table of Contents

Catalog

1 communication specifications

Table 1-1 Description of performance parameters

	name	Description
EtherCAT Communication function	**Physical layer**	100BASE-TX
	Communication connector	RJ45 ×2
	Network Architecture	Concatenation
	transfer speed	2 ×100Mbps (full duplex)
	Maximum data length	1484 bytes
	SyncManager	SM0: MailBox output SM1: MailBox input SM2: Periodic output SM3 : Periodic data input
	FMMU (Bus Memory Management Unit)	FMMU0: Periodic data input area FMMU1: Periodic data output area FMMU2 : MailBox status area
	Application layer protocol	COE : CANOpen Over EtherCAT
	Synchronous mode	DC sync mode (SYNC0)
	Communication object	SDO: Service Data Object (non-periodic data) PDO: Process Data Object (Periodic Data) EMCY : Emergency
	Application layer specification	IEC61800-7 CIA402 Driver Profile
Supported CIA402 operating modes		Contour position mode Contour speed mode Contour torque mode Periodic sync position mode Cycle synchronous speed mode Periodic synchronous torque mode Zero return mode

2 wiring

2.1 Interface Information

The servo driver uses the dual RJ45 terminal as the EtherCAT protocol communication port, and the terminal interface is as shown below. The first interface (the upper interface) is the input interface, and the second interface (the lower interface) is the output interface.

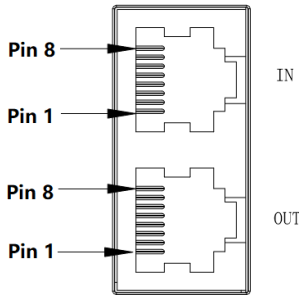


Figure 1.2 Pin Definition

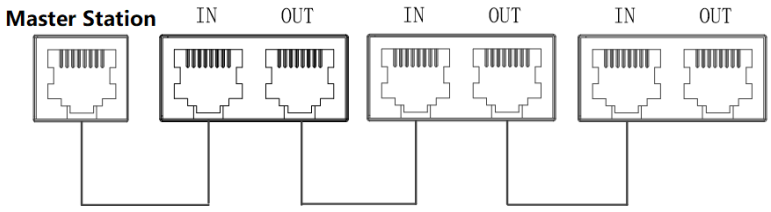
Table 2-1 Communication Signal Connector Pin Definitions

Pin number	name	Features
1	T X +	Receiver signal
2	T X -	Receiver signal
3	RX+	Send signal
4	--	--
5	--	--
6	R X -	Send signal
7	--	--
8	--	--
	shell	shield

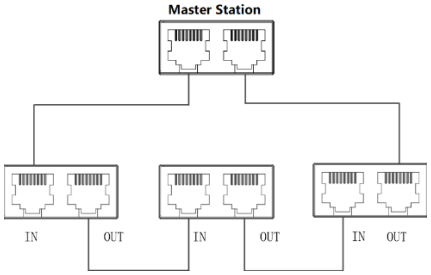
2.2 Topological connection

The SD700-EtherCAT servo drive topology communication connection is flexible and basically unlimited. The connection topology is as follows.

Linear connection:



Ring redundant connection:



2.3 communication cable

The EtherCAT communication cable uses an Ethernet Category 5 (100BASE-TX) network cable or a high-strength shielded network cable. currently using When using the SD700-EtherCAT servo drive, a shielded network cable is also required to enhance the anti-jamming capability.

2.4 EMC standard

The SD700-EtherCAT servo drive implements the IEC/EN61800-3:2004 (Adjustable speed electrical power drive systems-part3: EMC requirements and specific test methods) standard and the GB/t12668.3 national standards.

3 EtherCAT communication network settings

The EtherCAT communication network setting steps are as follows:

1. Import the XML file; (refer to the main site of the field application, determine the import method)
2. EtherCAT mode parameter setting; (refer to section 3.1)
3. EtherCAT communication parameter setting; (refer to Section 4.4)
4. Start the remote node. (You can observe section 3.2 to determine the current state)

3.1 Parameter setting of EtherCAT mode

In order to enable the servo driver to access the EthrCAT fieldbus network, it is necessary to set the relevant function code of the servo driver.

Table 3-3 System Settings Function Code Table

Pn000	**Function selection basic switch 0**	■	**Mailing address: 0x0000**
Factory default: 0000	**Setting range: 0000 ~ 001B**	**Unit: N/A**	**Control mode:**

Control mode selection	
0	Speed control mode
1	Analog speed mode
2	Torque mode
3	Internal speed
4	Internal speed <-> analog speed
5	Internal speed <-> position mode
6	Internal speed <-> torque mode
7	Position mode <-> analog speed
8	Position mode <-> torque mode
9	Torque mode <-> analog speed
10	Analog speed <-> zero speed mode
11	Position mode <-> pulse inhibit mode
16	**EtherCAT mode**

Note: When set to EtherCAT model, the control mode selection is invalid and the control mode is selected by the master station.

3.2 EtherCAT status monitoring

Control mode	
0	Reserved
1	Contour position mode (pp)
3	Contour speed mode (pv)
4	Contour torque mode (pt)
6	Zero return mode (hm)
8	Cycle sync position mode (csp)
9	Cycle Synchronous Speed Mode (csv)
10	Periodic synchronous torque mode (cst)

EtherCAT communication status	
1	Initialization status (I)
2	Pre-run status
4	Safe operating state
8	Operating status

Port connection status

Port 1 (In) connection status

Port 2 (Out) connection status

Status	Explanation	Status	Explanation
	On when the control power is ON, off when it is OFF		On when the main circuit power is ON, off when it is OFF
	Speed control : light up when speed is consistent output Position control: On when positioning is completed Torque control : always on		Lights when the rotation checkout (/TGON) output is on
			Speed control: light up when speed command is input Position control: lighted when the position command is being input
	On when servo is OFF, off when servo is ON		Torque control: light up when torque command is input Position control: light up when pulse clear signal is

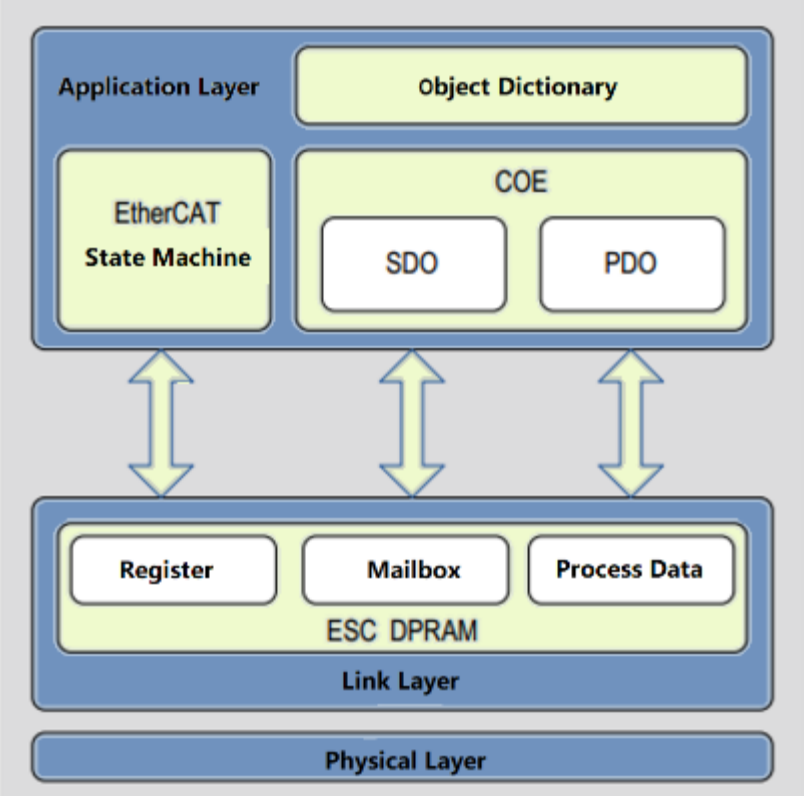
4. EtherCAT Communication Foundation

4.1 EtherCAT communication specification

Project		Specs
Communication protocol		IEC 61158 Type 12, IEC 61800-7 CiA 402 Drive Profile
Application layer	SDO	SDO request SDO response
	PDO	Variable PDO mapping
	CiA402	Contour position mode (pp) Contour velocity mode (pv) Contour torque mode (pt) Origin reset mode (hm) Synchronous periodic position mode (csp) Synchronous periodic velocity mode (csv) Synchronous periodic torque mode (cst)
Physical layer	Transport protocol	100BASE-TX (IEEE802.3)
	Maximum distance	100M
	Interface	RJ45 * 2 (INT、OUT)

4.2 Communication structure

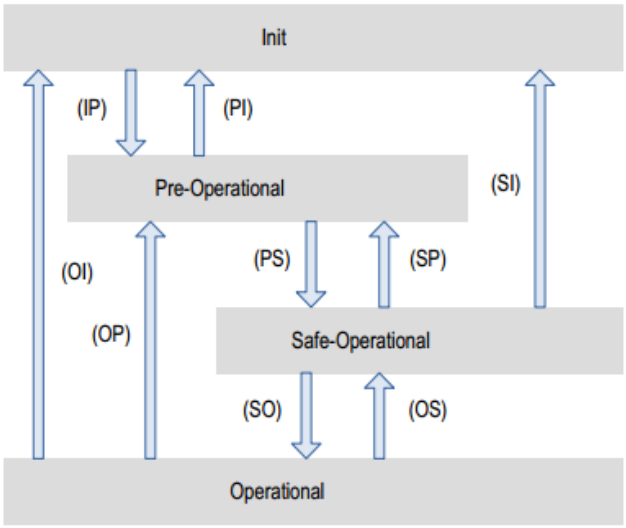
There are a variety of application layer protocols for using EtherCAT communication, but the SD700 servo driver described in this manual uses the IEC 61800-7 (CiA402)-CANopen motion control sub-protocol. The figure below shows the EtherCAT communication structure based on the CANopen application layer.



In the structure diagram, the application layer object dictionary contains communication parameters, service data objects (SDO), and process data objects (PDO). The PDO process data objects contain real-time data in the servo operation for periodic reading and writing. SDO mailbox communication is configured and accessed by aperiodic configuration of some communication parameter objects.

4.3 EtherCAT state machine

The following is a block diagram of the EtherCAT state transition:



The SD700-EtherCAT servo drive supports four state transitions and is responsible for coordinating the state transitions of the master and slave from initialization to runtime. The following table shows the relevant operations performed inside the initialization state to the running state servo.

State or state transition	Internal related operations
Initialization (I)	There is no communication on the application side, the master can only read and write the ESC register.
Initialize to pre-run conversion Init to Pre-Op (IP)	Master configuration slave station register Configure mailbox channel parameters Configuring distributed clock (DC) related registers The master writes a status control register to request a pre-run status
Pre-run (P)	Application layer mailbox data communication
Pre-run to convert to safe operation Pre-Op to Safe-Op (PS)	Master uses mailbox initialization process data mapping SM channel used by the master station configuration process data communication Master station configuration FMMU The master writes a status control register to request a safe operating state
Safe operation (S)	Application layer supports mailbox data communication There is process data communication, but only input data is allowed to be read, and no output signal is generated.
Safe running to run conversion Safe-Op to Op(SO)	The master sends valid output data The master writes a status control register to request a run status
Operating status (O)	Input and output are all valid Still using mailbox communication

4.4 Process Data

Real-time process data (PDO), following the producer - consumer model. The PDO can be divided into RPDO (Receive PDO) and TPDO (Transmit PDO). The slave receives the master command via RPDO and sends its own status information to the master via TPDO.

4.4.1 PDO Mapping Parameters

The PDO mapping is used to establish a mapping relationship between the object dictionary and the PDO. In the SD700-EtherCAT driver, 1600h~1603h is RPDO, and 1A00h~1A03h is TPDO. The following table shows information about the PDO mapping and mapping objects of the product, and the mapping objects can be changed.

PDO	index	Maximum number of maps	Longest mapped byte	Default mapping object
RPDO	1600h	8 Ge	32	6040h (control word) 607Ah (target location) 60FFh (target speed) 6071h (target torque) 6060h (operating mode)
	1601h	8 Ge	32	6040h (control word) 607Ah (target location)
	1602h	8 Ge	32	6040h (control word) 60FFh (target speed)
	1603h	8 Ge	32	6040h (control word) 6071h (target torque)
TPDO	1A00h	8 Ge	32	6041h (status word) 6064h (actual location) 606Ch (actual speed) 6077h (actual torque)
	1A01h	8 Ge	32	6041h (status word) 6064h (actual location)
	1A02h	8 Ge	32	6041h (status word) 6064h (actual location) 606Ch (actual speed)
	1A03h	8 Ge	32	6041h (status word) 6064h (actual location) 6077h (actual torque)

Note: It is recommended that the total number of RPDO and TPDO mapping bytes does not exceed 52 bytes, which may affect the servo performance.

4.4.2 PDO Synchronization Management Assignment Settings

In periodic data communication, the process data may include multiple PDO mapping objects. The CoE protocol uses the data objects 0x1C10~0x1C2F to define a list of PDO mapping objects of the corresponding synchronous management channels. Multiple PDOs may be mapped in different sub-indexes of 0x1C10~0x1C2F. In the SD700-EtherCAT servo drive, supports only. 1 th RPDO and. 1 th TPDO allocated in the following table:

Index	Subindex	Mapping assignment	Default map assignment
0x1C12	01h	Select one of 1600h~1603h as the RPDO application	1600h
0x1C13	01h	Select one of 1A00h~1A03h as the TPDO application	1A00h

4.4.3 PDO Configuration

The PDO mapping parameters (e.g., 1600h) contain pointers to the process data of the PDO that the PDO needs to send or receive, including the index of the mapping object, the sub-index, and the length of the object. The mapping parameter sub-index 0 records the number N of mapping objects of the PDO (for example, the maximum value of the SD700-EtherCAT servo N is 8), and one or more objects can be mapped at the same time. Sub-indexes 1 to 8 are mapping contents (mapping objects). The mapping parameter content is defined as follows.

Number of digits	31	.....	16	15	.....	8	7	.....	0
meaning	index			Subindex			Object length		

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

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

For example, the mapping parameter of object 6040h-00 is 60400010h.

The configuration process of PDO is as follows:

1. Cancel PDO. The 1C12h (or 1C13h) of 00h subindex write 0;
2. Clear the original mapping content. Write 0 to the 00h subindex of the mapping parameter (such as 1600h-00) to clear the original mapping content;
3. Write the PDO mapping content. Write the mapping parameter sub-index 1~N (N max is 8) according to the above definition;
4. Write the total number of PDO mapping objects. Step 3 The number of map N write mapping parameter subindex 0 (e.g. 1600h-00 write N);
5. Enable PDO. The 1C12h (or 1C13h) of 00h subindex write 1.

4.5 Mailbox Data SDO

The mailbox data SDO is used to transmit aperiodic data, such as the configuration of communication parameters, the configuration of servo drive operating parameters, and the like. In the SD700-EtherCAT servo drive, SDO requests and SDO are currently supported.

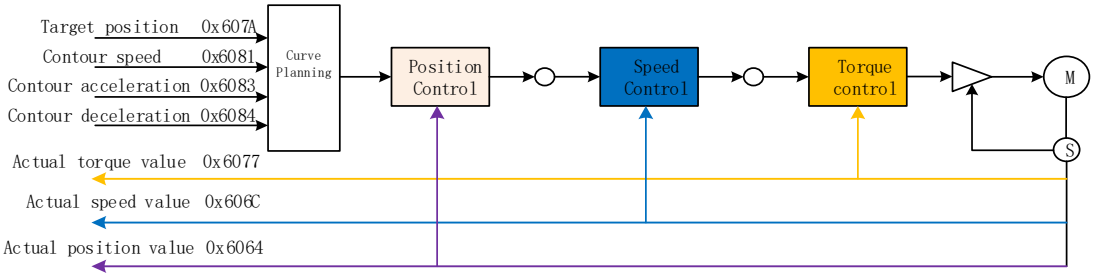
4.6 distributed clock

The distributed clock can make different servo devices use the same system clock, which can ensure that different servos receive instructions at the same time and execute instructions at the same time, which can achieve absolute time synchronization. The slave device can generate a synchronization signal based on the synchronization system time. The SD700-EtherCAT servo drive only supports DC synchronous mode. The synchronization period is controlled by SYNC0 and the period is 125us or an integer multiple of 250us.

5. Control mode

5.1 contour position mode (pp)

In the contour position mode, the master station sends a dictionary of related objects such as the required target position (absolute or relative), the speed of the position curve, acceleration and deceleration to the servo drive, and the servo driver generates a target curve based on the received related data and commands. instruction.



Related object dictionary

Control word 6040h		
Bit	**name**	**description**
0	Servo ready (Switch On)	0: invalid; 1: valid
1	Turn on the main circuit (Enable Voltage)	0: invalid; 1: valid
2	Fast stop (Quick Stop)	0: valid; 1: invalid
3	Servo operation (Enable Operation)	0: invalid; 1: valid
4	New target location (New Set-Point)	The rising edge triggers a new target position
5	Update Now (Change Set)	0: Not updated immediately; 1: Updated immediately
6	Absolute position command / relative position command (Abs/Rel)	0: The target position is an absolute position command 1: Target position is relative position command

Status word 6041h		
Bit	**name**	**Description**
10	Reaching the target (Target Reached)	0: The target location has not arrived 1: Target location arrives
12	Target location update (Set Point Acknowledge)	0: update target location 1: Cannot update the target location
13	Follow error (Following error)	0: No position deviation is too large 1: The position deviation is too large
15	Origin back to zero is completed (Home Find)	0: Zero return to origin is not completed 1: Home zero return is completed

index	Subindex	name	Read and write	type of data	unit	Predetermined area
0x603F	00	error code	RO	UINT 16	-	0~65535

0x6040	00	Control word	RW	UINT 16	-	0~65535
0x6041	00	Status word	RO	UINT 16	-	0~65535
0x6060	00	Operating mode	RW	INT 8	-	0~10
0x6061	00	Mode display	RO	INT 8	-	0~10
0x6062	00	Position command	RO	INT32	Command unit	-
0x6063	00	Position feedback	RO	INT 32	Encoder unit	-
0x6064	00	Position feedback	RO	INT 32	Command unit	-
0x6065	00	Position deviation excessive threshold	RW	UINT 32	Command unit	0~(2^{32}-1)
0x6067	00	Location arrival threshold	RW	UINT 32	Command unit	0~(2^{32}-1)
0x6068	00	Location arrival window	RW	UINT 16	2ms	0~65535
0x606C	00	Actual speed feedback	RO	INT 32	Command unit /s	-
0x6077	00	Actual torque	RO	UINT 16	1%	-
0x607A	00	target location	RW	INT 32	Command unit	-2^{31}~(2^{31}-1)
0x6081	00	Contour speed	RW	UINT 32	Command unit /s	0~(2^{32}-1)
0x6083	00	Acceleration	RW	UINT 32	Command unit / S^2	0~(2^{32}-1)
0x6084	00	decrease speed	RW	UINT 32	Command unit / S^2	0~(2^{32}-1)
0x6091	01	Motor resolution	RW	UINT 32	-	0~(2^{32}-1)
	02	Axis resolution	RW	UINT 32	-	1~(2^{32}-1)
0x60E0	00	Forward torque limit	RW	UINT 16	0.1%	0~65535

0x60E1	00	Negative torque limit	RW	UINT 16	0.1%	0~65535
0x60F4	00	Position deviation	RO	INT 32	Encoder unit	-
0x60FC	00	Position command	RO	INT 32	Encoder unit	-

Curve planning - not immediately updated

1 The main station sends the relevant information of the position command to the slave station according to the need (acceleration time 6083h, deceleration time 6084h, contour speed 6081h, target position 607A);

② master station 6040h of bit4 set to 1 , the detected slave station 6040h of bit4 after the rising edge of the signal, for receiving the displacement new processing instruction.

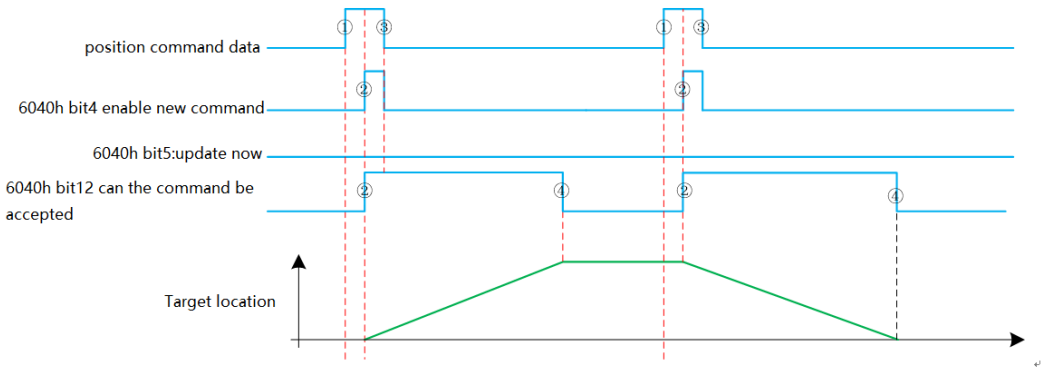
First, the station determines 6040h of bit5 whether 0, not 0 no correlation processing instruction information;

Next, the slave determines 6040h of bit5 is 0 , and 6041h of bit12 to 0 , from the station 6041h of bit12 is set to 1 , while executing the instruction ① related information. At this point, the slave is unable to continue to receive the new displacement command state.

③ the master station detects the status word 6041h of bit12 is set to 1 , the master station can release the displacement instruction data, the control word and 6040h of bit4 a 1 is set 0 .

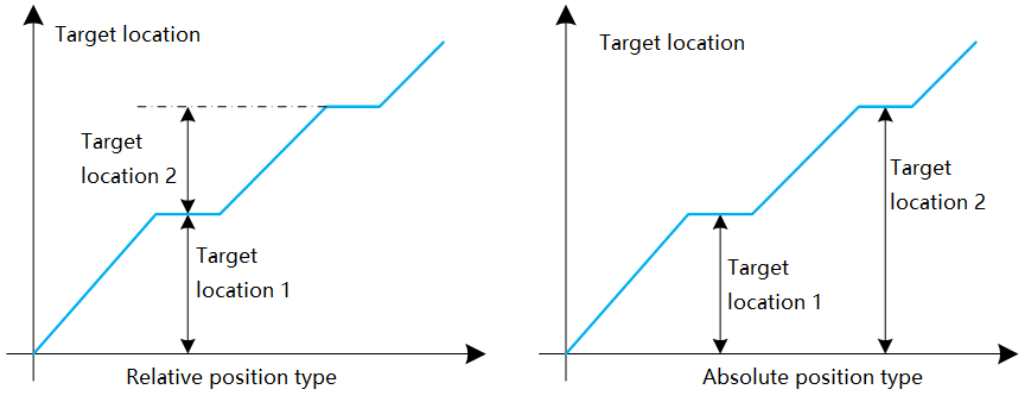
④ station detects a control word from 6040h to bit4 is 0 , positioning is completed after the current segment, the 6041h of bit12 is set 0 , indicating that the slave may receive a new shift command.

In the non-immediate update mode, the servo does not process the new displacement command while the current segment is running. The servo can receive and process the new displacement command only after the current segment positioning is completed.



Non-immediate update timing diagram

Control word 6040h of bit6 is 0, the absolute position of the control word 6040h to bit6 is. 1, the relative position.



Location type

Curve planning - update now

1 The main station sends the relevant information of the position command to the slave station according to the need (acceleration time 6083h, deceleration time 6084h, contour speed 6081h, target displacement 607A);

② master station 6040h of bit4 set to 1 , the detected slave station 6040h of bit4 after the rising edge of the signal, for receiving the displacement new processing instruction.

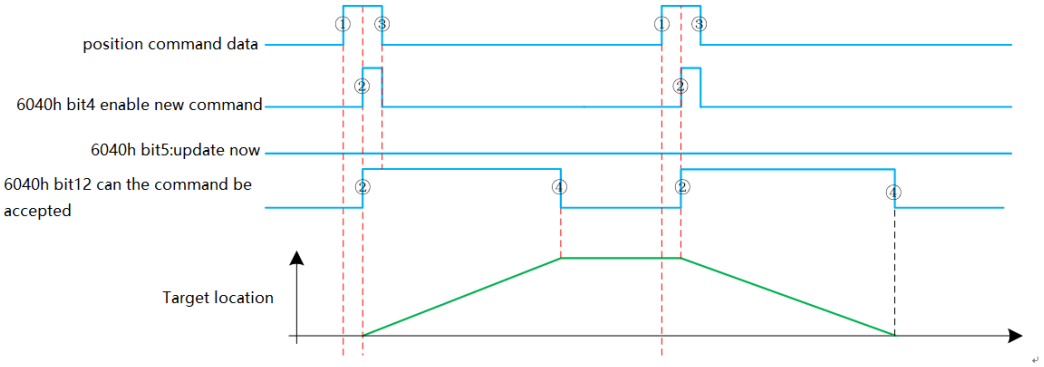
First, the station determines 6040h of bit5 whether one, not one is not related to the information processing instruction;

Next, the slave determines 6040h of bit5 is 1 , and 6041h of bit12 to 0 , from the station 6041h of bit12 is set to 1 , while executing the instruction ① related information. At this point, the slave is unable to continue to receive the new displacement command state.

③ the master station detects the status word 6041h of bit12 is set to 1 , the master station can release the displacement instruction data, the control word and 6040h of bit4 a 1 is set 0 .

④ detected from the station 6040h of bit4 a 1 is set to 0 , the then 6041h of bit12 is set to 0 . Indicates that the slave can receive a new displacement command.

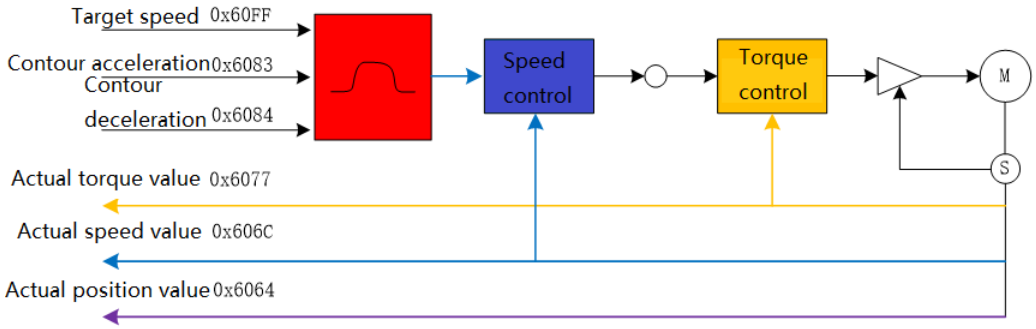
During the Immediate update mode, the current segment is running, is detected from the station to the 6040h of bit4 rising, while 604h of bit12 to 0, the servo displacement may receive a new command process.



Immediate update timing diagram

5.2 Contour speed mode (pv)

In the contour speed mode, the master sends the required target speed, acceleration time, and deceleration time to the servo drive, and the servo drive performs speed and torque adjustment.



Related object dictionary

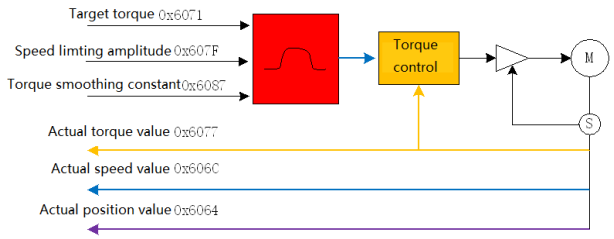
index	Subindex	name	Read and write	type of data	unit	Predetermined area
0x603F	00	error code	RO	UINT 16	-	-
0x6040	00	Control word	RW	UINT 16	-	0~65535
0x6041	00	Status word	RO	UINT 16	-	0~65535
0x6060	00	Operating mode	RW	INT 8	-	0~10

0x6061	00	Mode display	RO	INT 8	-	0~10
0x6063	00	Position feedback	RO	INT32	Encoder unit	-
0x6064	00	Position feedback	RO	INT32	Command unit	-
0x606C	00	Actual speed feedback	RO	INT 32	Command unit /s	-
0x6077	00	Actual torque	RO	UINT 16	1%	-
0x60E0	00	Forward torque limit	RW	UINT 16	0.1%	0~65535
0x60E1	00	Negative torque limit	RW	UINT 16	0.1%	0~65535
0x60FF	00	Target speed	RW	INT 32	Command unit / S	-2^{31}~(2^{31}-1)
0x607F	00	Maximum contour speed	RW	UINT 32	0.1rpm	0~(2^{32}-1)
0x6083	00	Acceleration	RW	UINT 32	Command unit / S^2	0 ~ (2^{32}-1)
0x6084	00	decrease speed	RW	UINT 32	Command unit / S^2	0 ~ (2^{32}-1)

Note: The speed limit value is determined by the smaller of 0x607F and the maximum motor speed.

5.3 Profile Torque Mode (pt)

In the contour torque mode, the primary station transmits the target torque command 6071h, the torque ramp constant 6087h, and the speed limit value 607Fh to the servo drive, which is executed internally by the servo driver. When the speed reaches the limit value of 607Fh, it will enter the speed regulation phase.



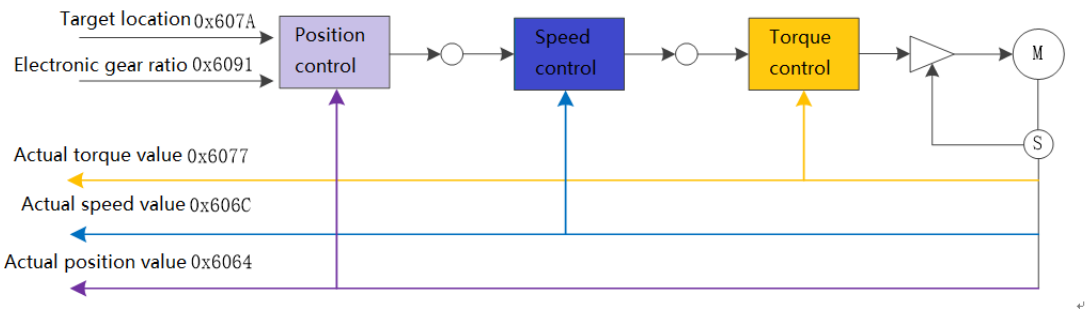
Related object dictionary

index	Subindex	Name	Read and write	type of data	unit	Predetermined area
0x603F	0x00	error code	RO	UINT 16	-	0~65535
0x6040	0x00	Control word	RW	UINT 16	-	0~65535

0x6041	0x00	Status word	RO	UINT 16	-	0~65535
0x6060	0x00	Operating mode	RW	INT 8	-	0~10
0x6061	0x00	Mode display	RO	INT 8	-	0~10
0x606C	0x00	Actual speed feedback	RO	INT 32	Command unit /s	-
0x6071	0x00	Target torque	RW	INT 16	0.1%	-3000~3000
0x6074	0x00	Torque command	RO	INT 16	1%	-
0x6077	0x00	Actual torque	RO	UINT 16	1%	-
0x607F	0x00	Maximum contour speed	RW	UINT 32	0.1rpm	0-50000
0x6087	0x00	Torque ramp time	RW	UINT 32	Ms	0-(2^{32}-1)
0x60E0	00	Forward torque limit	RW	UINT 16	0.1%	0~65535
0x60E1	00	Negative torque limit	RW	UINT 16	0.1%	0~65535

5.4 cycle synchronous position mode (csp)

In the cyclic synchronous position mode, the primary station transmits the planned target position 607Ah to the servo drive in a periodically synchronized manner according to the value of the target speed 60FFh. Position, speed and torque control are performed inside the servo drive.



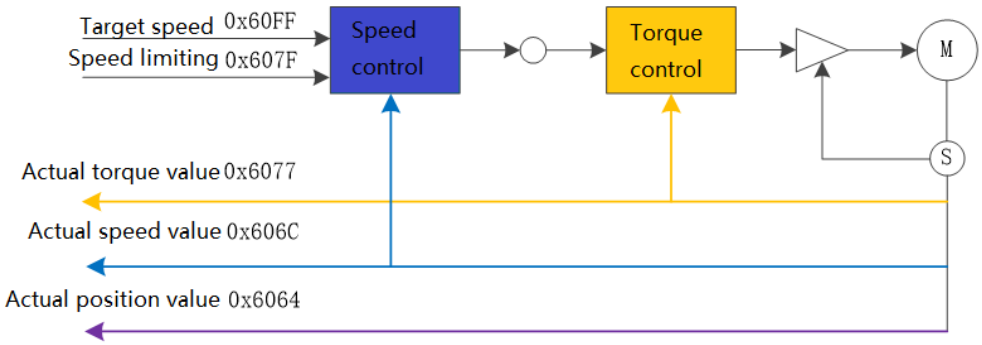
Related object dictionary

index	Subindex	Name	Read and write	type of data	unit	Predetermined area

0x603F	00	error code	RO	UINT 16	-	0~65535
0x6040	00	Control word	RW	UINT 16	-	0~65535
0x6041	00	Status word	RO	UINT 16	-	0~65535
0x6060	00	Operating mode	RW	INT 8	-	0~10
0x6061	00	Mode display	RO	INT 8	-	0~10
0x6062	00	Position command	RO	INT 32	Command unit	-
0x6063	00	Position feedback	RO	INT 32	Encoder unit	-
0x6064	00	Actual position feedback	RO	INT 32	Command unit	-
0x6065	00	Position deviation excessive threshold	RW	UINT 32	Command unit	$0\sim(2^{32}-1)$
0x6067	00	Location arrival threshold	RW	UINT 32	Command unit	$0\sim(2^{32}-1)$
0x6068	00	Location arrival window time	RW	UINT 16	2ms	0~65535
0x606C	00	Actual speed feedback	RO	INT 32	Command unit /s	-
0x607A	00	target location	RW	INT 32	Command unit	$-2^{31}\sim(2^{31}-1)$
0x6091	01	Motor resolution	RW	UINT 32	-	$0\sim(2^{32}-1)$
	02	Axis resolution	RW	UINT 32	-	$1\sim(2^{32}-1)$
0x60B0	00	Position offset	RW	INT 32	Command unit	$-2^{31}\sim(2^{31}-1)$
0x60F4	00	Position deviation	RO	INT 32	Encoder unit	-
0x60FC	00	Position command	RO	INT 32	Encoder unit	-

5.5 cycle synchronous speed mode (csv)

In the cycle synchronous speed mode, the master station sends the calculated target speed 60FFh periodically to the servo drive, and the speed and torque adjustment are internally adjusted by the servo.



Related object dictionary

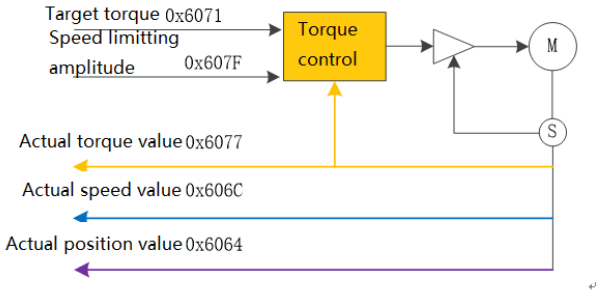
index	Subindex	Name	Read and write	type of data	unit	Predetermined area
0x603F	00	error code	RO	UINT 16	-	0~65535
0x6040	00	Control word	RW	UINT 16	-	0~65535
0x6041	00	Status word	RO	UINT 16	-	0~65535
0x6060	00	Operating mode	RW	INT 8	-	0~10
0x6061	00	Mode display	RO	INT 8	-	0~10
0x6063	00	Position feedback	RO	INT 32	Encoder unit	-
0x6064	00	Actual position feedback	RO	INT 32	Command unit	
0x606C	00	Actual speed feedback	RO	INT 32	Command unit /s	-
0x6077	00	Actual torque	RO	INT 16	1%	-
0x607F	00	Maximum speed	RW	UINT 32	0.1r/min	0~50000
0x60E0	00	Forward torque limit	RW	UINT 16	0.1%	0~65535
0x60E1	00	Negative torque limit	RW	UINT 16	0.1%	0~65535
0x6091	01	Motor resolution	RW	UINT 32	-	0~(2^{32}-1)

	02	Axis resolution	RW	UINT 32	-	1~(2 ^{32} -1)
0x60FF	00	Target speed	RW	INT 32	Command unit /s	-2 ^{31} ~(2 ^{31} -1)

Note: The speed limit value is determined by the smaller of 0x607F and the maximum motor speed.

5.6 cycle synchronous torque mode (cst)

In the cyclic synchronous torque mode, the master station periodically synchronizes the calculated target torque 6071h to the servo drive, and the torque adjustment is performed internally by the servo. When the speed reaches the limit value, it enters the speed regulation phase.



Related object dictionary

index	Subindex	Name	Read and write	type of data	unit	Predetermined area
0x603F	00	error code	RO	UINT 16	-	0~65535
0x6040	00	Control word	RW	UINT 16	-	0~65535
0x6041	00	Status word	RO	UINT 16	-	0~65535
0x6060	00	Operating mode	RW	INT 8	-	0~10
0x6061	00	Mode display	RO	INT 8	-	0~10
0x6064	00	Actual position feedback	RO	INT 32	Command unit	-
0x606C	00	Actual speed feedback	RO	INT 32	Command unit /s	-
0x6071	00	Target torque	RW	INT 16	0.1%	-3000~3000
0x6077	00	Actual torque	RO	INT 16	1%	-
0x607F	00	Maximum speed	RW	UINT 32	0.1r/min	0~(2 ^{32} -1)

0x60E0	00	Forward torque limit	RW	UINT 16	0.1%	0~65535
0x60E1	00	Negative torque limit	RW	UINT 16	0.1%	0~65535

Note: The speed limit value is determined by the smaller of 0x607F and the maximum motor speed.

5.7 zero return mode (hm)

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

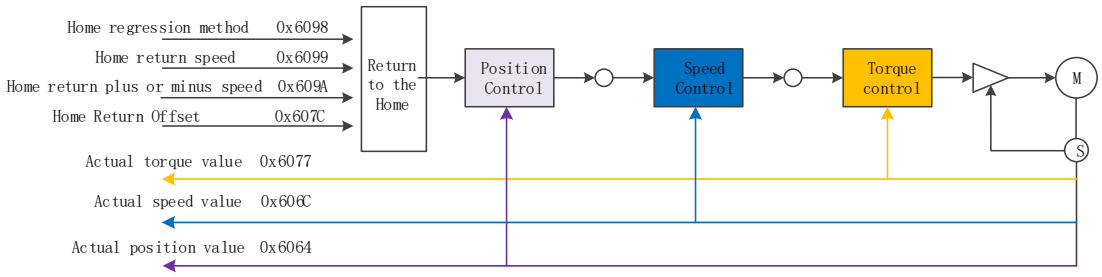
Mechanical origin: A fixed position on the machine that corresponds to a certain origin signal switch.

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

Mechanical zero point: Absolute 0 position on the machine.

After the servo drive returns to zero at the origin, the motor stops at the machine origin, and the positional relationship between the machine origin and the mechanical zero is adjusted by setting the value of the object dictionary 0x607C.

Zero return control block diagram



Related object dictionary

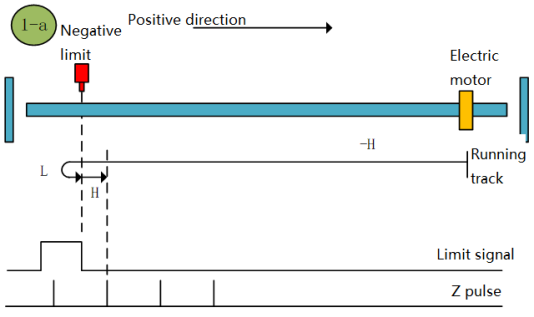
index	Subindex	name	Read and write	type of data	unit	Predetermined area
0x603F	00	error code	RO	UINT 16	-	0~65535
0x6040	00	Control word	RW	UINT 16	-	0~65535
0x6041	00	Status word	RO	UINT 16	-	0~65535
0x6060	00	Operating mode	RW	INT 8	-	0~10
0x6061	00	Mode display	RO	INT 8	-	0~10
0x6062	00	Command position	RO	INT 32	Command unit	-
0x6064	00	Actual position feedback	RO	INT 32	Command unit	-
0x606C	00	Actual speed feedback	RO	INT 32	Command unit /s	-

0x6067	00	Position arrival threshold	RO	UINT 32	Command unit	-
0x6068	00	Location arrival window	RW	UINT 16	2ms	-
0x6077	00	Actual torque	RO	INT 16	1%	-
0x6098	00	Origin return method	RW	INT8	-	1~35
0x6099	01	High speed search deceleration point	RW	UINT 32	Command unit /s	0~(2^{32}-1)
	02	Search origin low speed	RW	UINT 32	Command unit /s	1~(2^{32}-1)
0x609A	00	Acceleration	RW	UINT 32	Command unit / s^2	0~(2^{32}-1)
0x60F4	00	Position deviation	RO	INT 32	Encoder unit	-

Introduction to the zero return method:

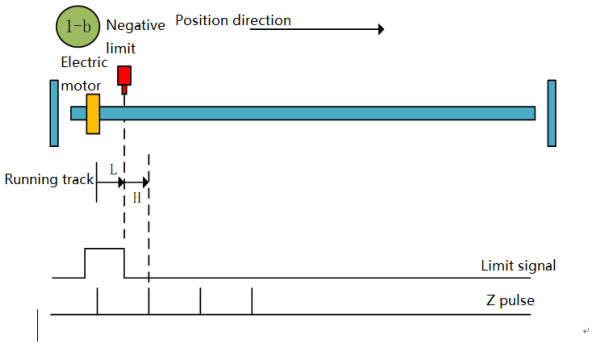
Zero return mode 1 (6098 00h=1)

a Start zero return → Reverse high speed to find negative limit → Hit negative limit rising edge → Deceleration to 0 → Forward low speed to find negative limit falling edge → Forward to find Z pulse



Zero return method 1-a

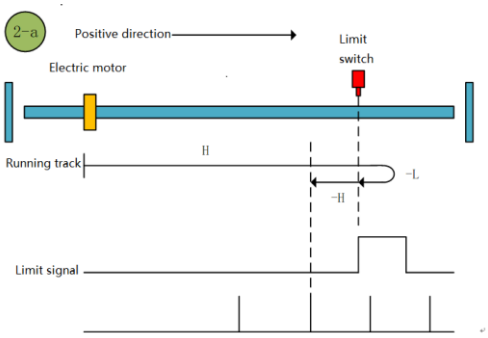
b Start origin return → Negative limit valid → Positive low speed find negative limit falling edge → Forward find Z pulse



Zero return method 1-b

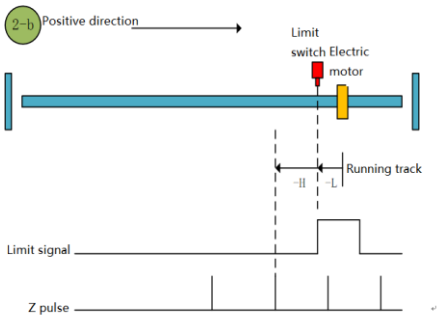
Zero return mode 2 (6098 00h = 2)

a Start zero return → Forward high speed correction limit → Hit the positive limit rising edge → Decelerate to 0 → Reverse low speed correction limit falling edge → Reverse find Z pulse



Zero return method 2-a

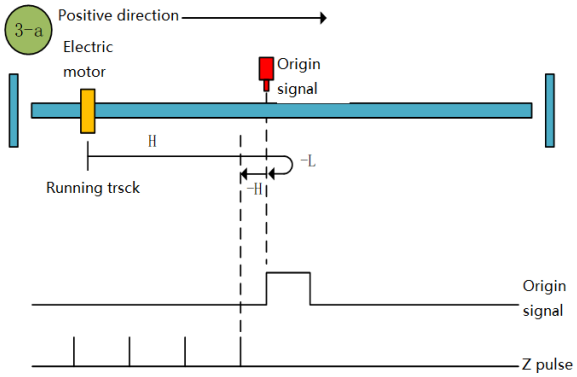
b Start the origin return → positive limit valid → reverse low speed correction limit falling edge → reverse find Z pulse



Origin return mode 2-b

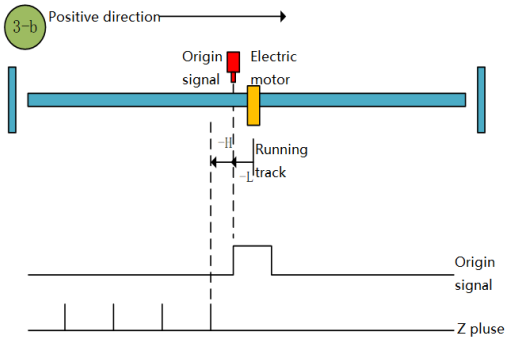
Zero return mode 3 (6098 00h = 3)

a. Start origin return to zero → Origin signal is OFF → Forward high speed finds the origin signal rising edge → Deceleration to 0 → Reverse low speed finds the origin signal falling edge → Reverse finds Z pulse



Origin return mode 3-a

b. Start origin return to zero → origin signal ON → reverse low speed to find the origin falling edge → reverse find Z pulse

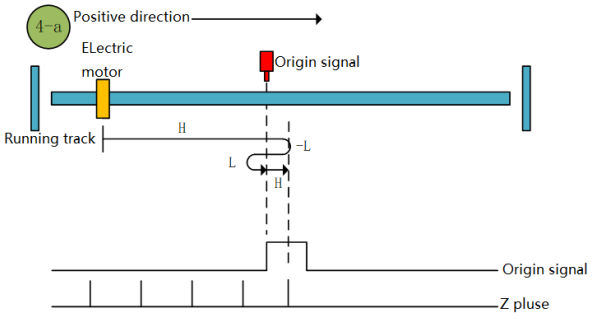


Origin return mode 3-b

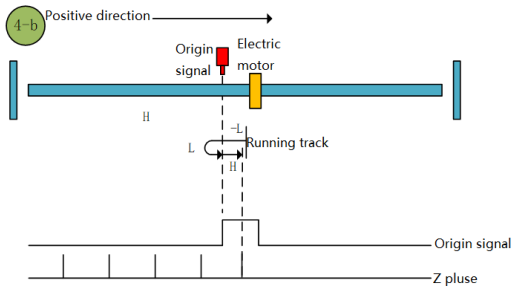
Zero return mode 4 (6098 00h = 4)

a. Start zero return → Origin signal OFF → Forward high speed to find the origin rising edge → Deceleration to 0 → Forward low speed to find the origin falling edge → Forward to find Z pulse.

b. Start the origin return → origin signal ON → reverse low speed to find the origin falling edge → positive low speed to find the origin rising edge → forward find Z pulse



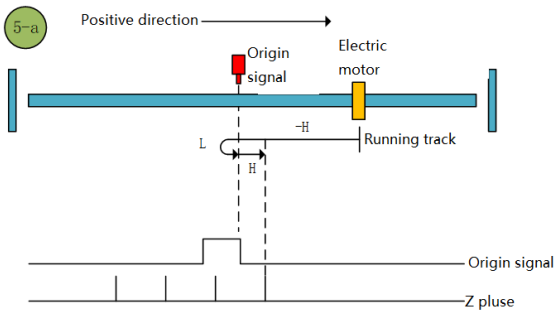
Origin return mode 4-a



Origin return mode 4-b

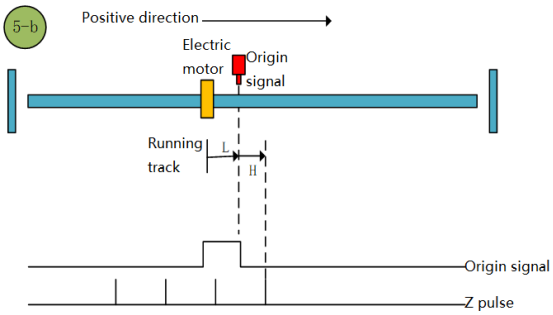
Zero return mode 5 (6098 00h = 5)

a. Start origin return to zero → Origin signal OFF → Reverse high speed to find the origin rising edge → Deceleration to 0 → Forward low speed to find the origin falling edge → Forward to find Z pulse



Zero return method 5-a

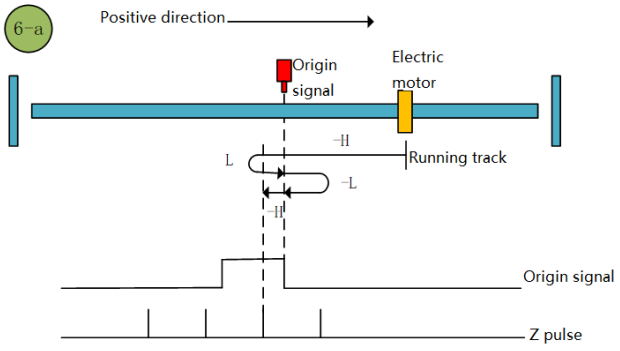
b. Start origin return to zero → origin signal ON → forward low speed to find the origin falling edge → forward find Z pulse



Origin return mode 5-b

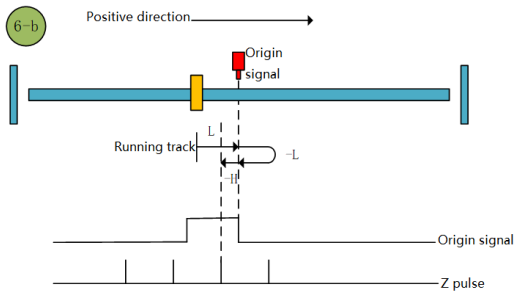
Origin return mode 6 (6098 00h = 6)

a. Start OPR → Origin signal OFF → Reverse high speed to find the origin rising edge → Deceleration to 0 → Forward low speed to find the origin falling edge → Reverse low speed to find the origin rising edge → Reverse to find Z pulse



Origin return mode 6-a

b. Start the origin return → the origin signal ON → the forward low speed to find the origin falling edge → the reverse low speed to find the origin rising edge → reverse to find the Z pulse



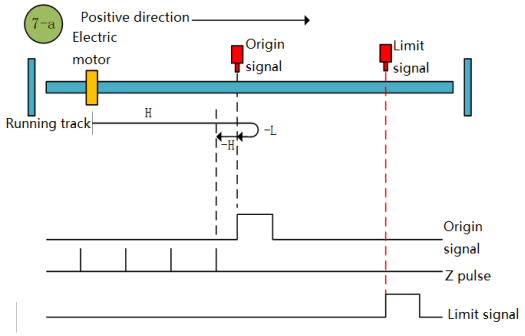
Origin return mode 6-b

Zero return mode 7 (6098 00h = 7)

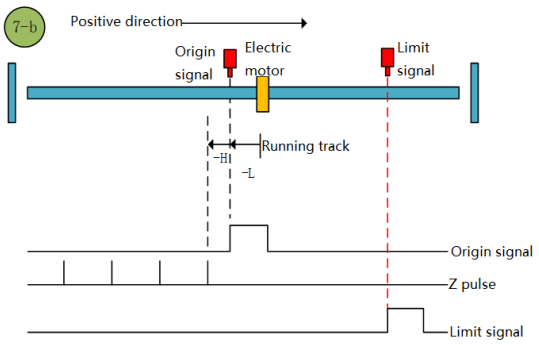
a. Start zero return → Origin signal OFF → Forward high speed to find the origin rising edge → Deceleration to 0 → Reverse low speed to find the origin falling edge → Reverse to find Z pulse

b. Start zero return → origin signal ON → reverse low speed to find the origin falling edge → reverse find Z pulse

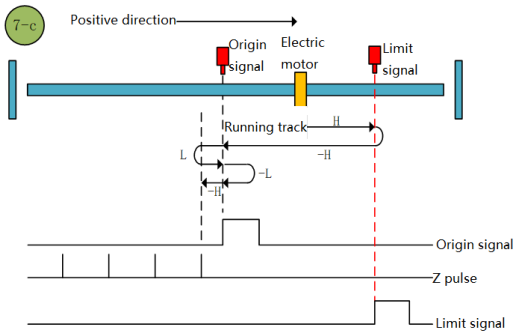
c. Start zero return → origin OFF → forward high speed to find the origin rising edge → hit the positive limit → reverse high speed to find the origin falling edge → decelerate to 0 → forward low speed to find the origin rising edge → reverse low speed to find the origin Find Z pulse along → reverse



Zero return method 7-a



Origin return mode 7-b



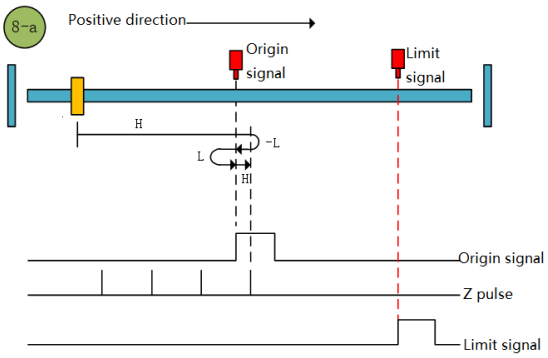
Origin return mode 7-c

Origin return mode 8 (6098 00h = 8)

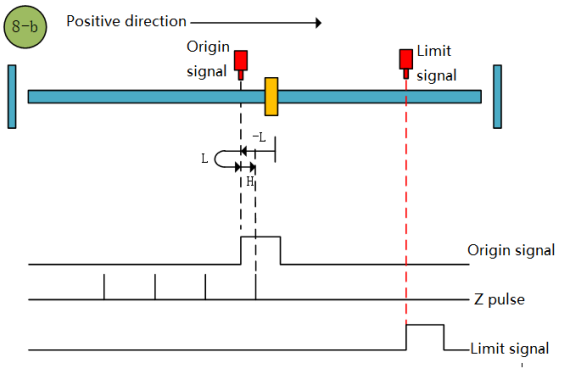
a. begins zeroing origin point signal → OFF → positive → rising speed seek decelerate to the origin 0 → → reverse slow to find the falling edge of positive origin to find the origin of the rising edge of the low-speed forward looking → Z pulse

b. Start zero return → origin signal ON → reverse low speed to find the origin falling edge → positive low speed to find the origin rising edge → forward find Z pulse

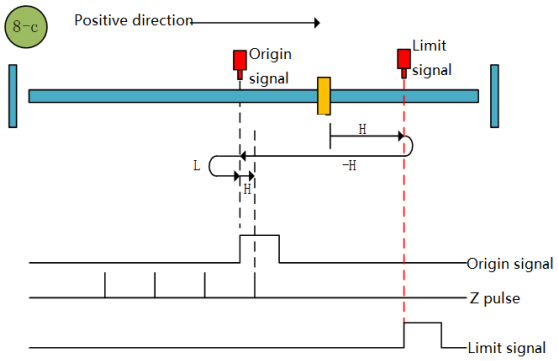
c. Start zero return → origin OFF → forward high speed to find the origin rising edge → hit the positive limit → reverse high speed to find the origin falling edge → decelerate to 0 → forward low speed to find the origin rising edge → forward find Z pulse



Origin return mode 8-a



Origin return mode 8-b



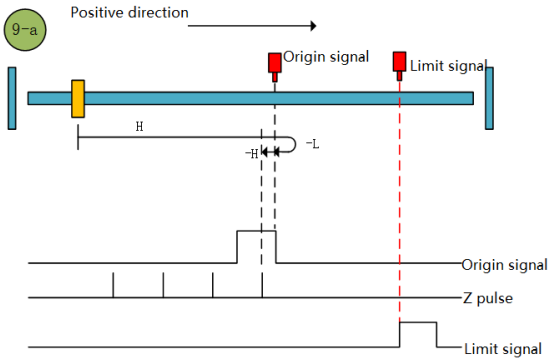
Origin return mode 8-c

Zero return mode 9 (6098 00h = 9)

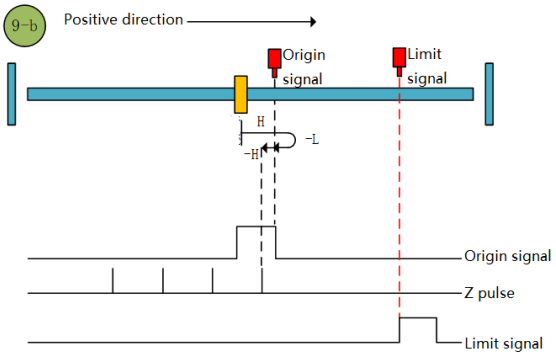
a. Start zero return → Origin signal OFF → Forward high speed to find the origin falling edge → Deceleration to 0 → Reverse low speed to find the origin rising edge → Reverse to find Z pulse

b. Start zero return → origin signal ON → forward high speed to find the origin falling edge → decelerate to 0 → reverse low speed to find the origin rising edge → reverse find Z pulse

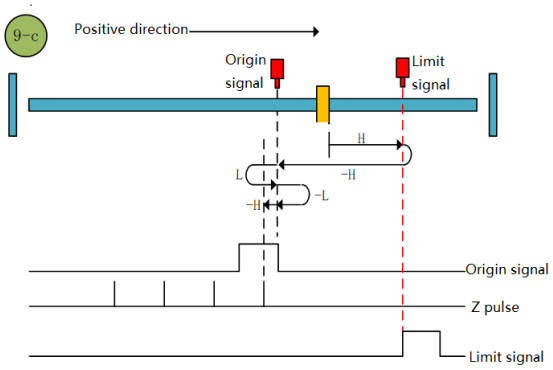
c. Start origin return → origin OFF → forward high speed to find the origin falling edge → hit the positive limit → reverse high speed to find the origin rising edge → decelerate to 0 → forward low speed to find the origin falling edge → reverse low speed to find the origin rising edge →Reverse looking for Z pulse



Zero return method 9-a



Zero return method 9-b



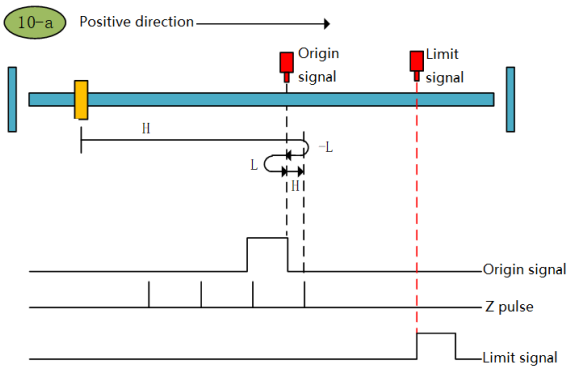
Zero return method 9-c

Zero return mode of origin 10 (6098 00h = 10)

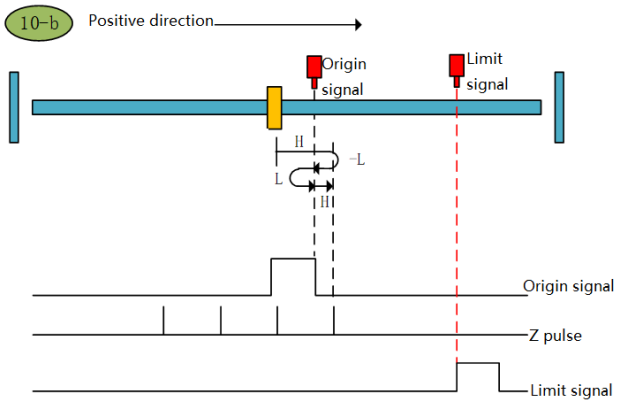
a. Start origin return → Origin signal OFF → Forward high speed to find the origin falling edge → Deceleration to 0 → Reverse low speed to find the origin rising edge → Forward low speed to find the origin falling edge → Forward to find Z pulse

b. Start origin return → origin signal ON → forward high speed to find the origin falling edge → decelerate to 0 → reverse low speed to find the origin rising edge → positive low speed to find the origin falling edge → forward find Z pulse

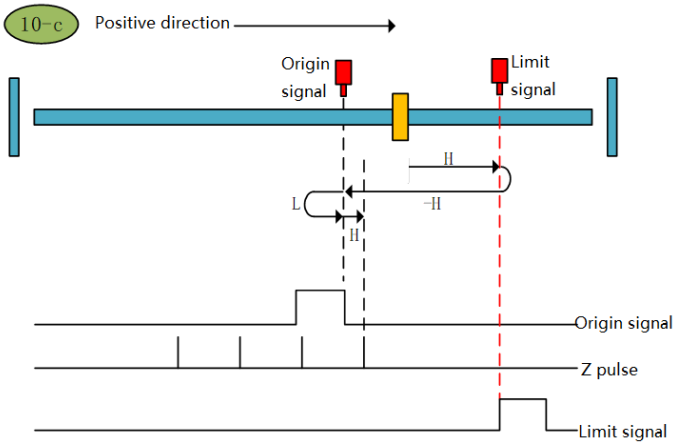
c. Start zero return → origin OFF → forward high speed to find the origin falling edge → hit the positive limit → reverse high speed to find the origin rising edge → decelerate to 0 → forward low speed to find the origin falling edge → forward find Z pulse



Zero return method 10-a



Zero return method 10-b



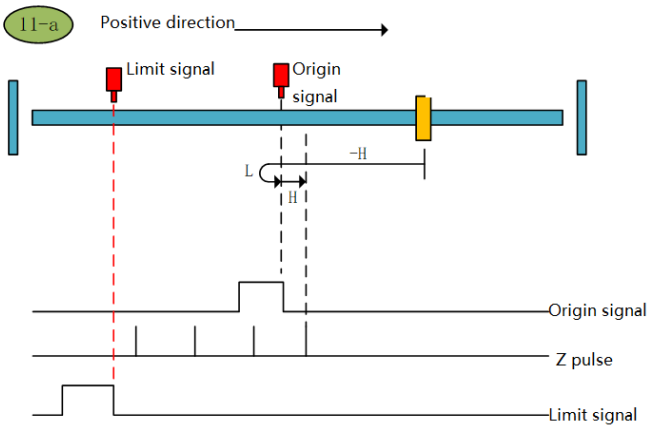
Zero return method of origin 0 -c

Zero return mode of origin (6098 00h = 11)

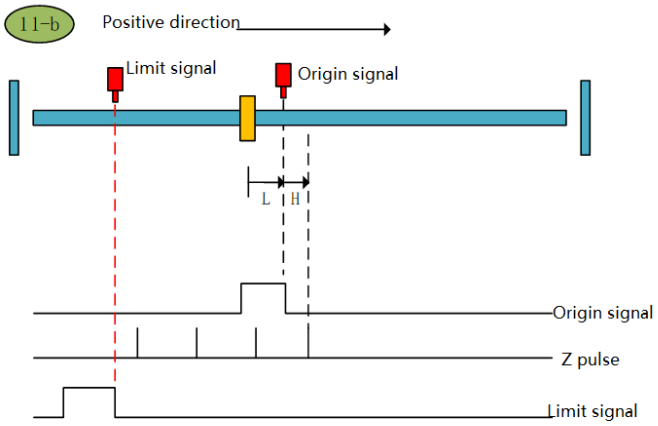
a. Origin return to zero → Origin signal OFF → Reverse high speed to find the origin rising edge → Deceleration to 0 → Forward low speed to find the origin falling edge → Forward to find Z pulse

b. Origin return to zero → Origin signal ON → Forward low speed to find the origin falling edge → Forward to find Z pulse

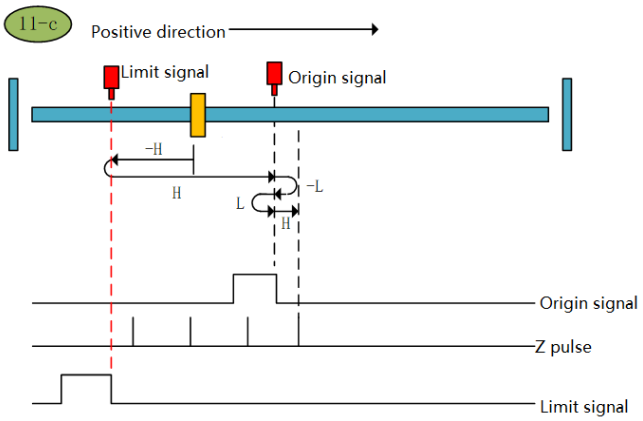
c. Origin return to zero → Origin signal OFF → Reverse high speed to find the origin rising edge → Hit the negative limit → Forward high speed to find the origin signal falling edge → Deceleration to 0 → Reverse low speed to find the origin rising edge → Forward to find Z pulse



Origin return mode 11-a



Origin return mode 11-b



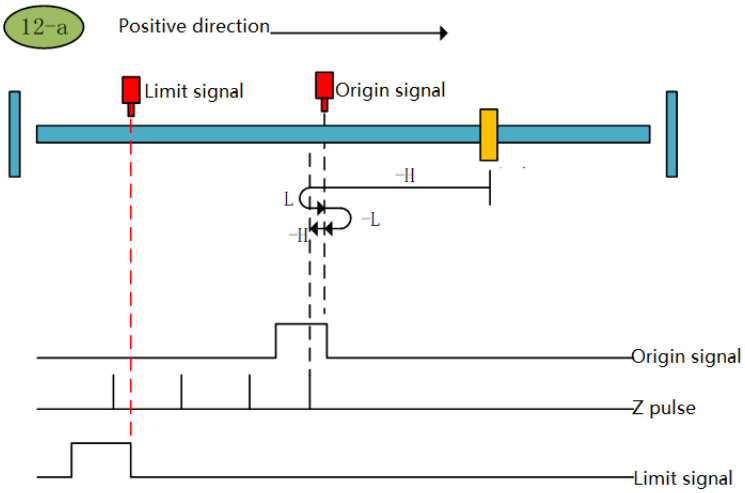
Origin return mode 11-c

Zero return mode of origin 12 (6098 00h = 12)

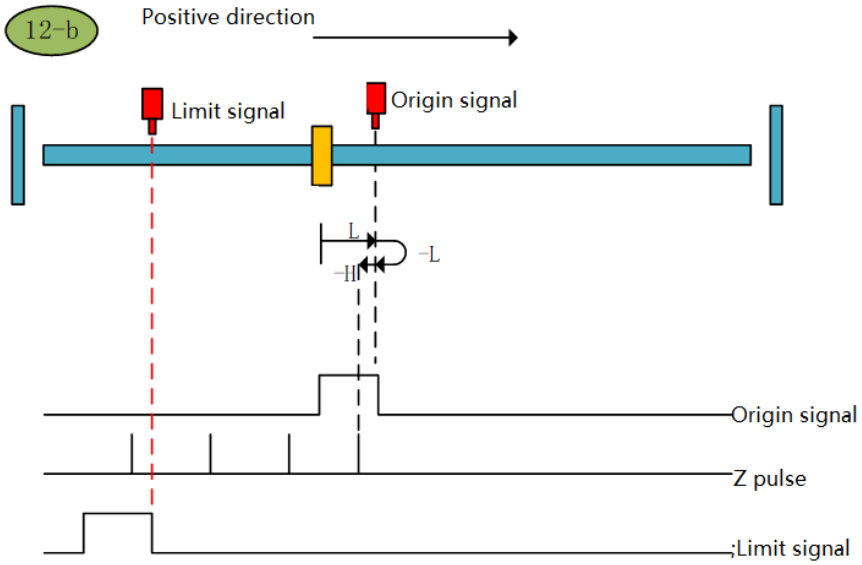
a. Origin return start → Origin signal OFF → Reverse high speed to find the origin rising edge → Deceleration to 0 → Forward low speed to find the origin falling edge → Reverse low speed to find the origin rising edge → Reverse to find the Z pulse

b. Origin return start → Origin signal ON → Forward low speed find origin falling edge → Reverse low speed find origin rising edge → Reverse find Z pulse

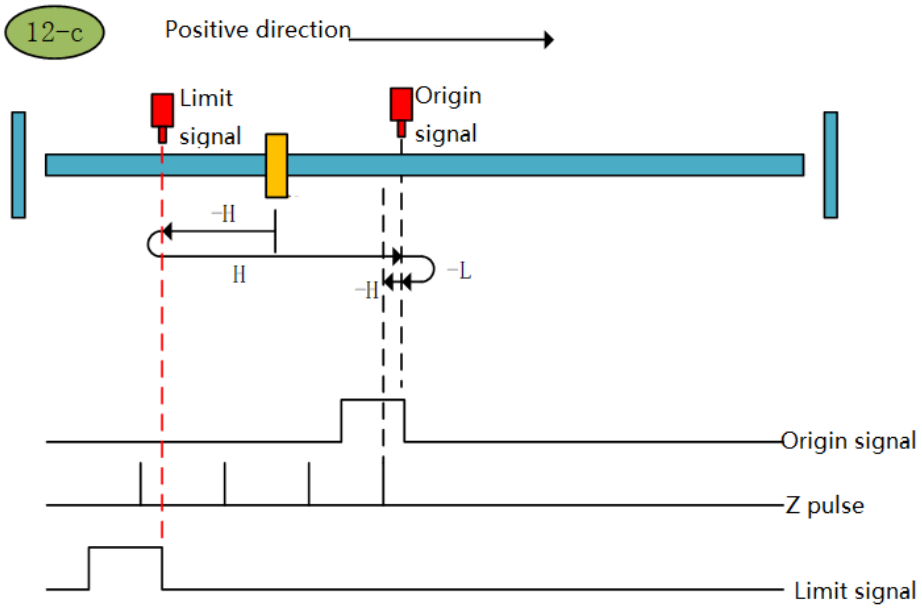
c. Origin return start → Origin signal OFF → Reverse high speed to find the origin rising edge → Hit the negative limit → Forward high speed to find the origin signal falling edge → Decelerate to 0 → Reverse low speed find the origin rising edge → Reverse find Z pulse



Zero return method 12-a



Zero return method 12-b



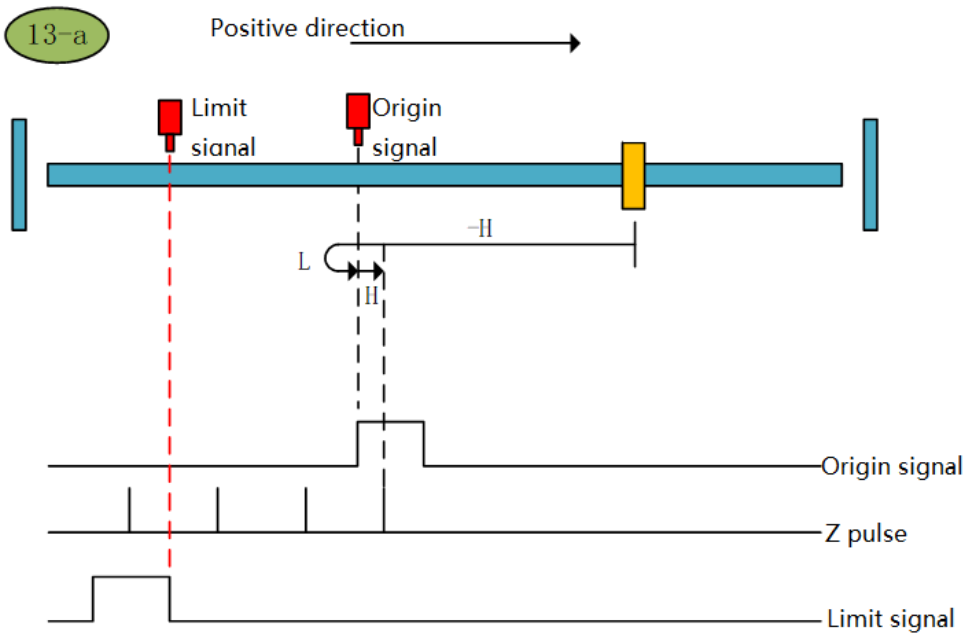
Zero return method 12-c

Zero return method of origin (6098 00h = 13)

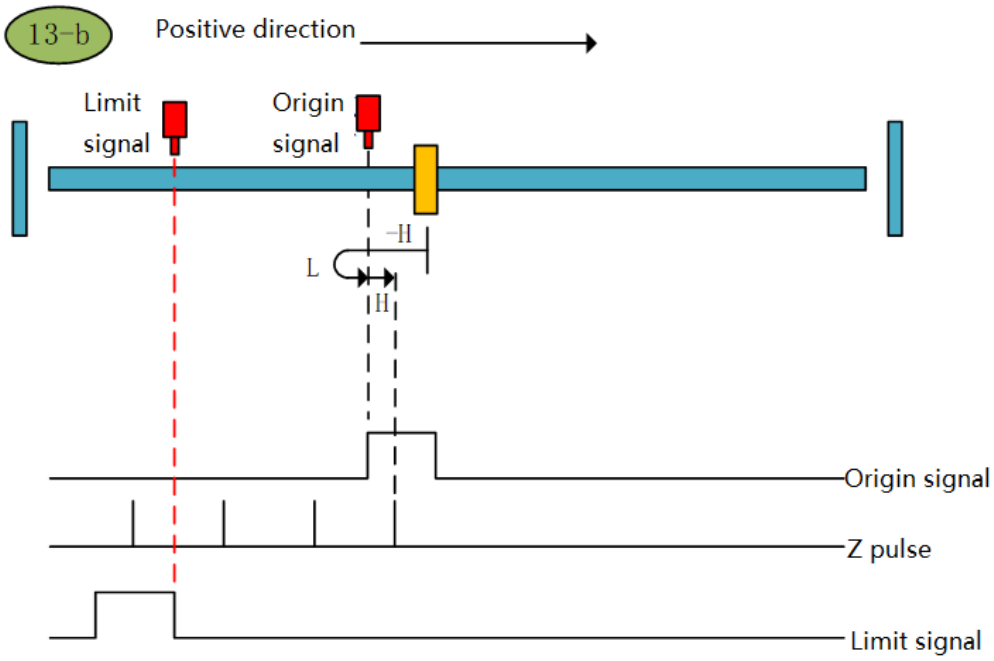
a. Origin return to zero → Origin signal OFF → Reverse high speed to find the origin falling edge → Deceleration to 0 → Forward low speed to find the origin rising edge → Forward to find Z pulse

b. Origin return to zero → Origin signal ON → Reverse high speed to find the origin falling edge → Deceleration to 0 → Forward low speed to find the origin rising edge → Forward to find Z pulse

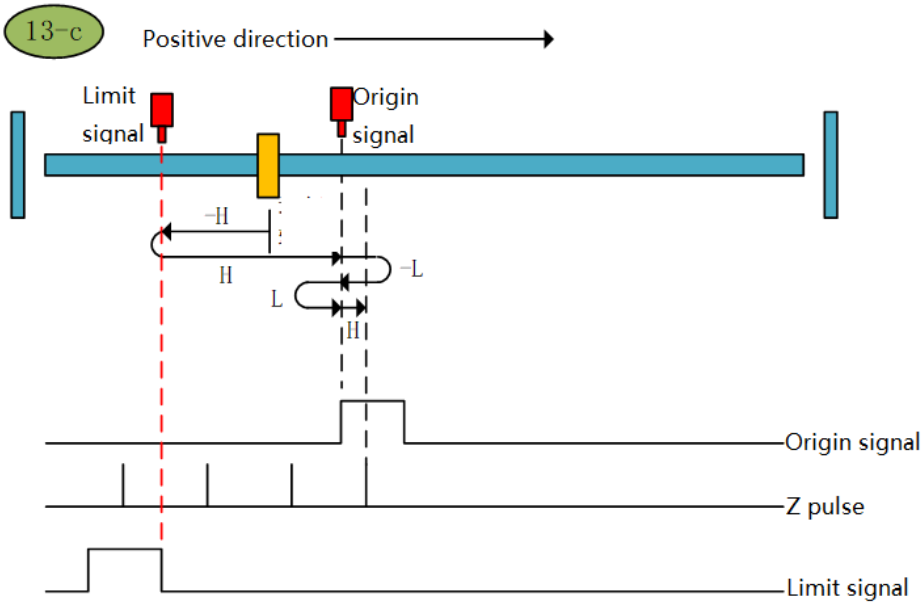
c. Origin return start → Origin signal OFF → Reverse high speed to find the origin falling edge → Hit the negative limit → Forward high speed to find the origin signal rising edge → Decelerate to 0 → Reverse low speed to find the origin signal falling edge → Forward low speed Origin signal rising edge → positive looking Z pulse



Origin return mode 13-a



Origin return mode 13-b



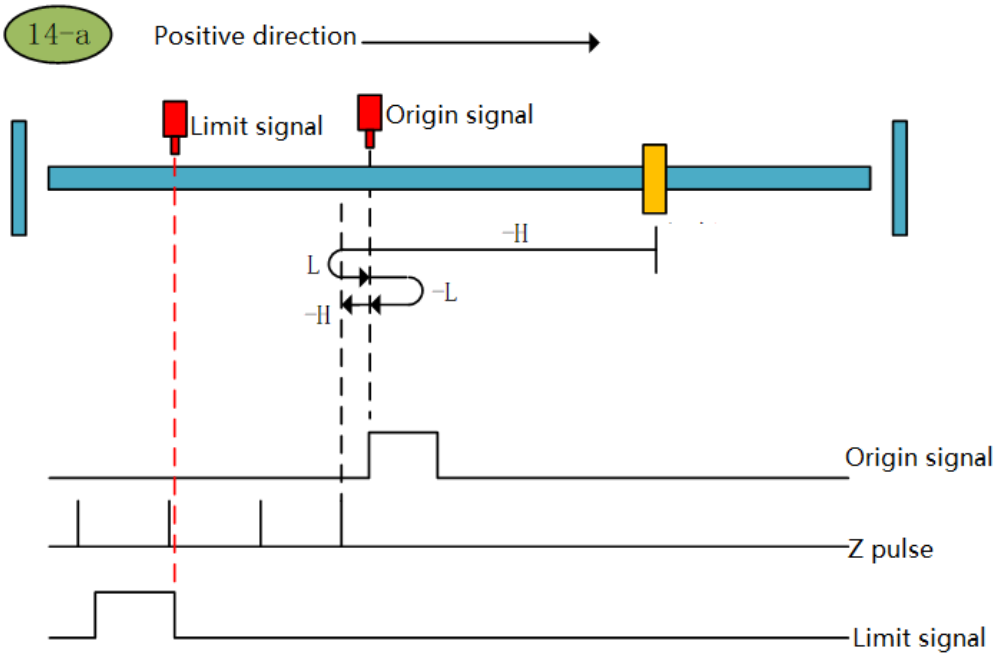
Origin return mode 13-c

Zero return method of origin (6098 00h = 14)

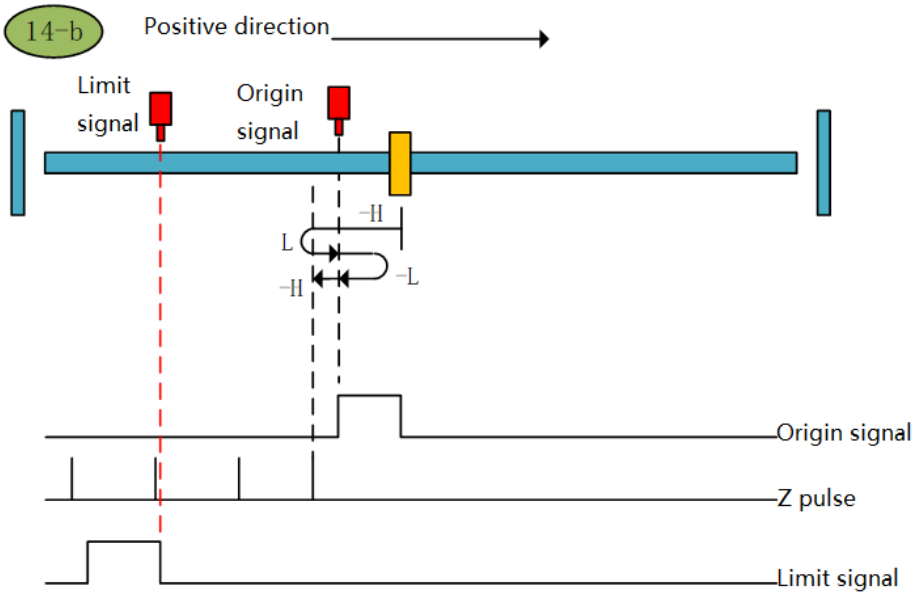
a. Origin return start → Origin signal OFF → Reverse high speed to find the origin falling edge → Deceleration to 0 → Forward low speed to find the origin rising edge → Reverse low speed to find the origin falling edge → Reverse to find the Z pulse

b. Origin return start → Origin signal ON → Reverse high speed to find the origin falling edge → Deceleration to 0 → Forward low speed to find the origin rising edge → Reverse low speed to find the origin falling edge → Reverse to find Z pulse

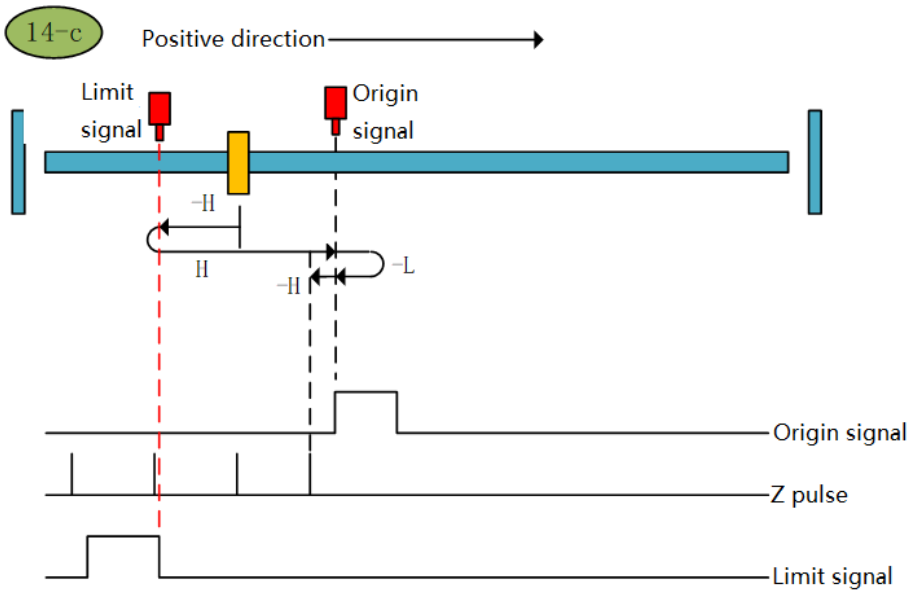
c. Origin return to zero → Origin signal OFF → Reverse high speed to find the origin falling edge → Hit the negative limit → Forward high speed to find the origin signal rising edge → Deceleration to 0 → Reverse low speed to find the origin signal falling edge → Reverse looking Z pulse



Origin return mode 14-a



Origin return mode 14-b



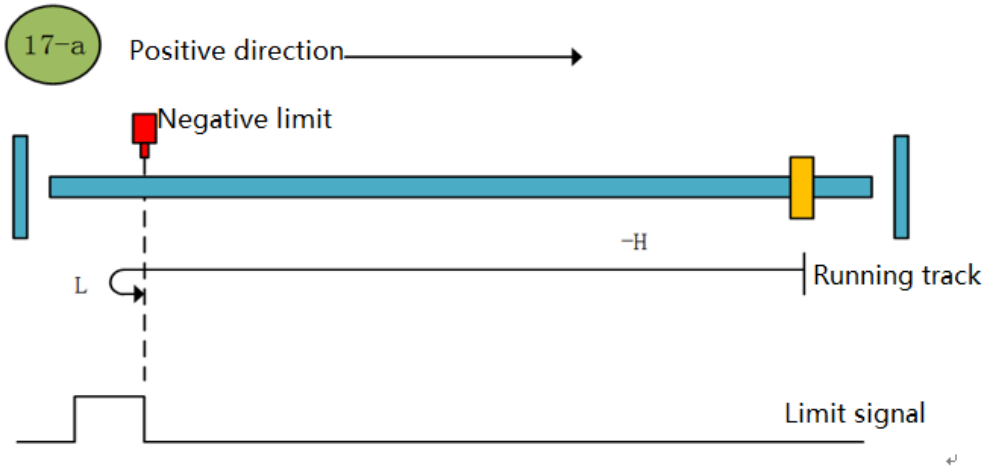
Origin return mode 14-c

Home zero return mode 15 (6098 00h = 15): Reserved.

Home zero return mode 16 (6098 00h = 16): Reserved.

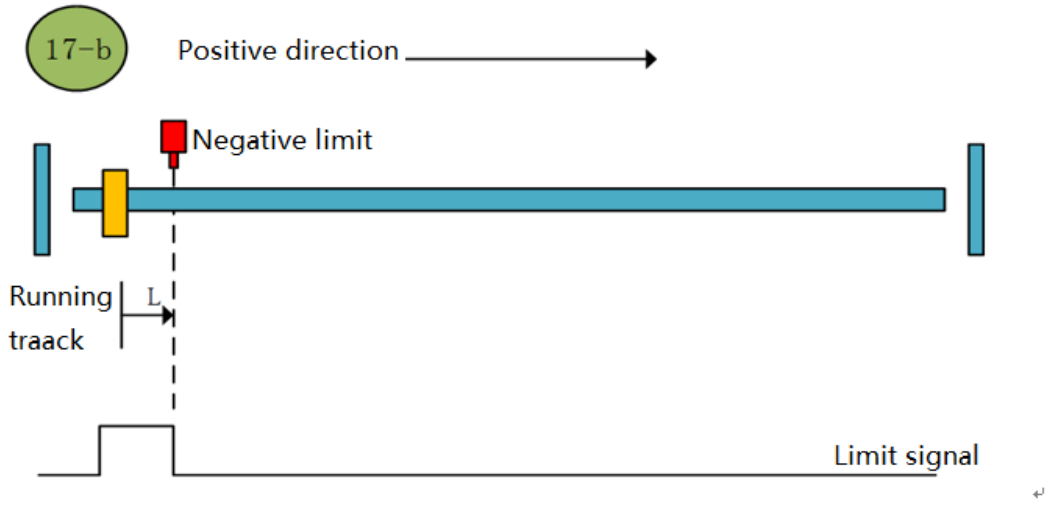
Zero return mode 17 (6098 00h = 17)

a. Start the origin return → reverse high speed to find the negative limit → hit the negative limit rising edge → decelerate to 0 → forward low speed to find the negative limit after the falling edge



Origin return mode 17-a

b. Start origin return → Negative limit is valid → Positive low speed finds negative limit and stops after falling

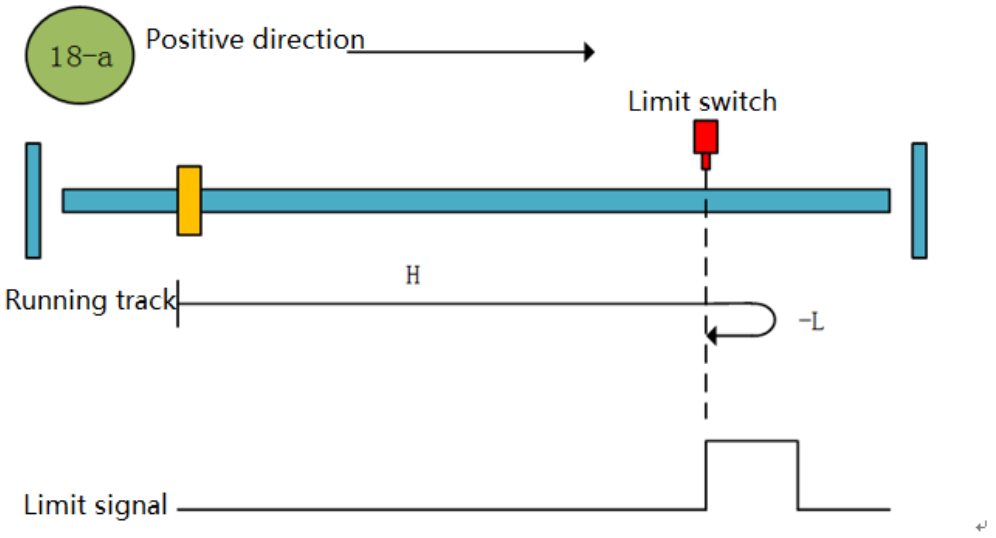


Origin return mode 17-b

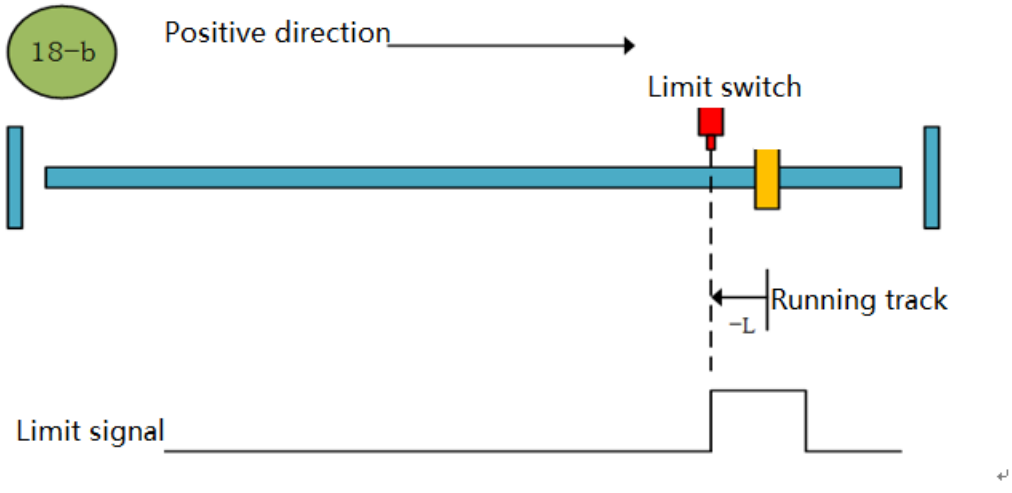
Zero return method of origin (6098 00h = 18)

a. Start the origin return → positive high speed correction limit → hit the positive limit rising edge → decelerate to 0 → reverse low speed to find the positive limit falling edge and stop

b. Start origin return → positive limit valid → reverse low speed correction limit



Zero return method 18-a

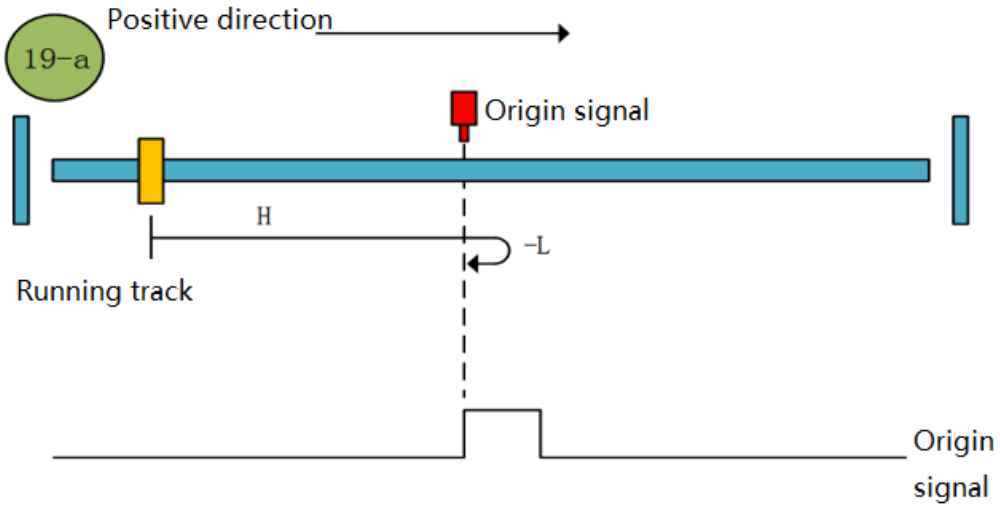


Zero return method 18-b

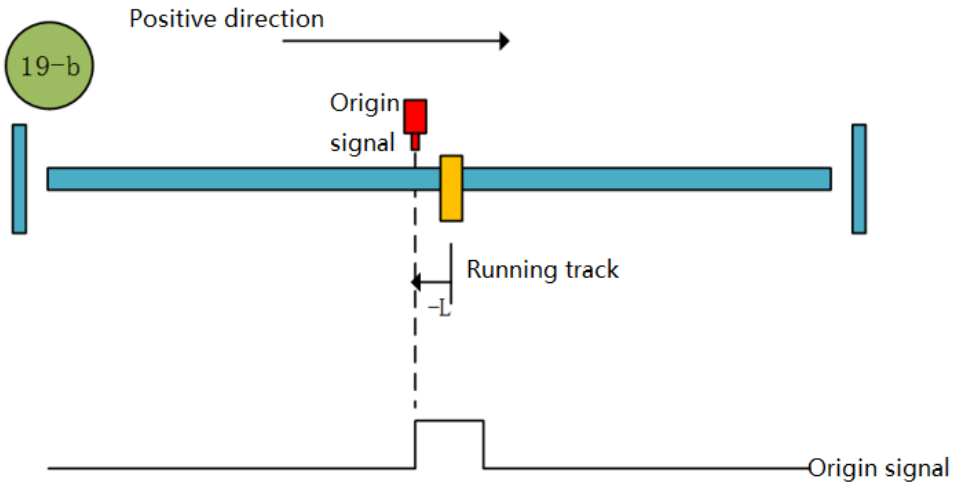
Zero return mode of origin 19 (6098 00h = 19)

a. Start the origin return → positive high speed correction limit → hit the positive limit rising edge → decelerate to 0 → reverse low speed to find the positive limit falling edge and stop

b. Start origin return → positive limit valid → reverse low speed correction limit



Zero return method 19-a

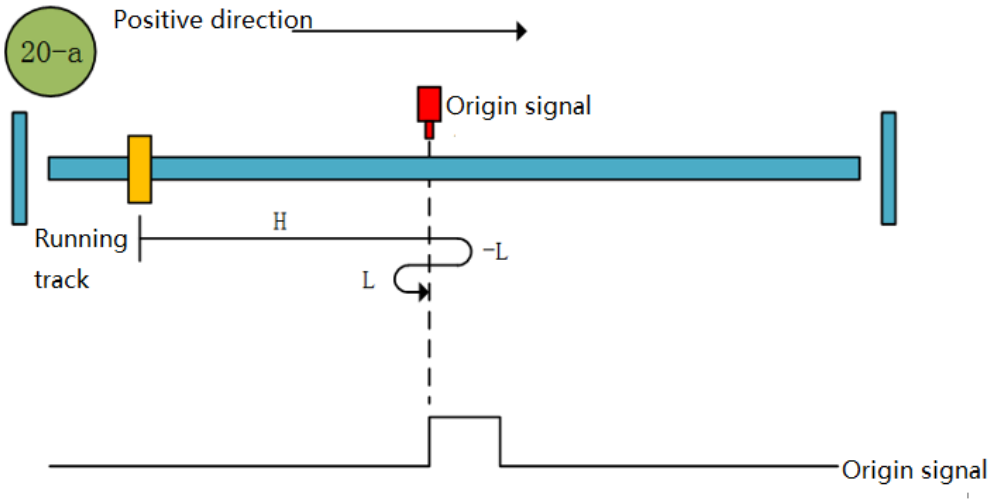


Zero return method 19-b

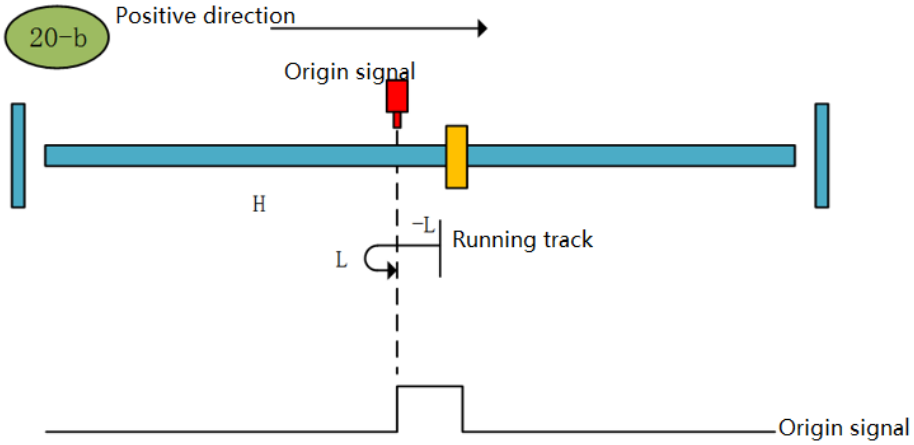
Zero return method of origin (6098 00h = 20)

a. Start OPR → Origin signal OFF → Forward high speed to find the origin rising edge → Deceleration to 0 → Reverse low speed to find the origin falling edge → Forward low speed to find the origin rising edge and stop

b. Start origin return → origin signal ON → reverse low speed to find the origin falling edge → positive low speed to find the original point rising edge and stop



Zero return method 20-a

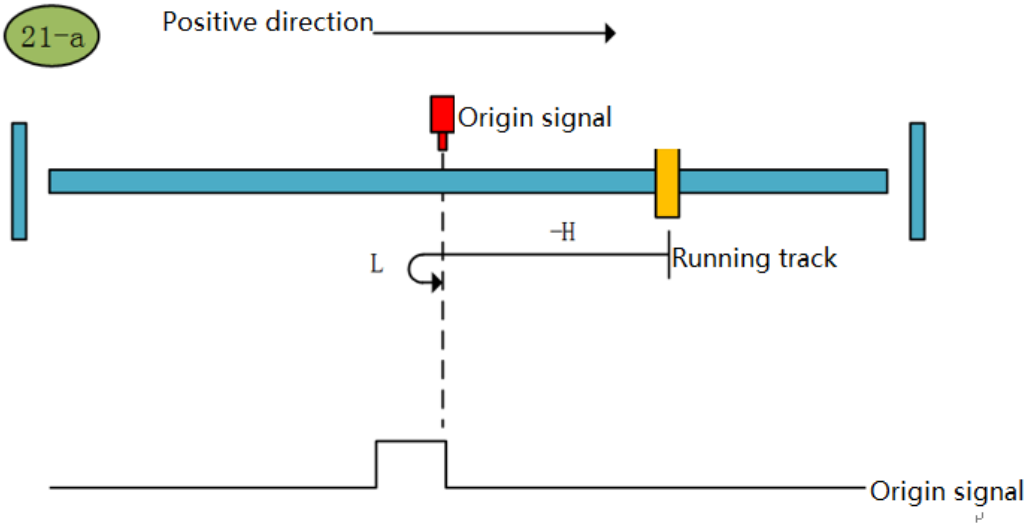


Origin return mode 20-b

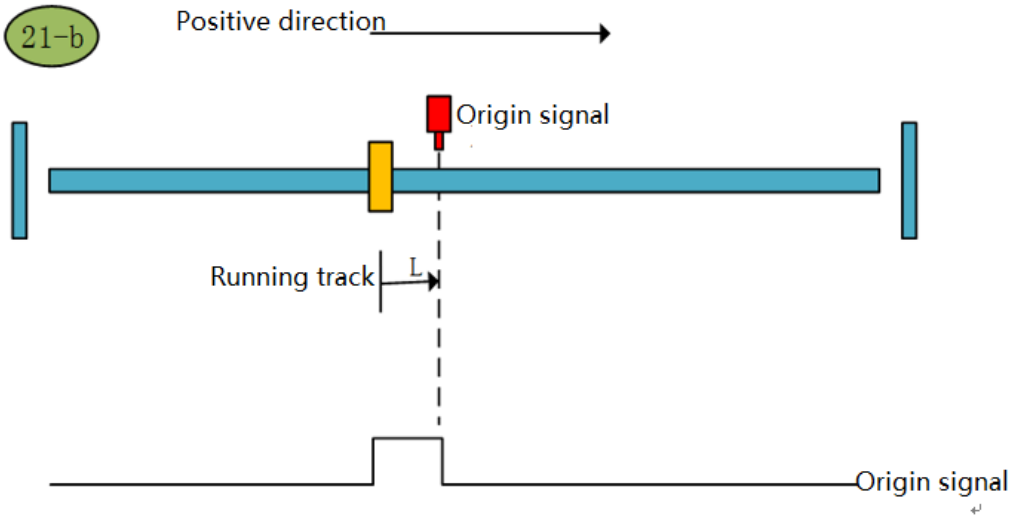
Zero return method of origin (6098 00h = 21)

a. Start OPR → Origin signal OFF → Reverse high speed to find the origin rising edge → Deceleration to 0 → Forward low speed to find the origin falling edge and stop

b. Start OPR → Origin signal ON → Stop at low speed and find the falling edge of the origin



Origin return mode 21-a

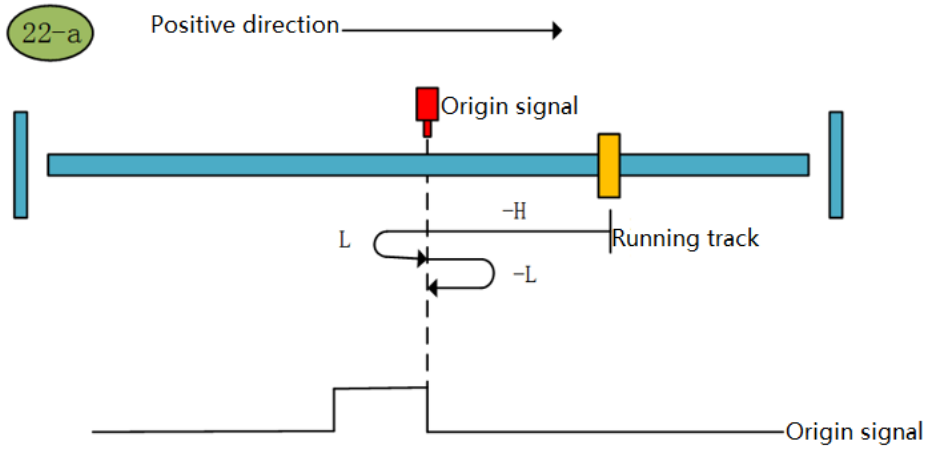


Origin return mode 21-b

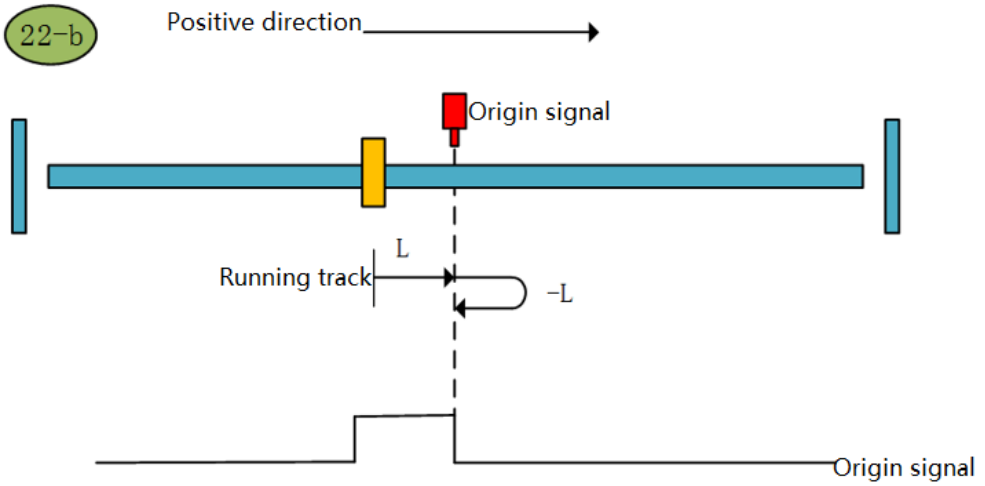
Origin return mode 22 (6098 00h = 22)

a Start OPR → Origin signal OFF → Reverse high speed to find the origin rising edge → Deceleration to 0 → Forward low speed to find the origin falling edge → Reverse low speed to find the origin rising edge and stop

b Start the origin return → the origin signal ON → the forward low speed finds the origin falling edge → the reverse low speed finds the origin rising edge and stops



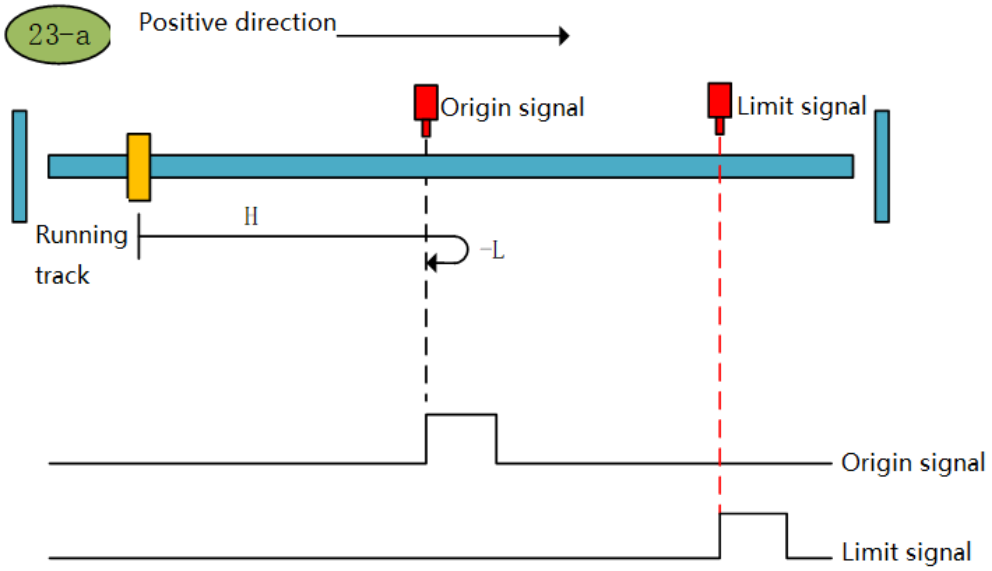
Origin return mode 22-a



Origin return mode 22-b

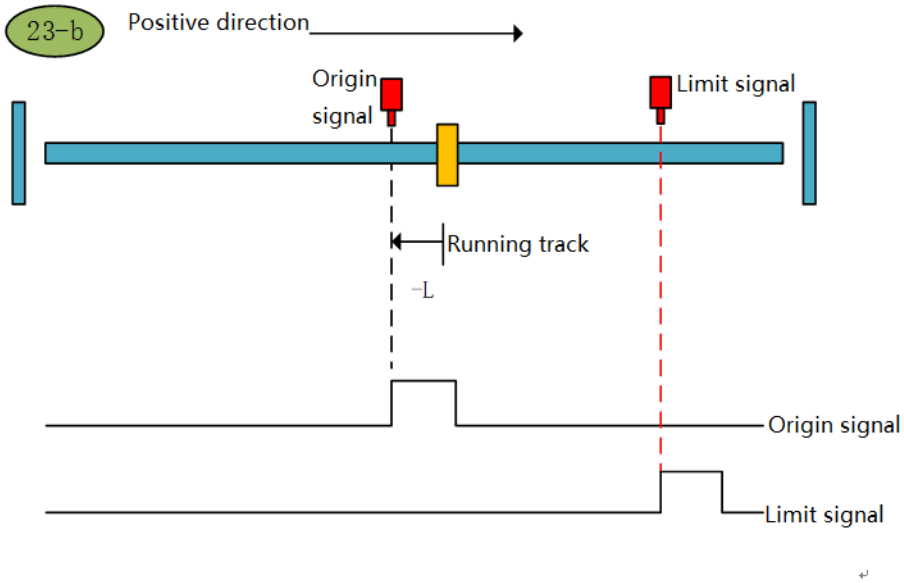
Zero return mode 23 (6098 00h = 23)

a Start OPR → Origin signal OFF → Forward high speed to find the origin rising edge → Deceleration to 0 → Reverse low speed to find the origin falling edge and stop



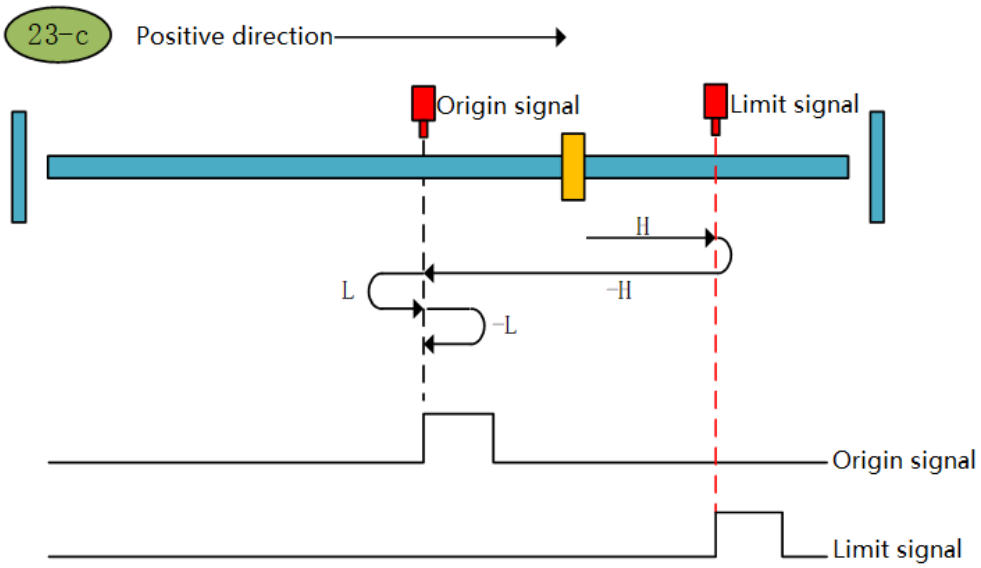
Origin return mode 23-a

b Start the origin return → origin signal ON → reverse low speed to find the original point after the falling edge



Origin return mode 23-b

c Start origin return → origin OFF → forward high speed to find the origin rising edge → hit the positive limit → reverse high speed to find the origin falling edge → decelerate to 0 → forward low speed to find the origin rising edge → reverse low speed to find the origin after the falling edge Downtime



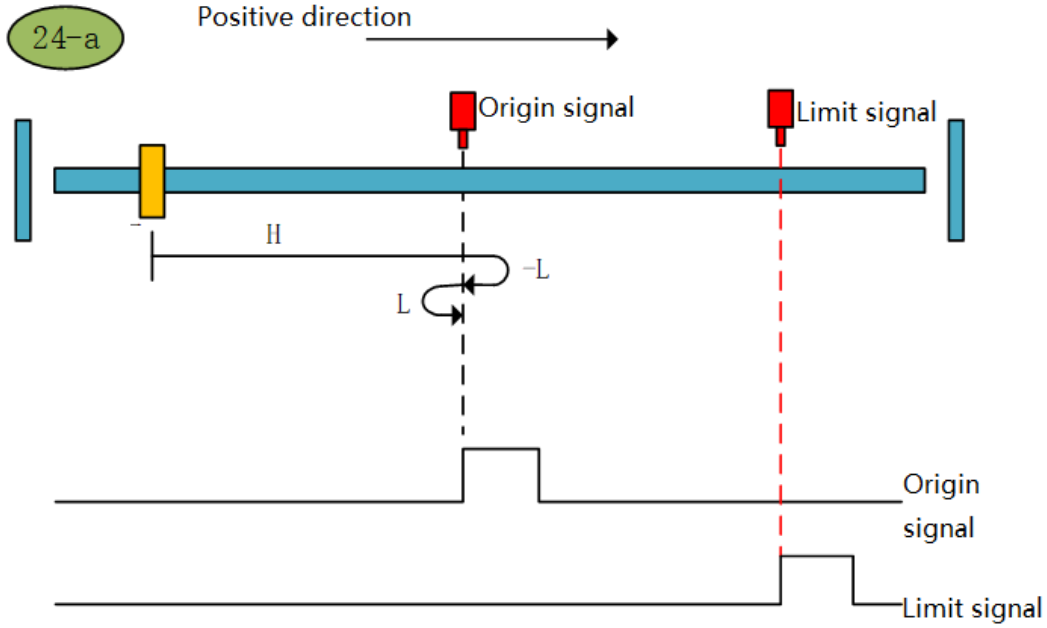
Origin return mode 23-c

Origin return mode 24 (6098 00h = 24)

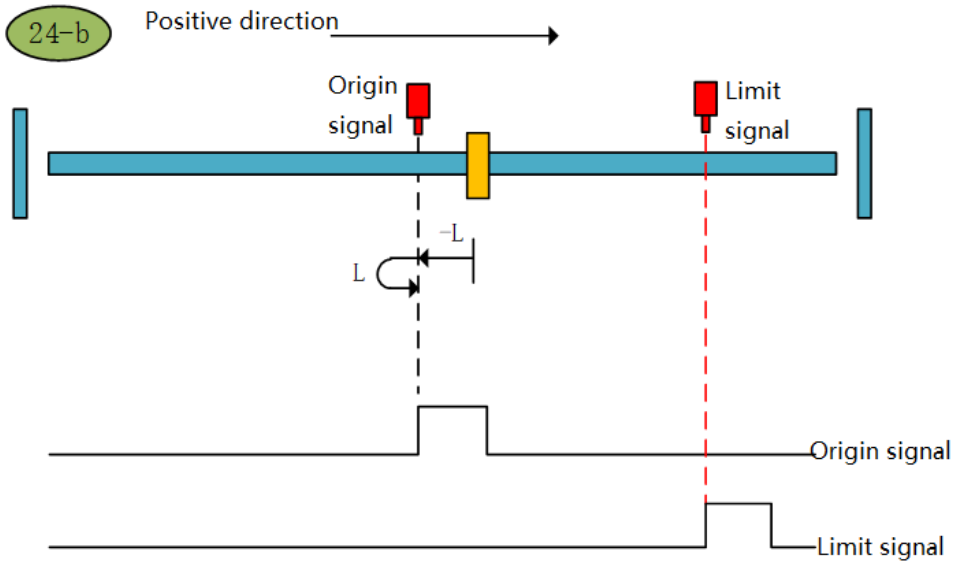
a. Start OPR → Origin signal OFF → Forward high speed to find the origin rising edge → Deceleration to 0 → Reverse low speed to find the origin falling edge → Forward low speed to find the origin rising edge and stop

b. Start origin return → origin signal ON → reverse low speed to find the origin falling edge → positive low speed to find the original point rising edge and stop

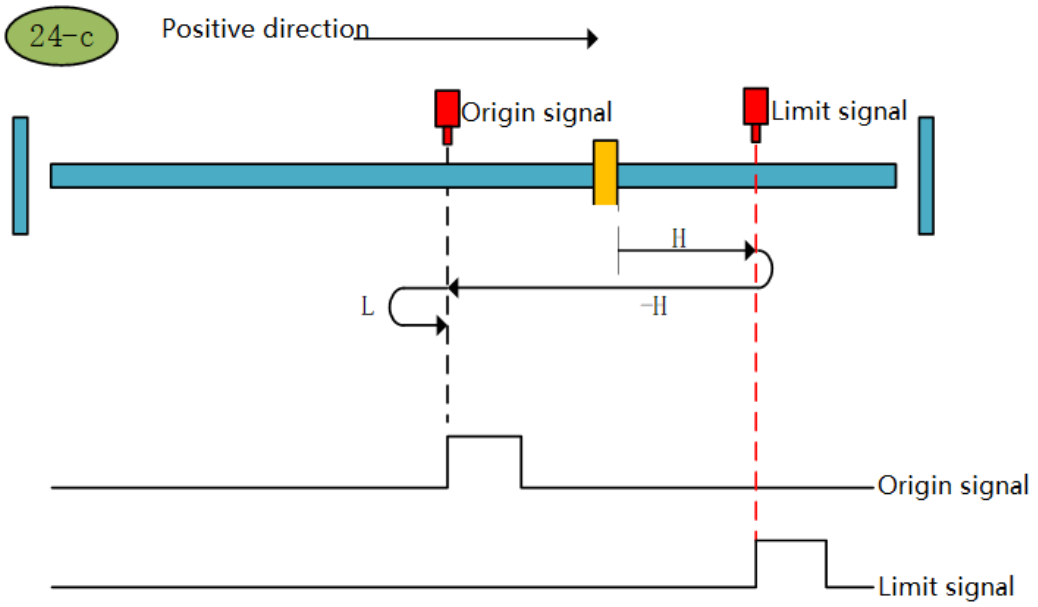
c. Start origin return → origin OFF → forward high speed to find the original rising edge → hit the positive limit → reverse high speed to find the origin falling edge → decelerate to 0 → forward low speed to find the original rising edge and stop



Origin return mode 24-a



Origin return mode 24-b



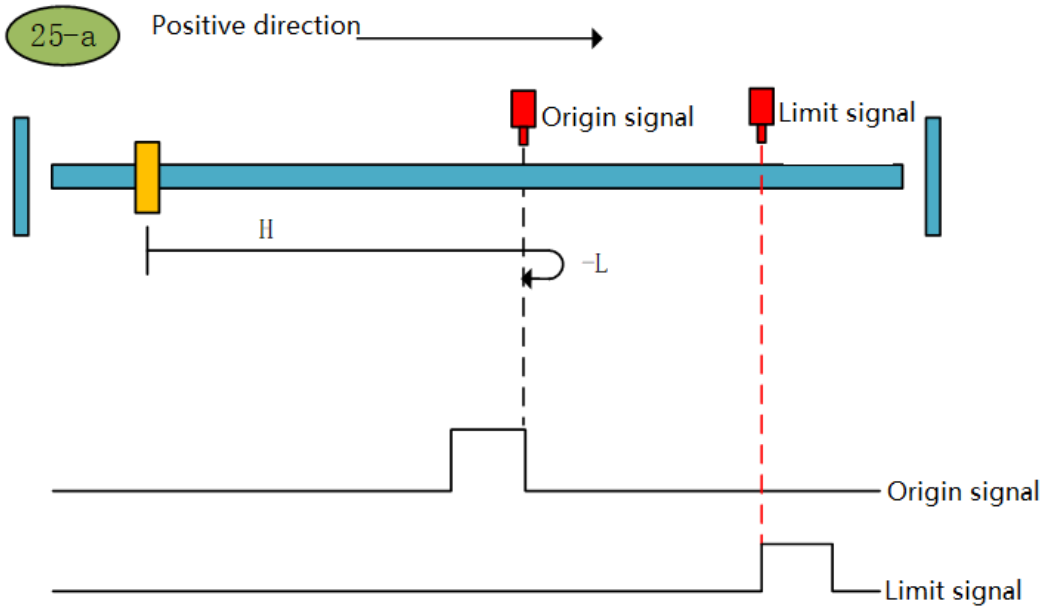
Origin return mode 24-c

Zero return mode of origin 25 (6098 00h = 25)

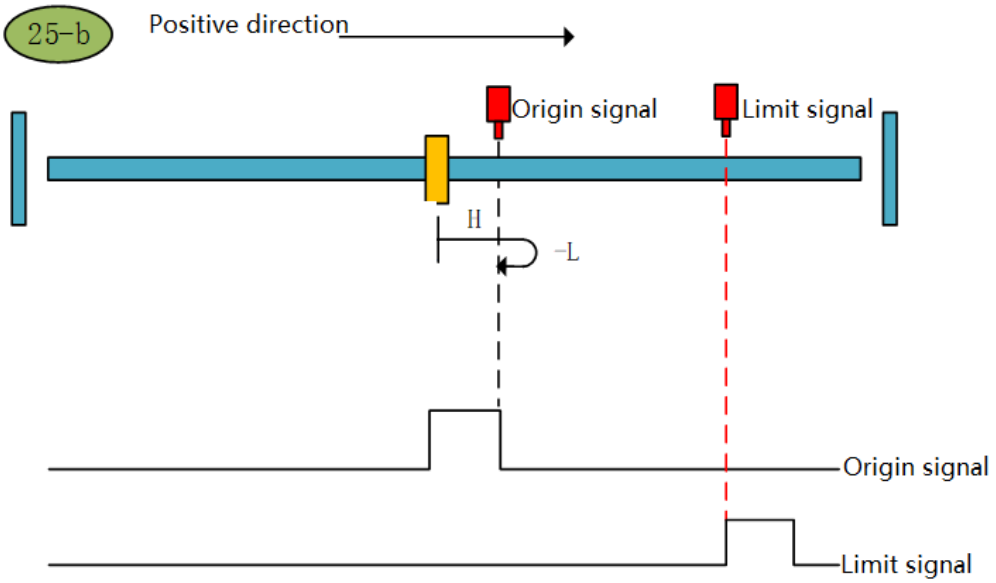
a. Start OPR → Origin signal OFF → Forward high speed to find the origin falling edge → Deceleration to 0 → Reverse low speed to find the original rising edge and stop

b. Start OPR → Origin signal ON → Forward high speed to find the origin falling edge → Deceleration to 0 → Reverse low speed to find the original rising edge and stop

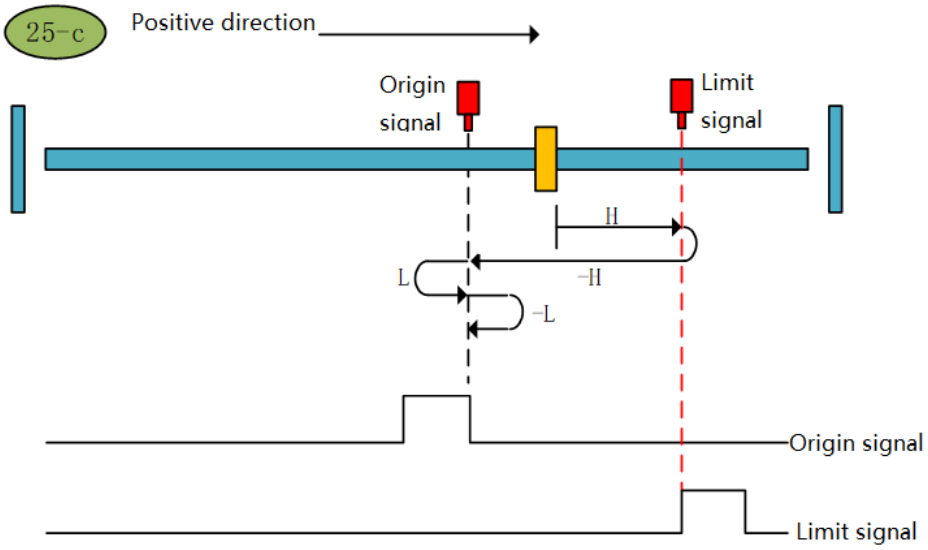
c. Start origin return → origin OFF → forward high speed to find the origin falling edge → hit the positive limit → reverse high speed to find the origin rising edge → decelerate to 0 → forward low speed to find the origin falling edge → reverse low speed to find the origin rising edge and stop



Origin return to zero 25-a



Origin return mode 25-b



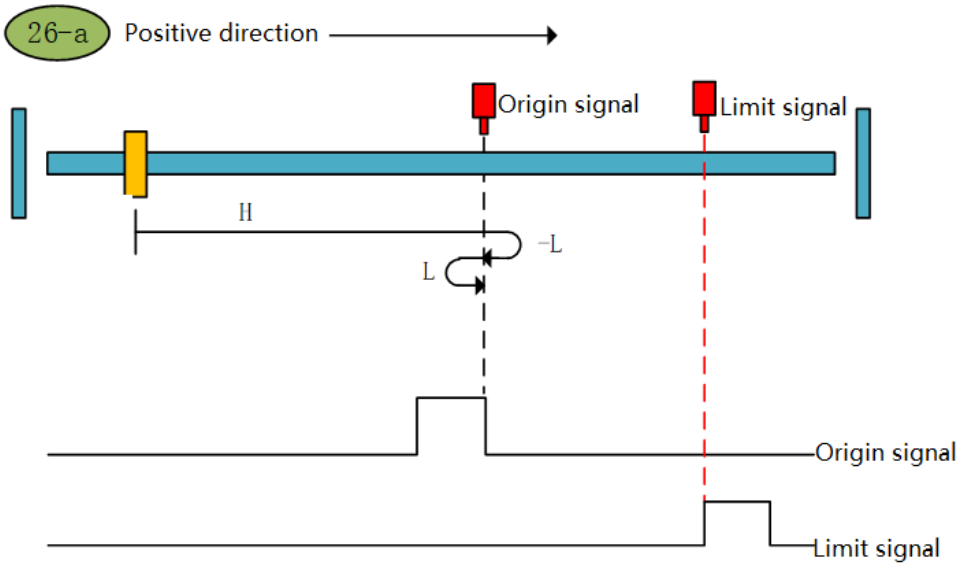
Origin return mode 25-c

Origin return mode 26 (6098 00h = 26)

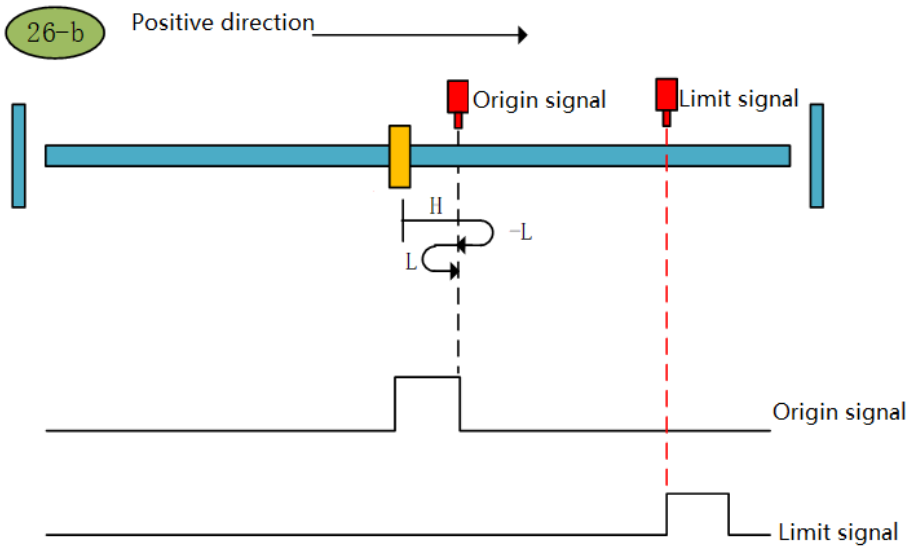
a. Start OPR → Origin signal OFF → Forward high speed to find the origin falling edge → Deceleration to 0 → Reverse low speed to find the origin rising edge → Forward low speed to find the origin falling edge and stop

b. Start OPR → Origin signal ON → Forward high speed to find the origin falling edge → Deceleration to 0 → Reverse low speed to find the origin rising edge → Forward low speed to find the origin falling edge and stop

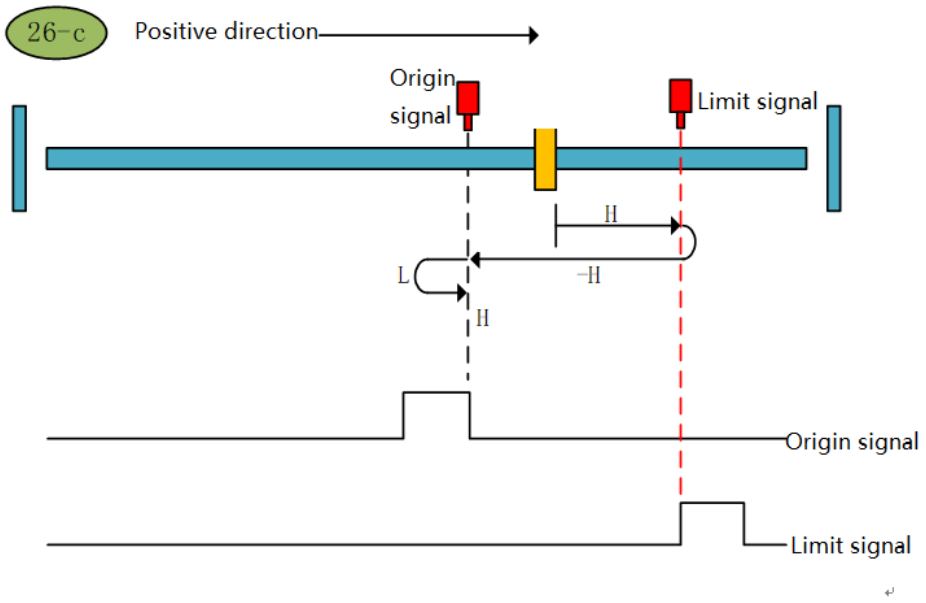
c. Start origin return → origin OFF → forward high speed to find the origin falling edge → hit the positive limit → reverse high speed to find the origin rising edge → decelerate to 0 → forward low speed to find the origin falling edge and stop



Origin return mode 26-a



Origin return mode 26-b



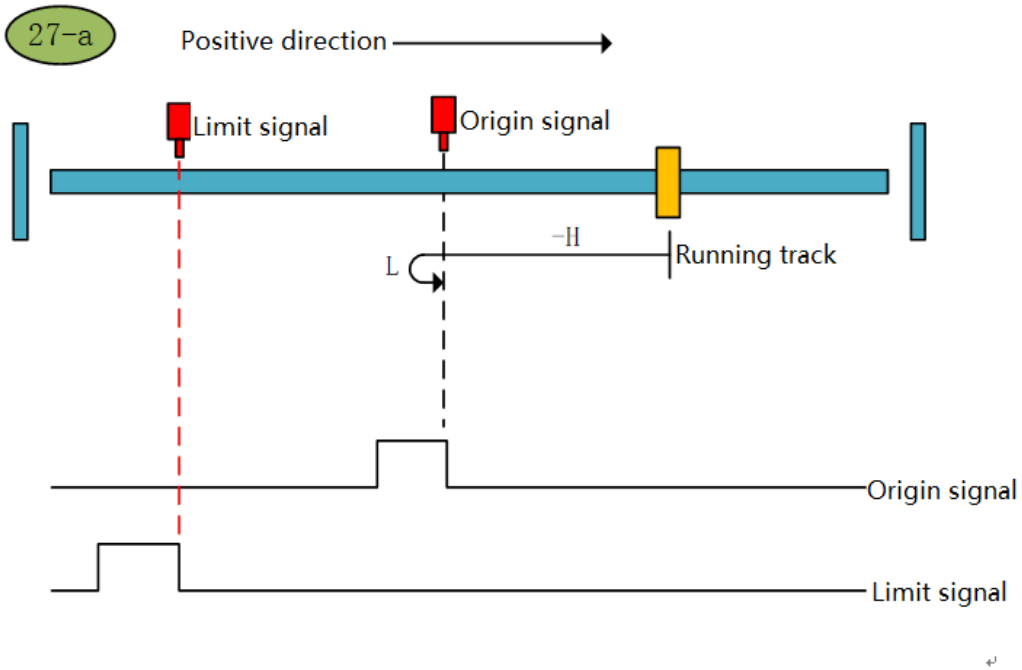
Origin return mode 26-c

Zero return mode 27 (6098 00h = 27)

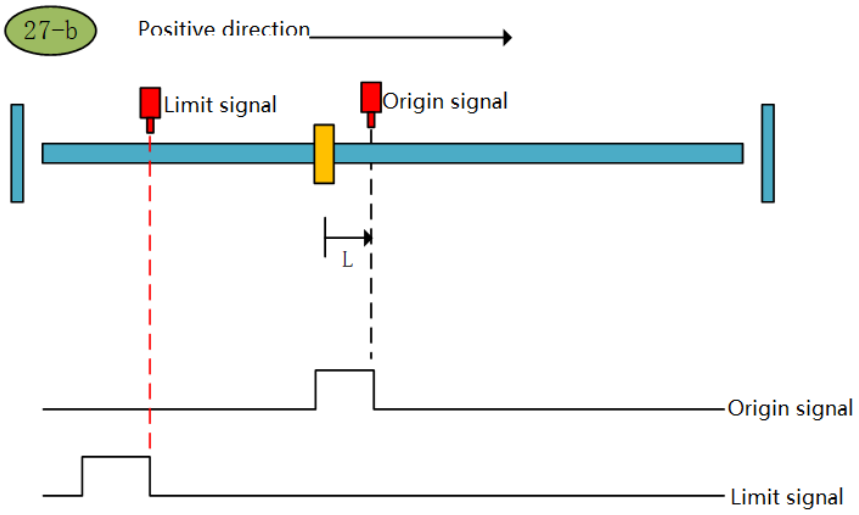
a. Origin return start → Origin signal OFF → Reverse high speed to find the origin rising edge → Deceleration to 0 → Forward low speed to find the origin falling edge and stop

b. Origin return start → origin signal ON → forward low speed to find the origin after the falling edge

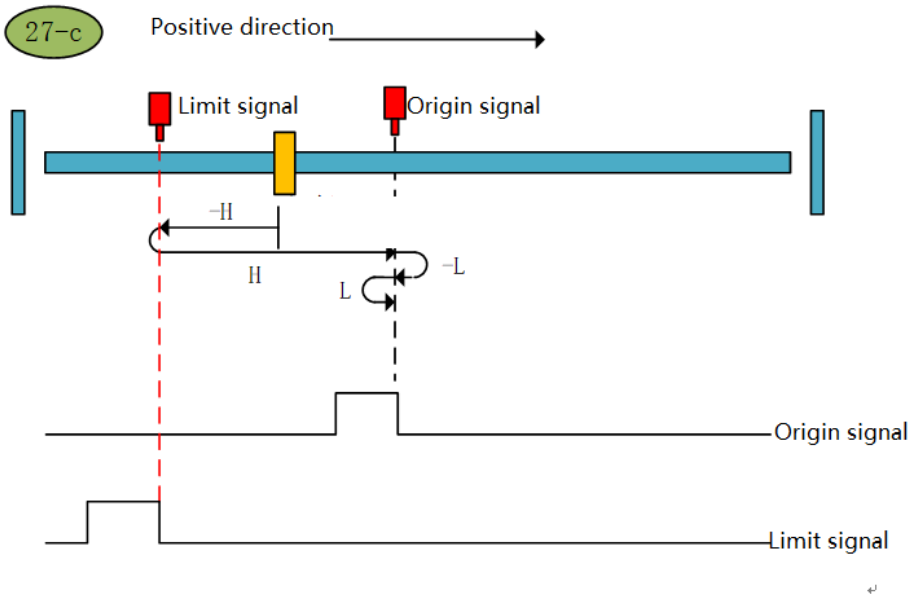
c. Origin return start → Origin signal OFF → Reverse high speed to find the origin rising edge → Hit the negative limit → Forward high speed to find the origin signal falling edge → Deceleration to 0 → Reverse low speed to find the origin rising edge and stop



Origin return mode 27-a



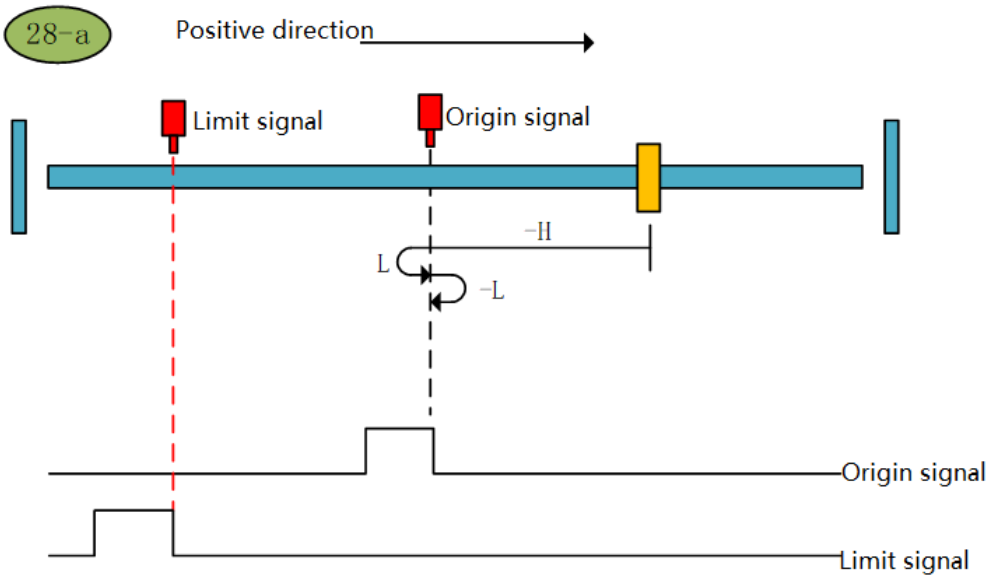
Origin return mode 27-b



Origin return mode 27-c

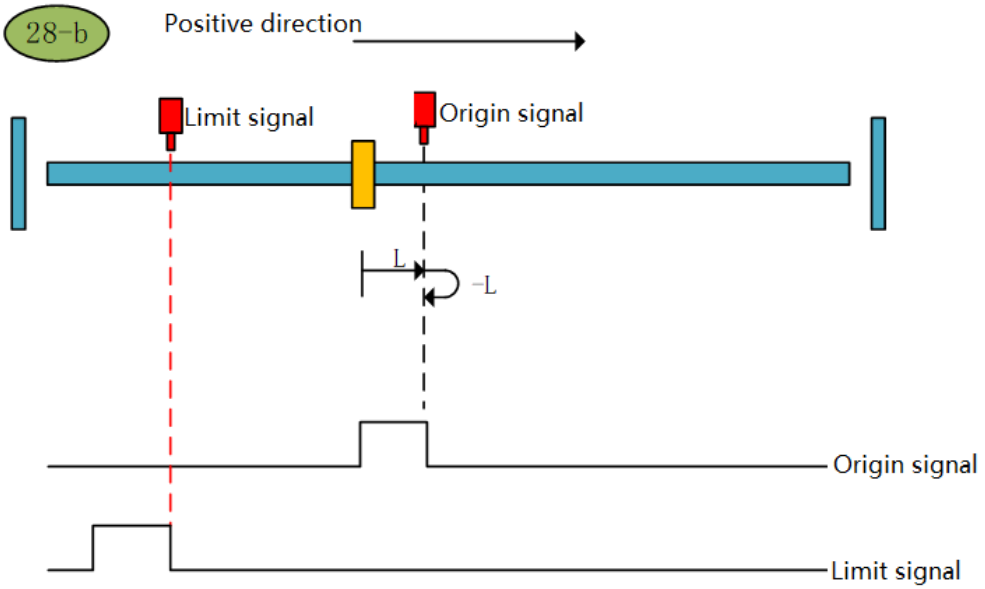
Zero return mode 28 (6098 00h = 28)

a. Origin return start → Origin signal OFF → Reverse high speed to find the origin rising edge → Deceleration to 0 → Forward low speed to find the origin falling edge → Reverse low speed to find the origin rising edge and stop



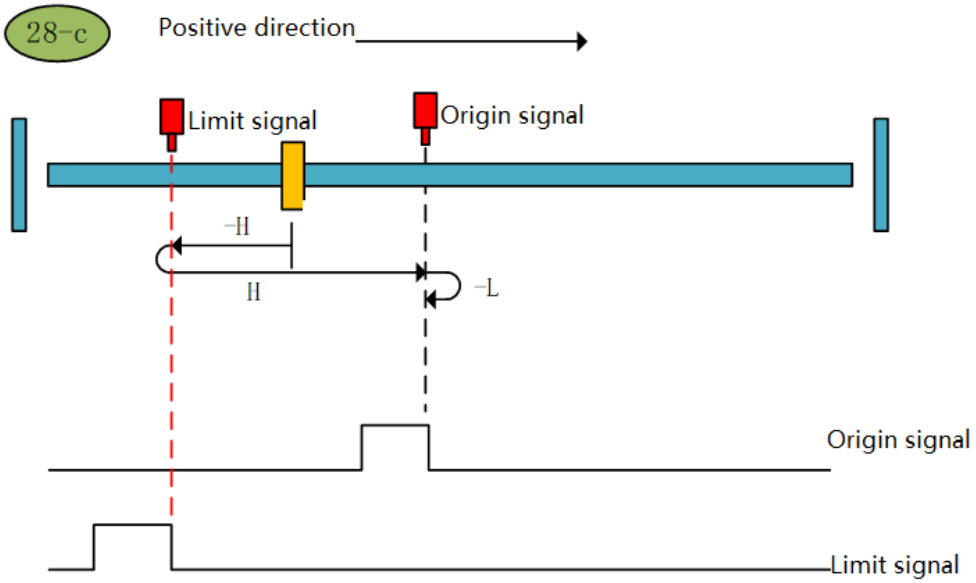
Origin return mode 28-a

b. Origin return start → Origin signal ON → Forward low speed to find the origin falling edge → Reverse low speed to find the origin rising edge and stop



Origin return mode 28-b

c. Origin return start → Origin signal OFF → Reverse high speed to find the origin rising edge → Hit the negative limit → Forward high speed to find the origin signal falling edge → Deceleration to 0 → Reverse low speed to find the origin rising edge and stop



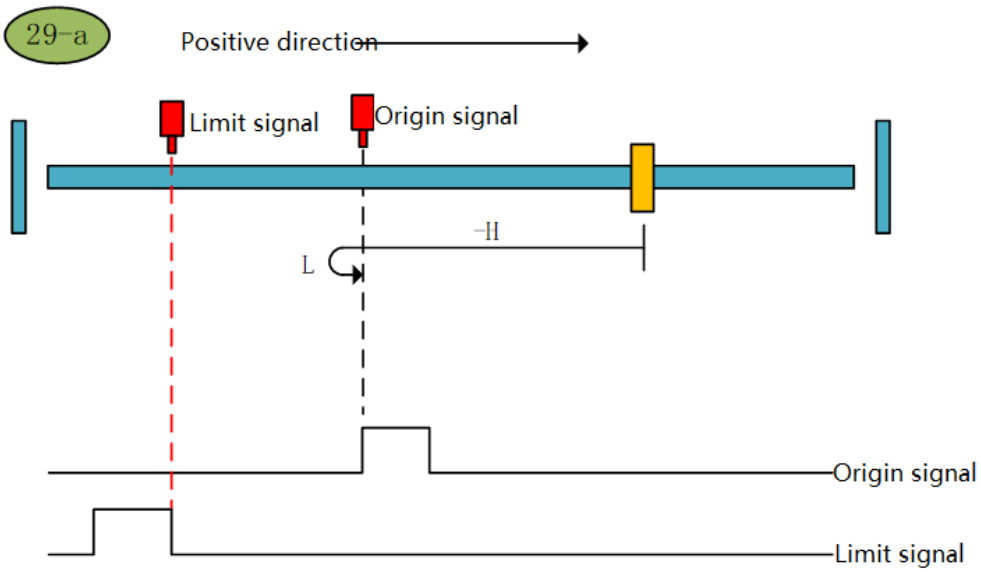
Origin return mode 28-c

Zero return mode 29 (6098 00h = 29)

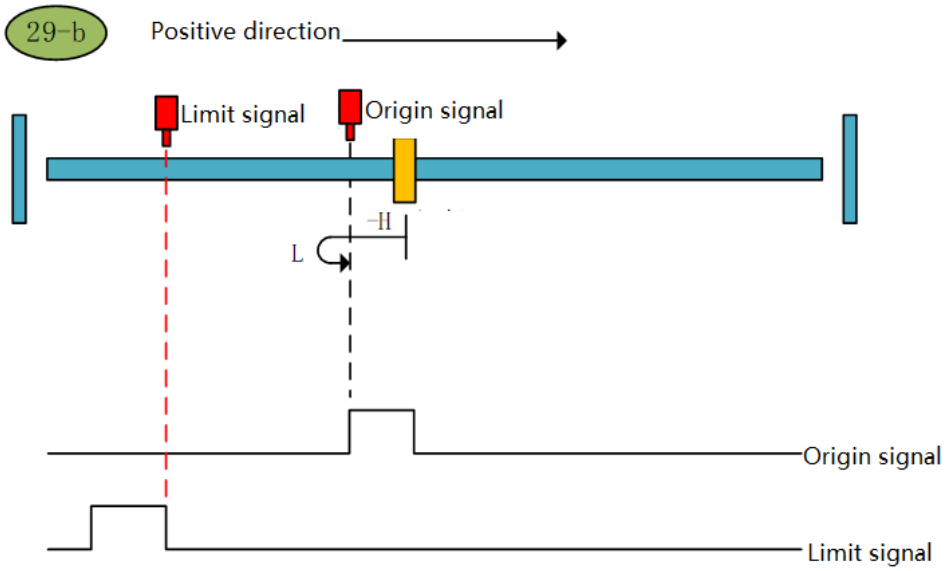
a. Origin return start → Origin signal OFF → Reverse high speed to find the origin falling edge → Deceleration to 0 → Forward low speed to find the origin rising edge and stop

b. Origin return start → Origin signal ON → Reverse high speed to find the origin falling edge → Deceleration to 0 → Forward low speed to find the origin rising edge and stop

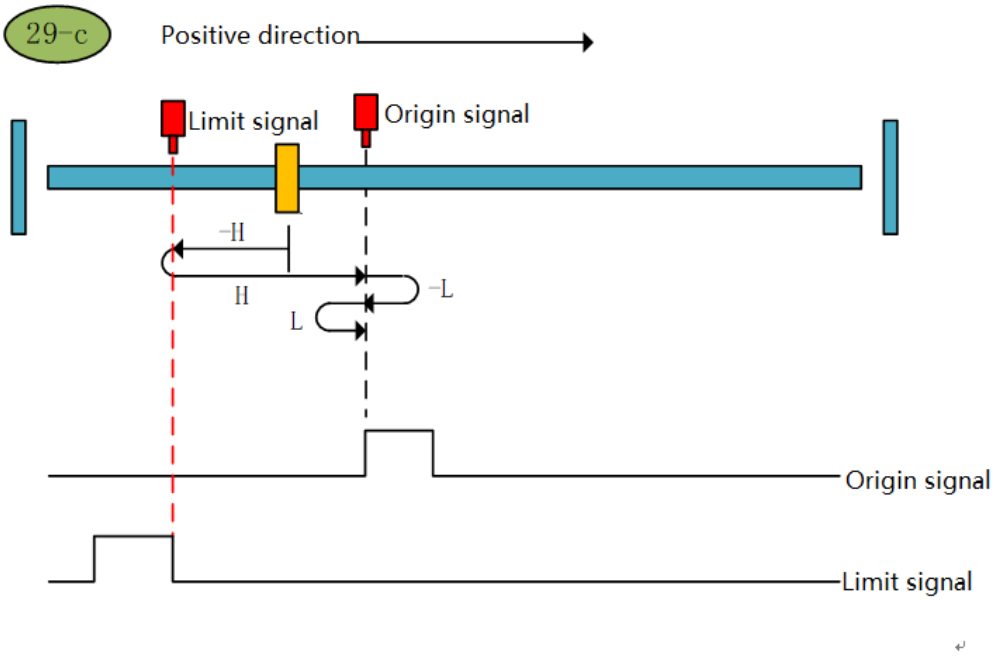
c. Origin return start → Origin signal OFF → Reverse high speed to find the origin falling edge → Hit the negative limit → Forward high speed to find the origin signal rising edge → Decelerate to 0 → Reverse low speed to find the origin signal falling edge → Forward low speed to find the origin signal Stop after rising edge



Zero return method 29-a



Origin return mode 29-b



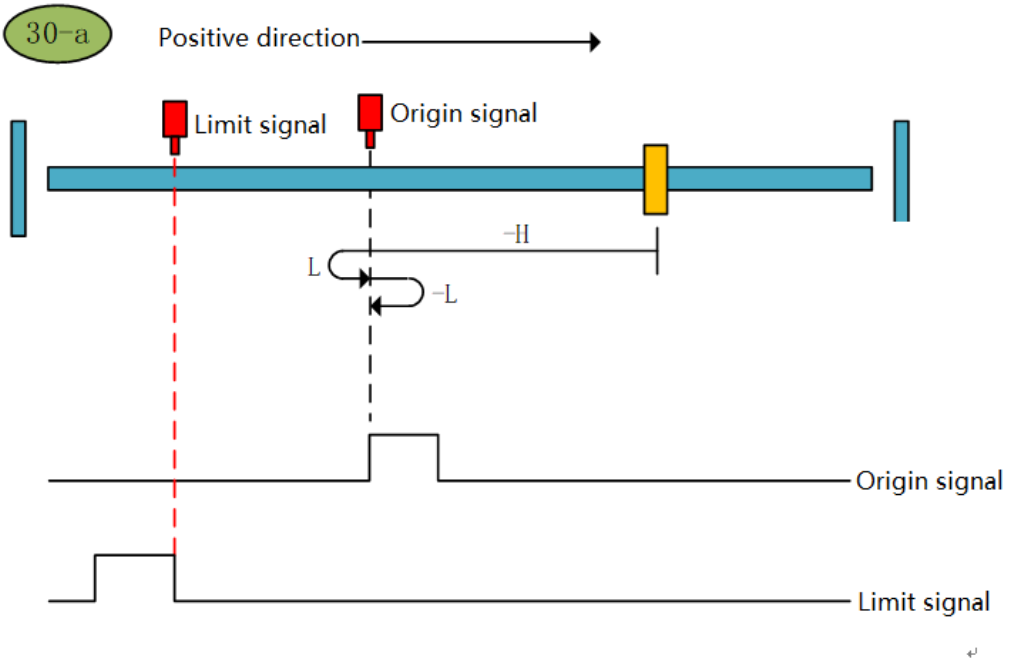
Origin return mode 29-c

Zero return mode 30 (6098 00h = 30)

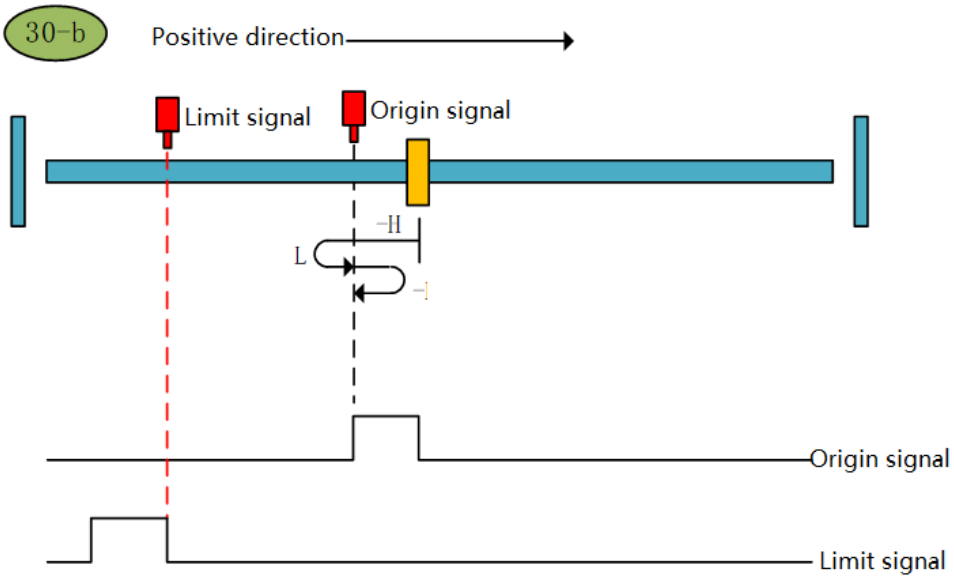
a. Origin return start → Origin signal OFF → Reverse high speed to find the origin falling edge → Deceleration to 0 → Forward low speed to find the origin rising edge → Reverse low speed to find the origin falling edge and stop

b. Origin return start → Origin signal ON → Reverse high speed to find the origin falling edge → Deceleration to 0 → Forward low speed to find the origin rising edge → Reverse low speed to find the origin falling edge and stop

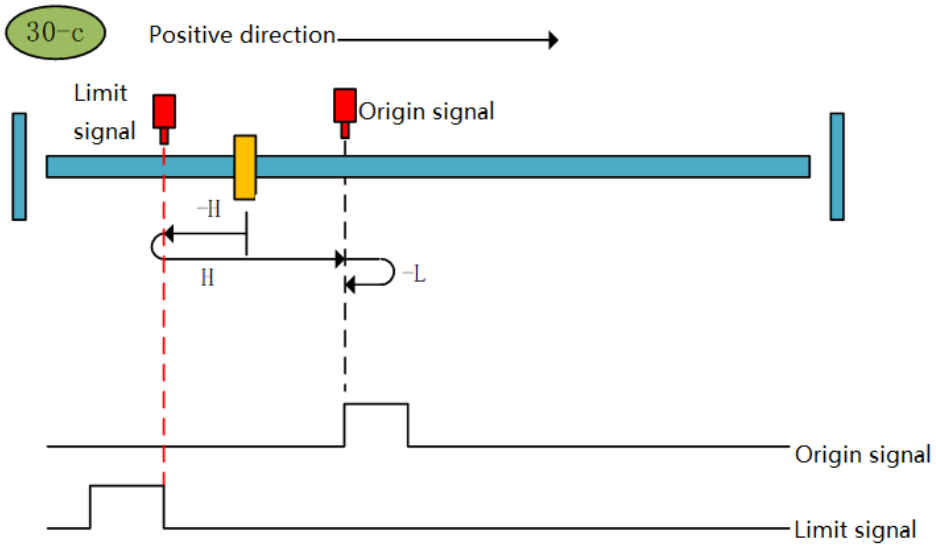
c. Origin return start → Origin signal OFF → Reverse high speed to find the origin falling edge → Hit the negative limit → Forward high speed to find the origin signal rising edge → Decelerate to 0 → Reverse low speed to find the origin signal falling edge and stop



Zero return method 30-a



Zero return method 30-b



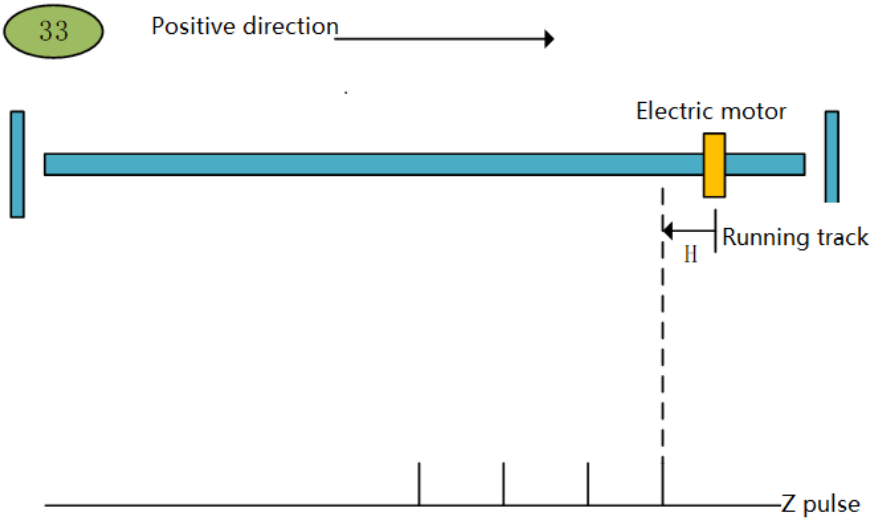
Zero return method 30-c

Zero return mode 31 (6098 00h = 31): Reserved.

Home zero return mode 32 (6098 00h = 32): Reserved.

Zero return mode of origin 33 (6098 00h = 33)

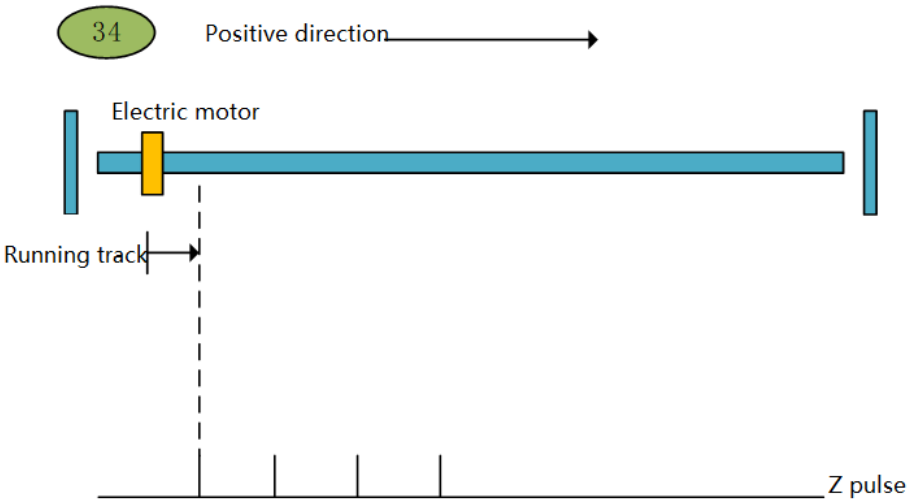
Starting zero return → negative looking for the first Z pulse



Origin return mode 33

Zero return mode of origin 34 (6098 00h = 34)

Origin return to zero → forward to find the first Z pulse



Origin return mode 34

6. Object Dictionary object dictionary

6.1 object attribute description

Glossary

" **Index** ": Specify the position of each object in the object dictionary, expressed in hexadecimal (h).

" **Subindex** ": Multiple objects under the same index, each object is offset under this class.

“ **Data Type** ”: See Table 5-1 for details.

Table 5-1 Description of data types

type of data	Numerical range	Data length	DS301 value
Int8	-128 ~ 127	1 byte	2
Uint8	0 ~ 255	1 byte	5
Int16	-32768 ~ +32767	2 bytes	3
Uint16	0 ~ 65535	2 bytes	6
Int32	-2147483648 ~ +2147483647	4 bytes	4
Uint32	0 ~ 4294967295	4 bytes	7
String	ASCII	-	9

“ **Read and write type** ”: See Table 4-1 for details.

Table 5-2 Description of read and write types

Read and write type	Description
RW	Read and write
WO	just write
RO	Read only
CONST	Constant, read only

" **Object structure** ": See Table 3-3 for details.

Table 5-3 Description of object structure

Object structure	Description
VAR	A single simple value containing the data types in Table 3-1
ARR	Multiple data field objects, the variables in the fields are all the same data shape
REC	Multiple data field objects, the variables in the fields are different data shapes

6.2 1000h group object list

index	Subindex	name	Accessibility	PDO mapping	type of data	unit	range	Factory default
1000h	00	Drive type	RO	NO	UINT32	-	-	0x00020192
100 1 h	00	error register	RO	NO	UINT8	-	-	-
100 8 h	00	Driver name	RO	NO	-	-	-	Servo Device
100 A h	00	Software version	RO	NO	-	-	-	3761
10 18 h	00	Maximum son Quote number	RO	NO	UINT8	-	-	4
	01	Vendor ID	RO	NO	UINT32	-	-	0x850104
	02	Device code	RO	NO	UINT32	-	-	0x26483052
	03	Revision	RO	NO	UINT32	-	-	0x39613708
	04	serial number	RO	NO	UINT32	-	-	0x20181123
1 600 h	00	Mapping object Number	RW	NO	UINT8	-	0~8	5
	01	Mapping object 1	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0x60400010
	02	Mapping object 2	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0x607A0020
	03	Mapping object 3	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0x60FF0020
	04	Mapping object 4	RW	NO	UINT32		0~(2 ^{32} - 1)	0x60710010
	05	Mapping object 5	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0x60600008
	06	Mapping object 6	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0
	07	Mapping object 7	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0
	08	Mapping object 8	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0
1 601 h	00	Mapping object	RW	NO	UINT8	-	0~8	2

		Number						
	01	Mapping object 1	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0x60400010
	02	Mapping object 2	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0x60FF0020
	03	Mapping object 3	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0
	04	Mapping object 4	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0
	05	Mapping object 5	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0
	06	Mapping object 6	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0
	07	Mapping object 7	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0
	08	Mapping object 8	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0
1 602 h	00	Mapping object Number	RW	NO	UINT8	-	0~8	2
	01	Mapping object 1	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0x60400010
	02	Mapping object 2	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0x60710020
	03	Mapping object 3	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0
	04	Mapping object 4	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0
	05	Mapping object 5	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0
	06	Mapping object 6	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0
	07	Mapping object 7	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0
	08	Mapping object 8	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0
1 603 h	00	Mapping object Number	RW	NO	UINT8	-	0~8	7
	01	Mapping object 1	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0x60400010
	02	Mapping object 2	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0x607A0020
	03	Mapping object 3	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0x60FF0020

	04	Mapping object 4	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0x60600008
	05	Mapping object 5	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0x60B80010
	06	Mapping object 6	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0x60E00010
	07	Mapping object 7	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0x60E10010
	08	Mapping object 8	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0
1 A00 h	00	Mapping object Number	RW	NO	UINT8	-	0~8	5
	01	Mapping object 1	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0x60410010
	02	Mapping object 2	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0x60640020
	03	Mapping object 3	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0x606C0020
	04	Mapping object 4	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0x60770010
	05	Mapping object 5	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0x60610008
	06	Mapping object 6	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0
	07	Mapping object 7	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0
	08	Mapping object 8	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0
1 A01 h	00	Mapping object Number	RW	NO	UINT8	-	0~8	2
	01	Mapping object 1	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0x60410010
	02	Mapping object 2	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0x60640020
	03	Mapping object 3	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0
	04	Mapping object 4	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0
	05	Mapping object 5	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0
	06	Mapping object 6	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0

	07	Mapping object 7	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0
	08	Mapping object 8	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0
1 A02 h	00	Mapping object Number	RW	NO	UINT8	-	0~8	3
	01	Mapping object 1	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0x60410010
	02	Mapping object 2	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0x60640020
	03	Mapping object 3	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0x606C0020
	04	Mapping object 4	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0
	05	Mapping object 5	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0
	06	Mapping object 6	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0
	07	Mapping object 7	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0
	08	Mapping object 8	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0
1 A03 h	00	Mapping object Number	RW	NO	UINT8	-	0~8	3
	01	Mapping object 1	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0x60410010
	02	Mapping object 2	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0x60640020
	03	Mapping object 3	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0x60770010
	04	Mapping object 4	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0
	05	Mapping object 5	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0
	06	Mapping object 6	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0
	07	Mapping object 7	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0
	08	Mapping object 8	RW	NO	UINT32	-	0~(2 ^{32} - 1)	0
1 C00 h	00	Maximum subindex	RO	NO	UINT8	-	-	4
	01	SM0 type	RO	NO	UINT8	-	-	1

	02	SM1 type	RO	NO	UINT8	-	-	2
	03	SM2 type	RO	NO	UINT8	-	-	3
	04	SM3 type	RO	NO	UINT8	-	-	4
1 C12 h	00	RPDO allocation Maximum subindex	RW	NO	UINT8	-	0~1	1
	01	RPDO allocation Object index	RW	YES	UINT16	-	0~65535	0x1601
1 C13 h	00	TPDO allocation Maximum subindex	RW	NO	UINT8	-	0~1	1
	01	TPDO allocation Object index	RW	YES	UINT16	-	0~65535	0x1A00
1 C32 h	00	Synchronous output Largest child Number of indexes	RO	NO	UINT8	-	-	32
	01	Synchronization type	RW	NO	UINT16	-	-	2
	02	Cycle Time	RO	NO	UINT32	Ns	-	0
	04	Supported synchronization Types of	RO	NO	UINT16	-	-	4
	05	Minimum period	RO	NO	UINT32	Ns	-	125000
	06	Calculation and replication time	RO	NO	UINT32	Ns	-	0
	08	Get loop time	RW	NO	UINT16	-	-	0
	09	delay	RO	NO	UINT32	Ns	-	0
	0A	SYN0 time	RW	NO	UINT32	Ns	-	-
	0B	Lost number	RO	NO	UINT16	-	-	0
	0C	Loop out count	RO	NO	UINT16	-	-	0
	20	Synchronization error	RO	NO	BOOL	-	-	FALSE
1 C33 h	00	Synchronous input Largest child Number of indexes	RO	NO	UINT8	-	-	32

	01	Synchronization type	RW	NO	UINT16	-	-	2
	02	Cycle Time	RO	NO	UINT32	Ns	-	0
	04	Supported synchronization Types of	RO	NO	UINT16	-	-	4
	05	Minimum period	RO	NO	UINT32	Ns	-	125000
	06	Calculation and replication time	RO	NO	UINT32	Ns	-	0
	08	Get loop time	RW	NO	UINT16	-	-	0
	09	delay	RO	NO	UINT32	Ns	-	0
	0A	SYN0 time	RW	NO	UINT32	Ns	-	-
	0B	Lost number	RO	NO	UINT16	-	-	0
	0C	Loop out count	RO	NO	UINT16	-	-	0
	20	Synchronization error	RO	NO	BOOL	-	-	0

6.3 2000h group object list

index	Subindex	name	Accessibility	PDO mapping	type of data	unit	range	Factory default
2000h	00	Control Mode	RO	NO	UINT16	-	-	-
2002h	00	Motor rotation direction	RW	NO	UINT16	-	0~1	0
2003h	00	Default monitoring parameters	RW	NO	UINT16	-	0~65535	0
2004h	00	Servo OFF and selection of stop method when the first type of alarm occurs	RW	NO	UINT16	-	0~2	0
2005h	00	Servo generation type 2 alarm stop method selection	RW	NO	UINT16	-	0~1	0
2006h	00	Overtravel warning detection option	RW	NO	UINT16	-	0~1	0

2007h	00	Stop method at servo overtravel (OT) Method selection	RW	NO	UINT16	-	0~1	1
2008h	00	Servo after electromagnetic brake hold Locking time	RW	NO	UINT16	10ms	0~50	10
2009h	00	Electromagnetic brake holding time delay	RW	NO	UINT16	10ms	10~100	50
2010h	00	Electromagnetic brake holding time delay release speed	RW	NO	UINT16	rpm	0~10000	100
2012h	00	External regenerative resistor power	RW	NO	UINT16	10W	0~65535	0
2013h	00	External regeneration resistor resistance value	RW	NO	UINT16	0.01Ω	0~65535	0
2015h	00	Overload warning value	RW	NO	UINT16	1%	1~100	20
2016h	00	Motor overload detection of base current Derating setting	RW	NO	UINT16	1%	0~100	100
2039h	00	Multi-turn upper limit	RW	NO	UINT16	Rev	0~65535	65535
2040h	00	How to use the absolute encoder	RW	NO	UINT16	-	0~1	0
2041h	00	Alarm/warning option for absolute encoder battery undervoltage	RW	NO	UINT16	-	0~1	0
2044h	00	Fully closed loop related application switches	RW	NO	UINT16	-	0~51	1

2045h	00	Function selection in case of undervoltage	RW	NO	UINT16	-	0~2	0
2046h	00	Torque limiting when main circuit voltage drops	RW	NO	UINT16	1%	0~100	50
2047h	00	Torque limit release time when main circuit voltage drops	RW	NO	UINT16	ms	0~1000	100
2050h	00	Torque limiting method setting	RW	NO	UINT16	-	0~3	0
2051h	00	Maximum torque limit 1	RW	NO	UINT16	1%	0~500	500
2052h	00	Maximum torque limit 2	RW	NO	UINT16	1%	0~500	500
2053h	00	Emergency stopping torque	RW	NO	UINT16	-	0~800	800
2070h	00	Number of encoder divider pulses	RW	NO	UINT16	-	16~4194304	2048
2072h	00	Inverse crossover output	RW	NO	UINT16	-	0~16384	0
2080h	00	EtherCAT site specific (OMRON)	RO	NO	UINT16	-	1~65535	1
2100h	00	Rotational inertia ratio	RW	NO	UINT16	1%	0~20000	100
2101h	00	No. 1 Speed Gain	RW	NO	UINT16	0.1Hz	10~20000	400
2102h	00	1st velocity integration time constant	RW	NO	UINT16	0.01ms	15~51200	2000
2103h	00	Position 1 gain	RW	NO	UINT16	0.1/s	10~20000	400
2104h	00	No. 1 Torque command filter	RW	NO	UINT16	0.01ms	0~65535	100
2105h	00	No. 2 Speed Gain	RW	NO	UINT16	0.1Hz	10~20000	400
2106h	00	No. 2 Velocity integration time constant	RW	NO	UINT16	0.01ms	15~51200	2000

2107h	08	Position 2 gain	RW	NO	UINT16	0.1/s	10~20000	400
2108h	00	No. 2 Torque command filter	RW	NO	UINT16	0.01ms	0~65535	100
2110h	00	Gain switching mode selection switch	RW	NO	UINT16	-	0~1	0
2111h	00	Position control gain automatic switching conditions	RW	NO	UINT16	-	0~5	0
2112h	00	Gain switching transition time 1	RW	NO	UINT16	ms	0~65535	0
2113h	00	Gain switching transition time 2	RW	NO	UINT16	ms	0~65535	0
2114h	00	Gain switching wait time 1	RW	NO	UINT16	ms	0~65535	0
2115h	00	Gain switching wait time 2	RW	NO	UINT16	ms	0~65535	0
2121h	00	Speed feedforward gain	RW	NO	UINT16	1%	0~100	0
2122h	00	Speed feed-forward filtering time	RW	NO	UINT16	0.01ms	0~6400	0
2123h	00	Using V-REF as a speed feedforward selection	RW	NO	UINT16	-	0~1	0
2124h	00	Speed/position control selection (T-REF assignment)	RW	NO	UINT16	-	0~1	0
2125h	00	Speed feedback filtering time constant	RW	NO	UINT16	0.01ms	0~655535	0
2126h	00	Velocity feedback moving average filter time constant	RW	NO	UINT16	125us	0~20	0
2130h	00	Control method of speed loop (PI/IP)	RW	NO	UINT16	PI/P	0~1	0

2131h	00	Speed loop P/PI switching condition selector switch	RW	NO	UINT16	-	0~4	0
2132h	00	Speed loop P/PI switching condition (torque command)	RW	NO	UINT16	1%	0~800	200
2133h	00	Speed loop P/PI switching condition (speed command)	RW	NO	UINT16	rpm	0~10000	0
2134h	00	Speed loop P/PI switching conditions (acceleration)	RW	NO	UINT16	rpm/s	0~30000	0
2135h	00	Speed loop P/PI switching condition (position deviation)	RW	NO	UINT16	Command unit	0~10000	0
2140h	00	Medium frequency vibration suppression control options	RW	NO	UINT16	-	0~17	16
2141h	00	Medium frequency vibration suppression inertia correction	RW	NO	UINT16	1%	1~1000	100
2142h	00	Medium frequency vibration suppression frequency	RW	NO	UINT16	0.1Hz	10~30000	1000
2143h	00	Medium frequency vibration suppression attenuation gain	RW	NO	UINT16	1%	0~300	0

2150h	00	Trap filter 1 automatic adjustment selection	RW	NO	UINT16	-	0~1	1
2151h	00	Trap filter 2 automatic adjustment selection	RW	NO	UINT16	-	0~1	1
2152h	00	Automatic trap resonance detection sensitivity	RW	NO	UINT16	1%	1~200	100
2153h	00	Trap filter 1 Frequency	RW	NO	UINT16	Hz	50~5000	5000
2154h	00	Trap filter 1Q value	RW	NO	UINT16	0.01	50~1000	70
2155h	00	Trap Filter 1 Depth	RW	NO	UINT16	0.001	0~1000	0
2156h	02	Trap filter 2 Frequency	RW	NO	UINT16	Hz	50~5000	5000
2157h	00	Trap filter 2Q value	RW	NO	UINT16	0.01	50~1000	70
2158h	00	Trap Filter 2 Depth	RW	NO	UINT16	0.001	0~1000	0
2160h	00	Disturbance compensation function selection	RW	NO	UINT16	-	0~1	0
2161h	00	Disturbance observer cut-off frequency	RW	NO	UINT16	0.1Hz	10~10000	1500
2163h	00	Disturbance compensation factor	RW	NO	UINT16	1%	0~100	0
2165h	00	Disturbance observer inertia correction factor	RW	NO	UINT16	1%	1~1000	100
2166h	00	Speed Observer Switch	RW	NO	UINT16	-	0~1	0
2167h	01	Speed observer cut-off frequency	RW	NO	UINT16	Hz	1~500	80

2168h	00	Speed observer inertia correction factor	RW	NO	UINT16	1%	0~1000	100
2169h	00	Speed observer filter time correction value	RW	NO	UINT16	0.01ms	0~65535	10
2170h	00	Frictional torque compensation cut-off speed	RW	NO	UINT16	rpm	0~1000	20
2171h	00	Frictional torque positive compensation factor	RW	NO	UINT16	1%/100 rpm	0~100	0
2172h	00	Frictional torque reversal compensation factor	RW	NO	UINT16	1%/100 rpm	0~100	0
2175h	00	Robust control selection	RW	NO	UINT16	-	0~1	0
2177h	00	Robust control of tuning values	RW	NO	UINT16	0.1Hz	100~800	400
2178h	00	Robust control of minimum load values	RW	NO	UINT16	1%	0~500	0
2185h	00	Vibration detection options	RW	NO	UINT16	-	0~2	0
2186h	00	Vibration detection sensitivity	RW	NO	UINT16	1%	50~500	100
2187h	00	Vibration detection value	RW	NO	UINT16	rpm	0~5000	50
2192h	00	Intelligent setting process position overshoot detection sensitivity (relative positioning completion)	RW	NO	UINT16	1%	0~100	100
2193h	00	Maximum gain setting value	RW	NO	UINT16	0.1Hz	10~4000	3000

		when exploring maximum gain						
2203h	00	Command pulse input multiplier	RW	NO	UINT16	-	0~100	1
2204h	00	Electronic gear ratio molecule	RW	NO	UINT32	-	0~1073741824	1
2206h	00	Electronic gear score master	RW	NO	UINT32	-	0~1073741824	1
2211h	00	Position command low-pass filter time constant	RW	NO	UINT16	0.1ms	0~6550	0
2212h	00	Position command sliding average filter time	RW	NO	UINT16	0.1ms	0~10000	0
2230h	00	Low frequency vibration suppression options	RW	NO	UINT16	-	0~2	0
2231h	04	Automatic adjustment selection of low frequency vibration suppression function	RW	NO	UINT16	-	0~1	1
2232h	05	Low frequency vibration detection sensitivity (relative positioning completion)	RW	NO	UINT16	0.1%	1~3000	400
2235h	00	Low Frequency Vibration Suppression 1 Frequency	RW	NO	UINT16	0.1Hz	10~2000	2000
2236h	00	Low-frequency vibration suppression 1 Correction	RW	NO	UINT16	1%	10~1000	100
2237h	00	Low Frequency Vibration	RW	NO	UINT16	0.1Hz	10~2000	2000

		Suppression 2 Frequency						
2238h	00	Low frequency vibration suppression 2 Correction	RW	NO	UINT16	1%	10~1000	100
2240h	00	Model tracking control options	RW	NO	UINT16	-	0~1	0
2241h	00	Model tracking control gain	RW	NO	UINT16	0.1/s	1~20000	500
2242h	00	Model tracking control attenuation coefficient	RW	NO	UNIT16	0.1%	500~2000	1000
2243h	00	Model tracking control speed feedforward Gain	RW	NO	UINT16	0.1%	0~10000	1000
2244h	00	Model tracking control forward torque feedforward gain	RW	NO	UINT16	0.1%	0~10000	1000
2245h	00	Model tracking control reverse torque feedforward gain	RW	NO	UINT16	0.1%	0~10000	1000
2246h	00	Model 2 tracking control gain	RW	NO	UINT16	0.1/s	1~20000	500
2247h	00	Model 2 tracking control attenuation coefficient	RW	NO	UINT16	0.1%	500~2000	1000
2248h	00	Model tracking control type selection (improved resolution)	RW	NO	UINT16	-	0~1	1
2249h	00	Speed feedforward / torque	RW	NO	UINT16	-	0~1	0

		feedforward selection						
2250h	00	Use of external encoder for fully closed-loop control	RW	NO	UINT16	-	0~3	0
2252h	00	Deviation factor between motor loads for 1 revolution of fully closed loop rotation	RW	NO	UINT16	1%	0~100	20
2253h	00	External scale resolution	RW	NO	UINT16	P/Rev	4~1048576	32768
2257h	00	Excessive deviation between motor and load setting	RW	NO	UINT32	-	0~1073741824	1000
2260h	00	Position proximity signal width	RW	NO	UINT16	-	1~1073741824	1073741824
2262h	00	Location completion range	RW	NO	UINT16	-	0~1073741824	7
2264h	00	Position deviation maximum threshold	RW	NO	UINT16	-	1~1073741824	25165824
2266h	00	Excessive position deviation warning setting	RW	NO	UINT16	1%	0~100	100
2267h	00	Excessive position deviation alarm threshold at servo ON	RW	NO	UINT16	-	1~1073741824	25165824
2269h	00	Excessive position deviation warning threshold when servo ON	RW	NO	UINT16	1%	10~100	100

2270h	00	Speed limit value when servo is ON	RW	NO	UINT16	rpm	0~10000	10000
2274h	00	/COIN output time	RW	NO	UINT16	-	0~2	0
2310h	00	Speed command trapezoidal acceleration time	RW	NO	UINT16	ms	0~10000	0
2311h	00	Speed command trapezoidal deceleration time	RW	NO	UINT16	ms	0~10000	0
2312h	00	Zero speed clamping mode	RW	NO	UINT16	-	0~3	3
2313h	00	Zero Speed Clamping Speed Threshold	RW	NO	UINT16	rpm	0~10000	10
2317h	00	Turning judgment threshold	RW	NO	UINT16	rpm	1~10000	20
2320h	00	Speed Consistency Range	RW	NO	UINT16	rpm	0~100	10
2401h	00	Torque command 2nd order low-pass filter cut-off frequency	RW	NO	UINT16	Hz	100~5000	5000
2402h	00	Torque command 2nd order low-pass filter Q value	RW	NO	UINT16	0.01	50~100	50
2411h	00	Setting of speed limiting method during torque control	RW	NO	UINT16	-	0~1	1
2412h	00	Speed limit selection	RW	NO	UINT16	-	0~1	0
2413h	00	Speed limitation during torque control	RW	NO	UINT16	rpm	0~10000	1000
2600h	00	Switching input signal	RW	NO	UINT16	-	0~1	1

		distribution mode						
2601h	00	CN1-40 Input configuration	RW	NO	UINT16	-	0~0x1FFF	0x001
2602h	00	CN1-42 Input configuration	RW	NO	UINT16	-	0~0x1FFF	0x102
2603h	00	CN1-43 Input configuration	RW	NO	UINT16	-	0~0x1FFF	0x103
2604h	00	CN1-41 Input configuration	RW	NO	UINT16	-	0~0x1FFF	0x1D
2605h	00	CN1-44 Input configuration	RW	NO	UINT16	-	0~0x1FFF	0x13
2606h	00	CN1-45 Input configuration	RW	NO	UINT16	-	0~0x1FFF	0x14
2607h	00	CN1-46 Input configuration	RW	NO	UINT16	-	0~0x1FFF	0x007
2610h	00	Switching input internal configuration 1	RW	NO	UINT16	-	0xFF	0x00
2611h	00	Switching input internal configuration 2	RW	NO	UINT16	-	0xFF	0x00
2612h	00	Switching input internal configuration 3	RW	NO	UINT16	-	0xFF	0x00
2613h	00	CN1-25,26 Output configuration	RW	NO	UINT16	-	0~0x109	0x006
2614h	00	CN1-27,28 Output configuration	RW	NO	UINT16	-	0~0x109	0x001
2615h	00	CN1-29,30 Output configuration	RW	NO	UINT16	-	0~0x109	0x002
2622h	00	Warning detection options	RW	NO	UINT16	-	0~1	0
2691h	00	Encoder type	RW	NO	UINT16	-	1~18	10
2692h	00	Absolute encoder operation	RW	NO	UINT16	-	0~2	0
2900h	00	Communication write 2000h	RW	NO	UINT16	-	0~1	1

		group function code EEPROM storage selection						
2901h	00	Feedback position (6064h) overflow function selection	RW	NO	UINT16	-	0~1	0
2902h	00	Home position storage function selection	RW	NO	UINT16	-	0~1	0
2903h	00	Ratio of number of linear axes	RW	NO	UINT16	-	1~256	1
2920h	00	Home position offset value L	RW	NO	UINT16	Com/s	0~65535	0
2921h	00	Home position offset value H	RW	NO	UINT16	Com/s	0~65535	0
2923h	00	Number of multi-turn overflows	RW	NO	UINT16	Rev	0~65535	0

6.4 6000h group object list

index	Subindex	name	Accessibility	PDO mapping	type of data	unit	range	Factory default
603F h	00	error code	RO	TPDO	UINT16	-	0~65535	0
6040 h	00	Control word	RW	RPDO	UINT16	-	0~65535	0
6041 h	00	Status word	RO	TPDO	UINT16	-	0~65535	0
6060 h	00	Control mode	RO	RPDO	INT8	-	0~10	8
6061 h	00	Mode display	RO	TPDO	INT8	-	0~10	-
6062 h	00	Position command	RO	TPDO	INT32	Command unit	-	-
6063 h	00	Position feedback	RO	TPDO	INT32	Encoder unit	-	-
6064h	00	Position feedback	RO	TPDO	INT32	Command unit	-	-
6065h	00	Position deviation	RW	RPDO	UINT32	Command unit	-	0xA0000000

		Excessive threshold						
6067h	00	Location arrival Threshold	RW	RPDO	UINT32	Command unit	0~(2 ^{32} -1)	100
6068h	00	Location arrival Window time	RW	RPDO	UINT16	2ms	0~65535	0
606Bh	00	Actual speed instruction	RO	TPDO	UINT32	r/min	-	-
606Ch	00	Actual speed	RO	TPDO	INT32	Command unit /s	-	-
606Dh	00	Speed arrival Threshold	RW	RPDO	UINT16	r/min	0~65535	100
606Eh	00	Speed arrival Window time	RW	RPDO	UINT16	Ms	0~65535	0
606Fh	00	Zero speed threshold	RW	RPDO	UINT16	r/min	0~65535	10
6070h	00	0 -speed time window	RW	RPDO	UINT16	Ms	0~65535	0
6071h	00	Target torque	RW	RPDO	INT16	0.1%	-3000~ 3000	0
6074h	00	Torque command	RO	TPDO	INT16	1%	- 300~300	0
6075h	00	Motor rating Current	RO	TPDO	UINT32	mA	0~(2 ^{32} -1)	-
6076h	00	Motor rating Torque	RO	TPDO	UINT32	mNm	0~(2 ^{32} -1)	-
6077h	00	Actual torque	RO	TPDO	UINT16	1%	- 300~300	0
607Ah	00	target location	RW	RPDO	INT32	Command unit	-2 ^{31} ~ (2 ^{31} -1)	0
607Ch	00	Origin offset	RW	RPDO	INT32	Encoder unit	-2 ^{31} ~ (2 ^{31} -1)	0
607Dh	00	Sub-index number	RO	NO	UINT8	-	-	2
	01	Minimum position limit	RW	RPDO	INT32	Command unit	-2 ^{31} ~ (2 ^{31} -1)	-2 ^{31}
	02	Maximum position limit	RW	RPDO	INT32	Command unit	-2 ^{31} ~ (2 ^{31} -1)	2 ^{31} -1

607Eh	00	Command polarity	RW	RPDO	UINT8	-	0~255	0
607Fh	00	Maximum profile speed	RW	RPDO	UINT32	0.1r/min	0~(2^{32}-1)	50000
6080h	00	Maximum motor speed	RW	RPDO	UINT32	1r/min	0~(2^{32}-1)	6000
6081h	00	Contour speed	RW	RPDO	UINT32	Command unit /s	0~(2^{32}-1)	10000000
6083h	00	Contour acceleration degree	RW	RPDO	UINT32	Command unit /s^2	0~(2^{32}-1)	200
6084h	00	Contour deceleration degree	RW	RPDO	UINT32	Command unit /s^2	0~(2^{32}-1)	200
6086h	00	Running curve select	RW	RPDO	INT16	-	-2^{15}~ (2^{15}-1)	0
6087h	00	Torque ramp time	RW	RPDO	UINT32	MS	0~(2^{32}-1)	0
6091h	00	Subindex Number	RO	NO	UINT8	-	-	2
	01	Motor Resolution	RW	RPDO	UINT32	-	1~(2^{32}-1)	1
	02	Load shaft Resolution	RW	RPDO	UINT32	-	1~(2^{32}-1)	1
6098h	00	Origin return the way	RW	RPDO	INT8	-	1~35	1
6099h	00	Subindex Number	RO	NO	UINT8	-	-	2
	01	High speed search Deceleration point	RW	RPDO	UINT32	Command unit /s	0~(2^{32}-1)	279620266
	02	Low speed search origin	RW	RPDO	UINT32	Command unit /s	0~(2^{32}-1)	5592405
609Ah	00	Return to zero Acceleration	RW	RPDO	UINT32	Command unit /s^2	0~(2^{32}-1)	16000

60B0h	00	Position offset	RW	RPDO	INT32	Command unit	-2^{31} ~ (2^{31}-1)	0
60B8h	00	Probe mode	RW	RPDO	UINT16	-	0~65535	0
60B9h	00	Probe status	RO	TPDO	UINT16	-	0~65535	0
60BAh	00	Probe 1 on Rising position	RO	TPDO	INT32	Command unit	-2^{31} ~ (2^{31}-1)	0
60BBh	00	Under probe 1 Falling edge position	RO	TPDO	INT32	Command unit	-2^{31} ~ (2^{31}-1)	0
60BCh	00	Probe 1 on Rising position	RO	TPDO	INT32	Command unit	-2^{31} ~ (2^{31}-1)	0
60BDh	00	Under probe 1 Falling edge position	RO	TPDO	INT32	Command unit	-2^{31} ~ (2^{31}-1)	0
60E0h	00	Forward torque limit	RW	RPDO	UINT16	0.1%	0~3000	3000
60E1h	00	Reverse torque limit	RW	RPDO	UINT16	0.1%	0~3000	3000
60F4h	00	Position deviation	RO	TPDO	INT32	Command unit	-	-
60FCh	00	Position command	RO	TPDO	INT32	Encoder unit	-	-
60FDh	00	DI status	RO	TPDO	UNT32	-	-	-
60FEh	00	Number of subindexes	RO	NO	UINT8	-	-	2
	01	Physical output	RO	TPDO	UINT32	-	0~(2^{32}-1)	0
	02	-	-	-	-	-	-	-
60FFh	00	Target speed	RW	RPDO	INT32	Command unit /s	-	-
6052h	00	Supported Drive mode	RO	NO	UINT32	-	-	0x3AD

6.5 Object Dictionary Details

6.5.1 Detailed Description of 1000h Group Objects

Object 1000h : Device Type					
Index	1000h				
Name	Equipment type				
Object structure	VAR	**type of data**	Uint32	**data range**	-
Can map	NO	**Accessibility**	RO	**Factory default**	0x00020 192

Describe the CoE device type:

BIT	name	description
0~15	Device subprotocol	402 (0x192): device sub-protocol
16~23	Types of	02: Servo drive
25~31	mode	Factory customization

Object 1001h : Error Register					
Index	1001h				
Name	Error register				
Object structure	VAR	**type of data**	Uint8	**data range**	-
Can map	NO	**Accessibility**	RO	**Factory default**	0

Bitwise to contain error messages

BIT	meaning	BIT	meaning
0	conventional	4	Communication
1	Current	5	Sub protocol
2	Voltage	6	Reserved
3	temperature	7	Vendor definition

Object 1008h : Manufacturer Device Name					
Index	1008h				
Name	Manufacturer equipment name				
Object structure	-	**type of data**	UINT8	**data range**	-
Can map	NO	**Accessibility**	RO	**Factory setting**	Servo Device

Describe the manufacturer's device name

Object 100Ah : Software Version (Software Version)					
Index	100Ah				
Name	Software version				
Object structure	-	**type of data**	-	**data range**	-

Can map	NO	**Accessibility**	RO	**Factory default**	3761
Describe the software version number					

Object 1018h : ID Object (Identity Object)					
Index	1018h				
Name	ID object				
Object structure	REC	**type of data**	-	**data range**	-
Can map	NO	**Accessibility**	RO	**Factory default**	-

Subindex	00h				
Name	Child index number (number of entries)				
Object structure	-	**type of data**	UINT8	**data range**	4
Can map	NO	**Accessibility**	R O	**Factory setting**	4

Subindex	01h				
Name	Vendor ID (Vendor-ID)				
Object structure	-	**type of data**	UINT 32	**data range**	-
Can map	NO	**Accessibility**	R O	**Factory setting**	0x850104
Uniform distribution by ETG manufacturers					

Subindex	02 $_{h}$				
name	Device code (Product Code)				
Object structure	-	**type of data**	UINT 32	**data range**	-
Can map	NO	**Accessibility**	RO	**Factory setting**	0x26483052

The device code corresponds to the product series and product model of the electronic tag, and the corresponding relationship is as follows:

Bit	0~15	16~31
meaning	Product number	Product Series

Subindex	03h				
name	Revision number (Revision Number)				
Object structure	-	**type of data**	UINT 32	**data range**	-
Can map	NO	**Accessibility**	R O	**Factory setting**	0x37613708

Indicates the upgrade record of the driver software

Bit	0~15	16~31
meaning	Minor revision	Main revision

Subindex	04h				
name	Serial number (Serial number)				
Object structure	-	**type of data**	UINT32	**data range**	-
Can map	NO	**Accessibility**	R O	**Factory setting**	-
Indicates the software update time (manufacturer's own definition)					

Object 1600h : RPDO1 mapping object (1st Receive PDO Mapping)					
index	1600h				
name	RPDO1 mapping object				
Object structure	REC	**type of data**	UINT32	**data range**	-
Can map	NO	**Accessibility**	RW	**Factory default**	-
Function description: This object can only be modified if the PDO is invalid.					

Subindex	00h				
name	RPDO1 map number (RPDO1 Mapping Numbers)				
Object structure	-	**type of data**	UINT8	**data range**	0~8
Can map	NO	**Accessibility**	R W	**Factory setting**	4
Function description: When writing 0 , other sub-index mapping objects are invalid					

Subindex	01h				
name	The first mapping object (Mapping entry (1))				
Object structure	-	**type of data**	UINT32	**data range**	0~ 429496729 5
Can map	NO	**Accessibility**	R W	**Factory setting**	0x60400010

Subindex	02h				
name	The second mapping object (Mapping entry (2))				
Object structure	-	**type of data**	UINT32	**data range**	0~ 429496729 5
Can map	NO	**Accessibility**	R W	**Factory setting**	0x607A0020

Subindex	03h
name	The third mapping object (Mapping entry (3))

Object structure	-	**type of data**	UINT32	**data range**	0~ 429496729 5
Can map	NO	**Accessibility**	R W	**Factory setting**	0x60FF0020

Subindex	04h				
name	The fourth mapping object (Mapping entry (4))				
Object structure	-	**type of data**	UINT32	**data range**	0~ 429496729 5
Can map	NO	**Accessibility**	R W	**Factory setting**	0x60710010

Subindex	05h~08h				
name	Of 5-8 mappings objects (Mapping entry (5) ~ (8))				
Object structure	-	**type of data**	UINT32	**data range**	0~ 429496729 5
Can map	NO	**Accessibility**	R W	**Factory setting**	-

Object 1601h : RPDO2 mapping object (2nd Receive PDO Mapping)					
index	1601h				
name	RPDO1 mapping object				
Object structure	REC	**type of data**	UINT32	**data range**	-
Can map	NO	**Accessibility**	RW	**Factory default**	-
Function description: This object can only be modified if the PDO is invalid.					

Subindex	00h				
name	RPDO2 map number (RPDO2 Mapping Numbers)				
Object structure	-	**type of data**	Uint 8	**data range**	0~8
Can map	NO	**Accessibility**	R W	**Factory setting**	2
Function description: When writing 0 , other sub-index mapping objects are invalid					

Subindex	01h				
name	The first mapping object (Mapping entry (1))				
Object structure	-	**type of data**	Uint 32	**data range**	0~ 429496729 5
Can map	NO	**Accessibility**	R W	**Factory setting**	0x60400010

Subindex	02h
name	The second mapping object (Mapping entry (2))

Object structure	-	**type of data**	Uint 32	**data range**	0~ 429496729 5
Can map	NO	**Accessibility**	R W	**Factory setting**	0x607A0020

Subindex	03h~08h				
name	Of 3 to 8 mappings objects (Mapping entry (3) ~ (8))				
Object structure	-	**type of data**	Uint 32	**data range**	0~ 429496729 5
Can map	NO	**Accessibility**	R W	**Factory setting**	-

Object 1602h : RPDO3 mapping object (3rd Receive PDO Mapping)					
index	1602h				
name	RPDO1 mapping object				
Object structure	REC	**type of data**	Uint 32	**data range**	-
Can map	NO	**Accessibility**	RW	**Factory default**	-
Function description: This object can only be modified if the PDO is invalid.					

Subindex	00h				
name	RPDO3 map number (RPDO3 Mapping Numbers)				
Object structure	-	**type of data**	Uint 8	**data range**	0~8
Can map	NO	**Accessibility**	R W	**Factory setting**	2
Function description: When writing 0 , other sub-index mapping objects are invalid					

Subindex	01h				
name	The first mapping object (Mapping entry (1))				
Object structure	-	**type of data**	Uint 32	**data range**	0~ 429496729 5
Can map	NO	**Accessibility**	R W	**Factory setting**	0x60400010

Subindex	02h				
name	The second mapping object (Mapping entry (2))				
Object structure	-	**type of data**	Uint 32	**data range**	0~ 429496729 5
Can map	NO	**Accessibility**	R W	**Factory setting**	0x60FF0020

Subindex	03h~08h
name	Of 3 to 8 mappings objects (Mapping entry (3) ~ (8))

Object structure	-	**type of data**	Uint 32	**data range**	0~ 429496729 5
Can map	NO	**Accessibility**	R W	**Factory setting**	-

Object 1603h : RPDO4 mapping object (4th Receive PDO Mapping)					
index	1603h				
name	RPDO4 mapping object				
Object structure	REC	**type of data**	Uint 32	**data range**	-
Can map	NO	**Accessibility**	RW	**Factory default**	-
Function description: This object can only be modified if the PDO is invalid.					

Subindex	00h				
name	RPDO4 map number (RPDO4 Mapping Numbers)				
Object structure	-	**type of data**	Uint 8	**data range**	0~8
Can map	NO	**Accessibility**	R W	**Factory setting**	2
Function description: When writing 0 , other sub-index mapping objects are invalid					

Subindex	01h				
name	The first mapping object (Mapping entry (1))				
Object structure	-	**type of data**	Uint 32	**data range**	0~ 429496729 5
Can map	NO	**Accessibility**	R W	**Factory setting**	0x60400010

Subindex	02h				
name	The second mapping object (Mapping entry (2))				
Object structure	-	**type of data**	Uint 32	**data range**	0~ 429496729 5
Can map	NO	**Accessibility**	R W	**Factory setting**	0x60710020

Subindex	03h~08h				
name	Of 3 to 8 mappings objects (Mapping entry (3) ~ (8))				
Object structure	-	**type of data**	Uint 32	**data range**	0~ 429496729 5
Can map	NO	**Accessibility**	R W	**Factory setting**	-

Object 1A00h : TPDO1 mapping object (1st Transmit PDO Mapping)					
index	1A00h				
name	TPDO1 mapping object				
Object structure	REC	**type of data**	Uint 32	**data range**	-
Can map	NO	**Accessibility**	RW	**Factory default**	-
Function description: This object can only be modified if the PDO is invalid.					

Subindex	00h				
name	TPDO1 map number (TPDO1 Mapping Numbers)				
Object structure	-	**type of data**	Uint 8	**data range**	0~8
Can map	NO	**Accessibility**	R W	**Factory setting**	4
Function description: When writing 0 , other sub-index mapping objects are invalid					

Subindex	01h				
name	The first mapping object (Mapping entry (1))				
Object structure	-	**type of data**	Uint 32	**data range**	0~ 429496729 5
Can map	NO	**Accessibility**	R W	**Factory setting**	0x60410010

Subindex	02h				
name	The second mapping object (Mapping entry (2))				
Object structure	-	**type of data**	Uint 32	**data range**	0~ 429496729 5
Can map	NO	**Accessibility**	R W	**Factory setting**	0x60640020

Subindex	03h				
name	The third mapping object (Mapping entry (3))				
Object structure	-	**type of data**	Uint 32	**data range**	0~ 429496729 5
Can map	NO	**Accessibility**	R W	**Factory setting**	0x606C0020

Subindex	04h				
name	The fourth mapping object (Mapping entry (4))				
Object structure	-	**type of data**	Uint 32	**data range**	0~ 429496729 5
Can map	NO	**Accessibility**	R W	**Factory setting**	0x60770010

Subindex	05h~08h

name	Of 3 to 8 mappings objects (Mapping entry (3) ~ (8))				
Object structure	-	**type of data**	Uint 32	**data range**	0~ 429496729 5
Can map	NO	**Accessibility**	R W	**Factory setting**	-

Object 1A01h : TPDO2 mapping object (2nd Transmit PDO Mapping)					
index	1A01h				
name	TPDO2 mapping object				
Object structure	REC	**type of data**	Uint 32	**data range**	-
Can map	NO	**Accessibility**	RW	**Factory default**	-
Function description: This object can only be modified if the PDO is invalid.					

Subindex	00h				
name	TPDO2 map number (TPDO2 Mapping Numbers)				
Object structure	-	**type of data**	Uint 8	**data range**	0~8
Can map	NO	**Accessibility**	R W	**Factory setting**	3
Function description: When writing 0 , other sub-index mapping objects are invalid					

Subindex	01h				
name	The first mapping object (Mapping entry (1))				
Object structure	-	**type of data**	Uint 32	**data range**	0~ 429496729 5
Can map	NO	**Accessibility**	R W	**Factory setting**	0x60410010

Subindex	02h				
name	The second mapping object (Mapping entry (2))				
Object structure	-	**type of data**	Uint 32	**data range**	0~ 429496729 5
Can map	NO	**Accessibility**	R W	**Factory setting**	0x60640020

Subindex	03h				
name	The third mapping object (Mapping entry (3))				
Object structure	-	**type of data**	Uint 32	**data range**	0~ 429496729 5
Can map	NO	**Accessibility**	R W	**Factory setting**	0x60780010

Subindex	04h~08h

name	Of 3 to 8 mappings objects (Mapping entry (3) ~ (8))				
Object structure	-	**type of data**	Uint 32	**data range**	0~ 429496729 5
Can map	NO	**Accessibility**	R W	**Factory setting**	-

Object 1A02h : TPDO3 mapping object (3rd Transmit PDO Mapping)					
index	1A02h				
name	TPDO3 mapping object				
Object structure	REC	**type of data**	Uint 32	**data range**	-
Can map	NO	**Accessibility**	RW	**Factory default**	-
Function description: This object can only be modified if the PDO is invalid.					

Subindex	00h				
name	TPDO3 map number (TPDO3 Mapping Numbers)				
Object structure	-	**type of data**	Uint 8	**data range**	0~8
Can map	NO	**Accessibility**	R W	**Factory setting**	3
Function description: When writing 0 , other sub-index mapping objects are invalid					

Subindex	01h				
name	The first mapping object (Mapping entry (1))				
Object structure	-	**type of data**	Uint 32	**data range**	0~ 429496729 5
Can map	NO	**Accessibility**	R W	**Factory setting**	0x60410010

Subindex	02h				
name	The second mapping object (Mapping entry (2))				
Object structure	-	**type of data**	Uint 32	**data range**	0~ 429496729 5
Can map	NO	**Accessibility**	R W	**Factory setting**	0x60640020
Subindex	03h				
name	The third mapping object (Mapping entry (3))				
Object structure	-	**type of data**	Uint 32	**data range**	0~ 429496729 5
Can map	NO	**Accessibility**	R W	**Factory setting**	0x606C0020

Subindex	04h~08h				
name	Of 3 to 8 mappings objects (Mapping entry (3) ~ (8))				
Object structure	-	**type of data**	Uint 32	**data range**	0~ 429496729 5

Can map	NO	**Accessibility**	R W	**Factory setting**	-

Object 1A03h : TPDO4 mapping object (4th Transmit PDO Mapping)					
index	1A03h				
name	TPDO4 mapping object				
Object structure	REC	**type of data**	Uint 32	**data range**	-
Can map	NO	**Accessibility**	RW	**Factory default**	-
Function description: This object can only be modified if the PDO is invalid.					

Subindex	00h				
name	TPDO3 map number (TPDO3 Mapping Numbers)				
Object structure	-	**type of data**	Uint 8	**data range**	0~8
Can map	NO	**Accessibility**	R W	**Factory setting**	3
Function description: When writing 0 , other sub-index mapping objects are invalid					

Subindex	01h				
name	The first mapping object (Mapping entry (1))				
Object structure	-	**type of data**	Uint 32	**data range**	0~ 429496729 5
Can map	NO	**Accessibility**	R W	**Factory setting**	0x60410010

Subindex	02h				
name	The second mapping object (Mapping entry (2))				
Object structure	-	**type of data**	Uint 32	**data range**	0~ 429496729 5
Can map	NO	**Accessibility**	R W	**Factory setting**	0x60640020
Sub index	03h				
name	The third mapping object (Mapping entry (3))				
Object structure	-	**type of data**	UINT 32	**data range**	0~ 429496729 5
Can map	NO	**Accessibility**	R W	**Factory setting**	0x60770020

Sub index	04h~08h				
name	Of 3 to 8 mappings objects (Mapping entry (3) ~ (8))				
Object structure	-	**type of data**	UINT 32	**data range**	0~ 429496729 5
Can map	NO	**Accessibility**	R W	**Factory setting**	-

Object 1C00h : Sync manager type

index	1C00h				
name	Synchronous management type				
Object structure	REC	**type of data**	-	**data range**	-
Can map	NO	**Accessibility**	RO	**Factory default**	-

Sub index	00h				
name	Synchronize management of the largest sub-index number				
Object structure	-	**type of data**	UINT 8	**data range**	-
Can map	NO	**Accessibility**	R O	**Factory setting**	4

Sub index	01h				
name	SM0 communication type (Communication Type SM0)				
Object structure	-	**type of data**	UINT 8	**data range**	-
Can map	NO	**Accessibility**	R O	**Factory setting**	1
SM0 communication type: receiving mailbox					

Subindex	02h				
name	SM1 communication type (Communication Type SM1)				
Object structure	-	**type of data**	Uint 8	**data range**	-
Can map	NO	**Accessibility**	R O	**Factory setting**	2
SM1 communication type: send mailbox					
Subindex	03h				
name	SM2 communication type (Communication Type SM2)				
Object structure	-	**type of data**	Uint 8	**data range**	-
Can map	NO	**Accessibility**	R O	**Factory setting**	3
SM2 communication type: process data output					

Subindex	04h				
name	SM3 communication type (Communication Type SM3)				
Object structure	-	**type of data**	Uint 8	**data range**	-
Can map	NO	**Accessibility**	R O	**Factory setting**	4
SM3 communication type: process data input					

Object 1C12h : Synchronous Management 2_RPDO Assignment (Sync Manager 2 RPDO Assignment)	
index	1C12h

name	Synchronous management 2_RPDO allocation				
Object structure	ARR	**type of data**	Uint 16	**data range**	-
Can map	NO	**Accessibility**	RW	**Factory default**	-
Set the object index of the RPDO allocation					

Subindex	00h				
name	Synchronize the maximum sub-index number assigned by 2_RPDO				
Object structure	-	**type of data**	Uint 8	**data range**	-
Can map	NO	**Accessibility**	R W	**Factory setting**	1

Subindex	01h				
name	RPDO assigned object index (Index of RPDO Assignment)				
Object structure	-	**type of data**	Uint 8	**data range**	-
Can map	NO	**Accessibility**	R W	**Factory setting**	1
Set the RPDO allocation index: 1. Must be configurable in pre-run state 2. If you use the TwinCAT software to directly select the RPDO allocation, otherwise: a. 1C12-00h write value 0 b. 1C12-01h writes the pre-used RPDOx (1600h~1603h) and configures the RPDOx mapping object (such as 1600h) c. 1C12-00h write value 1					

Object 1C13h : Synchronous Management 2_TPDO Assignment (Sync Manager 2 TPDO Assignment)					
index	1C13h				
name	Synchronous management 2_TPDO allocation				
Object structure	ARR	**type of data**	Uint 16	**data range**	-
Can map	NO	**Accessibility**	RW	**Factory default**	-
Set the object index of the TPDO allocation					

Subindex	00h				
name	Synchronous management of the maximum sub-index number assigned by 2_TPDO				
Object structure	-	**type of data**	Uint 8	**data range**	-
Can map	NO	**Accessibility**	R W	**Factory setting**	1

Subindex	01h
name	TPDO assigned object index (Index of TPDO Assignment)

Object structure	-	**type of data**	Uint 8	**data range**	-
Can map	NO	**Accessibility**	R W	**Factory setting**	1
Set the TPDO allocation index: 1. Must be configurable in pre-run state 2. If you use TwinCAT software to directly select the allocation of TPDO, otherwise: a. 1C13-00h write value 0 b. 1C13-01h writes the pre-used TPDOx (1A00h~1A03h) and configures the TPDOx mapping object (such as 1A00h) c. 1C13-00h write value 1					

Object 1C32h : synchronization management 2_ synchronization parameter output (Sync Output Manager Paramater 2)					
index	1C32h				
name	Synchronous management 2_sync output parameters				
Object structure	REC	**type of data**	-	**data range**	-
Can map	NO	**Accessibility**	RO	**Factory default**	-
Describe the output parameters of SM2					

Sub index	00h				
name	Maximum and maximum sub index number of the synchronization management 2 synchronization parameter				
Object structure	-	**type of data**	UINT 8	**data range**	-
Can map	NO	**Accessibility**	R O	**Factory setting**	32

Subindex	01h				
name	Type of synchronization (Synchronization Type)				
Object structure	-	**type of data**	Uint 16	**data range**	-
Can map	NO	**Accessibility**	R O	**Factory setting**	2
2 indicates that the synchronization type of SM2 is distributed clock synchronization 0 mode (DC SYNC0 Mode)					

Subindex	02h				
name	Cycle time (Cycle Time)				
Object structure	-	**type of data**	Uint 32	**data range**	-
Can map	NO	**Accessibility**	R O	**Factory setting**	0
Reflects the period of DC SYNC0					

Subindex	04h

name	Synchronization Types supported				
Object structure	-	**type of data**	Uint 16	**data range**	-
Can map	NO	**Accessibility**	R O	**Factory setting**	4
Reflect distribution clock type 4 indicates support for distributed clock 0 mode (DC SYNC0)					

Subindex	05h				
name	The minimum cycle time (Minimum Cycle Time)				
Object structure	-	**type of data**	Uint 32	**data range**	-
Can map	NO	**Accessibility**	R O	**Factory setting**	125000
Reflects the minimum synchronization period supported by the slave (ns)					

Subindex	06h				
name	Calculation and copy time (Calc and Copy Time)				
Object structure	-	**type of data**	Uint 32	**data range**	-
Can map	NO	**Accessibility**	R O	**Factory setting**	-
Reflects the time (ns) that the microprocessor copies data from synchronization management to the local					

Subindex	08h				
name	Acquisition cycle time (Get Cycle Time)				
Object structure	-	**type of data**	Uint 16	**data range**	-
Can map	NO	**Accessibility**	R O	**Factory setting**	-

Subindex	09h				
name	Delay Time (Delay Time)				
Object structure	-	**type of data**	Uint 32	**data range**	-
Can map	NO	**Accessibility**	R O	**Factory setting**	-
Unit (ns)					

Subindex	0Ah				
name	SYNC0 Cycle Time (SYNC0 Cycle Time)				
Object structure	-	**type of data**	Uint 32	**data range**	-
Can map	NO	**Accessibility**	R O	**Factory setting**	-
When the clock mode is distributed, the value of the ESC register 09A0h is set (ns)					

Subindex	0Bh				
name	Number of synchronization events lost (SM-Event Missed)				
Object structure	-	**type of data**	Uint 32	**data range**	-
Can map	NO	**Accessibility**	R O	**Factory setting**	-

Subindex	0Ch				
name	Loop count exceeded (Cycle Time Too Small)				
Object structure	-	**type of data**	Uint 32	**data range**	-
Can map	NO	**Accessibility**	R O	**Factory setting**	-
Setting period is too small					

Subindex	20h				
name	Sync Error (SYNC Error)				
Object structure	-	**type of data**	BOOL	**data range**	-
Can map	NO	**Accessibility**	R O	**Factory setting**	-
TURE: Synchronous activation with no errors FALSE : Synchronization is not activated or no synchronization error has occurred.					

Object 1C33h : synchronization management 2_ sync input parameters (Sync Manager Paramater INPUT 2)					
index	1C33h				
name	Synchronous management 2_synchronous input parameters				
Object structure	REC	**type of data**	-	**data range**	-
Can map	NO	**Accessibility**	RO	**Factory default**	-
Describe the input parameters of SM2					

Subindex	00h				
name	Maximum and maximum subindex number of the synchronization management 2 synchronization parameter				
Object structure	-	**type of data**	Uint 8	**data range**	-
Can map	NO	**Accessibility**	R O	**Factory setting**	32

Subindex	01h				
name	Type of synchronization (Synchronization Type)				
Object structure	-	**type of data**	Uint 16	**data range**	-
Can map	NO	**Accessibility**	R O	**Factory setting**	2

2 indicates that the synchronization type of SM2 is distributed clock synchronization 0 mode (DC SYNC0 Mode)

Subindex	02h				
name	Cycle time (Cycle Time)				
Object structure	-	**type of data**	Uint 32	**data range**	-
Can map	NO	**Accessibility**	R O	**Factory setting**	0
Reflects the period of DC SYNC0					

Subindex	04h				
name	Synchronization Types supported				
Object structure	-	**type of data**	Uint 16	**data range**	-
Can map	NO	**Accessibility**	R O	**Factory setting**	4
Reflect distribution clock type 4 indicates support for distributed clock 0 mode (DC SYNC0)					

Subindex	05h				
name	The minimum cycle time (Minimum Cycle Time)				
Object structure	-	**type of data**	Uint 32	**data range**	-
Can map	NO	**Accessibility**	R O	**Factory setting**	125000
Reflects the minimum synchronization period supported by the slave (ns)					

Subindex	06h				
name	Calculation and copy time (Calc and Copy Time)				
Object structure	-	**type of data**	Uint 32	**data range**	-
Can map	NO	**Accessibility**	R O	**Factory setting**	-
Reflects the time (ns) that the microprocessor copies data from synchronization management to the local					

Subindex	08h				
name	Acquisition cycle time (Get Cycle Time)				
Object structure	-	**type of data**	Uint 16	**data range**	-
Can map	NO	**Accessibility**	R O	**Factory setting**	-

Subindex	09h				
name	Delay Time (Delay Time)				
Object structure	-	**type of data**	Uint 32	**data range**	-
Can map	NO	**Accessibility**	R O	**Factory setting**	-

Unit (ns)

Subindex	0Ah				
name	SYNC0 Cycle Time (SYNC0 Cycle Time)				
Object structure	-	**type of data**	Uint 32	**data range**	-
Can map	NO	**Accessibility**	R O	**Factory setting**	-
When the clock mode is distributed, the value of the ESC register 09A0h is set (ns)					

Subindex	0Bh				
name	Number of synchronization events lost (SM-Event Missed)				
Object structure	-	**type of data**	Uint 32	**data range**	-
Can map	NO	**Accessibility**	R O	**Factory setting**	-

Subindex	0Ch				
name	Loop count exceeded (Cycle Time Too Small)				
Object structure	-	**type of data**	Uint 32	**data range**	-
Can map	NO	**Accessibility**	R O	**Factory setting**	-
Setting period is too small					

Subindex	20h				
name	Sync Error (SYNC Error)				
Object structure	-	**type of data**	BOOL	**data range**	-
Can map	NO	**Accessibility**	R O	**Factory setting**	-
TURE: Synchronous activation with no errors FALSE : Synchronization is not activated or no synchronization error has occurred.					

6.5.2 Detailed description of the 2000h group object

Object 2000h: Control Mode					
index	2000h				
name	Control Mode				
Object structure	VAR	**type of data**	Uint16	**data range**	-
Can map	NO	**Accessibility**	RO	**Factory default**	-
In the EtherCAT models, the EtherCAT control mode is already solidified and does not need to be set					

Object 2002h: Motor rotation direction					
index	2002h				
name	Motor rotation direction				
Object structure	VAR	**type of data**	Uint16	**data range**	0~1
Can map	NO	**Accessibility**	RW	**Factory default**	0
Facing the motor end face. 0-Counterclockwise is positive 1-Clockwise direction is positive					

Object 2003h: Default monitoring parameters					
index	2003h				
name	Default monitoring parameters				
Object structure	VAR	**type of data**	Uint16	**data range**	0~0xfff
Can map	NO	**Accessibility**	RW	**Factory default**	3
0xfff means no monitoring parameters are displayed and the system status is displayed. Note: When the alarm is displayed, the alarm is displayed first, and the setting parameter cannot be displayed normally.					

Object 2004h: Stop Methods of Servo OFF or 1st Error					
index	2004h				
name	Stop Methods of Servo OFF or 1st Error				
Object structure	VAR	**type of data**	Uint16	**data range**	0~2
Can map	NO	**Accessibility**	RW	**Factory default**	0
0-Stop the motor by DB 1-Stop the motor by DB, then release DB 2 - Leave the motor in free motion without DB Note: DB (dynamic brake) is the function to perform emergency stop. If the power is turned ON/OFF or the servo motor is started or stopped by servo ON in the state where the command is input, the DB circuit will operate frequently and may cause the internal components of the servo unit to deteriorate, so please execute the start and stop of the servo motor by the speed input command or position command.					

Object 2005h: Stop Methods of 2nd Error					
index	2005h				
name	Control Mode				
Object structure	VAR	**type of data**	Uint16	**data range**	0~1
Can map	NO	**Accessibility**	RW	**Factory default**	0
Facing the motor end face. 0-Counterclockwise is positive 1-Clockwise direction is positive					

Object 2006h: Over Distance Warning or Not					
index	2006h				
name	Over Distance Warning or Not				
Object structure	VAR	**type of data**	Uint16	**data range**	0~1
Can map	NO	**Accessibility**	RW	**Factory default**	0
0 - No overtravel warning is detected at the limit 1-Overtravel warning detected at limit					

Object 2007h: Stop Methods of Over Distance					
index	2007h				
name	Stop Methods of Over Distance				
Object structure	VAR	**type of data**	Uint16	**data range**	0~2
Can map	NO	**Accessibility**	RW	**Factory default**	0
Set the stop method when servo overtravel occurs and the status after stopping. 0-Same stop method as Pn004 1-The torque set by Pn053 is used as the maximum value, and the servo enters the locked state after deceleration and stop. 2-The torque set by Pn053 is used as the maximum value, and the servo enters the free running state after deceleration and stop.					

Object 2008h: Lock Time after Electromagnetic Brake(10ms)					
index	2008h				
name	Lock Time after Electromagnetic Brake(10ms)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~50
Can map	NO	**Accessibility**	RW	**Factory default**	0

When the brake (/BK) signal and the servo ON (/S-ON) signal are turned off at the same time when the servo motor is enabled but not running and is in the stop state, setting this parameter changes the time from when the brake (/BK) signal is turned off to when the motor actually enters the non-energized state.

Note: The action delay time of the brake varies slightly. Setting this parameter prevents the motor from being moved slightly by the self weight of the mechanical moving part of the vertical axis or by external forces that may cause mechanical movement when the brake is actuated.

Object 2009h: Electromagnetic brake delay(10ms)					
index	2009h				
name	Electromagnetic brake delay(10ms)				
Object structure	VAR	**type of data**	Uint16	**data range**	10~100
Can map	NO	**Accessibility**	RW	**Factory default**	0
The output time of the brake signal (/BK) OFF can be adjusted by this parameter and the brake holding delay release speed (any one of which is satisfied) when the servo motor is not energized in the event of servo OFF/alarm/main circuit OFF during servo motor rotation. Note: See the "Hold brake action" instructions for details of the logic.					

Object 2010h: Electromagnetic brake delay unhaloed speed(rpm)					
index	2010h				
name	Electromagnetic brake delay unhaloed speed(rpm)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~10000
Can map	NO	**Accessibility**	RW	**Factory default**	100
See "Electromagnetic brake holding time delay" for details.					

Object 2012h: External braking resistor powe(10w)					
index	2012h				
name	External braking resistor powe(10w)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~65535
Can map	NO	**Accessibility**	RW	**Factory default**	0
When connecting an external regenerative resistor, the regenerative resistor power should be set to a value that matches the permissible capacity of the connected external regenerative resistor. The setting value varies depending on the cooling condition of the external regenerative resistor. Note: See the "Setting the regenerative resistor" instructions for details on the settings.1-Overtravel warning detected at limit					

Object 2013h: External braking resistor value(0.01Ω)	
index	2013h

name	External braking resistor value(0.01Ω)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~65535
Can map	NO	**Accessibility**	RW	**Factory default**	0
When connecting an external regenerative resistor, the regenerative resistor resistance value should be set to match the value of the external regenerative resistor connected. Note: The minimum regenerative resistor resistance value allowed to be connected is different for each power section, see "Setting Regenerative Resistor" for details, otherwise it may cause damage to the internal components of the servo unit.					

Object 2015h: Overload warning value(1%)					
index	2015h				
name	Overload warning value(1%)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~100
Can map	NO	**Accessibility**	RW	**Factory default**	30
Set this parameter to change the overload warning detection time, for example, the factory overload warning detection time is 20% of the overload alarm detection time. Note: The overload alarm detection time is detailed in the "Overload Alarm" description.					

Object 2016h: Base current derating setting at motor overload detection(1%)					
index	2016h				
name	Base current derating setting at motor overload detection(1%)				
Object structure	VAR	**type of data**	Uint16	**data range**	10~100
Can map	NO	**Accessibility**	RW	**Factory default**	100
This parameter allows you to change the motor current threshold at which the overload alarm starts to be calculated, which can shorten the overload alarm detection time. Note: This value is not valid at motor currents above 200% of nominal.					

Object 2039h: Upper limit value of multi rotating circles(Rev)					
index	2039h				
name	Upper limit value of multi rotating circles(Rev)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~65535
Can map	NO	**Accessibility**	RW	**Factory default**	65535
Reserved					

Object 2040h: Application method of Absolute Encoder					
index	2040h				
name	Application method of Absolute Encoder				
Object structure	VAR	**type of data**	Uint16	**data range**	0~1
Can map	NO	**Accessibility**	RW	**Factory default**	0
0 - Use absolute encoder as absolute encoder: If the motor is an absolute multi-turn encoder, this parameter is set to 1 to use the multi-turn absolute function 1 - Use absolute encoder as incremental encoder: When used as incremental encoder, the power-down position will not be recorded, and the alarm or warning corresponding to multi-turn will not be reported if the battery is under-voltage or the drive is powerless.					

Object 2041h: Low voltage Warning or Error					
index	2041h				
name	Low voltage Warning or Error				
Object structure	VAR	**type of data**	Uint16	**data range**	0~1
Can map	NO	**Accessibility**	RW	**Factory default**	0
0-Sets low battery voltage as fault: Drive power-up/reset monitors battery status for 4 to 9 seconds, undervoltage will report undervoltage warning (Er.830), no detection after time. 1-Set low battery voltage as warning: battery undervoltage (below 3V) will report undervoltage warning (Al.930), always monitor battery voltage, self-recoverable, enable operation without restriction.					

Object 2044h: switche of fully closed-loop application					
index	2044h				
name	switche of fully closed-loop application				
Object structure	VAR	**type of data**	Uint16	**data range**	0~51
Can map	NO	**Accessibility**	RW	**Factory default**	1

Object 2045h: Function selection in case of undervoltage					
index	2045h				
name	Function selection in case of undervoltage				
Object structure	VAR	**type of data**	Uint16	**data range**	0x00~0x02
Can map	NO	**Accessibility**	RW	**Factory default**	0
0-No main circuit down warning detected 1-Detect main circuit down warning					

2-Detect main circuit down warning and perform torque limiting, see "Main circuit under-voltage torque limiting" for details.

Object 2046h: Torque limit while main circuit voltage drops(1%)					
index	2046h				
name	Torque limit while main circuit voltage drops(1%)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~100
Can map	NO	**Accessibility**	RW	**Factory default**	50
According to the undervoltage warning, the torque limit is applied internally to the servo unit, as described in "Torque limit in undervoltage".					

Object 2047h: Delay time of torque limit release while main circuit voltage drops(ms)					
index	2047h				
name	Delay time of torque limit release while main circuit voltage drops(ms)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~1000
Can map	NO	**Accessibility**	RW	**Factory default**	100
After the undervoltage warning signal is released, the torque limit value is controlled internally by the servo unit according to the set time, as described in "Main circuit undervoltage torque limit".					

Object 2050h: Torque limit method setting					
index	2050h				
name	Torque limit method setting				
Object structure	VAR	**type of data**	Uint16	**data range**	0~3
Can map	NO	**Accessibility**	RW	**Factory default**	1
0-Analog torque (torque mode invalid) 1-Maximum torque limit 1 2-Maximum torque limit for forward rotation 1; Maximum torque limit for reverse rotation 2 3-Maximum torque limit 1 when the switch "torque limit switch" is OFF; Maximum torque limit 2 when it is ON					

Object 2051h: Max torque limitation 1					
index	2051h				
name	Max torque limitation 1				
Object structure	VAR	**type of data**	Uint16	**data range**	0~500
Can map	NO	**Accessibility**	RW	**Factory default**	500
In 1%, effective immediately					

Object 2052h: Max torque limitation 2					
index	2052h				
name	Max torque limitation 2				
Object structure	VAR	**type of data**	Uint16	**data range**	0~500
Can map	NO	**Accessibility**	RW	**Factory default**	500
In 1%, effective immediately					

Object 2053h: Emergency stop torque					
index	2053h				
name	Emergency stop torque				
Object structure	VAR	**type of data**	Uint16	**data range**	0~800
Can map	NO	**Accessibility**	RW	**Factory default**	800
The unit is 1% and takes effect immediately. Set the motor stop method to set the torque at deceleration stop.					

Object 2070h: Encoder frequency-dividing pulse					
index	2070h				
name	Encoder frequency-dividing pulse				
Object structure	VAR	**type of data**	Uint16	**data range**	16~4194304
Can map	NO	**Accessibility**	RW	**Factory default**	2048
The number of pulses per revolution from the encoder is divided according to this parameter setting, which is set according to the system specifications of the machine and the host device. Note: The set value is the number of A/B quadrature output pulses per revolution. The number of divided pulses from the encoder is limited by the resolution of the encoder, as described in "Divided Pulse Output Setting".					

Object 2072h: Reverse direction of frequency division output					
index	2072h				
name	Reverse direction of frequency division output				
Object structure	VAR	**type of data**	Uint16	**data range**	0~1
Can map	NO	**Accessibility**	RW	**Factory default**	0
Set the A/B pulse phase sequence logic for forward and reverse rotation. 0-Pulse output is not reversed: A overtakes B in forward rotation 1-Pulse output is reversed: When forward rotation, B overtakes A					

Object 2100h: Ratio of moments of inertia(1%)					
index	2100h				
name	Ratio of moments of inertia(1%)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~20000
Can map	NO	**Accessibility**	RW	**Factory default**	100
Rotational inertia ratio = load rotational inertia converted from motor shaft / rotational inertia of servo motor * 100%					

Object 2101h: 1st speed gain(0.1Hz)					
index	2101h				
name	1st speed gain(0.1Hz)				
Object structure	VAR	**type of data**	Uint16	**data range**	10~20000
Can map	NO	**Accessibility**	RW	**Factory default**	400
The parameter that determines the responsiveness of the velocity ring. Since the lower responsiveness of the velocity loop will become the delay element of the outer position loop, overshoot or vibration of the velocity command will occur. Within the range of the mechanical system without vibration, the larger the setting value, the more stable the servo system is and the better the responsiveness.					

Object 2102h: 1st Velocity-time integral(0.01ms)					
index	2102h				
name	1st Velocity-time integral(0.01ms)				
Object structure	VAR	**type of data**	Uint16	**data range**	15~51200
Can map	NO	**Accessibility**	RW	**Factory default**	2000
In order to respond to even small inputs, the velocity loop contains an integral element. Since this integral element is a delay element for the servo system, when the time parameter is set too large, overshoot occurs, or the positioning time is extended, making the response poor.					

Object 2103h: 1st position gain(0.1/s)					
index	2103h				
name	1st position gain(0.1/s)				
Object structure	VAR	**type of data**	Uint16	**data range**	10~20000
Can map	NO	**Accessibility**	RW	**Factory default**	400
The responsiveness of the position ring is determined by the position ring gain. The higher the setting of the position loop gain, the higher the responsiveness and the shorter the positioning time. The position loop gain cannot be increased beyond the mechanical system rigidity, and to set the position loop gain to a larger value, the machine rigidity needs to be increased.					

Object 2104h: 1st Torque command filter					
index	2104h				
name	1st Torque command filter				
Object structure	VAR	**type of data**	Uint16	**data range**	0~65535
Can map	NO	**Accessibility**	RW	**Factory default**	100
The smaller the value, the more responsive the control can be, subject to machine conditions.					

Object 2105h: 2nd speed gain(0.1Hz)					
index	2105h				
name	2nd speed gain(0.1Hz)				
Object structure	VAR	**type of data**	Uint16	**data range**	10~20000
Can map	NO	**Accessibility**	RW	**Factory default**	400
The parameter that determines the responsiveness of the velocity ring. Since the lower responsiveness of the velocity loop will become the delay element of the outer position loop, overshoot or vibration of the velocity command will occur. Within the range of the mechanical system without vibration, the larger the setting value, the more stable the servo system is and the better the responsiveness.					

Object 2106h: 2nd Velocity-time integral(0.01ms)					
index	2106h				
name	2nd Velocity-time integral(0.01ms)				
Object structure	VAR	**type of data**	Uint16	**data range**	15~51200
Can map	NO	**Accessibility**	RW	**Factory default**	2000
The parameter that determines the responsiveness of the velocity ring. Since the lower responsiveness of the velocity loop will become the delay element of the outer position loop, overshoot or vibration of the velocity command will occur. Within the range of the mechanical system without vibration, the larger the setting value, the more stable the servo system is and the better the responsiveness.					

Object 2107h: 2nd position gain(0.1/s)					
index	2107h				
name	2nd position gain(0.1/s)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~65535
Can map	NO	**Accessibility**	RW	**Factory default**	100
The responsiveness of the position ring is determined by the position ring gain. The higher the setting of the position loop gain, the higher the responsiveness and the shorter the positioning time. The position loop gain cannot be					

increased beyond the mechanical system rigidity, and to set the position loop gain to a larger value, the machine rigidity needs to be increased.

Object 2108h: 2nd position gain(0.1/s)					
index	2108h				
name	2nd position gain(0.1/s)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~65535
Can map	NO	**Accessibility**	RW	**Factory default**	100
The responsiveness of the position ring is determined by the position ring gain. The higher the setting of the position loop gain, the higher the responsiveness and the shorter the positioning time. The position loop gain cannot be increased beyond the mechanical system rigidity, and to set the position loop gain to a larger value, the machine rigidity needs to be increased.					

Object 2110h: The gain switching function					
index	2110h				
name	The gain switching function				
Object structure	VAR	**type of data**	Uint16	**data range**	0~1
Can map	NO	**Accessibility**	RW	**Factory default**	0
There are two types of gain switching function, "Manual gain switching" which uses external input signal and "Automatic gain switching" which switches automatically. The gain switching function can be used to increase the gain and shorten the positioning time during positioning and to reduce the gain and suppress vibration when the motor stops. 0 - Manual gain switching is performed by manually switching the external input signal (G-SEL). 1 - When the automatic switching condition is established (Pn111), the gain is automatically switched from the first gain to the second gain, otherwise, the gain is switched back to the first gain.					

Object 2111h: Automatic switching condition of position control gain					
index	2111h				
name	Automatic switching condition of position control gain				
Object structure	VAR	**type of data**	Uint16	**data range**	0~5
Can map	NO	**Accessibility**	RW	**Factory default**	0
0-Positioning completion signal ON 1-Positioning completion signal OFF 2-Positioning approach signal ON 3-Positioning proximity signal OFF 4-Position command filtered to 0 and pulse input OFF 5-Position command pulse input ON					

If the condition is satisfied, then switch to the 2nd gain, otherwise switch to the 1st gain.

Object 2112h: Automatic switching condition of position control gain					
index	2112h				
name	Automatic switching condition of position control gain				
Object structure	VAR	**type of data**	Uint16	**data range**	0~65535
Can map	NO	**Accessibility**	RW	**Factory default**	0
After waiting for a wait time from the time when the switching condition has been established, the position 1 loop gain is changed linearly to the position 2 loop gain during the switching transition time.					

Object 2113h: Gain switching transition time 2(ms)					
index	2113h				
name	Gain switching transition time 2(ms)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~65535
Can map	NO	**Accessibility**	RW	**Factory default**	0
After waiting for a wait time from the time when the switching condition has been established, the 2nd position loop gain is changed linearly to the 1st position loop gain during the switching transition time.					

Object 2114h: Gain switching delay time 1(ms)					
index	2114h				
name	Gain switching delay time 1(ms)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~65535
Can map	NO	**Accessibility**	RW	**Factory default**	0
The time to wait between when the first gain to second gain switching condition is established and when the switching actually starts.					

Object 2115h: Gain switching delay time 2(ms)					
index	2115h				
name	Gain switching delay time 2(ms)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~65535
Can map	NO	**Accessibility**	RW	**Factory default**	0
The time to wait between when the second gain to first gain switching condition is established and when the switching actually starts.					

Object 2121h: Speed feedforward gain(1%)					
index	2121h				
name	Speed feedforward gain(1%)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~100
Can map	NO	**Accessibility**	RW	**Factory default**	0
The speed feedforward is a function to shorten the positioning time, and it is effective when the servo unit performs position control. Note: If the feedforward command is too large, position overshoot will occur, so please observe the response while making appropriate settings.					

Object 2122h: Speed feedforward filter time(0.01ms)					
index	2122h				
name	Speed feedforward filter time(0.01ms)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~6400
Can map	NO	**Accessibility**	RW	**Factory default**	0
Speed feedforward low-pass filtering time constants mitigate position overshoot and torque runout caused by feedforward.					

Object 2123h: Use V-REF as the speed feedforward selection					
index	2123h				
name	Use V-REF as the speed feedforward selection				
Object structure	VAR	**type of data**	Uint16	**data range**	0~1
Can map	NO	**Accessibility**	RW	**Factory default**	0
Velocity feedforward is a function to shorten the positioning time, optionally giving velocity feedforward via external analog V-REF. 0-None 1 - Use V-REF as speed feed forward input					

Object 2124h: Speed/Position control selection (T-REF distribution)					
index	2124h				
name	Speed/Position control selection (T-REF distribution)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~1
Can map	NO	**Accessibility**	RW	**Factory default**	0
Torque feedforward is a function to shorten the positioning time, and optionally the torque feedforward can be given via the external analog T-REF.					

0-None 1 - Use T-REF as torque feed forward input

Object 2125h: Time constant of speed feedback filtering(0.01ms)					
index	2125h				
name	Time constant of speed feedback filtering(0.01ms)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~65535
Can map	NO	**Accessibility**	RW	**Factory default**	0

Object 2126h: Time constant of speed feedback moving average filtering(125us)					
index	2126h				
name	Time constant of speed feedback moving average filtering(125us)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~0x14
Can map	NO	**Accessibility**	RW	**Factory default**	0

Object 2130h: Speed loop control method (PI/IP)					
index	2130h				
name	Speed loop control method (PI/IP)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~1
Can map	NO	**Accessibility**	RW	**Factory default**	0
0-PI control 1-I-P control					

Object 2131h: The selection switch of speed loop P/PI switch condition					
index	2131h				
name	The selection switch of speed loop P/PI switch condition				
Object structure	VAR	**type of data**	Uint16	**data range**	0~4
Can map	NO	**Accessibility**	RW	**Factory default**	0
The mode switch is a function to automatically switch between P control and PI control. By setting the switching conditions with this parameter, the overshoot during acceleration and deceleration is suppressed and the stabilization time is shortened when the corresponding switching condition value is met. 0-Conditional on internal torque command 1-Conditional on the speed command					

2-Conditional on acceleration 3-Conditional on position deviation pulse 4-No mode switching function

Object 2132h: Speed loop P/PI switching condition (torque command)(1%)					
index	2132h				
name	Speed loop P/PI switching condition (torque command)(1%)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~800
Can map	NO	**Accessibility**	RW	**Factory default**	200
If the torque command exceeds the torque set by this parameter, the speed loop will switch to P control, otherwise it is PI control.					

Object 2133h: Speed loop P/PI switching condition (speed command)(rpm)					
index	2133h				
name	Speed loop P/PI switching condition (speed command)(rpm)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~10000
Can map	NO	**Accessibility**	RW	**Factory default**	0
When the speed command exceeds the speed set by this parameter, the speed loop will switch to P control, otherwise it is PI control.					

Object 2134h: Speed loop P/PI switching condition (acceleration speed)(rpm/s)					
index	2134h				
name	Speed loop P/PI switching condition (acceleration speed)(rpm/s)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~30000
Can map	NO	**Accessibility**	RW	**Factory default**	0
When the speed command exceeds the speed set by this parameter, the speed loop will switch to P control, otherwise it is PI control.					

Object 2135h: Speed loop P/PI switching condition (position deviation)(Command unit)					
index	2135h				
name	Speed loop P/PI switching condition (position deviation)(Command unit)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~10000
Can map	NO	**Accessibility**	RW	**Factory default**	0
If the position deviation exceeds the value set in this parameter, the speed loop will switch to P control, otherwise PI control.					

Object 2140h: Medium frequency vibration suppression control selection					
index	2140h				
name	Medium frequency vibration suppression control selection				
Object structure	VAR	**type of data**	Uint16	**data range**	0~0x11
Can map	NO	**Accessibility**	RW	**Factory default**	0x10
The mid-frequency vibration suppression control function can effectively suppress the continuous vibration around 100 to 1000Hz that occurs when the control gain is increased. 0x1#:Automatic setting of IF vibration suppression frequency by intelligent setting and bandwidth setting 0x0#:Not set automatically by intelligent setting, bandwidth setting, only can be set manually 0x#1:Medium frequency vibration suppression frequency setting is effective 0x#0: IF suppression frequency setting is invalid					

Object 2141h: Inertial modification of intermediate frequency vibration suppression(1%)					
index	2141h				
name	Inertial modification of intermediate frequency vibration suppression(1%)				
Object structure	VAR	**type of data**	Uint16	**data range**	1~1000
Can map	NO	**Accessibility**	RW	**Factory default**	100

Object 2142h: First medium frequency vibration suppression frequency(0.1Hz)					
index	2142h				
name	First medium frequency vibration suppression frequency(0.1Hz)				
Object structure	VAR	**type of data**	Uint16	**data range**	10~30000
Can map	NO	**Accessibility**	RW	**Factory default**	1000
Set the IF suppression frequency value.					

Object 2143h: First medium frequency vibration suppression attenuation gain(1%)					
index	2143h				
name	First medium frequency vibration suppression attenuation gain(1%)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~300
Can map	NO	**Accessibility**	RW	**Factory default**	0
Increase this parameter can improve the vibration suppression effect, but too large a setting may increase the vibration, please check the vibration suppression effect while gradually increasing the setting value by 10% in the					

range of 0% to 200%, and if the vibration suppression effect is still not obtained after reaching 200%, please stop the setting and reduce the control gain appropriately.

Object 2150h: Notch filter 1 automatic adjustment selection					
index	2150h				
name	Notch filter 1 automatic adjustment selection				
Object structure	VAR	**type of data**	Uint16	**data range**	0~1
Can map	NO	**Accessibility**	RW	**Factory default**	1
0-Automatic adjustment without auxiliary function 1-Automatic adjustment by auxiliary function					

Object 2151h: Notch filter 2 automatic adjustment selection					
index	2151h				
name	Notch filter 2 automatic adjustment selection				
Object structure	VAR	**type of data**	Uint16	**data range**	0~1
Can map	NO	**Accessibility**	RW	**Factory default**	1
0-Automatic adjustment without auxiliary function 1-Automatic adjustment by auxiliary function					

Object 2152h: Automatic notch resonance detection sensitivity(1%)					
index	2152h				
name	Automatic notch resonance detection sensitivity(1%)				
Object structure	VAR	**type of data**	Uint16	**data range**	1~200
Can map	NO	**Accessibility**	RW	**Factory default**	100
Used to set the sensitivity of automatic detection of resonance frequency. When the setting value is smaller, the more sensitive to resonance, the easier to detect the vibration, too small may be false detection of resonance frequency.					

Object 2153h: Notch filter 1 frequency(Hz)					
index	2153h				
name	Notch filter 1 frequency(Hz)				
Object structure	VAR	**type of data**	Uint16	**data range**	50~5000
Can map	NO	**Accessibility**	RW	**Factory default**	5000
Set the frequency of the 2nd trap filter for resonance suppression. When this parameter is set to 5000, the function of the trap filter is disabled.					

Note: Do not set the trap filter frequency close to the response frequency of the speed loop, but at least 4 times the gain of the speed loop. Otherwise, the overall system performance may be affected.

Object 2154h: Notch filter 1 Q value(0.01)					
index	2154h				
name	Notch filter 1 Q value(0.01)				
Object structure	VAR	**type of data**	Uint16	**data range**	50~1000
Can map	NO	**Accessibility**	RW	**Factory default**	70
The trap filter Q value is the set value that determines the width of the filter frequency relative to the trap filter frequency. The width of the depression varies depending on the trap filter Q value; the larger the value of the trap filter Q value, the more powerful the depression and the narrower the width of the filter frequency.					

Object 2155h: Notch filter 1 depth(0.001)					
index	2155h				
name	Notch filter 1 depth(0.001)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~1000
Can map	NO	**Accessibility**	RW	**Factory default**	0
The trap filter depth is the set value that determines the depth of the filter frequency relative to the trap filter frequency. The depth of the depression varies depending on the depth of the trap filter. The smaller the value of the trap filter depth, the deeper the depression and the higher the vibration suppression effect. But too small will increase the vibration.					

Object 2156h: Notch filter 2 frequency(Hz)					
index	2156h				
name	Notch filter 2 frequency(Hz)				
Object structure	VAR	**type of data**	Uint16	**data range**	50~5000
Can map	NO	**Accessibility**	RW	**Factory default**	5000
The trap filter Q value is the set value that determines the width of the filter frequency relative to the trap filter frequency. The width of the depression varies depending on the trap filter Q value; the larger the value of the trap filter Q value, the more powerful the depression and the narrower the width of the filter frequency.					

Object 2157h: Notch filter 2 Q value(0.01)					
index	2157h				
name	Notch filter 2 Q value(0.01)				
Object structure	VAR	**type of data**	Uint16	**data range**	50~1000

Can map	NO	**Accessibility**	RW	**Factory default**	70
The trap filter Q value is the set value that determines the width of the filter frequency relative to the trap filter frequency. The width of the depression varies depending on the trap filter Q value; the larger the value of the trap filter Q value, the more powerful the depression and the narrower the width of the filter frequency.					

Object 2158h: Notch filter 2 depth(0.001)					
index	2158h				
name	Notch filter 2 depth(0.001)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~1000
Can map	NO	**Accessibility**	RW	**Factory default**	0
The trap filter depth is the set value that determines the depth of the filter frequency relative to the trap filter frequency. The depth of the depression varies depending on the depth of the trap filter. The smaller the value of the trap filter depth, the deeper the depression, the higher the vibration suppression effect, but too small will increase the vibration.					

Object 2160h: Disturbance compensation function selection					
index	2160h				
name	Disturbance compensation function selection				
Object structure	VAR	**type of data**	Uint16	**data range**	0~1
Can map	NO	**Accessibility**	RW	**Factory default**	0
Set the disturbance compensation function switch. 0-No use 1-Use					

Object 2161h: Disturbance observer cutoff frequency(0.1Hz)					
index	2161h				
name	Disturbance observer cutoff frequency(0.1Hz)				
Object structure	VAR	**type of data**	Uint16	**data range**	10~10000
Can map	NO	**Accessibility**	RW	**Factory default**	1500
Set the disturbance compensation gain, increase can improve the effect of suppressing the effect of disturbance, but too much noise will become larger.					

Object 2163h: Disturbance observer cutoff frequency(0.1Hz)	
index	2163h
name	Disturbance observer cutoff frequency(0.1Hz)

Object structure	VAR	**type of data**	Uint16	**data range**	0~100
Can map	NO	**Accessibility**	RW	**Factory default**	0
Set the disturbance compensation coefficient, set the received position command or speed command, and add the disturbance torque compensation value to the torque command.					

Object 2165h: Disturbance observer inertia correction coefficient(1%)					
index	2165h				
name	Disturbance observer inertia correction coefficient(1%)				
Object structure	VAR	**type of data**	Uint16	**data range**	1~1000
Can map	NO	**Accessibility**	RW	**Factory default**	100
This parameter sets the disturbance observer inertia and is used to adjust the discrimination error caused by inaccurate inertia settings. Note: This value is set to 100 when the inertia ratio is set correctly.					

Object 2166h: Speed observer switch					
index	2166h				
name	Speed observer switch				
Object structure	VAR	**type of data**	Uint16	**data range**	0~1
Can map	NO	**Accessibility**	RW	**Factory default**	0
Set the speed observation function switch. 0-Invalid 1-Effective					

Object 2167h: Speed observer cutoff frequency(Hz)					
index	2167h				
name	Speed observer cutoff frequency(Hz)				
Object structure	VAR	**type of data**	Uint16	**data range**	1~500
Can map	NO	**Accessibility**	RW	**Factory default**	80
Set the speed observer bandwidth by this parameter, increasing the set value will improve the response speed of the speed feedback value tracking the real speed, too large is prone to vibration and noise.					

Object 2168h: Inertial correction coefficient in speed observer(1%)	
index	2168h
name	Inertial correction coefficient in speed observer(1%)

Object structure	VAR	**type of data**	Uint16	**data range**	0~10000
Can map	NO	**Accessibility**	RW	**Factory default**	100

Object 2169h: Inertial correction coefficient in speed observer(1%)					
index	2169h				
name	Inertial correction coefficient in speed observer(1%)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~65535
Can map	NO	**Accessibility**	RW	**Factory default**	10

Object 2170h: Friction torque compensation cutoff speed(rpm)					
index	2170h				
name	Friction torque compensation cutoff speed(rpm)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~1000
Can map	NO	**Accessibility**	RW	**Factory default**	20
The friction compensation function is a function to complement the variation of viscous friction and fixed load. It is adjusted with the friction compensation coefficient. Usually, the friction compensation coefficient should be set below 95%.					

Object 2171h: Friction torque forward compensation coefficient(1%/100rpm)					
index	2171h				
name	Friction torque forward compensation coefficient(1%/100rpm)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~100
Can map	NO	**Accessibility**	RW	**Factory default**	0
The friction compensation function is a function to complement the variation of viscous friction and fixed load. It is adjusted with the friction compensation coefficient. Usually, the friction compensation coefficient should be set below 95%.					

Object 2172h: Friction torque reverse compensation coefficient(1%/100rpm)	
index	2172h
name	Friction torque reverse compensation coefficient(1%/100rpm)

Object structure	VAR	**type of data**	Uint16	**data range**	0~100
Can map	NO	**Accessibility**	RW	**Factory default**	0
The higher the setting value, the better the result, but the higher the setting value, the more likely the response will vibrate.					

Object 2175h: Robust control options					
index	2175h				
name	Robust control options				
Object structure	VAR	**type of data**	Uint16	**data range**	0~1
Can map	NO	**Accessibility**	RW	**Factory default**	0
The robust control function is a function that can obtain a stable response by automatic adjustment within a certain range, regardless of mechanical type and load fluctuation and inertia change. Setting the robust control function switch. 0-Invalid 1-Effective					

Object 2177h: Robust control tuning values(0.1Hz)					
index	2177h				
name	Robust control tuning values(0.1Hz)				
Object structure	VAR	**type of data**	Uint16	**data range**	100~800
Can map	NO	**Accessibility**	RW	**Factory default**	400
Set the gain tuning value for robust control. The larger the value is set, the faster the system response, but there may be system overshoot and excessive noise.					

Object 2178h: Robust control tuning values(0.1Hz)					
index	2178h				
name	Robust control tuning values(0.1Hz)				
Object structure	VAR	**type of data**	Uint16	**data range**	100~800
Can map	NO	**Accessibility**	RW	**Factory default**	400
Set the gain tuning value for robust control. The larger the value is set, the faster the system response, but there may be system overshoot and excessive noise.					

Object 2185h: vibration detection selection					
index	2185h				
name	vibration detection selection				
Object structure	VAR	**type of data**	Uint16	**data range**	0~2
Can map	NO	**Accessibility**	RW	**Factory default**	0
This function is to be able to detect machine vibration in the normal operation state after the detection value can automatically detect the function of the relevant alarm or warning. Set the vibration detection after the performance mode. 0-No vibration detection 1-Warning after vibration detection 2 - Alarm after vibration detection					

Object 2186h: Vibration detection sensitivity(1%)					
index	2186h				
name	Vibration detection sensitivity(1%)				
Object structure	VAR	**type of data**	Uint16	**data range**	50~500
Can map	NO	**Accessibility**	RW	**Factory default**	100
Set the sensitivity of vibration detection, the smaller the setting the more sensitive it is, too small a setting may detect vibration incorrectly during normal operation. Note: Depending on the state of the machinery used, the detection sensitivity of vibration alarm and vibration warning may vary.					

Object 2187h: Vibration detection value(rpm）					
index	2187h				
name	Vibration detection value(rpm）				
Object structure	VAR	**type of data**	Uint16	**data range**	0~5000
Can map	NO	**Accessibility**	RW	**Factory default**	50
Set the threshold value for vibration detection. The smaller the setting, the easier it is to detect vibration, and too small a setting may falsely detect vibration during normal operation. Note: Depending on the state of the machinery used, the vibration detection values for vibration alarms and vibration warnings may vary.					

Object 2192h: Detection sensitivity (relative positioning completed) during position overshoot in advance tuning(1%)

index	2192h				
name	Detection sensitivity (relative positioning completed) during position overshoot in advance tuning(1%)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~100
Can map	NO	**Accessibility**	RW	**Factory default**	100

Object 2193h: Maximum gain setting value when searching for maximum gain(0.1Hz)					
index	2193h				
name	Maximum gain setting value when searching for maximum gain(0.1Hz)				
Object structure	VAR	**type of data**	Uint16	**data range**	10~4000
Can map	NO	**Accessibility**	RW	**Factory default**	3000

Object 2203h: Command pulse input rate					
index	2203h				
name	Command pulse input rate				
Object structure	VAR	**type of data**	Uint16	**data range**	0~100
Can map	NO	**Accessibility**	RW	**Factory default**	1
Set the command pulse input multiplier value, and use it with the ON/OFF of the command pulse multiplier switch signal to switch the input multiplier of position command pulse to 1 times and the multiplier set by this parameter. Note: If the input pulse frequency is too low and the value is set too large, unstable speed may occur.					

Object 2204h: Electronic gear ratio numerator					
index	2204h				
name	Electronic gear ratio numerator				
Object structure	VAR	**type of data**	Uint32	**data range**	0~1073741824
Can map	NO	**Accessibility**	RW	**Factory default**	1
Power down effective					

Object 2206h: Electronic gear ratio denominator					
index	2206h				
name	Electronic gear ratio denominator				
Object structure	VAR	**type of data**	Uint32	**data range**	1~1073741824

Can map	NO	**Accessibility**	RW	**Factory default**	1
Power down effective					

Object 2211h: position command low-pass filtering time(0.1ms)					
index	2211h				
name	position command low-pass filtering time(0.1ms)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~6550
Can map	NO	**Accessibility**	RW	**Factory default**	0
This parameter sets the time constant of the first-order low-pass filter corresponding to the position command. Setting this parameter reduces mechanical shocks in case of sudden changes in the frequency of the input pulse command.					

Object 2212h: position command slide filtering time(0.1ms)					
index	2212h				
name	position command slide filtering time(0.1ms)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~10000
Can map	NO	**Accessibility**	RW	**Factory default**	0
This parameter sets the time constant of the sliding average filter for the corresponding position command. Setting this parameter reduces mechanical shocks in case of sudden changes in the frequency of the input pulse command.					

Object 2230h: Low frequency vibration suppression selection					
index	2230h				
name	Low frequency vibration suppression selection				
Object structure	VAR	**type of data**	Uint16	**data range**	0~2
Can map	NO	**Accessibility**	RW	**Factory default**	0
This parameter is used in conjunction with Pn231 and is set as an auto tuning method in the case of auto tuning through. 0-no vibration suppression 1-Additional vibration suppression for specific frequencies 2-Additional vibration suppression for 2 different frequencies					

Object 2231h: Low frequency vibration suppression function automatic tuning selection					
index	2231h				
name	Low frequency vibration suppression function automatic tuning selection				
Object structure	VAR	**type of data**	Uint16	**data range**	0~1

Can map	NO	**Accessibility**	RW	**Factory default**	1
This parameter sets the selection of whether low frequency vibration suppression is automatically set under the auxiliary functions such as intelligent setting and bandwidth setting. 0-Vibration suppression function is not automatically adjusted by auxiliary function 1-Vibration suppression function is automatically adjusted by the auxiliary function					

Object 2232h: Low-frequency vibration detection sensitivity (relative positioning is completed)(0.1%)					
index	2232h				
name	Low-frequency vibration detection sensitivity (relative positioning is completed)(0.1%)				
Object structure	VAR	**type of data**	Uint16	**data range**	1~3000
Can map	NO	**Accessibility**	RW	**Factory default**	400
This parameter sets the sensitivity of low-frequency vibration detection when positioning is completed. The smaller the sensitivity setting, the easier it is to automatically detect low-frequency vibration frequency points.					

Object 2235h: Low frequency vibration suppression 1 frequency(0.1Hz)					
index	2235h				
name	Low frequency vibration suppression 1 frequency(0.1Hz)				
Object structure	VAR	**type of data**	Uint16	**data range**	10~2000
Can map	NO	**Accessibility**	RW	**Factory default**	2000
This parameter is used to set the frequency of Low Frequency Vibration Suppression 1.					

Object 2236h: Low frequency vibration suppression 1 correction(1%)					
index	2236h				
name	Low frequency vibration suppression 1 correction(1%)				
Object structure	VAR	**type of data**	Uint16	**data range**	10~1000
Can map	NO	**Accessibility**	RW	**Factory default**	100
This parameter is used to set the correction factor of low frequency vibration suppression 1. The larger the value is set, the more obvious the low frequency suppression effect is, too small may lead to too long positioning time.					

Object 2237h: Low frequency vibration suppression 2 frequency(0.1Hz)					
index	2237h				
name	Low frequency vibration suppression 2 frequency(0.1Hz)				
Object structure	VAR	**type of data**	Uint16	**data range**	10~2000
Can map	NO	**Accessibility**	RW	**Factory default**	2000

This parameter is used to set the frequency of Low Frequency Vibration Suppression 2.

Object 2238h: Low frequency vibration suppression 2 correction(1%)					
index	2238h				
name	Low frequency vibration suppression 2 correction(1%)				
Object structure	VAR	**type of data**	Uint16	**data range**	10~1000
Can map	NO	**Accessibility**	RW	**Factory default**	100
This parameter is used to set the correction factor of low-frequency vibration suppression 2. The larger the value is set, the more obvious the low-frequency suppression effect is, while too small may lead to too long positioning time.					

Object 2240h: Model tracking control selection					
index	2240h				
name	Model tracking control selection				
Object structure	VAR	**type of data**	Uint16	**data range**	0~1
Can map	NO	**Accessibility**	RW	**Factory default**	0
Model tracking control is a function dedicated to position positioning, Model tracking control selector switch. 0-No model tracking control 1 - Use model tracking control					

Object 2241h: Gain of model tracking control(0.1/s)					
index	2241h				
name	Gain of model tracking control(0.1/s)				
Object structure	VAR	**type of data**	Uint16	**data range**	10~20000
Can map	NO	**Accessibility**	RW	**Factory default**	500
The size of the model tracking control gain determines how fast the servo system responds. By increasing the model tracking control gain, the responsiveness becomes faster and the positioning time becomes shorter. When the model tracking control is effective, the position response, deviation of the servo system is determined by this parameter, not the position gain.					

Object 2242h: Attenuation coefficient of model tracking control(0.1%)					
index	2242h				
name	Attenuation coefficient of model tracking control(0.1%)				
Object structure	VAR	**type of data**	Uint16	**data range**	500~2000
Can map	NO	**Accessibility**	RW	**Factory default**	1000

When the model tracking attenuation coefficient decreases, the position rectification section is prone to overshoot overshoot, and when the setting is too small, the position overshoot is reduced, but when it is too large, the position is prone to rebound, resulting in longer positioning time, and it is recommended to keep the value at 100 for general use.

Object 2243h: Speed forwardback gain of model tracking(0.1%)					
index	2243h				
name	Speed forwardback gain of model tracking(0.1%)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~10000
Can map	NO	**Accessibility**	RW	**Factory default**	1000
Model tracking speed feedforward gain decreases and responsiveness becomes slower, but is not prone to position overshoot and overshoot, too small tends to lead to long closing time of position deviation.					

Object 2244h: Forward torque feedforward gain of model tracking control(0.1%)					
index	2244h				
name	Forward torque feedforward gain of model tracking control(0.1%)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~10000
Can map	NO	**Accessibility**	RW	**Factory default**	1000
Positive position command, use this parameter when adjusting the positive response alone, when adjusting larger, the torque feedforward rises faster and the positioning time can be shortened appropriately.					

Object 2245h: Speed feedforward gain of model tracking control(0.1%)					
index	2245h				
name	Speed feedforward gain of model tracking control(0.1%)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~10000
Can map	NO	**Accessibility**	RW	**Factory default**	1000
Negative position command, use this parameter when adjusting the negative response alone, when adjusting larger, the torque feedforward rises faster and the positioning time can be shortened appropriately.					

Object 2246h: Gain of 2nd model tracking control(0.1/s)					
index	2246h				
name	Gain of 2nd model tracking control(0.1/s)				
Object structure	VAR	**type of data**	Uint16	**data range**	10~20000
Can map	NO	**Accessibility**	RW	**Factory default**	100
Used at gain 2 when model tracking is valid.					

Object 2247h: attenuation coefficient of model tracking control(0.1%)					
index	2247h				
name	attenuation coefficient of model tracking control(0.1%)				
Object structure	VAR	**type of data**	Uint16	**data range**	500~2000
Can map	NO	**Accessibility**	RW	**Factory default**	1000
Used at gain 2 when model tracking is valid.					

Object 2248h: Model tracking control type selection					
index	2248h				
name	Model tracking control type selection				
Object structure	VAR	**type of data**	Uint16	**data range**	0~1
Can map	NO	**Accessibility**	RW	**Factory default**	1

Object 2249h: Speed feedforward (VFF) / torque feedforward selection)					
index	2249h				
name	Speed feedforward (VFF) / torque feedforward selection)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~1
Can map	NO	**Accessibility**	RW	**Factory default**	0
0-No simultaneous use of model tracking control and external speed and torque feedforward 1 - Simultaneous use of model tracking control and external speed and torque feedforward When using model tracking control, the optimal feedforward is set inside the servo, and it is usually not recommended to use both the "velocity feedforward (V-REF) input" and "torque feedforward (T-REF) input" from the upper unit. However, it is possible to use both inputs as needed, when incorrect feedforward inputs may cause overshoot and system instability.					

Object 2250h: Instructions of external encoders under full closed loop control mode					
index	2250h				
name	Instructions of external encoders under full closed loop control mode				
Object structure	VAR	**type of data**	Uint16	**data range**	0~3
Can map	NO	**Accessibility**	RW	**Factory default**	0
0-No use of full closed loop function 1-Use in the standard running direction 2-Use in the reverse direction of operation					

Set the direction of scale movement when the motor is in forward rotation. Wrong setting of this direction may lead to flying car or alarm of excessive deviation between motor and load, so you can move the load manually before running and change this parameter to make the monitoring parameter Un007 (feedback pulse counter) and Un012 (external encoder feedback pulse counter) change in the same direction before running.

Object 2252h: Motor load division coefficient while tuning 1 cycle under full closed loop(1%)					
index	2252h				
name	Motor load division coefficient while tuning 1 cycle under full closed loop(1%)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~100
Can map	NO	**Accessibility**	RW	**Factory default**	20
Set the coefficient processing of the deviation between motor and load after 1 turn of motor operation, such as this parameter is set to 0%, the deviation is 1000 after 1 turn of operation, and the deviation is accumulated on the basis of 1000 at the beginning of the 2nd turn of operation, set to 20%, the deviation is accumulated on the basis of 200 (1000*20%=200) at the beginning of the 2nd turn of operation. This value is too large and may not be able to detect Er.d10 properly. It should be set according to the load and the allowable error between motor installation.					

Object 2253h: External grating ruler resolution(P/Rev)					
index	2253h				
name	External grating ruler resolution(P/Rev)				
Object structure	VAR	**type of data**	Uint32	**data range**	4~1048576
Can map	NO	**Accessibility**	RW	**Factory default**	32768
1 rotation of the motor, external scale resolution (after 4x frequency)					

Object 2257h: Large mixed deviation setting					
index	2257h				
name	Large mixed deviation setting				
Object structure	VAR	**type of data**	Uint32	**data range**	0~1073741824
Can map	NO	**Accessibility**	RW	**Factory default**	1000
Set the alarm threshold for excessive deviation between motor-load, the value is set too small, which may lead to false detection of Er.d10.					

Object 2260h: NEAR signal width					
index	2260h				
name	NEAR signal width				
Object structure	VAR	**type of data**	Uint32	**data range**	1~1073741824

Can map	NO	**Accessibility**	RW	**Factory default**	1073741824
In position control, the upper unit can receive the positioning proximity signal before confirming the positioning completion signal to prepare for the action sequence after positioning completion, which can shorten the time required for action at positioning completion. The signal is output when the difference between the number of command pulses of the upper unit and the amount of servo motor movement (position deviation) is lower than the set value.					

Object 2262h: Position arrival range					
index	2262h				
name	Position arrival range				
Object structure	VAR	**type of data**	Uint32	**data range**	1~1073741824
Can map	NO	**Accessibility**	RW	**Factory default**	7
When the difference between the number of command pulses from the upper unit and the amount of servo motor movement (position deviation) is lower than the set value of this parameter, the positioning completion signal is output for the upper unit to confirm that positioning has been completed.					

Object 2264h: Maximum tolerance value of position deviation					
index	2264h				
name	Maximum tolerance value of position deviation				
Object structure	VAR	**type of data**	Uint32	**data range**	1~1073741824
Can map	NO	**Accessibility**	RW	**Factory default**	25165824
When the motor action does not match the command, an abnormal condition can be detected and the motor can be stopped by setting an appropriate maximum threshold value for position deviation.					

Object 2266h: Large position deviation alarm setting(1%)					
index	2266h				
name	Large position deviation alarm setting(1%)				
Object structure	VAR	**type of data**	Uint16	**data range**	10~100
Can map	NO	**Accessibility**	RW	**Factory default**	100
This parameter sets the excessive position deviation warning threshold and generates an excessive position deviation warning when the position deviation is greater than the product of the "position deviation maximum threshold" and this parameter.					

Object 2267h: Alarm threshold of position deviation excess when servo is on	
index	2267h
name	Alarm threshold of position deviation excess when servo is on

Object structure	VAR	**type of data**	Uint32	**data range**	1~1073741823
Can map	NO	**Accessibility**	RW	**Factory default**	25165824
This parameter is used to set the threshold value for the Servo ON Instant Position Deviation Excessive Alarm. When the servo is ON, a position deviation value exceeding this setting value will generate the Servo ON Instant Position Deviation Excessive Alarm.					

Object 2269h: Warning threshold of large position deviation when servo is on(1%)					
index	2269h				
name	Warning threshold of large position deviation when servo is on(1%)				
Object structure	VAR	**type of data**	Uint16	**data range**	10~100
Can map	NO	**Accessibility**	RW	**Factory default**	100
This parameter sets the servo-on momentary excessive position deviation warning threshold. When the position deviation during servo-on is greater than the product of the "excessive position deviation warning threshold during servo-on" and this parameter, the excessive position deviation warning during servo-on is generated.					

Object 2270h: Speed limitation when servo on(rpm)					
index	2270h				
name	Speed limitation when servo on(rpm)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~10000
Can map	NO	**Accessibility**	RW	**Factory default**	10000
If the servo is ON in the state of position deviation accumulation, the speed limit is executed by this parameter. When the command pulse is input in this state, the alarm "Excessive position deviation caused by speed limit when servo ON alarm" is displayed when the set value of "Position deviation maximum threshold" is exceeded.					

Object 2274h: / COIN output time					
index	2274h				
name	/ COIN output time				
Object structure	VAR	**type of data**	Uint16	**data range**	0~2
Can map	NO	**Accessibility**	RW	**Factory default**	0
Set the output timing of the positioning completion signal/COIN. 0 - Output when the absolute value of position deviation is less than the position completion range (Pn262) 1 - Output when the absolute value of position deviation is less than the position completion range (Pn262) and the command after position command filtering is 0 2 - Output when the absolute value of position deviation is less than the position completion range (Pn262) and the position instruction input is 0					

Object 2310h: speed command acceleration time(ms)					
index	2310h				
name	speed command acceleration time(ms)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~10000
Can map	NO	**Accessibility**	RW	**Factory default**	0
The time required to accelerate the set speed from 0r/min to the maximum speed (corresponding to the motor model) is scaled to calculate the actual acceleration time when the given speed is greater or less than the maximum speed.					

Object 2311h: speed command deceleration time(ms)					
index	2311h				
name	speed command deceleration time(ms)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~10000
Can map	NO	**Accessibility**	RW	**Factory default**	0
The time required to set the maximum speed (corresponding to the motor model) to decelerate from 0r/min is calculated proportionally to the actual deceleration time when the given speed is greater or less than the maximum speed.					

Object 2312h: Zero-speed clamping mode					
index	2312h				
name	Zero-speed clamping mode				
Object structure	VAR	**type of data**	Uint16	**data range**	0~3
Can map	NO	**Accessibility**	RW	**Factory default**	3
Speed mode, operating mode when the switching zero speed clamping signal (/ZCLAMP) is set ON. 0-Invalid 1-Speed command set to 0, no clamping after shutdown 2-Speed command set to 0, clamping after stop 3-Speed command is set to 0 after the speed command is lower than the "zero speed clamping speed threshold", and then clamping after stopping					

Object 2313h: Zero-speed clamping Speed threshold(rpm)					
index	2313h				
name	Zero-speed clamping Speed threshold(rpm)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~10000
Can map	NO	**Accessibility**	RW	**Factory default**	10

Set the zero control switching threshold when "Zero speed clamping mode" is set to 3.

Object 2317h: Rotation judgment value(rpm)					
index	2317h				
name	Rotation judgment value(rpm)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~10000
Can map	NO	**Accessibility**	RW	**Factory default**	20
When the motor speed is above this setting value, the switching rotation detection signal (/TGON) is output.					

Object 2320h: Speed matching range(rpm)					
index	2320h				
name	Speed matching range(rpm)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~100
Can map	NO	**Accessibility**	RW	**Factory default**	10
If the difference between the motor speed and the commanded speed is below this setting, the switching speed agreement signal (/V-CMP) is output.					

Object 2401h: Torque command LP limiting frequency(Hz)					
index	2401h				
name	Torque command LP limiting frequency(Hz)				
Object structure	VAR	**type of data**	Uint16	**data range**	100~5000
Can map	NO	**Accessibility**	RW	**Factory default**	5000
This parameter is used to set the cutoff frequency of the second-order torque filter. When this parameter is set to 5000, the function of the filter is disabled.					

Object 2402h: Torque command LP Q(0.01)					
index	2402h				
name	Torque command LP Q(0.01)				
Object structure	VAR	**type of data**	Uint16	**data range**	50~100
Can map	NO	**Accessibility**	RW	**Factory default**	50
This parameter is used to set the Q value of the second-order torque filter. Increasing the Q value improves the system responsiveness, but noise is generated when the setting is too large.					

Object 2411h: Torque Cotrl speed limit method	
index	2411h

name	Torque Cotrl speed limit method				
Object structure	VAR	**type of data**	Uint16	**data range**	0~1
Can map	NO	**Accessibility**	RW	**Factory default**	1
0-Speed corresponding to the analog voltage (V-REF) and the speed set by Pn413, the smaller of the two 1-Speed set by Pn413					

Object 2412h: speed limit selection					
index	2412h				
name	speed limit selection				
Object structure	VAR	**type of data**	Uint16	**data range**	0~1
Can map	NO	**Accessibility**	RW	**Factory default**	0
0-Motor maximum speed (determined internally by the motor model) + Torque mode speed limit (Pn411) 1-Overspeed detection alarm speed (determined internally by the motor model) + Torque mode speed limit (Pn411)					

Object 2413h: Torque Cotrl speed limit(rpm)					
index	2413h				
name	Torque Cotrl speed limit(rpm)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~10000
Can map	NO	**Accessibility**	RW	**Factory default**	1000
Set the speed limit for torque control, used in conjunction with Pn411.					

Object 2600h: on-off input allocation model					
index	2600h				
name	on-off input allocation model				
Object structure	VAR	**type of data**	Uint16	**data range**	0~1
Can map	NO	**Accessibility**	RW	**Factory default**	1
Setting of the switching input signal assignment method. 0- Internal fixed: Use according to the pins and functions fixed inside the servo unit, see "CN1 terminal" input description for details 1-Parameter configuration: Use according to the function configured on each pin, configured by function codes Pn601~Pn609					

Object 2601h: CN1-40 input configuration **Object 2602h: CN1-42 input configuration** **Object 2603h: CN1-43 input configuration**

Object 2604h: CN1-41 input configuration **Object 2605h: CN1-44 input configuration** **Object 2606h: CN1-45 input configuration** **Object 2607h: CN1-46 input configuration**					
index	2601h、2602h、2603h、2604h、2605h、2606h、2607h				
name	input configuration				
Object structure	VAR	**type of data**	Uint16	**data range**	0~276
Can map	NO	**Accessibility**	RW	**Factory default**	1、258、259、5、4、6、7

0x00:Invalid	
0x01:Servo enable	0x101:Servo enable reversed
0x02:Positive limit	0x102:Positive limit release
0x03:Negative limit	0x103:Negative limit release
0x04:Alarm clearing	0x104:Alarm clearing reversed
0x05:Manual PI-P control	0x105:Manual PI-P control reversed
0x06:Torque limit switching	0x106:Torque limit switching reversed
0x07:Reserved	0x107:Reserved
0x08:Internal speed command direction selection	0x108:Internal speed command direction selection D reversed
0x09:Internal speed command selection A	0x109:Internal speed command selection A reversed
0x0A:Internal speed instruction selection B	0x10A:Internal speed instruction selection B inverted
0x0B:Control mode switch	0x10B:Control mode switch inverse
0x0C:Zero speed clamping	0x10C:Zero speed clamping reversed
0x0D:Command pulse disable	0x10D:Command pulse disable inverse
0x0E:Gain switching	0x10E:Gain switching inverse
0x0F:Torque command direction selection	0x10F:Torque command direction selection is reversed
0x10:Command pulse multiplier switching	0x110:Command pulse multiplier switching inverse

Object 2610h: on-off input Interior Options 1 **Object 2611h: on-off input Interior Options 2** **Object 2612h: on-off input Interior Options 3**					
index	2610h、2611h、2612h				
name	on-off input Interior Options				
Object structure	VAR	**type of data**	Uint16	**data range**	0~20
Can map	NO	**Accessibility**	RW	**Factory default**	0

0x00:Invalid	
0x01:Servo enable	0x02:Positive limit release
0x03:Negative limit release	0x04:Alarm clear
0x05:Manual PI-P control	0x06:Torque limit switching
0x08:Internal speed command direction selection	0x09:Internal speed command selection A
0x0A:Internal speed command selection B	0x0B:Control mode switching 0x0C:Zero speed clamping
0x0D:Pulse input disable	0x0E:Gain switching

0x0F:Torque command direction selection	0x10:Command pulse multiplier switching

Object 2613h: CN1-25/26 output configuration **Object 2614h: CN1-27/28 output configuration** **Object 2615h: CN1-29/30 output configuration**					
index	2613h、2614h、2615h				
name	output configuration				
Object structure	VAR	**type of data**	Uint16	**data range**	0~256
Can map	NO	**Accessibility**	RW	**Factory default**	0，1，2

0x00:Servo ready	0x100:Servo ready signal is reversed
0x01:Positioning complete	0x101:Positioning complete signal inverse
0x02:Speed agreement	0x102:Speed agreement signal inverse
0x03:Rotation detection signal	0x103:Rotation detection signal reversed
0x04:Torque limiting in progress	0x104:Torque limiting in progress signal reversed
0x05:Speed limiting in	0x105:Speed limiting in signal reversal
0x06:Brake interlock	0x106:Brake interlock signal reversed
0x07:Warning	0x107:Warning signal reversed
0x08:Positioning proximity signal	0x108:Positioning proximity signal reversed
0x09:Command pulse input multiplier switching signal	
0x109:Command pulse input multiplier switching signal is reversed	

Object 2622h: Warning detection selection					
index	2622h				
name	Warning detection selection				
Object structure	VAR	**type of data**	Uint16	**data range**	0~1
Can map	NO	**Accessibility**	RW	**Factory default**	0

Function selection switch.

0: No detection warning

1: Detection warning

Object 2792h: Clear absolute encoder operation					
index	2792h				
name	Clear absolute encoder operation				
Object structure	VAR	**type of data**	Uint16	**data range**	0~2
Can map	NO	**Accessibility**	RW	**Factory default**	0

0-No action

1-Write motor parameters to encoder EEPROM

2-Clear multi-turn encoder turns: When the battery is replaced/plugged during the first use or drive power down, the encoder backup alarm (Er.810) will be reported when the power is reapplied. This parameter is set to 2 and can only be cleared after re-powering.

Object 2900h: Store Para or Not					
index	2900h				
name	Store Para or Not				
Object structure	VAR	**type of data**	Uint16	**data range**	0~1
Can map	NO	**Accessibility**	RW	**Factory default**	0
0-Parameters changed by the master are not stored in power-down 1-Master changed parameters are stored					

Object 2901h: Prevent overflow function					
index	2901h				
name	Prevent overflow function				
Object structure	VAR	**type of data**	Uint16	**data range**	0~1
Can map	NO	**Accessibility**	RW	**Factory default**	0
0-Do not use the change function 1-Use the function					

Object 2902h: Set PosAc 0					
index	2902h				
name	Set PosAc 0				
Object structure	VAR	**type of data**	Uint16	**data range**	0~1
Can map	NO	**Accessibility**	RW	**Factory default**	0
The rise along the clear 6064h is 0					

Object 2903h: Linear axis number(Rev)					
index	2903h				
name	Linear axis number(Rev)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~65535
Can map	NO	**Accessibility**	RW	**Factory default**	0
When running in one direction all the time, the number of axes ratio can be set (e.g., gearbox 5:1, etc.) for use with 2901h.					

Object 2920h: Origin position L					
index	2920h				
name	Origin position L				
Object structure	VAR	**type of data**	Uint16	**data range**	0~65535
Can map	NO	**Accessibility**	RW	**Factory default**	0
The low 16-bit position value stored when the clear feedback position 0x6064 is 0					

Object 2921h: Origin position H					
index	2921h				
name	Origin position H				
Object structure	VAR	**type of data**	Uint16	**data range**	0~256
Can map	NO	**Accessibility**	RW	**Factory default**	1
The high 16-bit position value stored when the clear feedback position 0x6064 is 0					

Object 2923h: Times of multi circle overflow(Rev)					
index	2923h				
name	Times of multi circle overflow(Rev)				
Object structure	VAR	**type of data**	Uint16	**data range**	0~256
Can map	NO	**Accessibility**	RW	**Factory default**	1
Record the number of multi-turn overflows and monitor the variables.					

6.5.3 Detailed description of the 6000h group object

Object 603Fh : Error Code (Error Code)					
index	603Fh				
name	error code				
Object structure	VAR	**type of data**	Uint16	**data range**	0~65535
Can map	NO	**Accessibility**	RO	**Factory default**	-
The most recent error in the fault code drive. See the fault list for details.					

Object 6040h : Control Word (Control Word)	
index	6040h
name	Control word

Object structure	VAR	**type of data**	Uint16	**data range**	0~65535
Can map	RPDO	**Accessibility**	RW	**Factory default**	0

Control word definition:

Bit	meaning	description
0	Servo ready	0- invalid ; 1- valid
1	Turn on the main loop	0- invalid; 1- valid
2	Rapid shutdown	0- valid; 1- invalid
3	Servo operation	0- invalid ; 1- valid
4~6	Related to mode	-
7	Fault reset	Bit7 rising edge is valid Bit7 remains at 1 , other control commands are invalid.
8	Pause	
9~10	NA	Reserved
11~15	Factory customization	Reserved

Bits 4~6 are defined:

Bit	Operating mode			
	Contour position	Contour speed	Contour torque	Zero return mode
4	New location Rising edge trigger	Reserved	Reserved	Zero return
5	0: Not updated immediately 1 : Update now	Reserved	Reserved	Reserved
6	0: absolute position 1 : Relative position	Reserved	Reserved	Reserved

Object 6041h : status word (the Status Word)					
index	6041h				
name	Status word				
Object structure	VAR	**type of data**	Uint16	**data range**	0~65535
Can map	RPDO	**Accessibility**	RW	**Factory default**	0

Reflect the servo status:

Bit	name	Bit definition
0	Servo no fault	1- valid; 0- invalid
1	Waiting to turn on the servo enable	1- valid; 0- invalid
2	Servo operation	1- valid; 0- invalid
3	malfunction	0: no fault 1 : There is a fault
4	Turn on the main loop	1- valid; 0- invalid
5	Rapid shutdown	0- valid; 1- invalid
6	Servo is not operational	1- valid; 0- invalid
7	caveat	0 : no warning; 1 : warning
8	Factory customization	
9	remote control	0 - Non- CAN open mode. 1-CANopen remote control mode.
10	Target arrival	0 - The target position or speed has not arrived. 1- Target position or speed arrives.
11	Software internal location is out of limits	The 0- position command or feedback did not reach the software internal position limit. 1- position command or feedback reaches the software internal position limit
12	Zero speed signal	Speed mode: 0 -non-zero speed 1- zero speed Position mode: 0- positioning completed 1- positioning is not completed
13		Origin zero return failure flag: 0: zero error does not occur 1: Error occurs when returning to zero (zero return mode, zero return timeout)
14	NA	Reserved
15	Zero return to origin	0- Home return is not performed or not completed. 1- The zero point of the origin has been completed and the reference point has been found.

Object 6060h : mode selection (Modes of Operation)	
index	6060h
name	Mode selection

Object structure	VAR	**type of data**	Int8	**data range**	0~10
Can map	RPDO	**Accessibility**	RW	**Factory default**	8

Select the servo operation mode:

Set value	Description
0	NA
1	Contour position mode (pp)
2	NA
3	Contour speed mode (pv)
4	Contour torque mode (pt)
5	NA
6	Zero return mode (hm)
7	NA
8	Cycle sync position mode (csp)
9	Cycle Synchronous Speed Mode (csv)
10	Periodic synchronous torque mode (cst)

Object 6061h : Modes of Operation Display					
index	6061h				
name	Operating mode display				
Object structure	VAR	**type of data**	Int8	**data range**	0~10
Can map	TPDO	**Accessibility**	RO	**Factory default**	0

Select the servo operation mode:

Set value	Description
0	NA
1	Contour position mode (pp)
2	NA
3	Contour speed mode (pv)
4	Contour torque mode (pt)
5	NA
6	Zero return mode (hm)
7	NA
8	Cycle sync position mode (csp)
9	Cycle Synchronous Speed Mode (csv)
10	Periodic synchronous torque mode (cst)

Object 6062h : Position Demand Value

index	6062h				
name	Position command				
Object structure	VAR	**type of data**	Int32	**data range**	-2^{31}~$(2^{31}-1)$
Can map	TPDO	**Accessibility**	RO	**Factory default**	-
Reflect real-time position command (unit: command unit)					

Object 6063h : Position Actual Value					
index	6063h				
name	Motor position feedback				
Object structure	VAR	**type of data**	Int32	**data range**	-2^{31}~$(2^{31}-1)$
Can map	TPDO	**Accessibility**	RO	**Factory default**	-
Reflect real-time position feedback (unit: encoder unit)					

Object 6064h : Position Actual Value					
index	6064h				
name	Position feedback				
Object structure	VAR	**type of data**	Int32	**data range**	-2^{31}~$(2^{31}-1)$
Can map	TPDO	**Accessibility**	RO	**Factory default**	-
Reflect real-time position feedback (unit: command unit)					

Object 6065h : Position deviation excessive threshold (Following error Window)					
index	6065h				
name	Position deviation excessive threshold				
Object structure	VAR	**type of data**	Int32	**data range**	-2^{31}~$(2^{31}-1)$
Can map	TPDO	**Accessibility**	RW	**Factory default**	-
Set the position deviation too large threshold (command unit). When the difference between the user position command 6062h and the user position feedback 6064h exceeds ±6065h , a positional deviation excessive fault occurs. When 6065h is set to ($2^{32}-1$), the servo does not perform excessive position deviation monitoring.					

Object 6067h : Positional arrival threshold (Position Window)					
index	6067h				
name	**Location arrival threshold**				
Object structure	VAR	**type of data**	Uint32	**data range**	0~$(2^{32}-1)$

Can map	RPDO	**Accessibility**	RW	**Factory default**	-
Set the threshold for the arrival of the position (unit: command unit). The difference between the user position command 6062h and the user actual position feedback 6064h is within ±6067h , and when the time reaches 6068h , the position is considered to arrive, and in the contour position mode, the bit 10 of the status word 6041h is =1 .					

Object 6068h : Motor Position Time Window (Position Window Time)					
index	6068h				
name	Motor position arrival window time				
Object structure	VAR	**type of data**	Uint16	**data range**	0~65535
Can map	RPDO	**Accessibility**	RW	**Factory default**	0
Set the time window for the decision position to arrive (unit: 2ms). The difference between the user position command 6062h and the user actual position feedback 6064h is within ±6067h , and when the time reaches 6068h , the position is considered to arrive, and in the contour position mode, the bit 10 of the status word 6041h is =1 .					

Object 606Bh : User's actual speed command (Velocity Demand Value)					
index	6068h				
name	User actual speed command				
Object structure	VAR	**type of data**	Int32	**data range**	-2^{31}~$(2^{31}-1)$
Can map	TPDO	**Accessibility**	RO	**Factory default**	0
Reflects the user's actual speed command (unit: 1rpm). In the position class mode, a speed command corresponding to the output of the position adjuster is reflected; In the speed mode, the input command of the speed regulator is reflected.					

Object 606Ch : User's actual speed feedback (Velocity Actual Value)					
index	606Ch				
name	User actual speed feedback				
Object structure	VAR	**type of data**	Int32	**data range**	-2^{31}~$(2^{31}-1)$
Can map	TPDO	**Accessibility**	RO	**Factory default**	0
Reflects the user's actual speed feedback value (unit: command unit / s).					

Object 606Dh : Speed reaches the threshold speed (Velocity Window)	
index	606Dh
name	Speed reaches threshold speed

Object structure	VAR	**type of data**	Uint16	**data range**	0~3000
Can map	RPDO	**Accessibility**	RW	**Factory default**	0
Set the threshold for speed arrival (unit: 1 rpm). Target speed 60FFh user actual speed 606Ch is a difference ±606Dh less, and the time to reach 606Eh time, that the speed reaches the speed profile mode, the status word 6041h of Bit10 = . 1. Otherwise the status word 6061h of Bit10 = 0 .					

Object 606Eh : Velocity Window Time					
index	606Eh				
name	Speed arrival time window				
Object structure	VAR	**type of data**	Uint16	**data range**	0~65535
Can map	RPDO	**Accessibility**	RW	**Factory default**	0
Set the time window in which the judgment speed reaches the valid (unit: ms). Target speed 60FFh user actual speed 606Ch is a difference ±606Dh less, and the time to reach 606Eh time, that the speed reaches the speed profile mode, the status word 6041h of Bit10 = . 1. Otherwise the status word 6061h of Bit10 = 0 .					

Object 606Fh : zero speed threshold (Velocity Threshold)					
index	606Fh				
name	Zero speed threshold				
Object structure	VAR	**type of data**	Uint16	**data range**	0~65535
Can map	RPDO	**Accessibility**	RW	**Factory default**	10
Set the threshold (unit: 1 rpm) for determining whether the user speed is 0. User feedback speed 606Ch in ± 606Fh within, and the time to reach 6070h set value indicates that the user rate is 0 , case status word 6041h of bit12 = 1; do not satisfy any of the conditions in both, speed is not considered that the user is 0 , this when the status word 6041h of bit1 2 = 0 .					

Object 6070h : zero speed the time window (Zero the Velocity the Threshold Time)					
index	6070h				
name	Zero speed time window				
Object structure	VAR	**type of data**	Uint16	**data range**	0~65535
Can map	RPDO	**Accessibility**	RW	**Factory default**	0
Set the time window (unit: 2 ms) used to determine if the user's speed is zero.					

User feedback speed 606Ch in ± 606Fh within, and the time to reach 6070h set value indicates that the user rate is 0 , case status word 6041h of bit12 = 1; do not satisfy any of the conditions in both, speed is not considered that the user is 0 , this when the status word 6041h of bit1 2 = 0 .

Object 6071h : target torque (Target torque)					
index	6071h				
name	Target torque				
Object structure	VAR	**type of data**	Int16	**data range**	**-3000~3000**
Can map	RPDO	**Accessibility**	RW	**Factory default**	0
Only used for contour torque mode, torque command target value (unit: 0.1%).					

Object 6074h : a torque command (the Torque Demand the Value)					
index	6074h				
name	Torque command				
Object structure	VAR	**type of data**	Int16	**data range**	-
Can map	TPDO	**Accessibility**	RO	**Factory default**	-
Displays the current torque command (unit: 1%).					

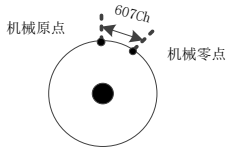
Object 6075h : nominal motor current (Motor Rated Current)					
index	6075h				
name	Motor rated current				
Object structure	VAR	**type of data**	Uint32	**data range**	-
Can map	TPDO	**Accessibility**	RO	**Factory default**	-
Rated current on the motor nameplate (unit: mA). All current-related parameter values are related to this parameter.					

Object 6076h : Motor rated torque (Motor Rated Torque)					
index	6076h				
name	Motor rated torque				
Object structure	VAR	**type of data**	Uint32	**data range**	0~(2^{32}-1)
Can map	TPDO	**Accessibility**	RO	**Factory default**	-
Rated torque on the motor nameplate (unit: m Nm). All torque related parameter values are related to this parameter.					

Object 6077h : motor feedback torque (T orque Actual the Value)	
index	6077h
name	Motor feedback torque

Object structure	VAR	**type of data**	Int16	**data range**	-
Can map	TPDO	**Accessibility**	RO	**Factory default**	-
The instantaneous torque output of the response servo motor (unit: 1%).					

Object 607Ah : target position (Target Position)					
index	607Ah				
name	target location				
Object structure	VAR	**type of data**	Int32	**data range**	-2^{31}~$(2^{31}-1)$
Can map	TPDO	**Accessibility**	RO	**Factory default**	-
Set the servo target position (unit: command unit) in the contour position mode and cycle synchronization position mode .					

Object 607Ch : Home Offset (Hmoe Offset)					
index	607Ch				
name	Origin offset				
Object structure	VAR	**type of data**	Int32	**data range**	-2^{31}~$(2^{31}-1)$
Can map	RPDO	**Accessibility**	RW	**Factory default**	0
In the position class control mode, the mechanical zero deviates from the physical position of the motor origin (unit: encoder unit). Mechanical origin = mechanical zero + 607Ch (origin offset). When set to 0, the origin is unbiased.  					

Object 607Dh : Software absolute position limit (Software Limit position)					
index	607Dh				
name	Software absolute position limit				
Object structure	VAR	**type of data**	Int32	**data range**	-
Can map	RPDO	**Accessibility**	RW	**Factory default**	-
Set the minimum and maximum values for the absolute position limit of the software. Minimum absolute position limit = (607D-1h) Maximum absolute position limit = (607D-2h) Software absolute position limit setting: 1. When both (607D-1h) and (607D-2h) are set to the default values, the software limit does not take effect.					

2. AL.931 warning will occur when the minimum absolute position limit (607D-1h) is greater than the maximum absolute position limit (607D-2h).

3. When the position command or position feedback reaches the software limit value, in the position mode, the servo will operate with the position limit as the target position, stop at the position limit, and prompt the overtravel warning AL.950. Entering the reverse command will cause the motor to exit the position overrun condition.

Subindex	00h				
name	Number of sub-indexes for absolute position restrictions				
Object structure	-	**type of data**	Uint8	**data range**	-
Can map	NO	**Accessibility**	R O	**Factory setting**	2

Subindex	01h				
name	Minimum software absolute position limit (Min Position Limit)				
Object structure	-	**type of data**	Uint8	**data range**	-2^{31}~$(2^{31}-1)$
Can map	RPDO	**Accessibility**	R W	**Factory setting**	-2^{31}

Subindex	02h				
name	Maximum software absolute position limit (Max Position Limit)				
Object structure	-	**type of data**	Uint8	**data range**	-2^{31}~$(2^{31}-1)$
Can map	RPDO	**Accessibility**	R W	**Factory setting**	$2^{31}-1$

Object 607Eh : Command Polarity (Polarity)					
index	607Eh				
name	Command polarity				
Object structure	VAR	**type of data**	Uint8	**data range**	0-255
Can map	RPDO	**Accessibility**	RW	**Factory default**	0

Reflects the polarity of the position command or speed command.

B it	7	6	5	4~0
meaning	Position command polarity	Speed command polarity	Torque command polarity	Reserved

Bit7=1, indicating the standard position mode, the position command ×(-1), the motor is reversed. In the contour position mode and the cycle synchronous position mode, the position command and the target position are inverted.

Bit6=1 means that the speed command (60FFh) ×(-1) will be reversed in the speed mode .

Object 607Fh : Max Profile Velocity	
index	607Fh
name	Maximum contour speed

Object structure	VAR	**type of data**	Uint32	**data range**	0~(2^{32}-1)
Can map	RPDO	**Accessibility**	RW	**Factory default**	50000
Maximum user speed (unit: 0.1rpm). The set value takes effect when the slave speed command changes.					

Object 6080h : Max Motor Velocity					
index	6080h				
name	Maximum motor speed				
Object structure	VAR	**type of data**	**Uint32**	**data range**	0~(2^{32}-1)
Can map	RPDO	**Accessibility**	RW	**Factory default**	6000
The maximum permissible operating speed of the motor can be obtained from the instruction manual of the servo motor (unit: rpm)					

Object 6081h : Profile Velocity					
index	6081h				
name	Contour speed				
Object structure	VAR	**type of data**	**Uint32**	**data range**	0~(2^{32}-1)
Can map	RPDO	**Accessibility**	RW	**Factory default**	10000000
In the contour position mode, the running speed of the uniform speed section reached after the acceleration section is completed (unit: command unit / s). $$\text{motor speed} = \frac{6081\text{h} \times \frac{6091\text{h}-1}{6091\text{h}-2}}{\text{Encoder resolution}} \times 60$$					

Object 6083h : Acceleration profile (Profile Acceleration)					
index	6083h				
name	Contour acceleration				
Object structure	VAR	**type of data**	**Uint32**	**data range**	0~(2^{32}-1)
Can map	RPDO	**Accessibility**	RW	**Factory default**	200
Lower profile mode, from the motor 0rpm accelerated to the speed reference acceleration (unit: Command / $S^{2)}$. After the slave receives the segment shift command, the set value takes effect.					

Object 6084h : Profile Deceleration	
index	6084h

name	Profile deceleration				
Object structure	VAR	**type of data**	**Uint32**	**data range**	0~(2^{32}-1)
Can map	RPDO	**Accessibility**	RW	**Factory default**	200
The speed profile mode, from the motor speed is decelerated to the current 0rpm deceleration (unit: Command / S [2)]. After the slave receives the segment shift command, the set value takes effect.					

Object 6086h : Motor Run Curve Type					
index	6086h				
name	Motor running curve type				
Object structure	VAR	**type of data**	**Int16**	**data range**	-2^{15}~(2^{15}-1)
Can map	RPDO	**Accessibility**	RW	**Factory default**	0
The type of curve for the motor position command or speed command. 0- linear					

Object 6087h : torque ramp (the Torque S Lope)					
index	6087h				
name	Torque ramp				
Object structure	VAR	**type of data**	**UInt32**	**data range**	0~(2^{32}-1)
Can map	RPDO	**Accessibility**	RW	**Factory default**	0
In the contour torque mode, it indicates the time (in ms) that the torque given amount has risen from 0% of the rated torque to 100% .					

Object 6091h : gear ratio (Gear Ratio)					
index	6091h				
name	Gear ratio				
Object structure	VAR	**type of data**	**UInt32**	**data range**	-
Can map	RPDO	**Accessibility**	RW	**Factory default**	-
The position factor is used to establish the proportional relationship between the load displacement specified by the user and the motor displacement: Motor displacement (motor unit) = load displacement (user unit) × position factor The position factor setting is related to the mechanical reduction ratio, mechanical size related parameters, and motor resolution. The calculation method is as follows:					

$$\text{Position factor} = \frac{\text{Motor resolution} \times \text{gear ratio}}{\text{Load feed}}$$

Subindex	00h				
name	**Number of sub-indexes**				
Object structure	VAR	**type of data**	Uint8	**data range**	-
Can map	NO	**Accessibility**	R O	**Factory setting**	2

Subindex	01h				
name	**Motor resolution (Motor Revolutions)**				
Object structure	VAR	**type of data**	Uint32	**data range**	1~(2^{32}-1)
Can map	RPDO	**Accessibility**	R W	**Factory setting**	1

Subindex	02h				
name	**Resolution shaft (Shaft Revolutions)**				
Object structure	VAR	**type of data**	Uint32	**data range**	1~(2^{32}-1)
Can map	RPDO	**Accessibility**	R W	**Factory setting**	1

Object 6098h : zero modes (Homing Method,)					
index	6098h				
name	**Zero return mode**				
Object structure	VAR	**type of data**	Int8	**data range**	1~35
Can map	RPDO	**Accessibility**	RW	**Factory default**	1

Select the origin return method:

value	Description
1	Regression with negative limit switch and Z pulse signal
2	Regression with positive limit switch and Z pulse signal
3,4	Regression with positive origin switch and Z pulse signal
5,6	Regression to negative origin switch and Z pulse signal
7~14	Regression with the origin switch and Z pulse signal
15~16	Reserved
17~30	Regression without reference to the Z pulse signal
31~32	Reserved
33~34	Regression without reference to the Z pulse signal
35	Reset with current position

Object 6099h : Homing Speeds

index	6099h				
name	**Zero return speed**				
Object structure	ARR	**type of data**	-	**data range**	-
Can map	YES	**Accessibility**	RW	**Factory default**	-
In the zero return mode, the two speed value settings are included: 6099-1h search deceleration point signal speed (unit: command unit / s); 6099-2h Search for the origin signal speed (unit: command unit / s).					

Subindex	00h				
name	**Number of sub-indexes**				
Object structure	VAR	**type of data**	Uint8	**data range**	-
Can map	NO	**Accessibility**	R O	**Factory setting**	2

Subindex	01h				
name	Search deceleration point signal speed (Speed During Search for Switch)				
Object structure	VAR	**type of data**	Uint32	**data range**	**0 ~(2 ^{32} -1)**
Can map	RPDO	**Accessibility**	R W	**Factory setting**	27962026

Subindex	02h				
name	Search for the speed of the origin signal (Speed During Search for Zero)				
Object structure	VAR	**type of data**	Uint32	**data range**	**0 ~(2 ^{32} -1)**
Can map	RPDO	**Accessibility**	R W	**Factory setting**	5592405

Object 609Ah : Homing Acceleration					
index	609Ah				
name	**Zero return acceleration**				
Object structure	ARR	**type of data**	**Uint32**	**data range**	**0 ~(2 ^{32} -1)**
Can map	RPDO	**Accessibility**	RW	**Factory default**	16000
Set the acceleration in the home return mode. The object dictionary unit is defined as the position instruction increment per second (command unit / s ^{2}) .					

Object 60B0h : position offset (the Position offset)					
index	60B0h				
name	**Position offset**				
Object structure	ARR	**type of data**	**Int32**	**data range**	**-2 ^{31} ~(2 ^{31} -1)**

Can map	RPDO	Accessibility	RW	Factory default	0
Used to synchronize the position command offset amount in the cycle position mode, servo target position = 607Ah + 60B0h .					

Object 60B8h : Probe Function (Touch Probe Function)					
index	60B8h				
name	Probe function				
Object structure	VAR	type of data	Uint16	data range	0
Can map	RPDO	Accessibility	RW	Factory default	0

I.e. the probe feature position latch function, can latch the external DI signal or the electrical machine Z position information of the signals change. This servo supports two probe functions to latch 4 position information. Probe 1 can be selected DI44 or Z signal as a probe signal, the probe 2 can select DI45 or Z signal as a probe signal.

The function of probe 1 and probe 2:

Bit	description	Range
0	Probe 1 enable	0--- Probe 1 is not enabled 1--- Probe 1 enabled
1	Probe 1 trigger mode	0--- single trigger 1--- Continuous trigger
2	Probe 1 trigger signal selection	0---DI44 input signal 1---Z signal
3	NA	
4	Probe 1 rising edge, falling edge selection	0--- falling edge latch 1--- rising edge latch
5-7	NA	
8	Probe 2 enable	0--- Probe 2 is not enabled 1--- Probe 2 enabled
9	Probe 2 trigger mode	0--- single trigger 1--- Continuous trigger
10	Probe 2 trigger signal selection	0---DI45 input signal 1---Z signal
11	NA	
12	Probe 2 rising edge, falling edge selection	0--- falling edge latch 1--- rising edge latch
13-15	NA	

Object 60B9h : Probe Status (Touch Probe Status)

index	60B9h				
name	Probe status				
Object structure	VAR	type of data	Uint16	data range	0
Can map	TPDO	Accessibility	RO	Factory default	0

The state of probe 1 and probe 2.

Bit	description	Remarks
0	0--- Probe 1 is not enabled 1--- Probe 1 enabled	
1	0 — Probe 1 rising edge latch is not executed 1 — Probe 1 rising edge latch has been executed	
2	0 — Probe 1 falling edge latch is not executed 1 — Probe 1 falling edge latch has been executed	
3~5	NA	
6	0 — DI44 input signal 1 — Z signal	
7	0 — DI44 is low 1 — DI44 is high	
8	0--- Probe 2 is not enabled 1--- Probe 2 enabled	
9	0 — Probe 2 rising edge latch is not executed 1 — Probe 2 rising edge latch has been executed	
10	0 — Probe 2 falling edge latch is not executed 1 — Probe 2 falling edge latch has been executed	
11~13	NA	
14	0 — DI45 input signal 1 — Z signal	
15	0 — DI45 is low 1 — DI45 is high	

Object 60BAh : Probe 1 rising edge position feedback (Touch Probe Pos1 Pos Value)	
index	60Bah
name	**Probe 1 rising edge position feedback**

Object structure	VAR	**type of data**	**Int32**	**data range**	-
Can map	TPDO	**Accessibility**	RO	**Factory default**	-
Displays the rising edge of the probe 1 signal and the position feedback (command unit).					

Object 60BBh : Probe 1 Falling Edge Position Feedback (Touch Probe Pos1 Neg Value)					
index	60BBh				
name	**Probe 1 falling edge position feedback**				
Object structure	VAR	**type of data**	**Int32**	**data range**	-
Can map	TPDO	**Accessibility**	RO	**Factory default**	-
Displays the falling edge of the probe 1 signal and the position feedback (command unit).					

Object 60BCh : Probe 2 rising edge position feedback (Touch Probe Pos1 Pos Value)					
index	60BCh				
name	**Probe 2 rising edge position feedback**				
Object structure	VAR	**type of data**	**Int32**	**data range**	-
Can map	TPDO	**Accessibility**	RO	**Factory default**	-
Displays the rising edge of the probe 2 signal and the position feedback (command unit).					

Object 60BDh : Probe 1 Falling Edge Position Feedback (Touch Probe Pos2 Neg Value)					
index	60BDh				
name	**Probe 2 falling edge position feedback**				
Object structure	VAR	**type of data**	**Int32**	**data range**	-
Can map	TPDO	**Accessibility**	RO	**Factory default**	-
Displays the falling edge of the probe 2 signal and the position feedback (command unit).					

Object 60E0h : Forward torque limit value (Positive Torque limit)					
index	**60E0h**				
name	Forward torque limit				
Object structure	VAR	**type of data**	**Uint16**	**data range**	**0~3000**
Can map	RPDO	**Accessibility**	RW	**Factory default**	3000
Limit the maximum forward torque (in 0.1%).					

Object 60E1h : Negative to the torque limit value (Negtive Torque limit)

index	60E1h				
name	Negative torque limit				
Object structure	VAR	type of data	Uint16	data range	0~3000
Can map	RPDO	Accessibility	RW	Factory default	3000
Limit the maximum negative torque (in 0.1%).					

Object 60F4h : positional deviation (Following Error Actual Value)					
index	60F4h				
name	Position deviation				
Object structure	VAR	type of data	Int32	data range	-2^{31}~$(2^{31}-1)$
Can map	TPDO	Accessibility	RO	Factory default	-
Display position deviation (command unit).					

Object 60FCh : position command (the Position the Value Demand)					
index	60FCh				
name	Position command				
Object structure	VAR	type of data	Int32	data range	-2^{31}~$(2^{31}-1)$
Can map	TPDO	Accessibility	RO	Factory default	-
Motor real-time position command Position command (6062h) × electronic gear ratio (609 1 h) = motor position command (60FCh)					

Object 60FDh : Digital Input					
index	60FDh				
name	Digital input				
Object structure	VAR	type of data	Uint32	data range	0~0xFFFFFFFF
Can map	TPDO	Accessibility	RO	Factory default	0

Reflects the current DI terminal logic of the drive, 0 means invalid, 1 means valid

The DI signals indicated by each of the following are as follows:

Bit	31~16	15~ 5	4	3	2	1	0
description	Factory customization	Reserved	DI 45 level	DI 44 level	Origin switch	Forward overtravel switch	Reverse overtravel switch

Object 60FEh : Digital Output (Digital Out put)	
index	60FEh

name	Digital output				
Object structure	ARR	**type of data**	Uint32	**data range**	-
Can map	TPDO	**Accessibility**	RO	**Factory default**	-
Reflects the current D O terminal logic of the drive , 0 means invalid, 1 means valid					

Subindex	00h				
name	**Number of sub-indexes**				
Object structure	VAR	**type of data**	Uint8	**data range**	-
Can map	NO	**Accessibility**	R O	**Factory setting**	2

Sub index	01h				
name	**Physical output**				
Object structure	VAR	**type of data**	Uint32	**data range**	0~0xFFFFFFFF
Can map	TPDO	**Accessibility**	R O	**Factory setting**	0

The D O signals indicated by each of the following are as follows:

Bit	description
0	Brake output
1~15	NA
16~31	Factory customization

Subindex	02h				
name	Physical output enable				
Object structure	VAR	**type of data**	Uint32	**data range**	0~0xFFFFFFFF
Can map	TPDO	**Accessibility**	R O	**Factory setting**	0
invalid					

Object 60FFh : Target Velocity					
index	**60FFh**				
name	Target speed				
Object structure	VAR	**type of data**	Int32	**data range**	-2^{31}~$(2^{31}-1)$
Can map	RPDO	**Accessibility**	RW	**Factory default**	-
Set the speed command (in unit: command unit / s) in the contour speed mode and synchronous cycle speed mode .					

Object 6052h : Support servo operation mode (Supported Drive Modes)

index	6052h				
name	Support servo operation mode				
Object structure	VAR	type of data	Uint32	data range	0~(2^{32}-1)
Can map	TPDO	Accessibility	RO	Factory default	-

The servo operation mode supported by the drive, 0 means not supported, 1 means support.

Bit	Described above	value
0	Contour position mode	1
1	Frequency control mode	0
2	Contour speed mode	1
3	Contour torque mode	1
4	NA	0
5	Zero return mode	1
6	Interpolation position mode	0
7	Cycle sync position mode (csp)	1
8	Cycle Synchronous Speed Mode (csv)	1
9	Periodic synchronous torque mode (cst)	1
1 0 ~ 31	Manufacturers custom defined	Pre- stay

7 Troubleshooting information

7.1 fault code

Fault display	Fault name	error code	Auxiliary code
Er.020	Parameter and check exception	0x6000	0x00000020
Er.021	Parameter formatting exception	0x6001	0x00000021
Er.022	System and check exception	0x600 2	0x00000022
Er. 023	XML file not burned	0x6003	0x00000023
Er.030	Main circuit detection part is abnormal	0x600 4	0x00000030
Er.040	Abnormal parameter setting	0x600 5	0x00000040
Er.041	Distribution pulse output setting is abnormal	0x600 6	0x00000041
Er.042	Abnormal combination of parameters	0x600 7	0x00000042
Er.044	Semi-closed loop / full closed loop parameter setting exception	0x600 8	0x00000044
Er.050	Drive does not match motor capacity	0x600 9	0x00000050
Er.051	Product does not support alarm	0x600 A	0x00000051
Er.080	Encoder unit pulse distance setting is abnormal	0x600B	0x00000080
Er.08A	Position sensor resolution setting is abnormal	0x600 C	0x0000008A

Er.0B0	Servo on command invalid alarm	0x600 D	0x000000B0
Er.100	Overcurrent (OC)	0x600 E	0x00000100
Er.300	Regeneration failure	0x600 F	0x00000300
Er.320	Regenerative overload	0x6010	0x00000320
Er.330	Main circuit power supply wiring error	0x601 1	0x00000330
Er.400	Main circuit overvoltage (OV)	0x601 2	0x00000400
Er.410	Main circuit under voltage	0x601 3	0x00000410
Er.42A	Converter exception	0x601 4	0x0000042A
Er.450	Main circuit capacitor overvoltage	0x601 5	0x00000450
Er.510	Over speed (OS)	0x601 6	0x00000510
Er.511	Crossover pulse output overspeed	0x601 7	0x00000511
Er.520	Vibration alarm	0x601 8	0x00000520
Er.521	Automatic tuning alarm	0x601 9	0x00000521
Er.550	The highest speed setting is abnormal	0x601 A	0x00000550
Er.710	Overload (instantaneous maximum load)	0x601 B	0x00000710
Fault display	**Fault name**	**error code**	**Auxiliary code**
Er.720	Overload (continuous maximum load)	0x601 C	0x00000720
Er.730	DB overload 1	0x601 D	0x00000730
Er.731	DB overload 2	0x601 E	0x00000731
Er.740	Inrush current limiting resistor overload	0x601 F	0x00000740
Er.7A0	Heat sink overheating	0x6020	0x000007A0
Er.7AA	Control the substrate temperature abnormality	0x602 1	0x000007AA
Er.7AB	Servo unit built-in fan stop	0x602 2	0x000007AB
Er.810	Encoder backup alarm	0x602 3	0x00000810
Er.820	Encoder and number verification alarm	0x602 4	0x00000820
Er.830	Encoder battery alarm	0x602 5	0x00000830
Er.840	Encoder data is abnormal	0x602 6	0x00000840
Er.850	Encoder overspeed	0x602 7	0x00000850
Er.860	Encoder overheating	0x602 8	0x00000860
Er.870	Encoder write error	0x60 78	0x00000870
Er.900	CAN main station dropped (life factor)	0x607 A	0x00000900
Er.901	CAN main station dropped (consumer time)	0x607 B	0x00000901
Er.902	CAN master station off monitoring line configuration conflict	0x607 C	0x00000902
Er.910	Critical point number error	0x606 D	0x00000910
Er.911	Spindle data error	0x606 E	0x00000911
Er.912	Spindle speed error	0x606 F	0x00000912
Er.913	Slave speed error	0x6070	0x00000913
Er.914	Line type error	0x607 1	0x00000914
Er.915	System operation error	0x607 2	0x00000915
Er.B10	Speed command A/D exception	0x603 1	0x00000B10

Er.B11	Speed command A/D conversion data is abnormal	0x603 2	0x00000B11
Er.B20	Torque command A/D exception	0x603 3	0x00000B20
Er.B31	Current detection fault 1 (U phase)	0x603 4	0x00000B31
Er.B32	Current detection fault 2 (V phase)	0x603 5	0x00000B32
Er.B33	Current detection fault 3 (current detector)	0x603 6	0x00000B33
Er.B40	Reserved	0x603 7	0x00000B40
Er.BE0	Reserved	0x603 8	0x00000BE0
Fault display	**Fault name**	**error code**	**Auxiliary code**
Er.BF0	System alarm 0	0x603 9	0x00000BF0
Er.BF1	System alarm 1	0x603 A	0x00000BF1
Er.BF2	System alarm 2	0x603 B	0x00000BF2
Er.BF3	System alarm 3	0x603 C	0x00000BF3
Er.BF4	Hardware overcurrent	0x603 D	0x00000BF4
Er.C10	Prevent out of control detection	0x603 E	0x00000C10
Er.C20	Phase error detection	0x603 F	0x00000C20
Er.C21	Hall sensor is abnormal	0x60 4 0	0x00000C21
Er.C22	Inconsistent phase information	0x6041	0x00000C22
Er.C50	Magnetic pole monitoring failed	0x604 2	0x00000C50
Er.C51	Magnetic pole monitoring stopped	0x604 3	0x00000C51
Er.C52	Magnetic pole monitoring is not finished	0x604 4	0x00000C52
Er.C53	Magnetic pole monitoring overtravel	0x604 5	0x00000C53
Er.C54	Magnetic pole monitoring failure 2	0x604 6	0x00000C54
Er.C80	Encoder clear exception	0x604 7	0x00000C80
Er.C90	Encoder communication failure	0x604 8	0x00000C90
Er.C91	Encoder communication position data acceleration abnormal	0x604 9	0x00000C91
Er.C92	Encoder communication timer is abnormal	0x604 A	0x00000C92
Er.CA0	Encoder parameter exception	0x604 B	0x00000CA0
Er.CB0	Encoder loopback check exception	0x604 C	0x00000CB0
Er.CC0	Inconsistent upper and lower limits of the number of revolutions	0x604 D	0x00000CC0
Er.CF1	Reserved	0x604 E	0x00000CF1
Er.CF2	Reserved	0x604 F	0x00000CF2
Er.D00	Position deviation is too large	0x6050	0x00000D00
Er.D01	Excessive position deviation when servo ON	0x605 1	0x00000D01
Er.D02	Excessive positional deviation caused by speed limit when servo is ON	0x605 2	0x00000D02
Er.D10	Motor - load position deviation is too large	0x605 3	0x00000D10
Er.D30	Location data overflow	0x605 4	0x00000D30
Er.EB9	EtherCAT initialization exception	0x606 C	0x00000EB9
Er.EC6	Reserved	0x607 3	0x00000EC6

Er.F10	Phase loss when the main circuit selects three-phase power input	0x607 7	0x00000F10

7.2 warning code

Fault display	Fault name	error code	Auxiliary code
AL.900	Position deviation is too large	0x900	0x700
AL.901	Excessive position deviation when servo ON	0x901	0x701
AL.910	Overload warning	0x910	0x710
AL.911	Vibration warning	0x911	0x711
AL.920	Regenerative overload warning	0x920	0x720
AL.921	DB overload warning	0x921	0x7 21
AL.930	Battery undervoltage warning	0x930	0x7 3 0
AL.931	The software limit 607Dh is abnormally set.	0x931	0x7 31
AL.940	The origin is offset outside the soft limit	0x940	0x7 4 0
AL.941	Parameter change warning that requires power failure	0x941	0x7 41
AL.942	EtherCAT control mode setting is abnormal	0x942	0x7 42
AL.950	Position exceeds the soft limit	0x950	0x7 50
AL.971	Undervoltage warning	0x971	0x700
AL.9A0	Overtravel warning	0x9A0	0x7 A0
AL .9B0	SYNC0 sync frame is lost	0x9B0	0x7 B0
AL.9F0	Distributed clock cycle setting exception (125us integer multiple)	0x9F0	0x7 F0

7.3 Communication failure recovery method

For the fault of the servo drive itself, please refer to the corresponding manual. This section only describes the processing method of the EtherCAT communication part.

1) Er.023: Unburned xml configuration file

the reason	Treatment
Unburned configuration file	Burn configuration file
Drive failure	Replace the servo drive

2) Er.EB9: EtherCAT initialization exception

the reason	Treatment
Unburned configuration file	Burn configuration file
Drive failure	Replace the servo drive
Unburned FPGA code	Burn FPGA code

3) AL.931: The software limit 607D is set abnormally.

the reason	Treatment
The minimum limit value of 607Dh is greater than the maximum limit value.	Change 607Dh maximum limit value or minimum limit value

4) AL.940: The origin is offset outside the soft limit

the reason	Treatment
607Ch is set larger than the 607Dh limit value	Change the value of 607Dh or 607Ch

5) AL.942: EtherCAT control mode setting is abnormal

the reason	Treatment
Object dictionary 6060h is set incorrectly	Modify the setting value of 6060h

6) AL.950: Position exceeds the soft limit

the reason	Treatment
Target position 607Ah or actual position 6064h exceeds the limit of 607Dh	Change the value of 607Dh or 607Ah

7) AL.9b0: SYNC0 sync frame is lost

the reason	Treatment
Slave receives an exception during synchronous communication	1. Use shielded twisted pair 2. View the wiring status through the digital display of the keyboard
The master sends an exception when synchronizing communication	Replace the master station equipment

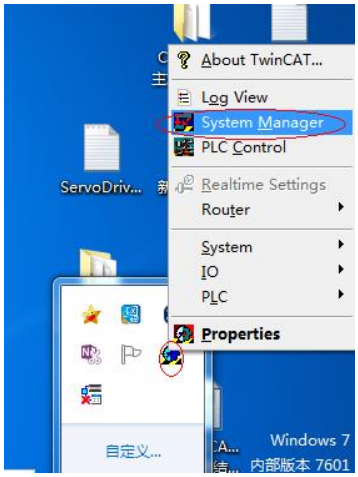
8) AL.9F0: Distribution clock cycle setting is abnormal

the reason	Treatment
The synchronization period is not an integer multiple of 125us	Modify the synchronization period setting to an integer multiple of 125us

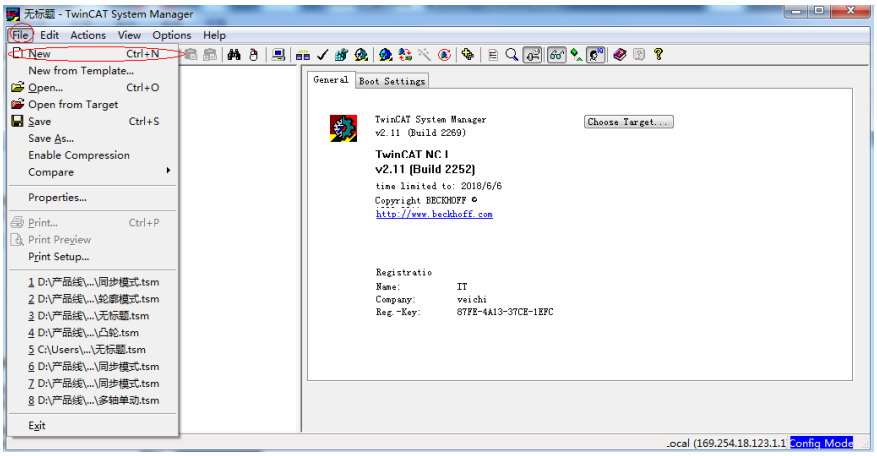
8 with the Beckhoff main station operation example

The following example is a system configuration using Beckhoff TwinCAT 2.110.2252 software with Beckhoff controller CX9020. Please install the TWinCAT software correctly before configuring the software.

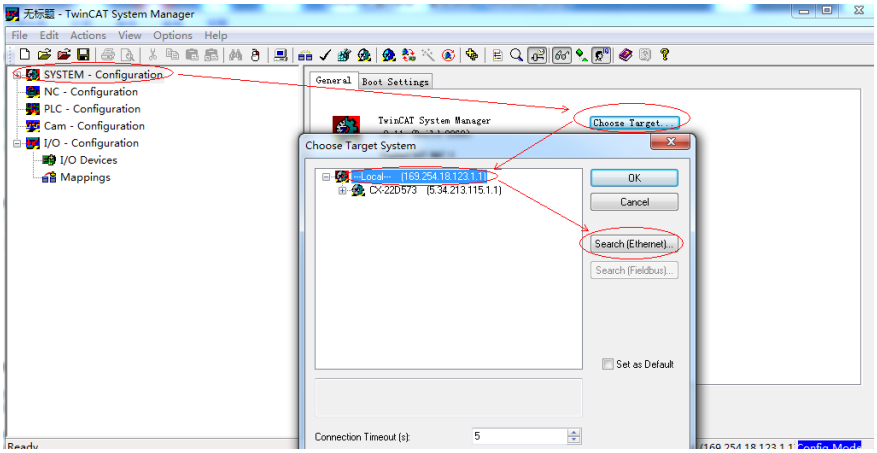
1. Firstly connect the servo driver correctly and supply power, and set the first digit of function code Pn000 to 1.
2. Copy the VEICHI XML file to the folder where TwinCAT is installed (the path is usually: C: \TwinCAT\IO\EtherCAT);
3. Restart TwinCAT.
4. Open the TwinCAT System Manager as shown below and start the configuration process.



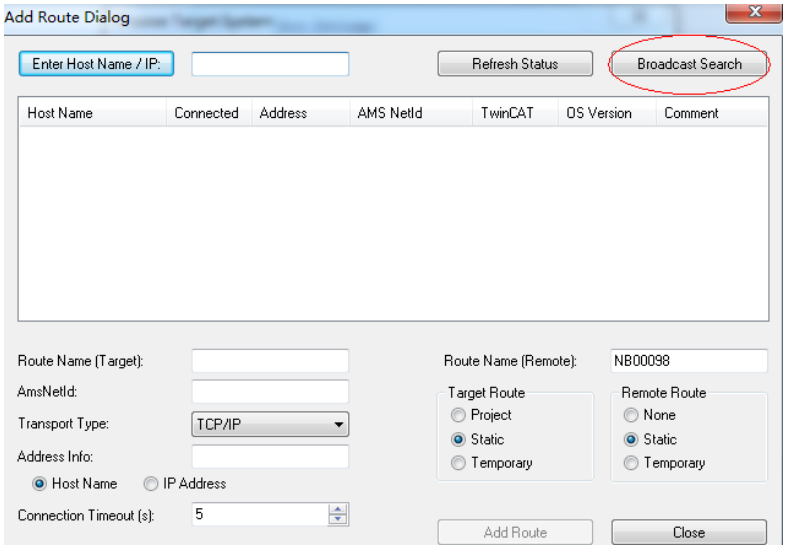
5. New project [" FILE " → " New "]



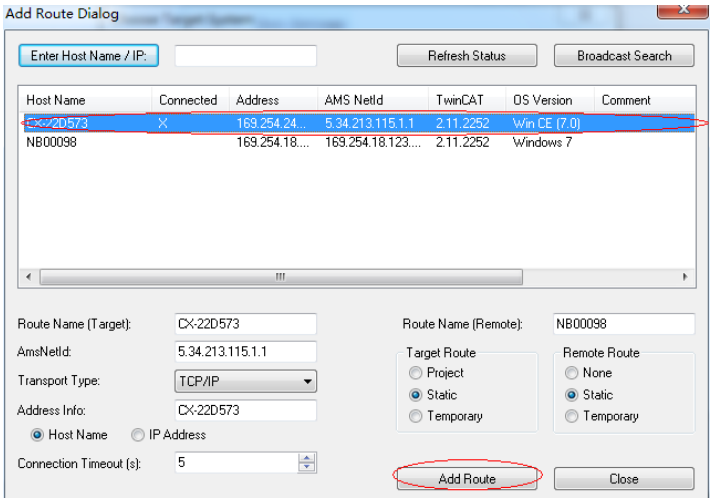
6. Connect the CX9020 and the servo drive and supply power.
7. Start searching for the CX9020 controller.



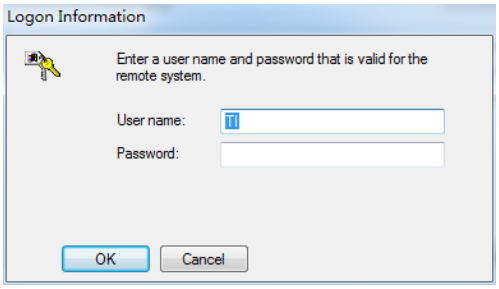
8. Click on " Broadcast Search "



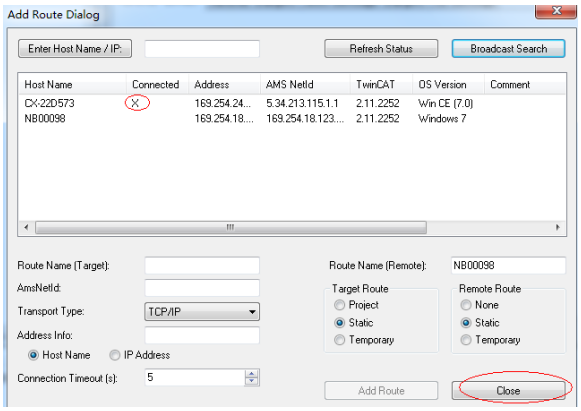
9. Select as shown below and click on “ Add Route ”



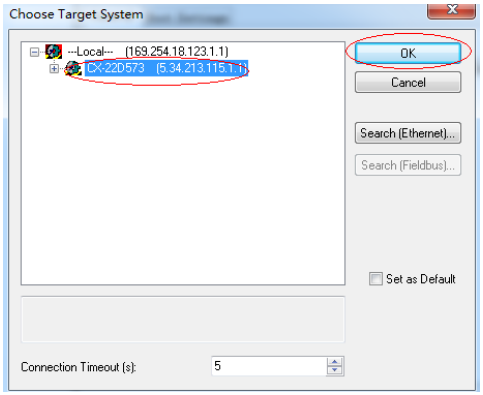
10.Click " OK "



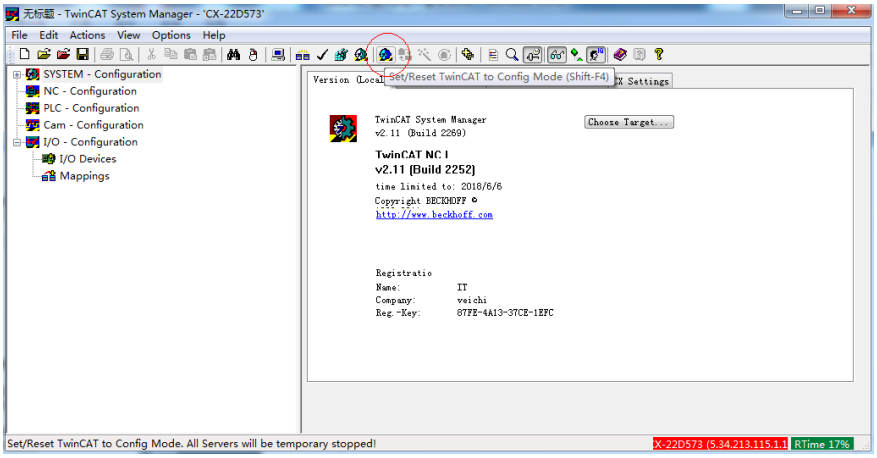
11. The " X " mark appears to indicate that it is connected, click " Close "



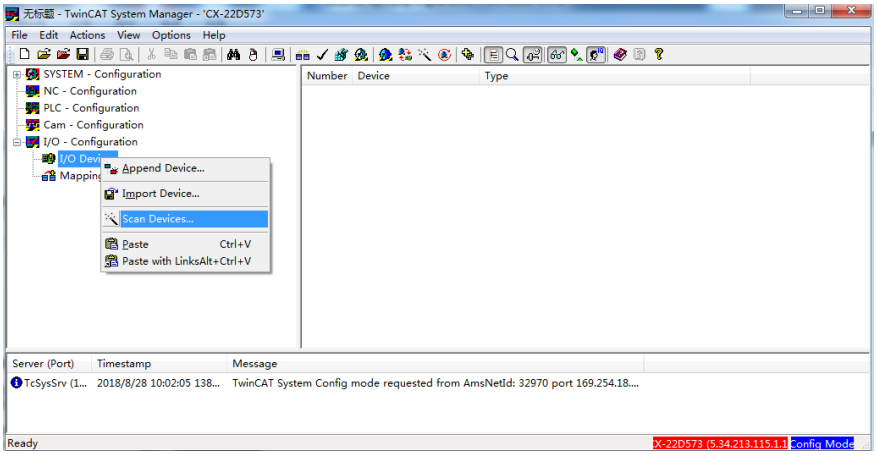
12. Select as shown below



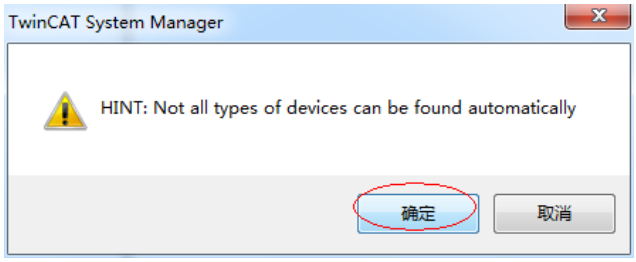
13. Put TwinCAT in configuration mode, click on the image below



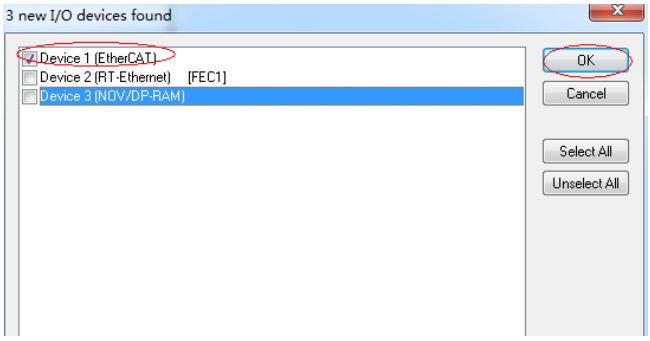
14. Right click on " I/O Device " to search for the servo device



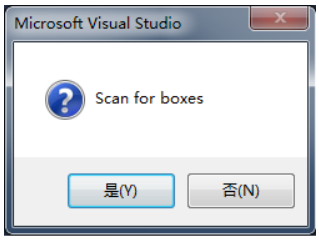
15. Click "OK"



16. Select " EtherCAT "

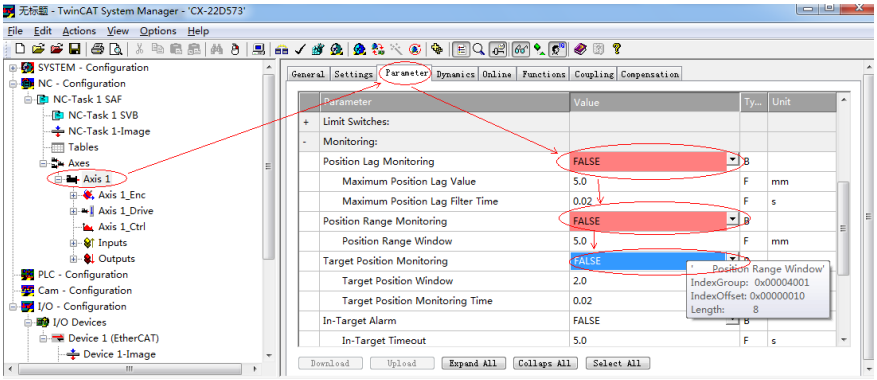


17. Select "Yes" to enable EtherCAT device detection.



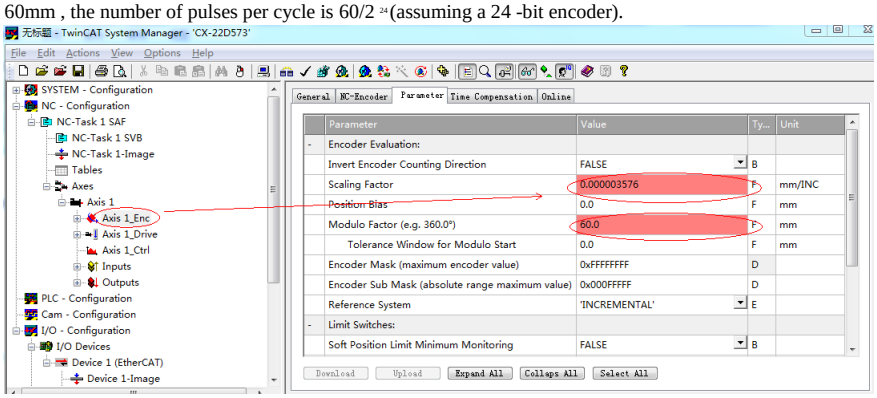
18. When the following window pops up, click No

19. Turn off the monitoring error function

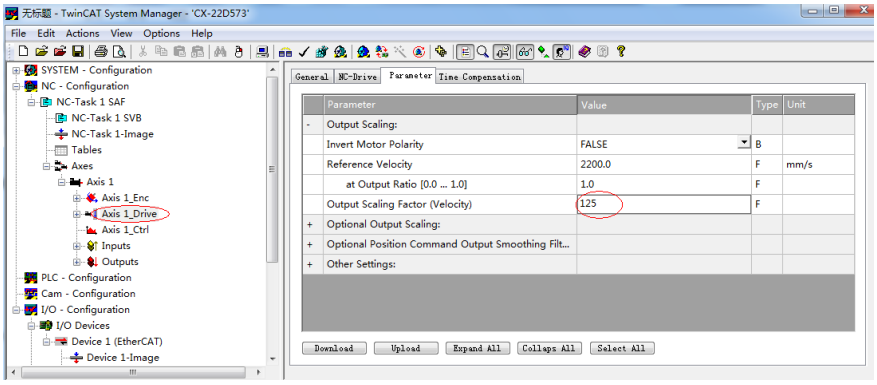


20. Set the pulse coefficient

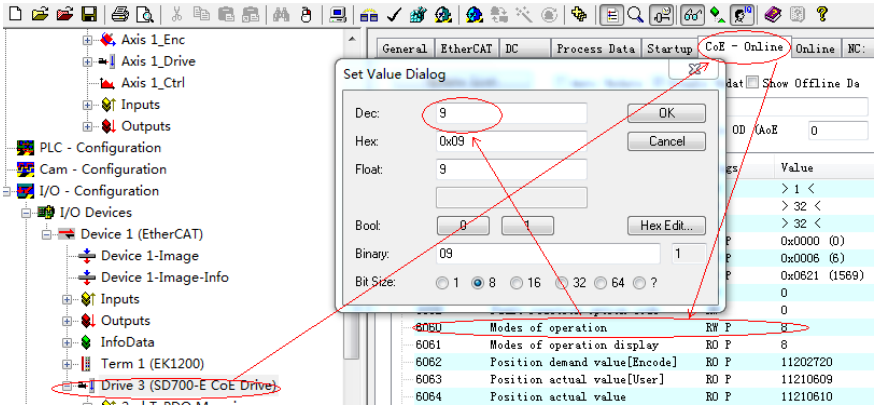
If the setting is 60mm , the number of pulses per cycle is $60/2^{24}$(assuming a 24 -bit encoder).



21. Set the speed output coefficient (used in speed mode)

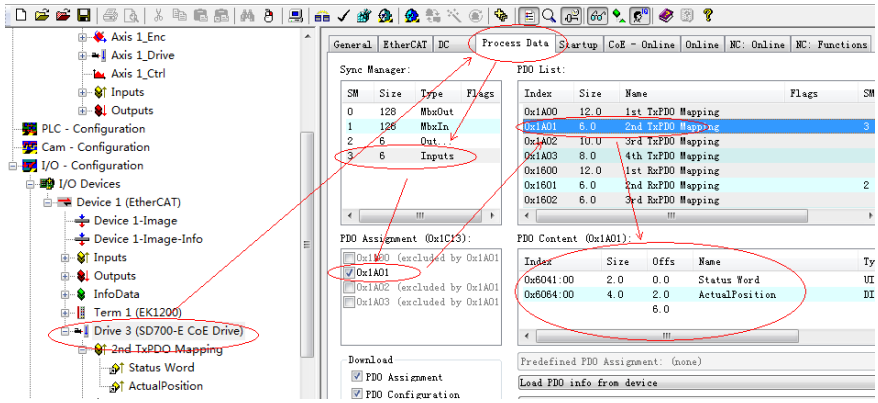


22. Set the relevant parameters through SDO (for example, change the control mode to synchronous cycle speed mode)

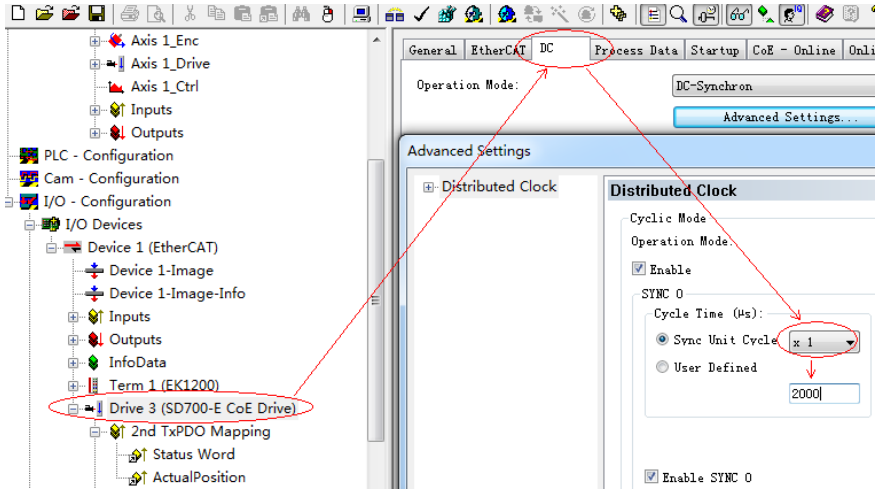


23. Configure PDO

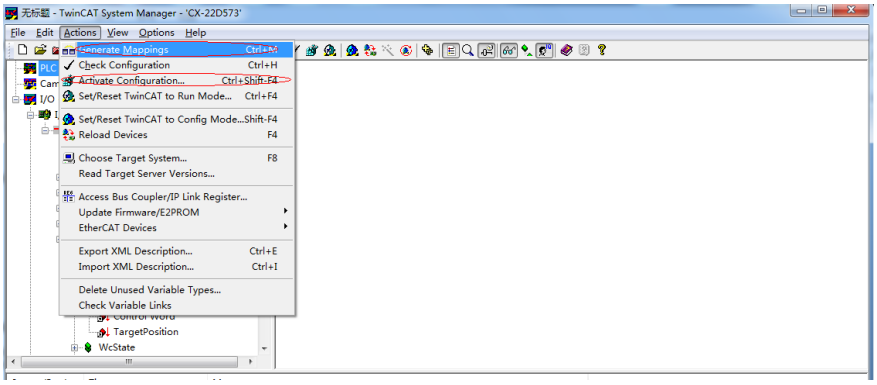
PDO mapping parameter defaults to 1601h and 1A01h , can arbitrarily change the RPDO and TPDO mapping parameters, but can only check 1 Ge RPDO and 1 Ge TPDO . The contents of the mapping object can be added and deleted, but it is recommended that the total number of bytes of the mapping object does not exceed 52 bytes.



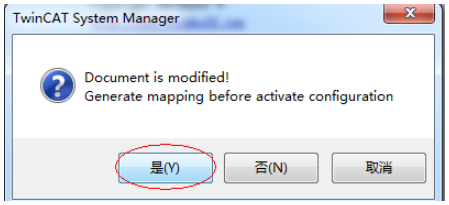
24. Distributed clock cycle setting (set cycle time according to specific conditions, default 2ms)



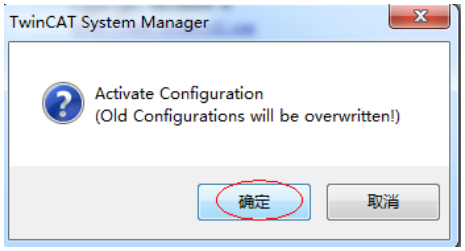
25. Activation configuration



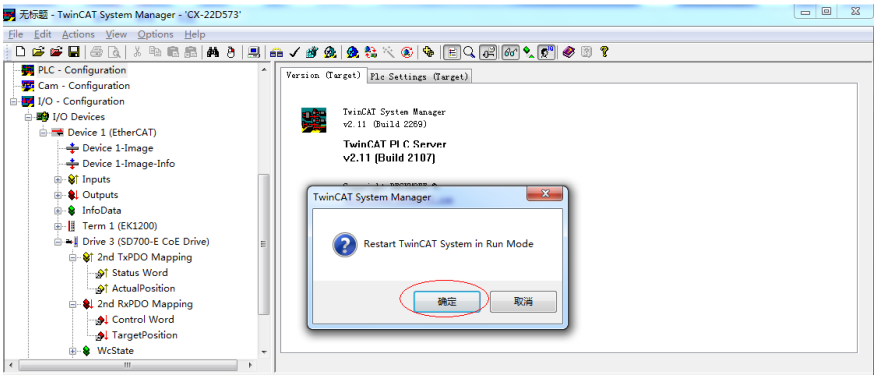
26. Click "Yes"



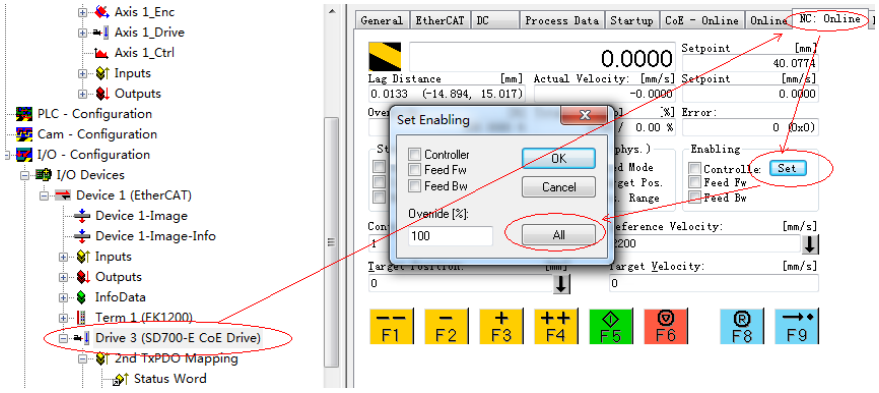
27. Click "OK"



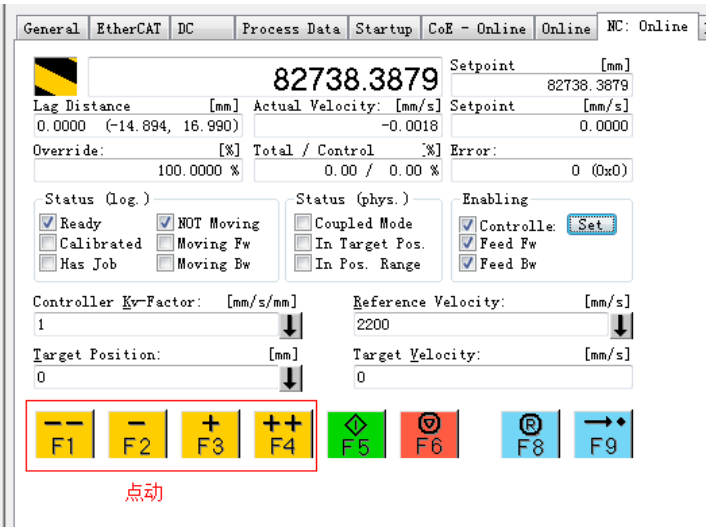
28. When the window below appears, click OK to make TwinCAT running.



29. Enable Servo Drive



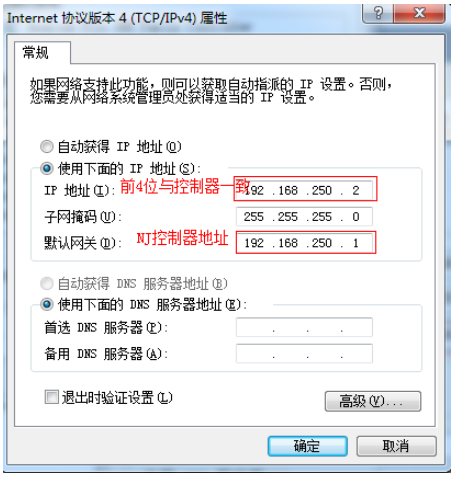
30. Jog operation



9 . With the OMRON main station operation example

9.1 Preparation

1. Install sysmac studio software, it is recommended to install version 1.09 and above.
2. Place the device description file of Flextronics " VEICHI_SD700-E_Rev4_XML_ET1100_20190228.xml " and above into the specified path : OMRON\Sysmac Studio\IO DeviceProfiles\EsiFiles\UserEsiFiles
For the first time, change the file to this path and you need to restart the sysmac studio software .
3. Set the network connection properties of the computer :
To connect the computer to the NJ controller via Etherenet , you need to set the TCP/IP properties of the computer :



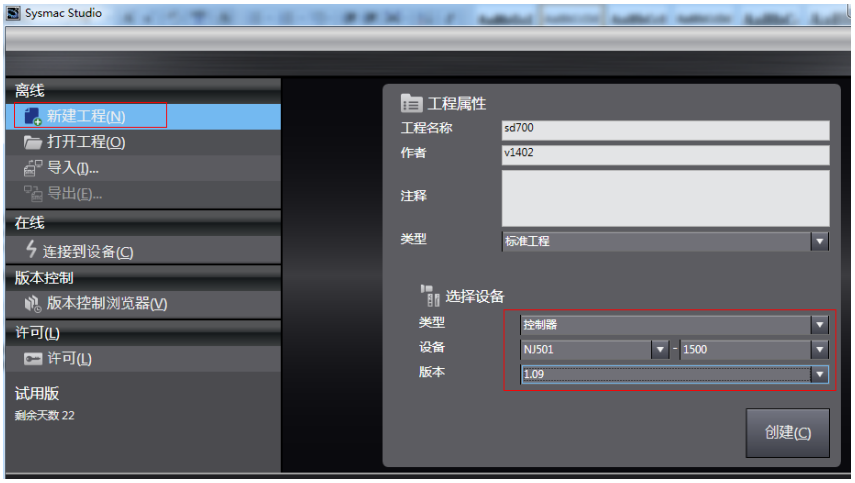
4. Confirm that the servo drive ARM software version is " 3762 " and above.

5. Since the OMRON master station cannot automatically assign the station address to the slave station, the slave station needs to manually set the node address. The SD700 sets the site alias , that is, the communication station number, through **Pn080** . It is recommended to set the station number according to the actual physical connection order in order to manage the configuration.

9.2 Omron background software configuration

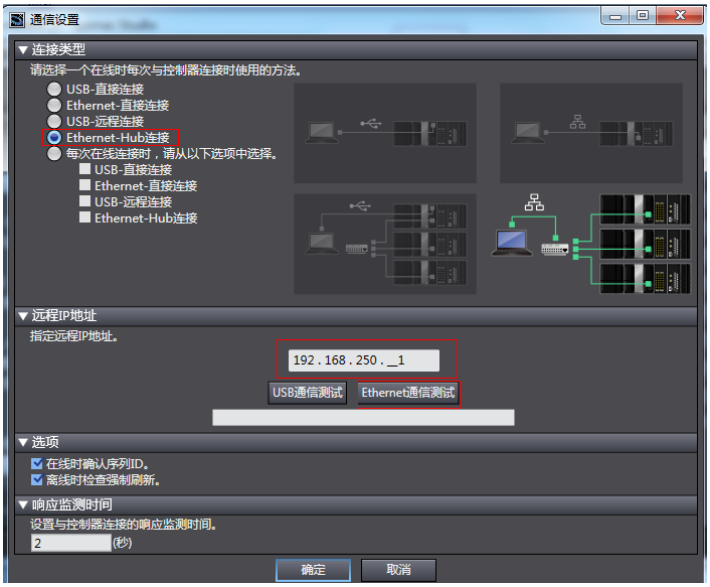
9.2 . 1 New construction

Select the controller type and software version, it is recommended that the software select version 1.09 or higher.



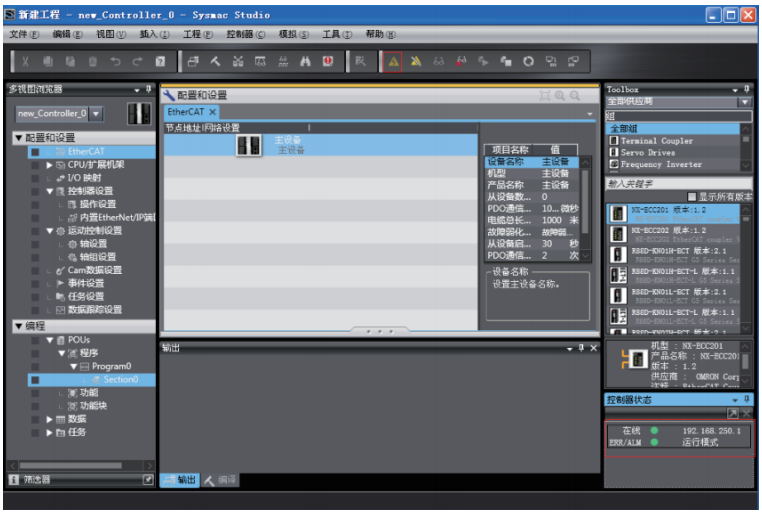
9.2.2 Communication Settings

After entering the main interface, click the menu bar "Controller" - "Communication Settings", select the Ethernet-Hub connection, type the controller network address, and click " Ethernet Communication Test". If there are no problems, the test will be successful.



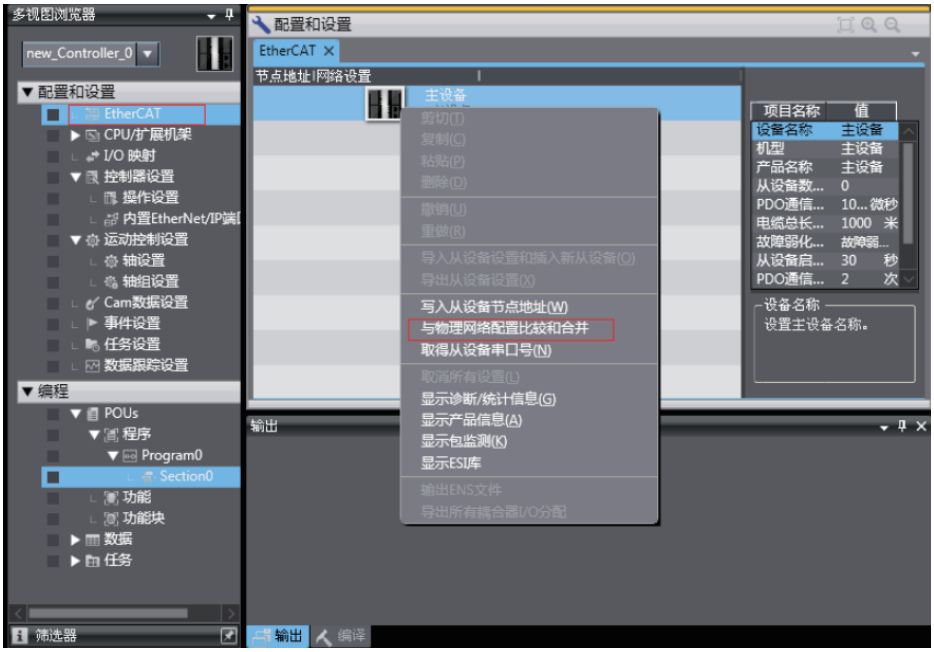
9.2.3 Scanning equipment

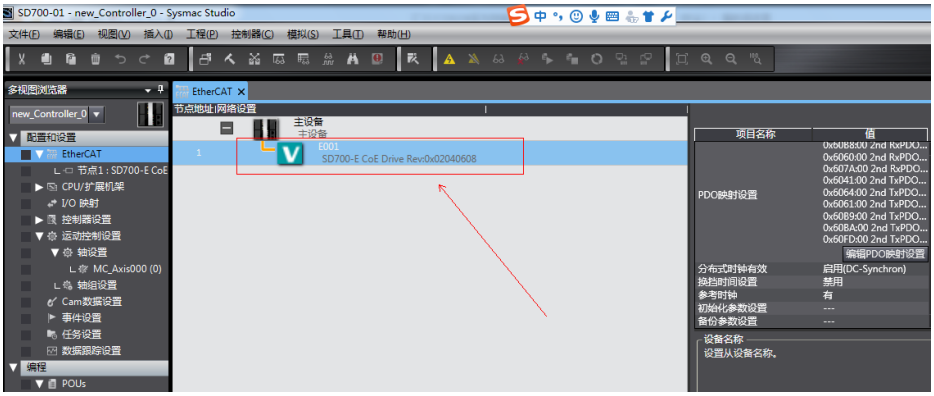
Switch the controller to "online" mode.



Scan the slave and add

In the left menu bar "Configuration and Settings", click " EtherCAT ", right-click "Master Device", and select "Compare with physical network configuration." The controller then automatically scans all slaves in the network. After scanning, click “Apply Physical Network Configuration”, then the slave is added. On the main page you can see the added SD700 slaves. As shown below.

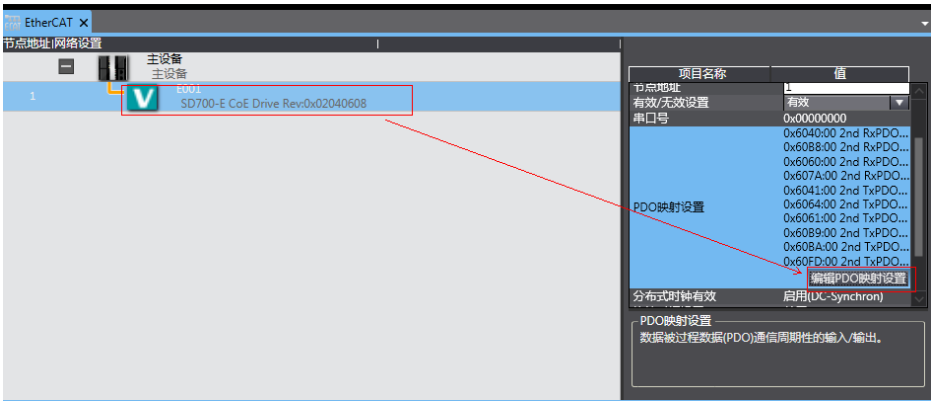


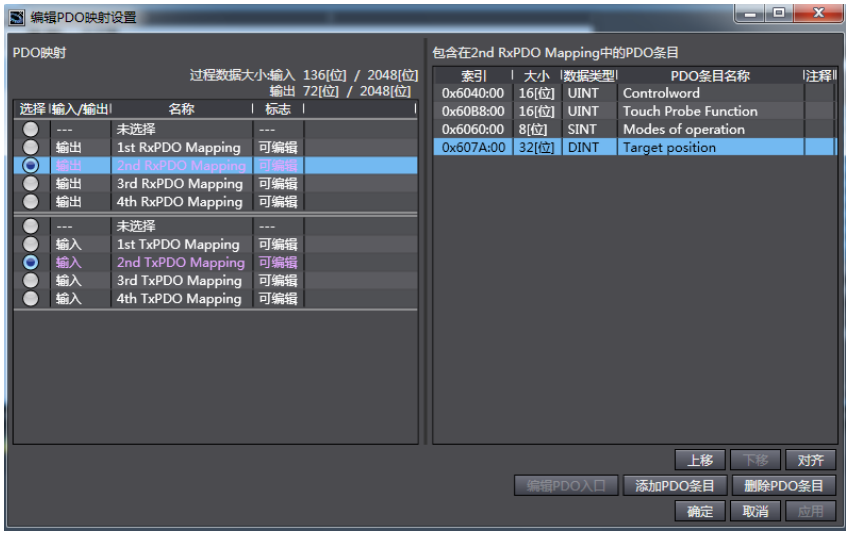


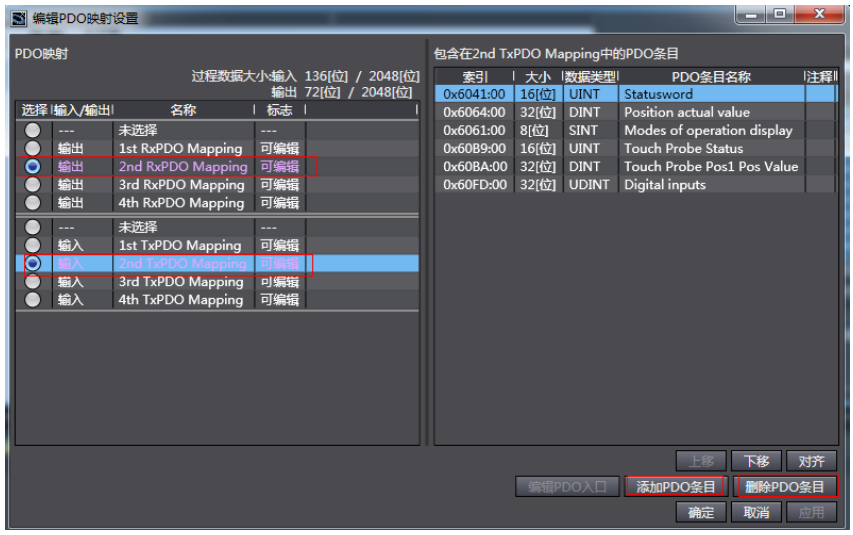
9.2.4 Parameter Configuration

Switch the controller to the offline mode and configure the PDO , DC clock, axis parameters, and so on.

Click on the scanned SD700 slave and click on "Compile PDO Mapping Configuration" on the right . After opening the can see the default choice TPDO and RPDO can change the input and output, change after, and can be for PDO content mapping that is added or deleted .



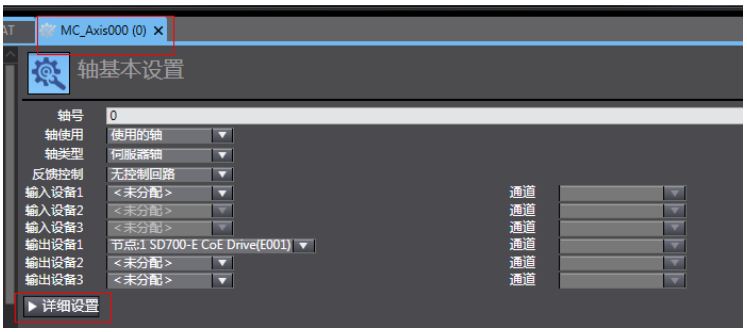




Among them, the object dictionary 0x6060 and the object dictionary 0x60B8 must be mapped, otherwise the primary station will report an error.

Right-click on "Motion Control Settings" - "Axis Settings" to add "Motion Control Axis". Then make the basic settings of the axis.

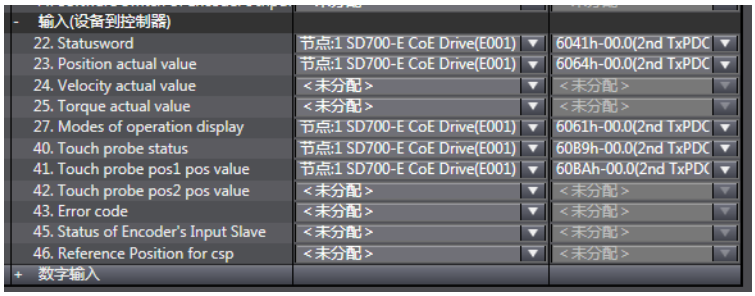
Set the basic settings of the axis as shown below, click "Detailed Settings"



After clicking on the detailed settings, you can expand the "Output", "Input" and "Digital Input". It is recommended to assign the corresponding content as shown in the figure below.

Output (controller to device):

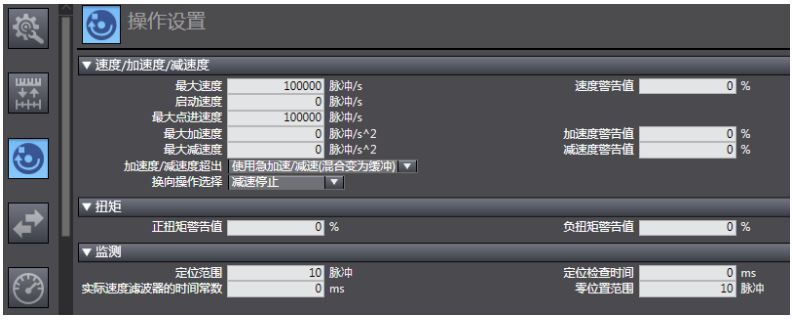
Input (device to controller) :



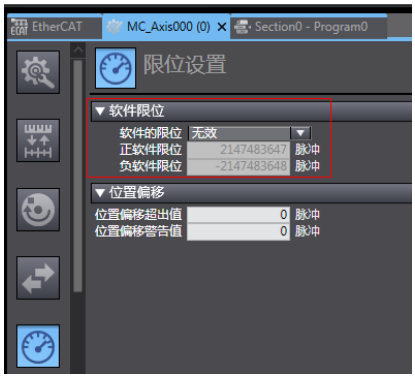
Digital input:

The actual motor resolution setting "motor revolution 1 the number of turns of the command pulse" (such as 23 is registration motor revolution is selected 8388608 pulses). The working stroke of the motor can be kept as default by one week. The electronic gear ratio is no longer set inside the servo.

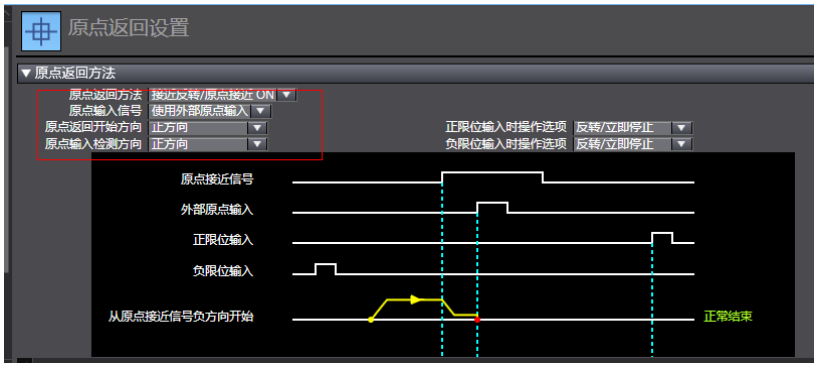
Set the speed, acceleration, deceleration, etc. according to the conversion of the unit. The acceleration or deceleration is 0 , indicating that the running curve is planned with the maximum acceleration or deceleration.



The limit function can be selected according to the actual situation.



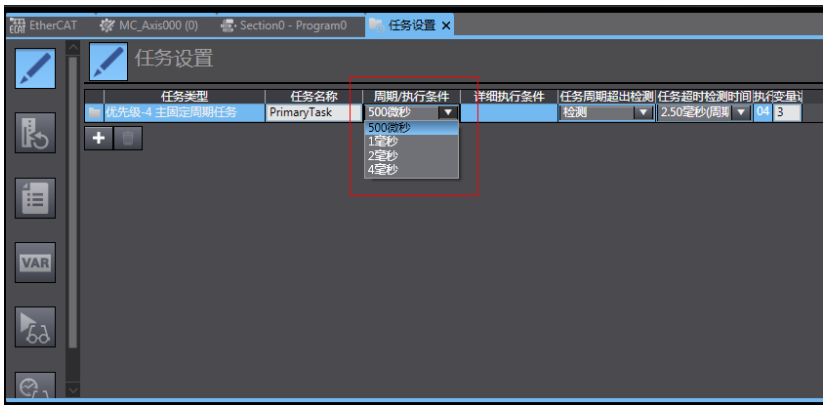
Use the following figure to set the OPR method and method.



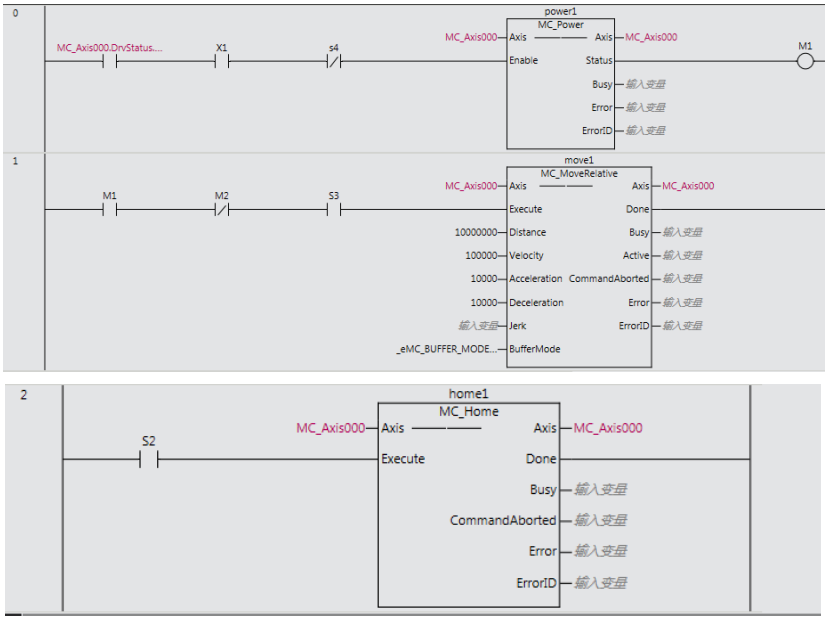
Setting the origin return method requires attention, the cooperation between the servo and the host computer function, refer to the following table.

NJ software description	**Servo response function**	**Terminal configuration**
Origin approach signal	Origin switch	DI38-DI41 , DI46
External origin input	Probe 1 or probe 2	DI44 or DI45
Z phase signal input	Motor encoder Z- phase signal	-
Positive limit input	Positive limit	DI38-DI41 , DI46
Negative limit input	Negative limit	DI38-DI41 , DI46

The distributed clock cycle can be set by the following figure



After the configuration is completed, the servo operation can be controlled by the PLC program. The following is a programming example, which can be referred to.



After all configuration and programming is completed, switch to the online state and execute the download to the controller to compare the difference between the current program and the program in the controller. After that, you can run the program and click the corresponding switch in the PLC program to drive the servo.