: command unit/s)	Int32	-	RO	-
6081	00	Profile speed (Unit: command pulse/s)	Uint32	$0 \sim 2^{32}-1$		1000
6083	00	Profile acceleration (Unit: command pulse/s ²)	Uint32	$0 \sim 2^{32}-1$		1000
6084	00	Profile deceleration (Unit: command pulse/s ²)	Uint32	$0 \sim 2^{32}-1$		1000

3.3.2. Position Curve Generator

1. Update immediately

- (1) The upper computer first updates and modifies other attributes of the displacement command as needed (acceleration time 6083h, deceleration time 6084h, contour speed 6081h, target displacement 607Ah).
- (2) The host computer sets the bit4 of 6040h from 0 to 1, indicating that the slave station has a new displacement command that needs to be enabled
- (3) After receiving the rising edge of bit 4 of 6040h, the slave station judges whether the new displacement command can be received: If the initial state of bit 5 of 6040 is 0, and bit 12 of 6041h is 0 at this time, it indicates that the slave station can receive a new displacement command①; after receiving the new displacement command, the slave station sets bit 12 of 6041 from 0 to 1, indicating the new displacement command ① has been received, and the current slave station is in a state where it cannot continue to receive new displacement commands. In immediate update mode, once a new displacement command is received (bit12 of 6041h changes from 0 to 1), the servo will immediately execute the displacement command.
- (4) After the upper computer receives the bit12 of the status word 6041h of the slave station becomes 1, it can release the displacement command data, and set the bit4 of the control word 6040h from 1 to 0, indicating that there is no new position command at present. Because bit4 of 6040h is valid for edge change, this operation will not interrupt the displacement command being executed.
- (5) When the slave station detects that the bit4 of the control word 6040h changes from 1 to 0, it can set the bit12 of the status word 6041h from 1 to 0, indicating that the slave station is ready to receive a new displacement command. In immediate update mode, when the slave detects that bit 4 of the control word 6040h changes from 1 to 0, it will always clear bit 12 of 6041h. In immediate update mode, during the execution of the current displacement command ①, a new displacement command ② is received, and the unexecuted displacement command in ① is not discarded. For the relative position command, after the positioning of the second displacement command is completed, the total displacement increment = target position increment 607Ah of ①+ target position increment 607Ah of ②, for the absolute position command, after the second stage of displacement command positioning is completed, the user's absolute position = target position 607Ah of ②.

2. Non-immediate update

- (1) The upper computer first updates and modifies other attributes of the displacement command as needed (acceleration time 6083h, deceleration time 6084h, contour speed 6081h, target displacement 607Ah)
- (2) The host computer sets the bit4 of 6040h from 0 to 1, indicating that the slave station has a new displacement instruction that needs to be enabled
- (3) After receiving the rising edge of bit 4 of 6040h, the slave station judges whether the new displacement command can be received: If the initial state of bit 5 of 6040 is 0, and bit 12 of 6041h is 0 at this time, it indicates that the slave station can receive a new displacement command①; after receiving the new displacement command, the slave station sets bit 12 of 6041 from 0 to 1, indicating the new displacement command ① has been received, and the current slave station is in a state where it cannot continue to receive new displacement commands.
- (4) After the upper computer receives the bit12 of the status word 6041h of the slave station becomes 1, it can release the displacement command data, and set the bit4 of the control word 6040h from 1 to 0, indicating that there is no new position command at present. Because bit4 of 6040h is valid for edge change, this operation will not interrupt the displacement command being executed.
- (5) The slave station detects that the bit 4 of the control word 6040 changes from 1 to 0, and releases the bit 12 of 6041 after the current segment positioning is completed, indicating that the slave station is ready to receive a new displacement command. In non-immediate update mode, while the current segment is running, the servo cannot receive a new displacement command. After the current segment positioning is completed, the servo can receive a new displacement command. Once received (bit12 of 6041 changes from 0 to 1), the servo executes the displacement command immediately.

3.3.3. Recommended configuration

RPDO	TPDO	Description
6040h: Control word(Control Word)	6041h: 状态字(Status Word)	Necessary
607Ah: 目标位置(Target Position)	6064h: 位置反馈(Position Actual Value)	Necessary
6060h: 模式选择(Modes of Operation)	6061h: 运行模式(Modes of Operation Display)	Optional
6081h: 轮廓速度(Profile Velocity)		Necessary
6083h: 轮廓加速度(Profile Acceleration)		Optional
6084h: 轮廓减速度(Profile Deceleration)		Optional

3.4. Profile Velocity Mode (PV)

In this mode, the host computer sends the target speed, acceleration, and deceleration to the servo driver, and the speed and torque adjustment is performed internally by the servo.

3.4.1. Related Objects

Control word 6040h		
Bit	Name	Description
0	伺服准备好(Switch On)	When the value of Bit0 to Bit3 is 1, the motor is enabled
1	接通主回路电(Enable Voltage)	
2	快速停机(Quick Stop)	
3	伺服运行(Enable Operation)	
4	新目标 Bit 置(New Set-Point)	This is the rising edge from 0 to 1 indicates the pre-triggered new target position 607Ah, profile velocity 6081h, acceleration 6083h, deceleration 6084h given
5	立即更新(Change Set Immediately)	0: Not immediately updated 1: Update immediately
6	绝对位置/相 BitBit 置(Absolute/Relative)	0: The target position is an absolute position command 1: The target position is a relative position command
7	复位故障(Reset Fault)	0: no effect 1: Reset driver failure
8	暂停 Halt	0: Servo is set according to Bit0~Bit3 1: Servo pauses according to 605Dh setting

Status word 6041h		
Bit	Name	Description
10	目标到达 Target Reached	0: Target speed not reached 1: Target speed reached
11	软件内部位置超限 Internal Limit Active	0: Neither the position command nor the position feedback exceeds the limit 1: Position command or position feedback overrun
15	原点回零完成 Home Find	0: Home not completed 1: Home completed

Index (Hex)	Sub-index (Hex)	Name	Data type	Setting range	Accessibility	Default
6040	00	Control word	Uint16	0~65535	RW	0
6041	00	Status word	Uint16	-	RO	0
6060	00	Operating mode	Int16	0~10	RW	1
6061	00	Mode display	Int16	-	RO	-
606C	00	Actual speed (Unit: command unit/s)	Int32	-	RO	-
6083	00	Profile acceleration (Unit: command pulse/s ²)	Uint32	0~2 ³² -1		1000
6084	00	Profile deceleration (Unit: command pulse/s ²)	Uint32	0~2 ³² -1		1000
60FF	00	Profile speed (Unit: command pulse/s)	Uint32	0~2 ³² -1	RW	0

3.4.2. Recommended Configuration

RPDO	TPDO	Description
6040h: 控制字(Control Word)	6041h: 状态字(Status Word)	Necessary
6060h: 模式选择(Modes of Operation)	6061h: 运行模式(Modes of Operation Display)	Optional
60FFh: 目标速度(Target Velocity)		Necessary
6083h: 轮廓加速度(Profile Acceleration)	606Ch: 速度反馈(Velocity Actual Value)	Optional
6084h: 轮廓减速度(Profile Deceleration)	603Fh: 错误代码(Error Code)	Optional
	60FDh: 数字输入(Digital Inputs)	Optional

3.5. Homing Mode (HM)

The homing mode is used to find the mechanical origin and locate the positional relationship between the mechanical origin and the mechanical zero point.

Mechanical origin: A fixed position on the machine can correspond to a certain origin switch or motor Z-phase signal.

Mechanical zero point: absolute zero position on the machine.

After the homing is completed, the stop position of the motor is the mechanical origin. By setting 607Ch, the relationship between the mechanical origin and the mechanical zero can be set:

Mechanical origin = mechanical zero + 607Ch (origin offset)

When 607Ch = 0, the mechanical origin coincides with the mechanical zero point.

3.5.1. Related Objects

Control word 6040h		
Bit	Name	Description
0	伺服准备好(Switch On)	When the value of Bit0 to Bit3 is 1, the motor is enabled
1	接通主回路电(Enable Voltage)	
2	快速停机(Quick Stop)	
3	伺服运行(Enable Operation)	
4	启动回零(Home Start)	0 -> 1: Start home 1: Homing 1 -> 0: End home
7	复位故障(Reset Fault)	0: no effect 1: Reset drive failure
8	暂停 Halt	0: The servo determines whether to start home according to Bit4 setting 1: Servo pauses according to 605Dh setting

Status word 6041h		
Bit	Name	Description
10	目标到达 Target Reached	0: target position not reached 1: target position reached
11	软件内部位置超限 Internal Limit Active	0: Neither the position command nor the position feedback exceeds the limit 1: Position command or position feedback overrun
12	回零 Homing Attained	0: Home unsuccessful 1: Home success, this flag is valid after the servo in homing mode operation (target reach signal) is set
13	回零错误 Homing Error	0: Home no error occurred 1: Home error occurred
15	原点回零完成 Home Find	0: Home not completed 1: Home completed

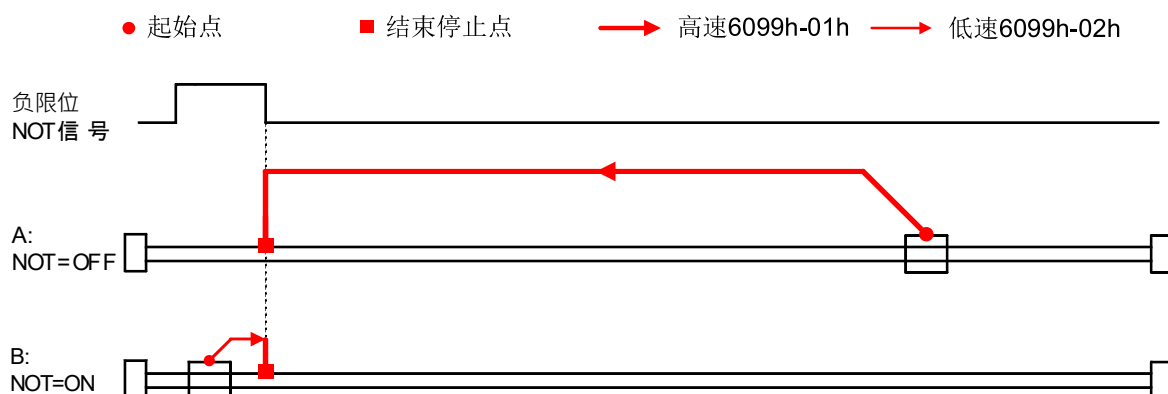
Index(Hex)	Sub-index (Hex)	Name	Data type	Setting range	Accessibility	Default
6040	00	Control word	Uint16	0~65535	RW	0
6041	00	Status word	Uint16	-	RO	0
6060	00	Operating mode	Int16	0~10	RW	1
6061	00	Mode display	Int16	-	RO	-
6062	00	Position command (Unit: command unit)	Int32	-	RO	-
6064	00	Position feedback (Unit: encoder unit)	Int32	-	RO	-
6098	00	Homing mode	Int8	0~127	RW	0
6099	01	Search deceleration point signal speed (Unit: command unit/s)	Uint32	$1 \sim 2^{31}-1$	RW	100
	02	Search origin signal speed (Unit: command unit/s)	Uint32	$1 \sim 2^{31}-1$	RW	10
609A	00	Acceleration (Unit: command unit/s ²)	Uint32	$0 \sim 2^{32}-1$	RW	100

3.5.2. Introduction to Homing Operation

1. Object 6098h = 17

Mechanical origin: Negative limit switch

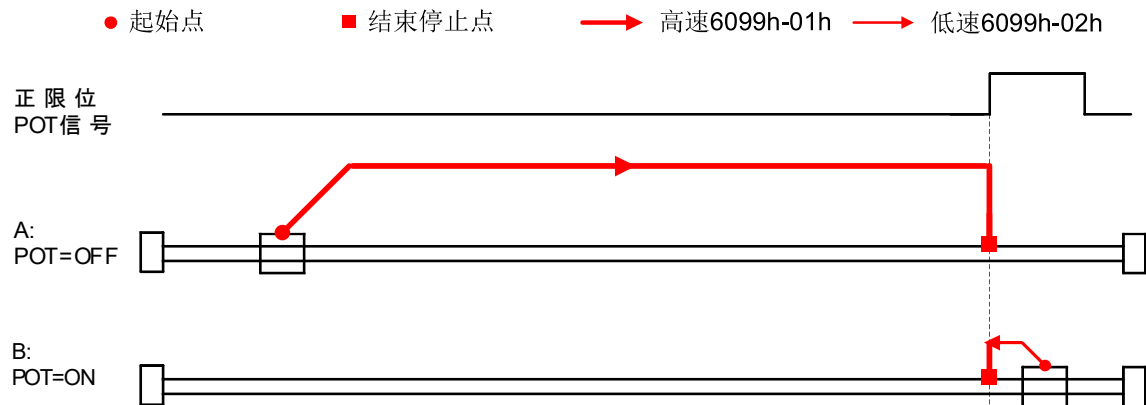
Deceleration point: Negative limit switch



2. Object 6098h = 18

Origin: Positive limit switch

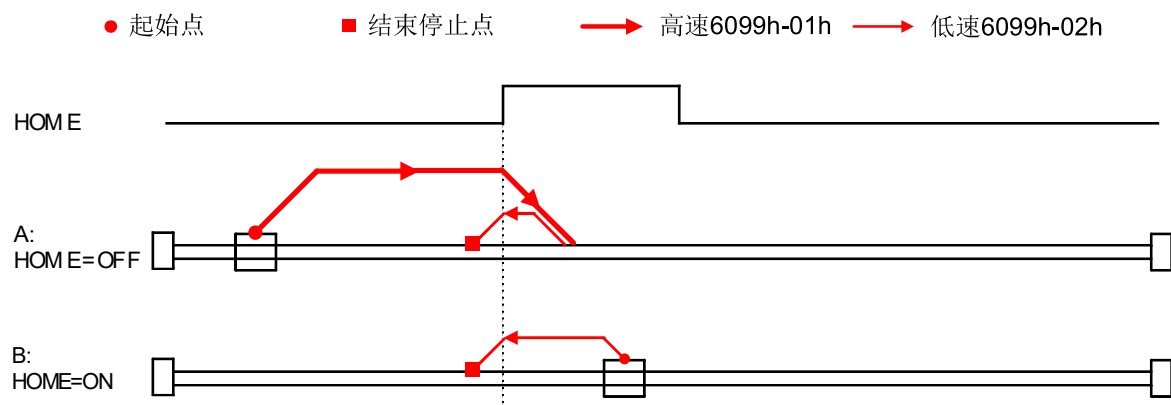
Deceleration point: Positive limit switch



3. Object 6098h = 19

Origin: Origin switch

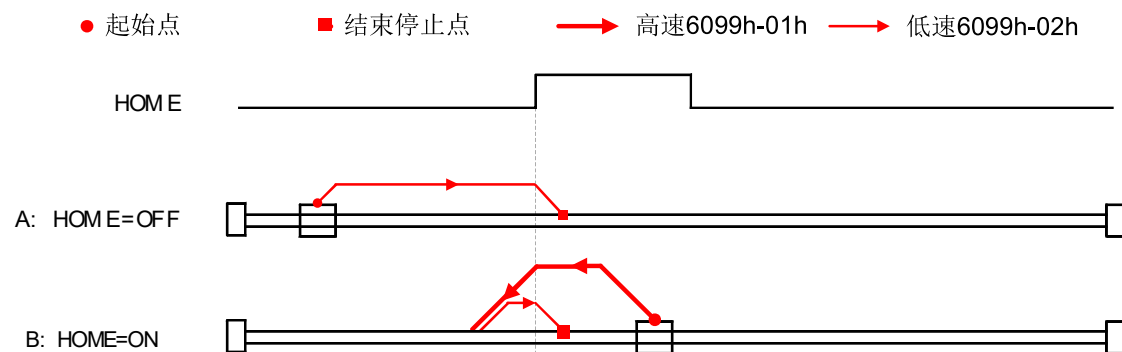
Deceleration point: Origin switch



4. Object 6098h = 20

Origin: Origin switch

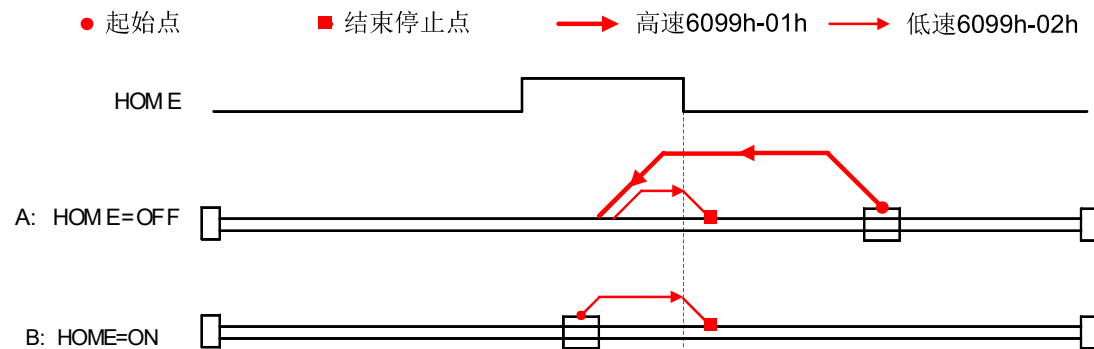
Deceleration point: Origin switch



5. Object 6098h = 21

Origin: Origin switch

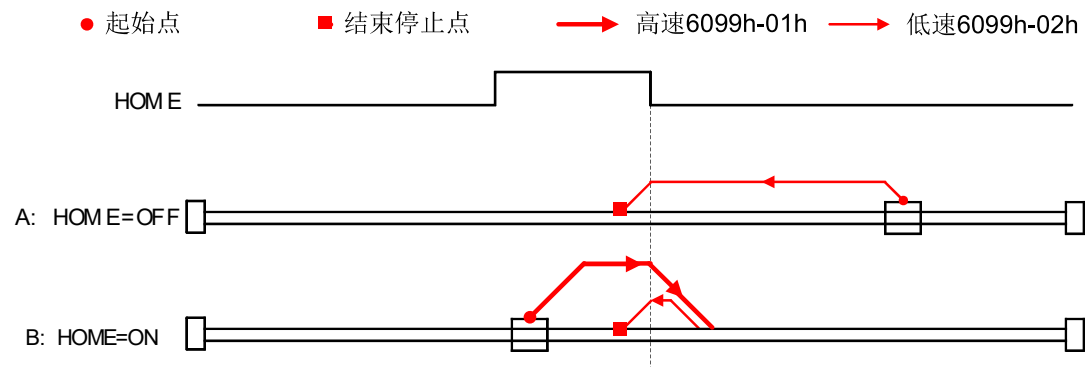
Deceleration point: Origin switch



6. Object 6098h = 22

Origin: Origin switch

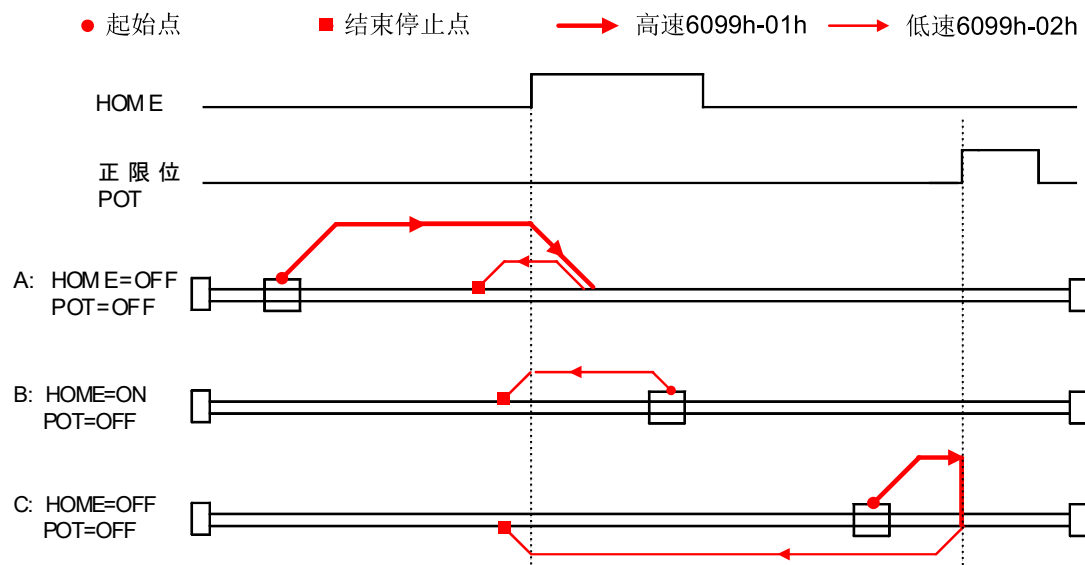
Deceleration point: Origin switch



7. Object 6098h = 23

Origin: Origin switch

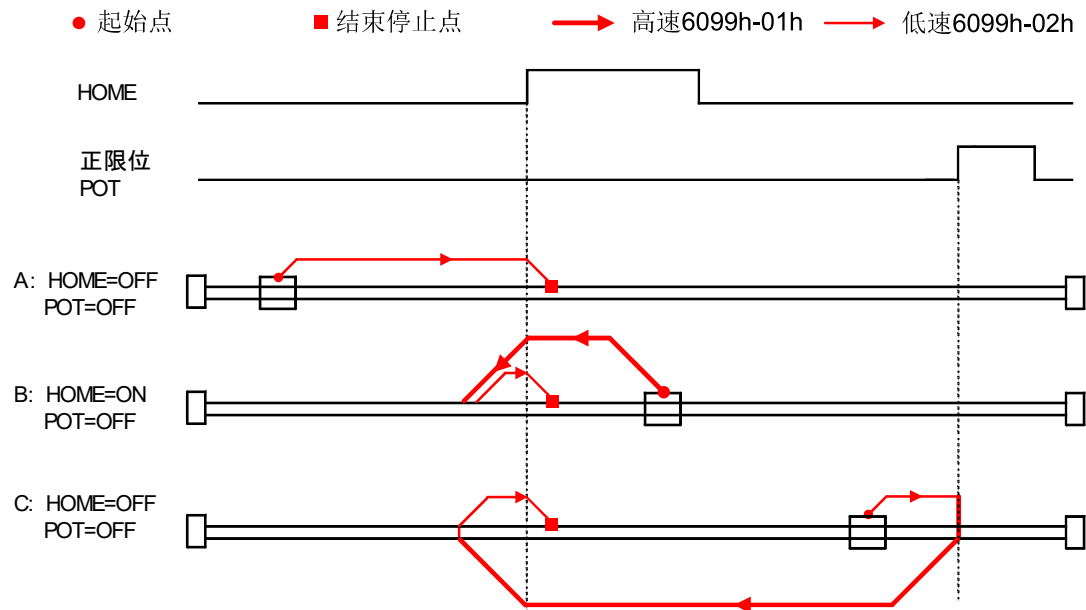
Deceleration point: Origin switch



8. Object 6098h = 24

Origin: Origin switch

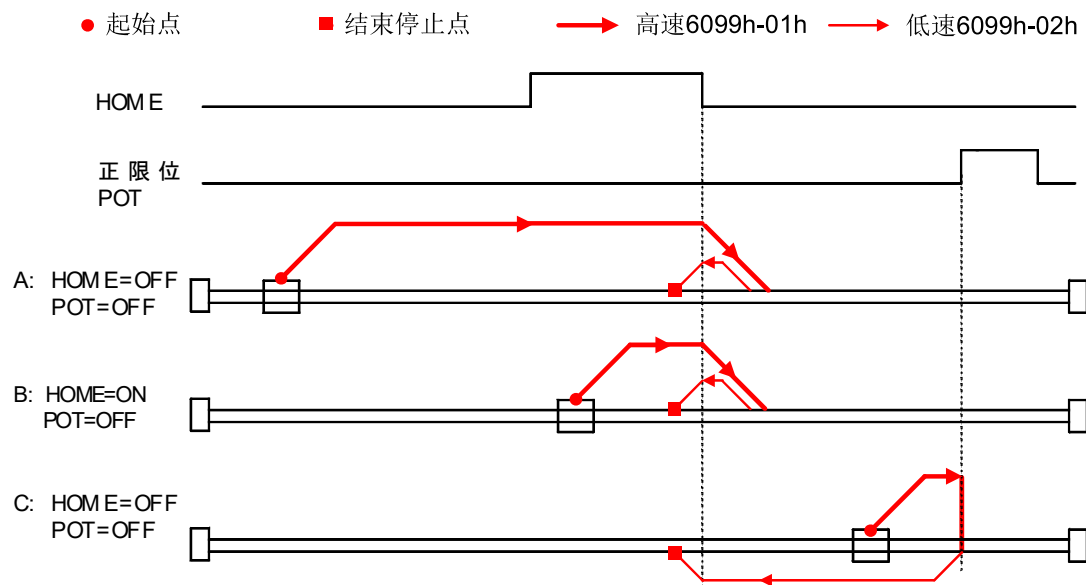
Deceleration point: Origin switch



9. Object 6098h = 25

Origin: Origin switch

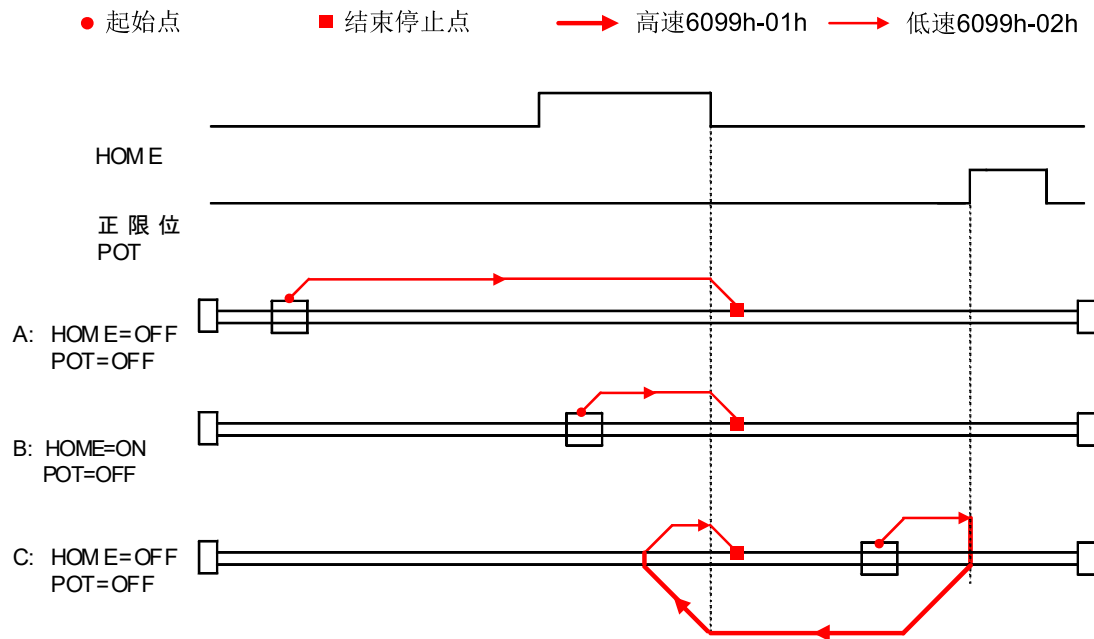
Deceleration point: Origin switch



10. Object 6098h = 26

Origin: Origin switch

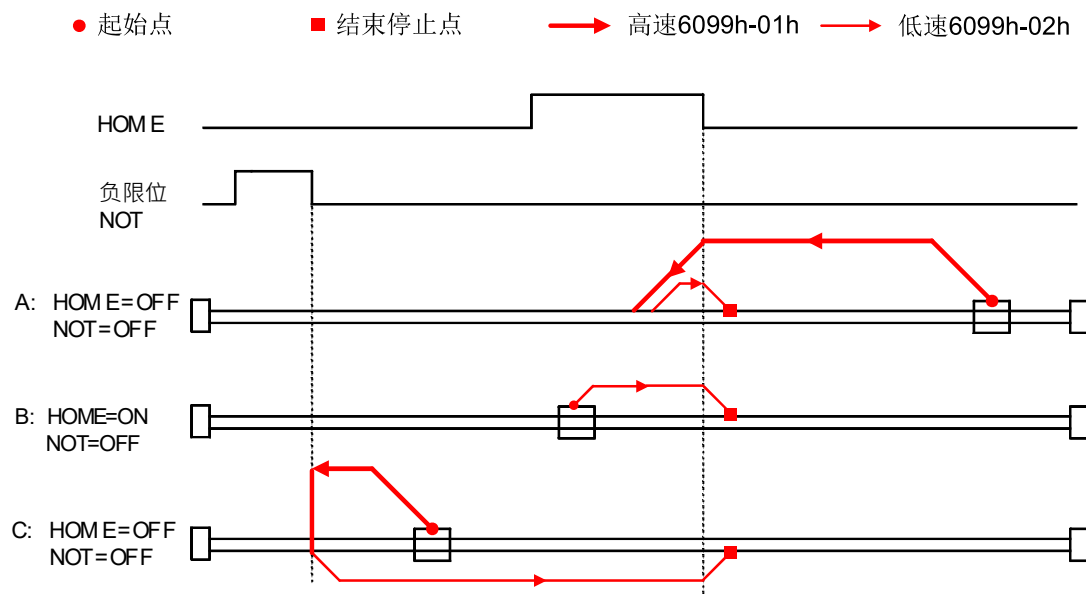
Deceleration point: Origin switch



11. Object 6098h = 27

Origin: Origin switch

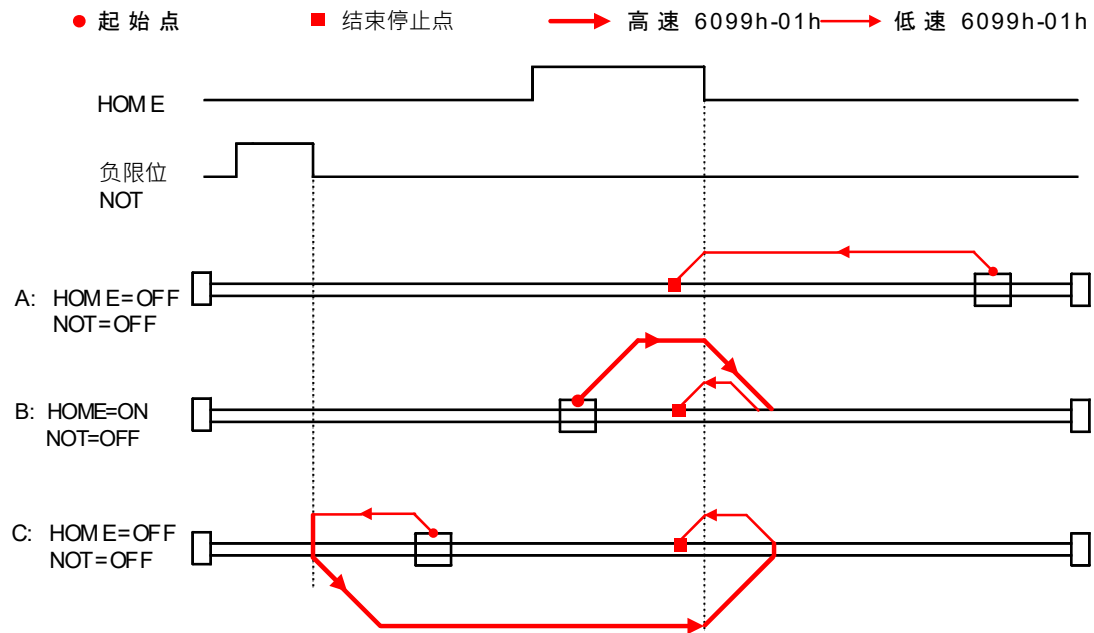
Deceleration point: Origin switch



12. Object 6098h = 28

Origin: Origin switch

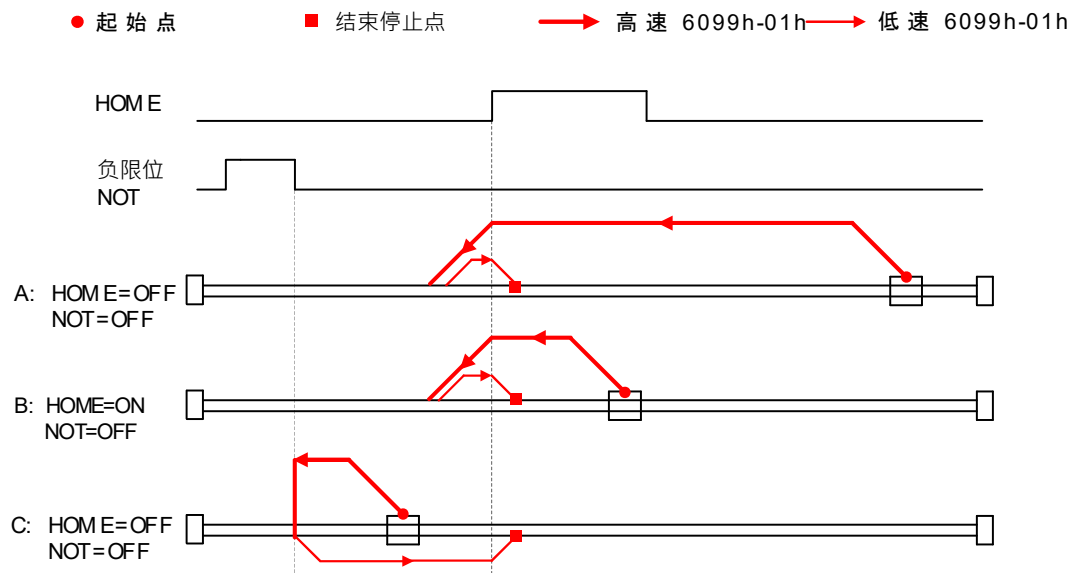
Deceleration point: Origin switch



13. Object 6098h = 29

Origin: Origin switch

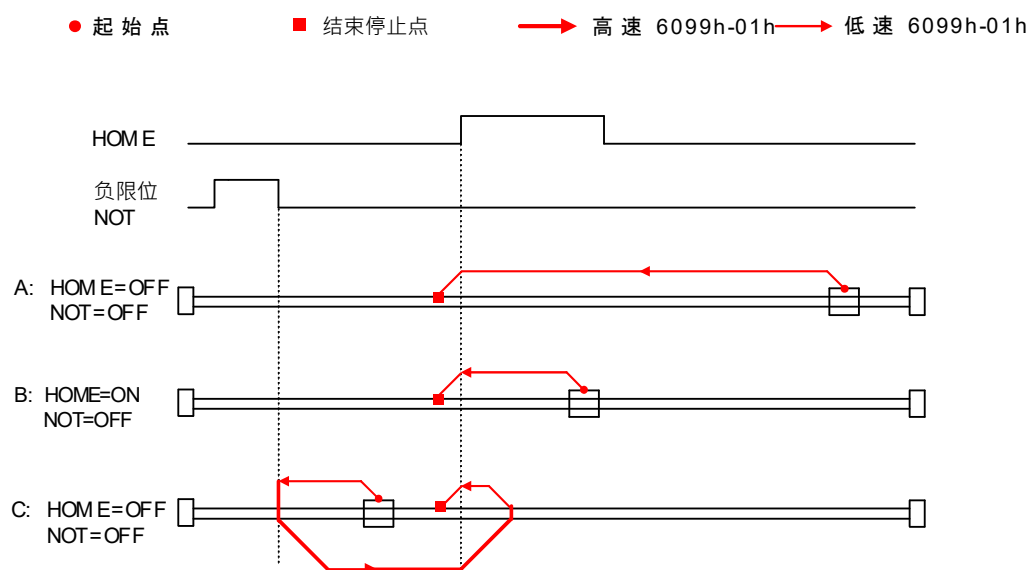
Deceleration point: Origin switch



14. Object 6098h = 30

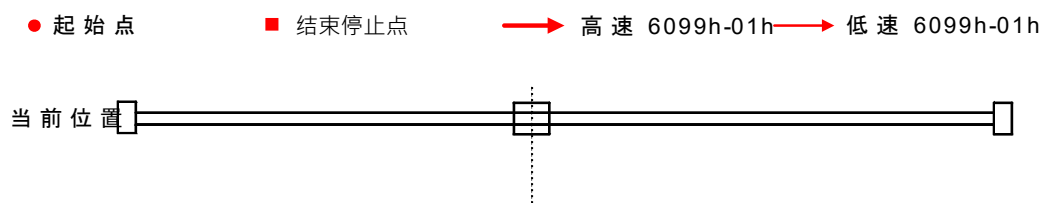
Origin: Origin switch

Deceleration point: Origin switch



15. Object 6098h = 35

Taking the current position as the mechanical origin, after triggering the origin return to zero (6040h control word: 0x0F → 0x1F), the position feedback 6064h is set to the origin offset 607Ch.



3.5.3. Recommended Configuration

RPDO	TPDO	Description
6040h: Control word(Control Word)	6041h: 状态字(Status Word)	Necessary
6060h: 模式选择(Modes of Operation)	6061h: 运行模式(Modes of Operation Display)	Optional
6098h: 回零方式(Homing Method)		Optional
6099-01h: 搜索减速点信号速度(Speed during search for switch)		Optional
6099-02h: 搜索原点信号速度(Speed during search for zero)	603Fh: 错误代码(Error Code)	Optional
609Ah: 回零加速度(Homing acceleration)	60FDh: 数字输入(Digital Inputs)	Optional

4. Object Dictionary

4.1. Object Dictionary Classification Description

The object dictionary is the most important part of the device specification. It is an ordered collection of a set of parameters and variables, including all parameters of device description and device network status, and a set of objects that can be accessed through the network in an orderly and predefined manner.

The CANopen protocol uses an object dictionary with a 16-bit index and an 8-bit sub-index. The structure of the object dictionary is shown in the following table.

Index	Object
0000h	Unused
0001h~001Fh	Static data types (standard data types, such as Boolean, Integer16)
0020h~003Fh	Complex data types (predefined structures composed of simple types such as PDOCommPar, SDOParmeter)
0040h~005Fh	Complex data types specified by the manufacturer
0060h~007Fh	Static data type specified by the device sub-protocol
0080h~009Fh	Complex data types specified in the device sub-protocol
00A0h~0FFFh	Reserve
1000h~1FFFh	Communication sub-protocol area (e.g. device type, error register, number of supported PDOs)
2000h~5FFFh	Manufacturer-specific sub-protocol area (e.g. function code mapping)
6000h~9FFFh	Standard equipment sub-protocol area (e.g. DSP-402 protocol)
A000h~FFFFh	Reserve

The NT series drive object contains the following attributes:

- ◆ Index
- ◆ Sub-index
- ◆ Data structure
- ◆ Data type
- ◆ Accessibility
- ◆ Map
- ◆ Setting effective
- ◆ Data range
- ◆ Factory setting

■ Noun Interpretation:

The position of the object dictionary in the parameter table is specified by "index" and "sub-index".

“Index”: Specify the position of the same type of object in the object dictionary, expressed in hexadecimal.

“Sub-index”: Under the same index, it contains multiple objects, and the offset of each object under this category.

The descriptions of each object in the object dictionary are categorized by description. For example, the object 6099h with homing speed set in the object dictionary describes the speed of searching for deceleration points and the speed of searching for the origin, respectively. The object definitions are as follows:

Index	Sub-index	Name	Meaning
6099h	00h	Number of elements	The number of object data, not including itself
6099h	01h	Index 1	Search for deceleration point speed
6099h	02h	Index 2	Search for origin speed

“Data structure”:

Data type	Value range	Data length	DS301 value
Int8	-128～+127	1 byte	0002
Int16	-32768～+32767	2 bytes	0003
Int32	-2147483648～+2147483647	4 bytes	0004
UInt8	0～255	1 byte	0005
UInt16	0～65535	2 bytes	0006
UInt32	0～4294967295	4 bytes	0007
String	ASCII	-	0009

8 “Data type”:

Data type	Value range	Data length	DS301 value
Int8	-128～+127	1 byte	0002
Int16	-32768～+32767	2 bytes	0003
Int32	-2147483648～+2147483647	4 bytes	0004
UInt8	0～255	1 byte	0005
UInt16	0～65535	2 bytes	0006
UInt32	0～4294967295	4 bytes	0007
String	ASCII	-	0009

“Accessibility”:

Accessibility	Description
RW	Can read and write
WO	Write only
RO	Read only

“Map”:

Map	Description
NO	Can not be mapped in PDO
RPDO	Can be used as RPDO
TPDO	Can be used as TPDO

“Data range”: Data upper and lower limits of parameters with writable attributes

“Factory setting”: Parameter default value

4.2. Object Group: 1000h

Index	Sub-index	Name	Accessibility	Map	Data type	Factory setting
1000h	-	Device type	RO	NO	UINT32	0x40192
1001h	-	Error register	RO	NO	UINT8	0x00
1003h	-	Predefined error register	-	-	-	-
	00h	Maximum sub- index number	RO	NO	UINT32	0x04
	01~04h	Error register	RO	NO	UINT32	0x00
1005h	-	Synchronization message COB-ID	RW	NO	UINT32	0x80
1008h	-	Manufacturer device name	CONST	NO	STRING	NT86-C / NT60-C
1009h	-	Hardware version	CONST	NO	STRING	V2.0
100Ah	-	Software version	CONST	NO	STRING	V2.00
100Ch	-	Node guardian time	RW	NO	UINT16	0
100Dh	-	Life factor	RW	NO	UINT8	0
1010h	-	Save parameters	-	-	-	-
	00h	Maximum sub- index number	RO	NO	UINT8	0x01
	01h	Save driver parameters, write 0x65766173 to the upper computer	RW	NO	UINT32	0x00
1011h	-	Factory reset	-	-	-	-
	00h	Maximum sub- index number	RO	NO	UINT8	0x01
	01h	Factory reset, write 0x64616F6C to the upper computer	RW	NO	UINT32	0x00
1014h	-	Emergency message COB-ID	RW	NO	UINT32	0x80
1016h	-	Consumer heartbeat time	-	-	-	-
	00h	Maximum sub- index number	RO	NO	UINT8	0x05
	01~05h	Consumer heartbeat time	RW	NO	UINT32	0x00
1017h	-	Producer heartbeat time	RW	NO	UINT16	0x00
1018h	-	Device object description	-	-	-	-
	00h	Maximum sub- index number	RO	NO	UINT8	0x03
	01h	Manufacturer ID	RO	NO	UINT32	0xA88
	02h	Device code	RO	NO	UINT32	0x100
	03h	Revision number	RO	NO	UINT32	0x200
1029h	-	Error behavior object	-	-	-	-
	00h	Maximum sub- index number	RO	NO	UINT8	0x01
	01h	Communication error behavior	RW	NO	UINT8	0x00
1200h	-	SDO service parameters	-	-	-	-
	00h	Maximum sub- index number	RO	NO	UINT8	0x02
	01h	Client to server COB-ID	RW	NO	UINT32	0x600 + Node_ID
	02h	Server to client COB-ID	RW	NO	UINT32	0x580 + Node_ID
1400h	-	RPDO1 parameter	RW	NO	PDO	-

					parameter	
	00h	Maximum sub- index number	RO	NO	UINT8	0x02
	01h	RPDO1's COB-ID	RW	NO	UINT32	0x00000200+Node_ID
	02h	RPDO1 transmission type	RW	NO	UINT8	255
1401h	-	RPDO2 parameter	RW	NO	PDO parameter	-
	00h	Maximum sub- index number	RO	NO	UINT8	0x02
	01h	RPDO2's COB-ID	RW	NO	UINT32	0x00000300+Node_ID
	02h	RPDO2 transmission type	RW	NO	UINT8	255
1402h	-	RPDO3 parameter	RW	NO	PDO parameter	-
	00h	Maximum sub- index number	RO	NO	UINT8	0x02
	01h	RPDO3's COB-ID	RW	NO	UINT32	0x00000400+Node_ID
	02h	RPDO3 transmission type	RW	NO	UINT8	255
1403h	-	RPDO4 parameter	RW	NO	PDO parameter	-
	00h	Maximum sub- index number	RO	NO	UINT8	0x02
	01h	RPDO4's COB-ID	RW	NO	UINT32	0x00000500+Node_ID
	02h	RPDO4 transmission type	RW	NO	UINT8	255
1600h	-	RPDO1 mapping parameter	RW	NO	PDO parameter	-
	00h	Maximum sub- index number	RW	NO	UINT8	0x08
	01~08h	RPDO1 mapping object	RW	NO	UINT32	-
1601h	-	RPDO2 mapping parameter	RW	NO	PDO parameter	-
	00h	Maximum sub- index number	RW	NO	UINT8	0x08
	01~08h	RPDO2 mapping object	RW	NO	UINT32	-
1602h	-	RPDO3 mapping parameter	RW	NO	PDO parameter	-
	00h	Maximum sub- index number	RW	NO	UINT8	0x08
	01~08h	RPDO3 mapping object	RW	NO	UINT32	-
1603h	-	RPDO4 mapping parameter	RW	NO	PDO parameter	-
	00h	Maximum sub- index number	RW	NO	UINT8	0x08
	01~08h	RPDO4 mapping object	RW	NO	UINT32	-
1800h	-	TPDO1 parameter	RW	NO	PDO parameter	-
	00h	Maximum sub- index number	RO	NO	UINT8	0x02
	01h	TPDO1's COB-ID	RW	NO	UINT32	0x80000180+Node_ID
	02h	TPDO1 transmission type	RW	NO	UINT8	255
	03h	Prohibited time	RW	NO	UINT16	0
	04h	-	-	-	-	-

	05h	Event timer	RW	NO	UINT16	0
1801h	-	TPDO2 parameter	RW	NO	PDO parameter	-
	00h	Maximum sub- index number	RO	NO	UINT8	0x02
	01h	TPDO2's COB-ID	RW	NO	UINT32	0x80000280+Node_ID
	02h	TPDO2 transmission type	RW	NO	UINT8	255
	03h	Prohibited time	RW	NO	UINT16	0
	04h	-	-	-	-	-
	05h	Event timer	RW	NO	UINT16	0
1802h	-	TPDO3 parameter	RW	NO	PDO parameter	-
	00h	Maximum sub- index number	RO	NO	UINT8	0x02
	01h	TPDO3's COB-ID	RW	NO	UINT32	0x80000380+Node_ID
	02h	TPDO3 transmission type	RW	NO	UINT8	255
	03h	Prohibited time	RW	NO	UINT16	0
	04h	-	-	-	-	-
	05h	Event timer	RW	NO	UINT16	0
1803h	-	TPDO4 parameter	RW	NO	PDO parameter	-
	00h	Maximum sub- index number	RO	NO	UINT8	0x02
	01h	TPDO4's COB-ID	RW	NO	UINT32	0x80000480+Node_ID
	02h	TPDO4 transmission type	RW	NO	UINT8	255
	03h	Prohibited time	RW	NO	UINT16	0
	04h	-	-	-	-	-
	05h	Event timer	RW	NO	UINT16	0
1A00h	-	TPDO1 mapping parameter	RW	NO	PDO parameter	-
	00h	Maximum sub- index number	RW	NO	UINT8	0x08
	01~08h	TPDO1 mapping object	RW	NO	UINT32	-
1A01h	-	TPDO2 mapping parameter	RW	NO	PDO parameter	-
	00h	Maximum sub- index number	RW	NO	UINT8	0x08
	01~08h	TPDO2 mapping object	RW	NO	UINT32	-
1A02h	-	TPDO3 mapping parameter	RW	NO	PDO parameter	-
	00h	Maximum sub- index number	RW	NO	UINT8	0x08
	01~08h	TPDO3 mapping object	RW	NO	UINT32	-
1A03h	-	TPDO4 mapping parameter	RW	NO	PDO parameter	-
	00h	Maximum sub- index number	RW	NO	UINT8	0x08
	01~08h	RPDO4 mapping object	RW	NO	UINT32	-

4.3. Object Group: 2000h

Index	Sub-index	Name	Accessibility	Map	Data type	Factory setting																																			
2000h	-	Running current (mA)	RW	YES	UINT16	3200																																			
2001h	-	Pulse per revolution (pulse/rev)	RW	YES	UINT16	10000																																			
2002h	-	Stand-by time (ms)	RW	YES	UINT16	500																																			
2003h	-	Stand-by current percentage (%)	RW	YES	UINT16	50																																			
2005h	-	Output port parameter	RW	YES	UINT16	-																																			
	00h	Maximum sub- index number	RO	NO	UINT8	0x02																																			
	01h	Output port 1 function	RW	YES	UINT16	0x01																																			
	02h	Output port 2 function	RW	YES	UINT16	0x02																																			
	The output port function settings and corresponding functions are as follows:																																								
		<table><tr><th>Set value</th><th>Function</th><th>Set value</th><th>Function</th></tr><tr><td>0</td><td>Universal output</td><td>8</td><td>Positive limit valid</td></tr><tr><td>1</td><td>Alarm output</td><td>9</td><td>Negative limit valid</td></tr><tr><td>2</td><td>Brake output</td><td>10</td><td>Power supply indicators</td></tr><tr><td>3</td><td>Positioning completed</td><td>11</td><td>Torque arrival</td></tr><tr><td>4</td><td>Speed arrival</td><td>12</td><td>Run to maximum position</td></tr><tr><td>5</td><td>Homing completed</td><td>13</td><td>USER1</td></tr><tr><td>6</td><td>Servo ready</td><td>14</td><td>USER2</td></tr><tr><td>7</td><td>Pulse command stop</td><td>15</td><td>-</td></tr></table>	Set value	Function	Set value	Function	0	Universal output	8	Positive limit valid	1	Alarm output	9	Negative limit valid	2	Brake output	10	Power supply indicators	3	Positioning completed	11	Torque arrival	4	Speed arrival	12	Run to maximum position	5	Homing completed	13	USER1	6	Servo ready	14	USER2	7	Pulse command stop	15	-			
	Set value	Function	Set value	Function																																					
	0	Universal output	8	Positive limit valid																																					
	1	Alarm output	9	Negative limit valid																																					
	2	Brake output	10	Power supply indicators																																					
3	Positioning completed	11	Torque arrival																																						
4	Speed arrival	12	Run to maximum position																																						
5	Homing completed	13	USER1																																						
6	Servo ready	14	USER2																																						
7	Pulse command stop	15	-																																						
2006h	-	Output port polarity	RW	YES	UINT16	3																																			
	Set the normally open/normally closed characteristics of the output port: Bit0 is the polarity setting for output port 1, and Bit1 is the polarity setting for output port 2.																																								
	0 – Normally open																																								
	1 – Normally closed																																								
	<table><tr><th>Bit15～Bit2</th><th>Bit1</th><th>Bit0</th></tr><tr><td>-</td><td>OUT2</td><td>OUT1</td></tr></table>		Bit15～Bit2	Bit1	Bit0	-	OUT2	OUT1																																	
Bit15～Bit2	Bit1	Bit0																																							
-	OUT2	OUT1																																							
2007h	-	Input port function	RW	YES	UINT16	-																																			
	00h	Maximum sub- index number	RO	NO	UINT8	06h																																			
	01h	Input port 1 function	RW	YES	UINT16	3																																			
	02h	Input port 2 function	RW	YES	UINT16	2																																			
	03h	Input port 3 function	RW	YES	UINT16	9																																			
	04h	Input port 4 function	RW	YES	UINT16	10																																			
	05h	Input port 5 function	RW	YES	UINT16	11																																			
	06h	Input port 6 function	RW	YES	UINT16	6																																			

	The input port function settings and corresponding functions are as follows:							
	Set value	Function	Set value	Function				
	0	Pulse	16	Multi-segment speed control 2				
	1	Direction	17	Multi-segment speed control 3				
	2	Encoder A phase	18	Multi-segment position control 0				
	3	Encoder B phase	19	Multi-segment position control 1				
	4	Motor offline	20	Multi-segment position control 2				
	5	Fault clearing	21	Multi-segment position control 3				
	6	Emergency stop	22	USER1				
	7	Jog forward/start-stop	23	USER2				
	8	Jog reverse/direction	24	USER3				
	9	Positive limit	25	USER4				
	10	Negative limit	26	Gear ratio				
	11	Origin signal	27	-				
	12	Homing	28	-				
	13	Reverse the direction of the motor	29	-				
	14	Multi-segment speed control 0	30	-				
	15	Multi-segment speed control 1	31	-				
2008h	-	Input port polarity	RW	YES	UINT16	0x3F		
	Set the normally open/normally closed characteristics of the input port: 0 – Normally open 1 – Normally closed							
	Bit15~Bit6		Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
	-		IN6	IN5	IN4	IN3	IN2	IN1
2009h	-	Filtering time (62.5us)	RW	YES	UINT16	128		
200Ah	-	Lock axis time (62.5us)	RW	YES	UINT16	5000		
200Bh	-	Current loop parameter	RW	YES	UINT16	-		
	00h	Maximum sub- index number	RO	NO	UINT8	04h		
	01h	Auto-PI enable 0 – Disable auto-PI Function 1 – Enable auto-PI Function	RW	YES	UINT16	1		
	02h	Current loop Kp	RW	NO	UINT16	1299		
	03h	Current loop Ki	RW	YES	UINT16	200		
	04h	Current loop Kc	RW	YES	UINT16	256		
200Ch	-	Motor parameter	RW	YES	UINT16	-		
	00h	Maximum sub- index number	RO	NO	UINT8	06h		
	01h	Motor type: 0 – Two phase stepper motor 1 – Three phase stepper motor	RW	NO	UINT16	0		

	02h	Self-identifying winding resistance value	RW	NO	UINT16	0		
	03h	Self-identifying winding inductance	RO	NO	UINT16	0		
	04h	Winding resistance value setting	RW	NO	UINT16	0		
	05h	Winding inductance setting	RW	NO	UINT16	0		
	06h	Back electromotive force coefficient	RW	NO	UINT16	0		
200Dh	-	Running direction: 0 – Normal running direction 1 – Reverse the running direction	RW	YES	UINT16	0		
200Eh	-	Internal alarm code	RO	YES	UINT16	0		
	This object displays the current fault code of the driver, with each bit of the object corresponding to an alarm status.							
	Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8
	-				CAN send full	CAN receive full	Pulse anomaly	Encoder phase
	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
	Out of tolerance	Motor phase loss	Parameter storage	Over temperature	Undervoltage	Overvoltage	Overcurrent	Internal voltage
200Fh	-	Internal status code	RO	YES	UINT16	0		
	This object displays the current status code of the driver, with each bit of the object corresponding to a status.							
	Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8
	-			Maximum position	Torque arrival	Power supply indicators	Negative limit valid	Positive limit valid
	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
	Brake valid	Speed arrival	Ready	Homing completed	Run	Position arrival	Alarm	Enable
2010h	-	Position clear: 1 – Reset object 0x6064 (actual position value) to zero	RW	YES	UINT16	0		
2011h	-	Control mode: 0 – Open loop operation 1 – Closed loop operation 2 – Closed loop operation / FOC mode	RW	YES	UINT16	0		
2020h	-	Encoder resolution (pulse/rev)	RW	YES	UINT16	4000		
2021h	-	Encoder single loop real-time position (pulse)	RO	YES	UINT16	0		
2022h	-	Position out of tolerance alarm threshold (pulse)	RW	YES	UINT16	4000		
2023h	-	Servo mode1control parameter	RW	YES	UINT16	-		
	00h	Maximum sub- index number	RO	NO	UINT8	05h		

	01h	Proportional gain Kp	RW	YES	UUINT16	2500
	02h	Integral gain Ki	RW	YES	UUINT16	0
	03h	Kd	RW	YES	UUINT16	200
	04h	Speed compensation Kvff	RW	YES	UUINT16	30
	05h	Kdi	RW	YES	UUINT16	0
2024h	-	Position arrival signal	RW	YES	UINT16	-
	00h	Maximum sub- index number	RO	NO	UINT8	03h
	01h	Position arrival signal determination mode: 0- Check at all times 1- Detection after pulse command stops	RW	YES	UINT16	0
	02h	Positioning completion output detection threshold	RW	YES	UINT16	10
	03h	Positioning completion output time window: When the position error is less than the set pulse value (02h) and continues for more than the set time (03h), it is judged to be in place.	RW	YES	UINT16	100
2025h	-	Servo speed filter (Servo mode 2 effective)	RW	YES	UINT16	-
	00h	Maximum sub- index number	RO	NO	UINT8	03h
	01h	FV1_HZ (peed feedback low-pass filtering cut-off frequency 1: Hz)	RW	YES	UINT16	200
	02h	FV2_HZ (peed feedback low-pass filtering cut-off frequency 2: Hz)	RW	YES	UINT16	600
	03h	FPOUT_HZ (FOC speed loop output low-pass frequency: Hz)	RW	YES	UINT16	5000
2026h	-	Servo mode 2 control parameter	RW	YES	UINT16	-
	00h	Maximum sub- index number	RO	NO	UINT8	05h
	01h	Position proportional gain Kp	RW	YES	UINT16	3000
	02h	Integral gain Ki	RW	YES	UINT16	1000
	03h	Speed feedback gain Kv1	RW	YES	UINT16	10
	04h	Speed feedback gain Kv2	RW	YES	UINT16	800
	05h	Speed feedforward gain Kvff	RW	YES	UINT16	600
2043h	-	Speed given (Current given speed of the motor: rpm)	RO	YES	UINT16	-
2044h	-	Speed feedback (Current actual motor speed: rpm)	RO	YES	UINT16	-
2048h	-	Bus voltage (10mV)	RO	YES	UINT16	-
2049h	-	Input signal status	RO	YES	UINT16	-

		Display the physical level status of the current input port: 0 – No input signal (Input optocoupler not conducting) 1 – With input signal (Input optocoupler conducting)													
		<table><tr><td>Bit15~Bit6</td><td>Bit5</td><td>Bit4</td><td>Bit3</td><td>Bit2</td><td>Bit1</td><td></td></tr><tr><td>-</td><td>IN6</td><td>IN5</td><td>IN4</td><td>IN3</td><td>IN2</td><td></td></tr></table>	Bit15~Bit6	Bit5	Bit4	Bit3	Bit2	Bit1		-	IN6	IN5	IN4	IN3	IN2
Bit15~Bit6	Bit5	Bit4	Bit3	Bit2	Bit1										
-	IN6	IN5	IN4	IN3	IN2										
204Ah	-	Output signal status		RO	YES	UINT16	-								
	Display the physical level status of the current output port: 0- No output signal (Output optocoupler not conducting) 1- With output signal (Output optocoupler conducting)														
	Bit15~Bit2		Bit1		Bit0										
	-		OUT2		OUT1										
2056h	-	Enable phase loss detection (effective in open-loop mode) Bit7 of the set value is used to enable phase loss detection function Bit7=0 - Disable phase loss detection Bit7=1 - Enable phase loss detection	RW	YES	UINT16	1									
2060h	-	First resonance point harmonic amplitude	RW	YES	UINT16	0									
2061h	-	First resonance point A phase harmonic phase	RW	YES	UINT16	0									
2062h	-	First resonance point B phase harmonic phase	RW	YES	UINT16	0									
2150h	-	CAN station alias	RW	NO	UINT16	0									
2151h	-	CAN communication baud rate (Reserved for use)	RO	NO	UINT16	0									

4.4. Object Group: 6000h

Index	Sub-index	Name	Accessibility	Map	Data type	Factory setting
603Fh	-	Fault code	RO	YES	UINT16	-
	When a fault occurs, first eliminate the fault condition, then write 0x0080 to Control word 0x6040 and clear 0x603F.					
	Fault code		Description			
	0x7500		Communication failure			
	0x3150		Internal voltage error in A-phase circuit			
	0x3151		Internal voltage error in B-phase circuit			
	0x8611		Closed loop mode tracking error exceeds the limit			
	0x2211		Overcurrent			
	0x3100		Overvoltage			

6040h

-	Control word	RW	YES	UINT16	0
---	--------------	----	-----	--------	---

This object is used to control the status of the driver and motion, and can enable/disable the driver, motor start/stop, clear faults, etc.

The definition of control word is as follows:

Bit	Description
0	Switch on
1	Enable voltage
2	Quick stop
3	Enable operation
4	Related to operation mode
5	Related to operation mode
6	Related to operation mode
7	Fault reset
8	Pause
9	Remote
10~15	Reserve

Detailed description of the combination of Bit0~Bit3 and Bit7:

Command	Control word Bit				
	Bit7	Bit3	Bit2	Bit1	Bit0
Shutdown	0	x	1	1	0
Switch On	0	0	1	1	1
Switch On + Enable Operation	0	1	1	1	1
Disable Voltage	0	x	x	0	x
Quick Stop	0	x	0	1	x
Disable Operation	0	0	1	1	1
Enable Operation	0	1	1	1	1
Fault Reset	0 -> 1	x	x	x	x

Definition of Bit4/Bit5/Bit6/Bit8 in related modes:

PP mode:

Bit	Name	Value	Description
4	A new target position	0 -> 1	Change from 0 to 1 and set a new target position
5	Immediate/non-immediate update	0	Non-immediate update
		1	Immediate update
6	Absolute/relative	0	Absolute position command
		1	Relative position command
8	Pause	0	Motor waiting to complete positioning
		1	Stop running

PV mode:

Bit	Name	Value	Description
8	Pause	0	Motor waiting to complete positioning
		1	Stop running

Homing mode:

Bit	Name	Value	Description
4	Start homing	0 -> 1	Start homing
8	Pause	0	Controlled by bit4
		1	Stop homing

-	Status word	RO	YES	UINT16	-
---	-------------	----	-----	--------	---

Reflect the current status of the servo driver:

Bit	Name	Description
0	伺服准备好	Ready to switch on 0: Invalid, 1: Effective
1	可以开启伺服运行	Switch on 0: Invalid, 1: Effective
2	伺服运行	Operation enabled 0: Invalid, 1: Effective
3	故障	Fault 0: Invalid, 1: Effective
4	主电路电接通	Voltage enabled 0: Invalid, 1: Effective
5	快速停机	Quick stop 0: Invalid, 1: Effective
6	伺服不可运行	Switch on disabled 0: Invalid, 1: Effective
7	警告	Warning 0: Invalid, 1: Effective
8	厂家自定义	Manufacturer specific Undefined function
9	远程控制	Remote 0: Invalid, 1: Effective(Control word effective)
10	目标到达	Target reach 0: Invalid, 1: Effective
11	内部限制有效	Internal limit active 0: Invalid, 1: Effective
12~13	运行模式相关	Operation limit active Related to each servo operation mode
14	厂家自定义	Manufacturer specific Undefined function
15	原点已找到	Home find 0: Invalid, 1: Effective

Display value (Binary value)	Description
xxxx xxxx x0xx 0000	未准备好(Not ready to switch on)
xxxx xxxx x1xx 0000	启动失效(Switch on disabled)
xxxx xxxx x01x 0001	准备好(Ready to switch on)
xxxx xxxx x01x 0011	启动(Switch on)
xxxx xxxx x01x 0111	操作使能(Operation enabled)
xxxx xxxx x00x 0111	快速停机有效(Quick stop active)
xxxx xxxx x0xx 1111	故障反应有效(Fault reaction active)
xxxx xxxx x0xx 1000	故障(Fault)

◆ Note:

- (1) Bit0 to Bit9 have the same meaning in each servo mode. After the control word 6040h sends commands in sequence, the servo feedback a certain state.

	(2) Bit12~Bit13 are related to various servo modes (Please refer to the control commands under different modes) (3) Bit10, Bit11 and Bit15 have the same meaning in each servo mode, and provide feedback on the status of the servo after executing a certain servo mode.					
605Ah	-	Quick stop mode selection	RW	NO	INT16	2
6060h	-	Operation mode: 1 – Profile Position Mode (PP) 3 – Profile Velocity Mode (PV) 6 – Homing Mode (HM)	RW	YES	INT8	1
6061h	-	Operation mode display (Defined as 0x6060)	RO	YES	INT8	0
6062h	-	User position command (Command unit)	RO	YES	INT32	-
6063h	-	Motor position feedback (Encoder unit)	RO	YES	INT32	-
6064h	-	User position feedback (Command unit)	RO	YES	INT32	-
6065h	-	User position deviation too large threshold (Command unit)	RW	YES	UINT32	10000
6067h	-	Position arrival threshold (Command unit)	RW	YES	UINT32	100
6068h	-	Position arrival time window (ms)	RW	YES	UINT16	10
606Bh	-	User actual speed command (Command unit/s)	RO	YES	INT32	-
606Ch	-	User actual speed feedback (Command unit/s)	RO	YES	INT32	-
606Dh	-	Speed arrival threshold (Command unit/s)	RW	YES	UINT16	10
606Eh	-	Speed arrival time window (ms)	RW	YES	UINT16	10
606Fh	-	Zero speed threshold (Command unit/s)	RW	YES	UINT16	10
6070h	-	Zero speed time window (ms)	RW	YES	UINT16	0
6071h	-	Target torque (mA)	RW	YES	INT16	0
6077h	-	Actual torque (mA)	RO	YES	INT16	0
607Ah	-	Target position (Command unit)	RW	YES	INT32	0
607Ch	-	Origin Offset (Command unit)	RW	YES	INT32	0
607Dh	-	Software position restriction	RW	YES	ARRAY	-
	00h	Maximum sub- index number	RO	NO	UINT8	02h
	01h	Minimum position restriction (Command unit)	RW	YES	INT32	0
	02h	Maximum position restriction (Command unit)	RW	YES	INT32	0
607Fh	-	Maximum profile velocity (Command unit/s)	RW	YES	UINT32	6000
6081h	-	Profile velocity (Command unit/s)	RW	YES	UINT32	1000
6083h	-	Profile acceleration (Command unit/s^2)	RW	YES	UINT32	1000
6084h	-	Profile deceleration (Command unit/s^2)	RW	YES	UINT32	1000
6085h	-	Quick stop deceleration (Command	RW	YES	UINT32	10000

		unit/s^2)																							
6098h	-	Homing mode	RW	YES	INT8	0																			
6099h	-	Homing speed	RW	YES	ARRAY	-																			
	00h	Maximum sub- index number	RO	YES	UINT8	0x02																			
	01h	Search for deceleration point signal speed (Command unit/s)	RW	YES	UINT32	100																			
	02h	Search for zero point signal speed (Command unit/s)	RW	YES	UINT32	10																			
609Ah	-	Homing acceleration/deceleration (Command unit/s^2)	RW	YES	UINT32	100																			
60F4h	-	Position deviation value (Command unit)	RO	YES	INT32	-																			
60FCh	-	Motor position command (Encoder unit)	RO	YES	INT32	-																			
60FDh	-	Digital input	RO	YES	UINT32	-																			
	Reflect the current IN terminal logic of the driver: 0 – Logic invalid 1 – Logic effective																								
		<table><tr><th>Bit</th><th>Description</th></tr><tr><td>0</td><td>IN1</td></tr><tr><td>1</td><td>IN2</td></tr><tr><td>2</td><td>IN3</td></tr><tr><td>3</td><td>IN4</td></tr><tr><td>4</td><td>IN5</td></tr><tr><td>5</td><td>IN6</td></tr><tr><td>6~31</td><td>NA</td></tr></table>	Bit	Description	0	IN1	1	IN2	2	IN3	3	IN4	4	IN5	5	IN6	6~31	NA							
Bit	Description																								
0	IN1																								
1	IN2																								
2	IN3																								
3	IN4																								
4	IN5																								
5	IN6																								
6~31	NA																								
60FEh	-	Digital output	RW	YES	ARRAY	-																			
	00h	Maximum sub- index number	RO	YES	UINT8	0x02																			
	01h	Digital output	RW	YES	UINT32	0																			
		Reflect the current OUT terminal logic of the driver:																							
<table><tr><th>Bit</th><th>Related OUT port</th><th>Description</th></tr><tr><td>0~15</td><td>NA</td><td></td></tr><tr><td>16</td><td>OUT1</td><td>Forced output (0: OFF, 1: ON), only effective when Bit16 is set to 1 on 60FE-02h.</td></tr><tr><td>17</td><td>OUT2</td><td>Forced output (0: OFF, 1: ON), only effective when Bit17 is set to 1 on 60FE-02h.</td></tr><tr><td>18</td><td>OUT3</td><td>Forced output (0: OFF, 1: ON), only effective when Bit18 is set to 1 on 60FE-02h.</td></tr><tr><td>19</td><td>OUT4</td><td>Forced output (0: OFF, 1: ON), only effective when Bit19 is set to 1 on 60FE-02h.</td></tr><tr><td>20~31</td><td>NA</td><td></td></tr></table>		Bit	Related OUT port	Description	0~15	NA		16	OUT1	Forced output (0: OFF, 1: ON), only effective when Bit16 is set to 1 on 60FE-02h.	17	OUT2	Forced output (0: OFF, 1: ON), only effective when Bit17 is set to 1 on 60FE-02h.	18	OUT3	Forced output (0: OFF, 1: ON), only effective when Bit18 is set to 1 on 60FE-02h.	19	OUT4	Forced output (0: OFF, 1: ON), only effective when Bit19 is set to 1 on 60FE-02h.	20~31	NA				
Bit		Related OUT port	Description																						
0~15		NA																							
16	OUT1	Forced output (0: OFF, 1: ON), only effective when Bit16 is set to 1 on 60FE-02h.																							
17	OUT2	Forced output (0: OFF, 1: ON), only effective when Bit17 is set to 1 on 60FE-02h.																							
18	OUT3	Forced output (0: OFF, 1: ON), only effective when Bit18 is set to 1 on 60FE-02h.																							
19	OUT4	Forced output (0: OFF, 1: ON), only effective when Bit19 is set to 1 on 60FE-02h.																							
20~31	NA																								

	02h	Digital output enable	RW	YES	UINT32	0	
		Set whether to enable OUT forced output					
		Bit	Related OUT port	Description			
		0~15	NA				
		16	OUT1	0: Disable OUT1 Forced output 1: EnableOUT1 Forced output			
		17	OUT2	0: Disable OUT2 Forced output 1: EnableOUT2 Forced output			
		18	OUT3	0: Disable OUT3 Forced output 1: EnableOUT3 Forced output			
		19	OUT4	0: Disable OUT4 Forced output 1: EnableOUT4 Forced output			
		20~31	NA				
60FFh	-	PV mode target speed (Command unit)	RW	YES	INT32	0	
6502h	-	Supported operation mode	RO	NO	UINT32	0x2D	
	The definition of Bit is as follows:						
	Bit	Description	Supported or not (0-unsupported/1-supported)				
	0	Profile Position Mode (PP)	1				
	1	Velocity Mode (VI)	0				
	2	Profile Velocity Mode (PV)	1				
	3	Profile Torque Mode (PT)	1				
	4	Reserved	0				
	5	Homing Mode (HM)	1				
	6	Interpolated Position Mode (IP)	0				
	7	Cyclic Sync Position Mode (CSP)	0				
	8	Cyclic Sync Velocity Mode (CSV)	0				
	9	Cyclic Sync Torque Mode (CST)	0				
	10~31	Reserved	0				

5. Appendix

Appendix A: SDO Transmission Abort Code

SDO Transmission Abort Code:

Stop Code	Function Description
0503 0000h	The trigger bit does not change alternately
0504 0000h	SDO protocol timeout
0504 0001h	Invalid or unknown client/server command word
0504 0005h	Memory overflow
0601 0000h	Object does not support access
0601 0001h	Attempt to read write-only object
0601 0002h	Attempt to write read-only object
0602 0000h	The object does not exist in the object dictionary
0604 0041h	Object cannot be mapped to PDO
0604 0042h	The number and length of the mapped objects exceed the PDO length
0604 0043h	Generic parameters are incompatible
0604 0047h	General device internal incompatibility
0606 0000h	Hardware error caused object access failure
0607 0010h	Data type does not match, service parameter length does not match
0607 0012h	The data type does not match, the length of the service parameter is too large
0607 0013h	The data type does not match, the length of the service parameter is too short
0609 0011h	Sub-index does not exist
0609 0030h	Value range outside the parameter value
0609 0031h	The write parameter value is too large
0609 0032h	The write parameter value is too small
0609 0036h	The maximum value is less than the minimum value
0800 0000h	General error
0800 0020h	Data cannot be transferred or saved to the application
0800 0021h	Data cannot be transferred or saved to the application due to local control
0800 0022h	Data cannot be transferred or saved to the application due to the current device status
0800 0023h	The object dictionary dynamically generates an error or the object dictionary does not exist
0800 0024h	Value does not exist