

# **AIMOTOR driver with CANOPEN Control manual**

**Shanghai Siheng Motor Co.,Ltd**

Address:1th,No 399 ,Songchun Road,zhaoxiang town,Qingpu

District,Shanghai,China

Website:[www.sihengmotor.com](http://www.sihengmotor.com)

## A: Product introduction

### 1.1 Product Overview

AIMOTOR driver uses CANOPEN bus communication interface, CANOPEN slave station technology, vector control technology, built-in micro-segmentation technology, adaptive filtering technology, closed-loop control technology integration, to achieve real-time control and real-time data transmission of the system, optimize the performance of the motor: has excellent stability and ultra-low noise; Smooth and accurate pure sinusoidal current vector control technology effectively reduces motor heating.

### 1.2 Product Characteristics

- The new generation of 32-bit ARM technology, high cost performance, good stability, noise, vibration performance is superior
- CANOPEN slave technology, support CIA301 and CIA402 sub-protocols, support CSP, PV, PP, PT, HM mode
- Users can set system parameters through CANOPEN bus or serial communication.

Item	Description
Link Layer protocol	CAN bus
Application layer protocol	CANopen protocol
CAN-ID type	11bit-CAN2.0A
Baud rate	500Kbit/s( Default ) 1Mbit/s 、 250 Kbit/s、 125 Kbit/s、 100 Kbit/s 、 50 Kbit/s、 20 Kbit/s
Maximum number of sites	63 pcs
CAN frame length	0~8 byte
Application layer CAN frame type	Data frames, remote frames
Terminal matching resistor	120Ω
Support sub-protocols	CiA-301 V4.02: CANopen Application Layer and Communication Protocol DSP-402 V3.0: Driver and Motion Control Sub-protocol
support services	NMT: Network Management System SDO: Service Data Object PDO: Process Data Object Device Monitoring: Including node protection and heartbeat SYNC: Including synchronization generator and synchronization reception, applied to PDO transmission
PDO transmission type	Time-triggered, event-triggered, and synchronously triggered
The number of Pdos supported	Four RPdos, four TPdos
SDO transmission mode	Accelerate SDO transmission and segment SDO transmission
Supports servo operation mode	PP (contour position mode) PV (contour speed mode), PT (contour torque mode) HM (origin back to zero mode) CSP (position synchronous mode) CSV (synchronous speed mode) CST (synchronous torque mode)

### 1.3 Networking scheme

The transmission distance of CAN bus is directly related to the baud rate and communication cable, and the relationship between the maximum bus line length and the baud rate is shown in the following table:

The default Nodeld=1, rate =500kbps, if you want to change, please use AIMotor series RS485 debugging PC to change H0C\_00 and H0C-08 parameters;

transmission distance	BAUD	Number of Nodes	Twisted pair cable diameter
25m	1Mbps	64	0.2mm <sup>2</sup>
95m	500kbps	64	0.3mm <sup>2</sup>
560m	100kbps	64	0.5mm <sup>2</sup>
1100m	50kbps	64	0.75mm <sup>2</sup>

Figure 1 Networking scheme 1

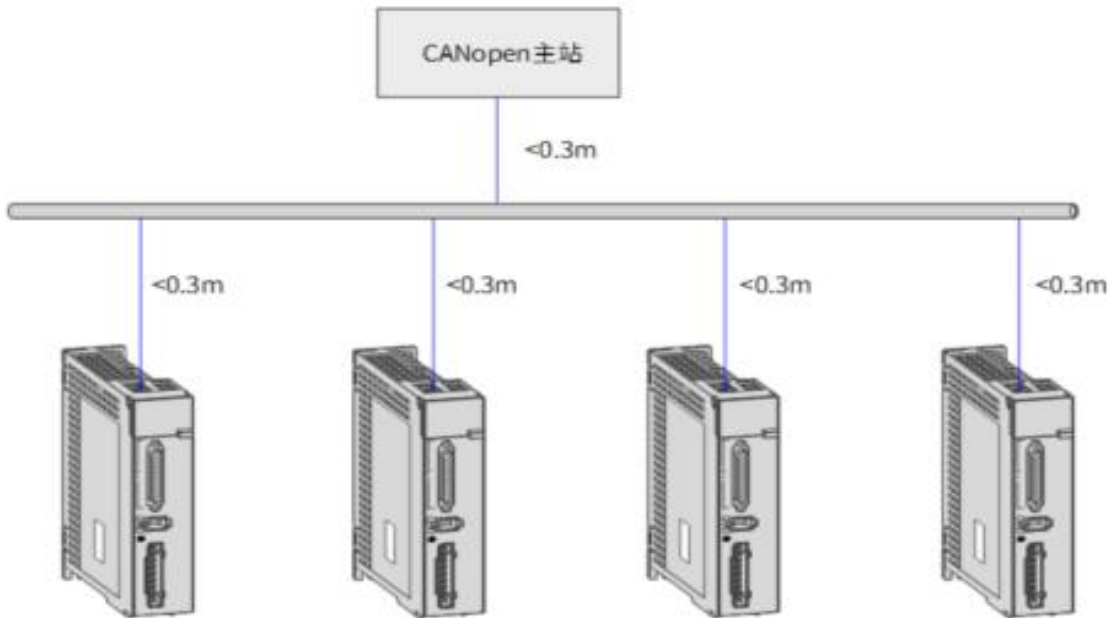
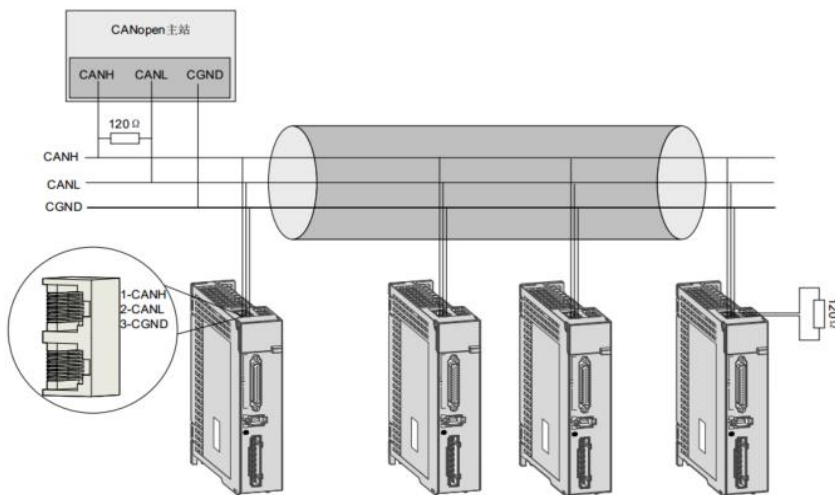


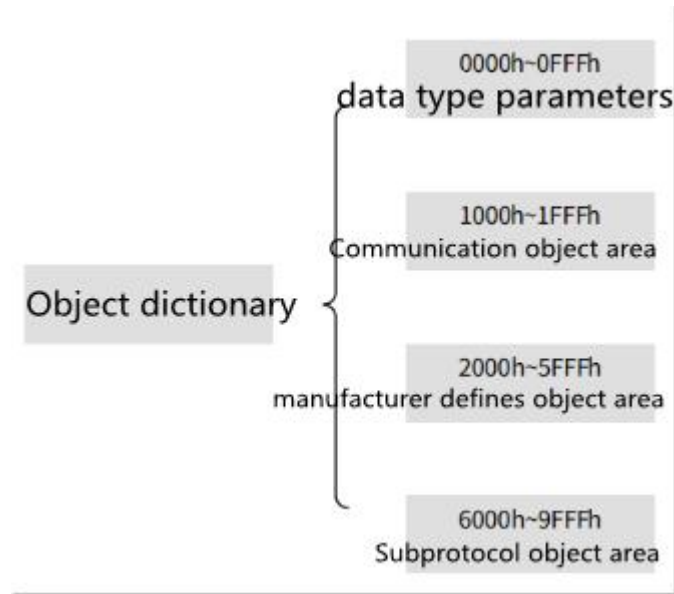
Figure 2 Networking scheme 2



## B: Parameter description and setting of object dictionary

### 2.1 Object dictionary

The object dictionary is the most important part of the device specification. It is an ordered set of parameters and variables, containing all parameters of the device description and the network state of the device. A set of objects that can be accessed through a network in an ordered, predefined manner. CANOpen protocol uses an object dictionary with a 16-bit index and an 8-subindex. The structure of the object dictionary is shown in the figure as below:



0000h-0FFFh User has no access significance.

1000h-1FFFh defines communication object parameters and device types according to CIA301 protocol.

2000h-5FFFh Manufacturer defined object area, we define all AIMOTOR drive parameter function code objects in this area;

6000h-9FFFh Some common CIA402 control objects defined by the CIA402 protocol

The corresponding relationship between AIMOTOR drive parameter function code and object dictionary is as follows: Object dictionary index =  $0x2000 + \text{function code group number}$ ; Object dictionary subindex = hexadecimal + 1 offset in function code set

Example of mapping relationship between drive parameter function code and object dictionary:

Parameter function code			Corresponding object dictionary address	
Parameter function code name	Function code group number (hexadecimal)	In-group bias	Index (hexadecimal)	Subindex (hexadecimal)
H02-00	0x02	00	0x2002	0x01
H03-09	0x03	09	0x2003	0x0A
H0B-26	0x0B	26	0x200B	0x1B

2.1.1 CIA301 subprotocol communication parameters

Index	Sub-Index	Name	Description	Type	Nature	False	Range
1000	0	device type	Consistent with CIA rules	Uint32	RO	-	-
1001	0	Error register	No effect, reserved to increase	Uint8	RO	0	-
1003	0	The current number of alarms	1. 00h indicates the current number of recorded alarms, and only 0 can be written to clear the records. 2-0, 01 h ah historical alarm cache entry data: bit00 - bit15, standard error code bit16 - bit23 manufacturers custom error code	Uint8	RW	0	0~9
	1-0A	Historical alarm cache entries		Uint32	RO	-	0~0xFFFFFFFF
1005	0	Sync Messages COB-ID	Used in the periodic synchronization function	Uint32	RW	0x80	0~0xFFFFFFFF
100A	0	Software version	AIMOTOR_CIA402	str	RO	-	-
100C	0	Node guard time	When used under the node daemon function, if 100C is not 0, the node daemon function will be enabled. The guardian timeout = 100C*100D	Uint16	RW	0	0~32767
100D	0	Life factor		Uint8	RW	0	0~255
1010	00	Subindex number	Write 0x65766173 (save) to the corresponding sub-index to save the current cia301 or cia402 object value to the device's eeprom storage. Power on again to retrieve the eeprom value as the object value	Uint8	RO	4	-
	01	Save all parameters					
	02	Save the cia301 object					
	03	Save the cia402 object		Uint32	RW	0	0~0xFFFFFFFF
1011	00	Subindex number	Writing 0x64616F6C (load) to the corresponding sub-index will restore the current cia301 or cia402 object value to the factory default	Uint8	RO	4	-
	01	Restore all objects					
	02	Restore the cia301 object					
	03	Restore the cia402 object		Uint32	RW	0	0~0xFFFFFFFF
1014	0	Emergency message COB-ID	Emergency packet identifier	Uint32	RO	0x80+ID	
1016	0	Subindex number	Sets the consumer heartbeat time bit0 - bit15, monitor the heartbeat timeout ms bit16 - bit23 monitoring node id	Uint8	RO	1	-
	1	Consumer heartbeat time		Uint32	RW	0	0~0xFFFFFFFF

1017	0	Producer heartbeat time	Set the producer heartbeat time (the interval between heartbeats generated by this machine) in ms	Uint32	RW	0	0~0xFFFFFFFF
1200	0	Subindex number		Uint8	RO	2	-
	1	Client to server COB-ID	SDO Communication COB-ID	Uint32	RO	0x600+ID	-
	2	Server to client COB-ID		Uint32	RO	0x580+ID	-
Index	Sub-Index	Name		Description	Type	Nature	False
1400	0	Subindex number	Set RPDO1 communication parameters, transmission type 000-240, for the synchronous transmission mode. The transmission type is set to 254-255, indicating asynchronous transmission mode. Synchronous transmission is triggered and takes effect by synchronous frames. Changes in asynchronous transmission data take effect immediately	Uint8	RO	6	-
	1	COB-ID of RPDO1		Uint32	RO	0x200+ID	-
	2	Transmission type of RPDO1		Uint8	RW	1	0~255
1401	0	Subindex number	Setup RPDO2 communication interface	Uint8	RO	6	-
	1	COB-ID of RPDO2		Uint32	RO	0x300+ID	0~0xFFFFFFFF
	2	Transmission type of RPDO2		Uint8	RW	1	0~255
1402	0	Subindex number	Setup RPDO3 communication interface	Uint8	RO	6	-
	1	COB-ID of RPDO3		Uint32	RO	0x400+ID	-
	2	Transmission type of RPDO3		Uint8	RW	1	0~255
1403	0	Subindex number	Setup RPDO4 communication interface	Uint8	RO	6	-
	1	COB-ID of RPDO4		Uint32	RO	0x500+ID	-
	2	Transmission type of RPDO4		Uint8	RW	1	0~255
1600	0	Number of RPDO1 mappings	Setup RPDO Group 1 Mapping object;	Uint8	RO	8	-
	1	RPDO1 Mapping object 1	Factory default mapping is as follows:	Uint32	RW	0x60400010	0~0xFFFFFFFF
	2	RPDO1 Mapping object 2	Subindex 1 maps 6040 control words;	Uint32	RW	0x60600008	0~0xFFFFFFFF

	3	RPDO1 Mapping object 3	Subindex 2 maps 6060 control mode;	Uint32	RW	0x607A0020	0~0xFFFFFFFF
	4-8	RPDO1 Mapping object 4~8	Subindex 3 maps 607A target location; Subindexes 4-8 are unmapped and can be allocated on demand.	Uint32	RW	0x00000000	0~0xFFFFFFFF
1601	0	Number of RPDO2 mappings	Set an RPDO group 2 mapping object.	Uint8	RO	8	-
	1	RPDO2 Mapping object 1	The factory default mapping is as follows:	Uint32	RW	60FF0020	0~0xFFFFFFFF
	2	RPDO2 Mapping object 2	Subindex 1 maps 60FF target speed;	Uint32	RW	60710010	0~0xFFFFFFFF
	3-8	RPDO2 Mapping object 3~8	Subindex 2 maps 6071 target torque; Subindexes 3-8 are unmapped and can be allocated on demand.	Uint32	RW	0x00000000	0~0xFFFFFFFF
1602	00	Number of RPDO3 mappings	Set the RPDO group 3 mapping object.	Uint8	RO	8	-
	01-08	RXPDO3 Maps objects 1 to 8	Subindexes 1-8 are unmapped and can be allocated on demand.	Uint32	RW	0x00000000	0~0xFFFFFFFF
1603	0	Number of RPDO4 mappings	Set an RPDO group4 mapping object.	Uint8	RO	8	-
	01-08	RXPDO4 Maps objects 1 to 8	Subindexes 1-8 are unmapped and can be allocated on demand.	Uint32	RW	0x00000000	0~0xFFFFFFFF

Index	Sub-Index	Name	Description	Type	Nature	False	Range
1800	0	Subindex number	Set TPDO1 communication parameters, transmission type	Uint8	RO	6	-
	1	COB-ID of TPDO1	000-240, for the synchronous transmission mode. The	Uint32	RO	0x180+ID	-
	2	Transport type of TPDO1	transmission type is set to 254-255, indicating asynchronous transmission mode. Synchronous transmission is triggered by synchronous frames for upload.	Uint8	RW	1	0~255
	5	The event timer of TPDO1	When the asynchronous transmission data changes or the event timer time is up, it is uploaded immediately	Uint16	RW	0	0-65535

1801	0	Subindex number	Set TPDO2 communication parameters	Uint8	RO	6	-
	1	COB-ID of TPDO2		Uint32	RO	0x280+ID	-
	2	Transport type of TPDO2		Uint8	RW	1	0~255
	5	The event timer of TPDO2		Uint16	RW	0	0-65535
1802	0	Subindex number	Set TPDO3 communication parameters	Uint8	RO	6	-
	1	COB-ID of TPDO3		Uint32	RO	0x380+ID	-
	2	Transport type of TPDO3		Uint8	RW	1	0~255
	5	The event timer of TPDO3		Uint16	RW	0	0-65535
1803	0	Subindex number	Set TPDO4 communication parameters	Uint8	RO	6	-
	1	COB-ID of TPDO4		Uint32	RO	0x480+ID	-
	2	Transport type of TPDO4		Uint8	RW	1	0~255
	5	The event timer of TPDO4		Uint16	RW	0	0-65535
1A00	0	Number of TPDO1 mappings	Set the TPDO group 1 mapping object.	Uint8	RO	8	-
	1	TPDO1 Mapping object 1	The factory default mapping is as follows:	Uint32	RW	0x60410010	0~0xFFFFFFFF
	2	TPDO1 Mapping object 2	Subindex 1 maps 6041 status words;	Uint32	RW	0x60610008	0~0xFFFFFFFF
	3	TPDO1 Mapping object 3	Subindex 2 maps 6061 display mode;	Uint32	RW	0x60640020	0~0xFFFFFFFF
	4-8	TPDO1 Mapping object 4~8	Subindex 3 maps 6064 actual location; Subindexes 4-8 are unmapped and can be allocated on demand.	Uint32	RW	0x00000000	0~0xFFFFFFFF
1A01	0	Number of TPDO2 mappings	Set the mapping object of TPDO group 2.	Uint8	RO	8	-
	1	TPDO2 Mapping object 1	The factory default mapping is as follows:	Uint32	RW	606C0020	0~0xFFFFFFFF
	2	TPDO2 Mapping object 2	Subindex 1 maps 606C actual speed;	Uint32	RW	60770010	0~0xFFFFFFFF
	3-8	TPDO2 Mapping object 3~8	Subindex 2 maps 6077 actual torque; Subindexes 3-8 are unmapped and can be allocated on demand.	Uint32	RW	0x00000000	0~0xFFFFFFFF

1A02	00	Number of RPDO3 mappings	Set the mapping object of TPDO group 3.	Uin8	RO	8	-
	01-08	RXPDO3 Maps objects 1 to 8	Subindexes 1-8 are unmapped and can be allocated on demand.	Uin32	RW	0x00000000	0~0xFFFFFFFF
1A03	0	Number of RPDO4 mappings	Set the mapping object of TPDO group 4.	Uin8	RO	8	-
	01-08	RXPDO4 Maps objects 1 to 8	Subindexes 1-8 are unmapped and can be allocated on demand.	Uin32	RW	0x00000000	0~0xFFFFFFFF

### 2.1.2 Manufacturer custom parameters

The 2000h-5FFFh manufacturer defines the object area. Our company has defined all the parameter function code objects of the AIMOTOR driver in this area. Each parameter has a corresponding object dictionary index and index, AIMOTOR drive parameter function code is as follows: the corresponding relationship between object dictionary object dictionary index = 0 x2000 + function block number; Object dictionary sub-index = hexadecimal offset within the function code group + 1;

Example: Mapping relationship between drive parameter function code and object dictionary

Parameter function code			Corresponding object dictionary address	
Parameter function code name	Function code group number (hexadecimal)	In-group bias	Index (hexadecimal)	Subindex (hexadecimal)
H02-00	0x02	00	0x2002	0x01
H03-09	0x03	09	0x2003	0x0A
H0B-26	0x0B	26	0x200B	0x1B
H12-20	0x12	20	0x2012	0x15

Due to the excessive parameter function code, it is not listed in the CANOPEN manual, please refer to the same series of standard pulse product manual parameter description.

◆ When using SDO communication to read and write the manufacturer's custom parameter object, please pay attention to the attribute of the parameter;

1.If the modify mode property is < Enable disconnect > parameter, you need to disconnect the motor enable and then write the parameter object value;

2.If the factory value attribute is < registered value > parameter, the user can only read but not write;

◆ H00 group, H01 group, H0B group parameters, users can only read but not write;

◆ SDO communication error abort code

Abort code	Tag	Description
0x00000000	OD_SUCCESSFUL	SDO communication successful
0x06010001	OD_READ_NOT_ALLOWED	Trying to read only write objects
0x06010002	OD_WRITE_NOT_ALLOWED	Attempting to write to a read-only object
0x06020000	OD_NO_SUCH_OBJECT	Object does not exist in the dictionary
0x06040041	OD_NOT_MAPPABLE	Objects cannot be mapped in PDO
0x06060000	OD_ACCES_FAILED	Access failed due to hardware error
0x06070010	OD_LENGTH_DATA_INVALID	Data type and length of the service parameter do not match
0x06090011	OD_NO_SUCH_SUBINDEX	Word index does not exist
0x06090030	OD_VALUE_RANGE_EXCEEDED	Out of parameter value range (write access only)
0x06090031	OD_VALUE_TOO_LOW	Parameter value is too high
0x06090032	OD_VALUE_TOO_HIGH	Parameter value is too low
0x05030000	SDOABT_TOGGLE_NOT_ALTERNED	Switching bits do not alternate
0x05040000	SDOABT_TIMED_OUT	SDO protocol communication timed out
0x05040001	SDOABT_CS_NOT_VALID	Command specifier is invalid
0x05040002	SDOABT_INVALID_BLOCK_SIZE	Invalid block size (block mode only)
0x05040005	SDOABT_OUT_OF_MEMORY	Memory Overflow
0x08000000	SDOABT_GENERAL_ERROR	SDO message size is incorrect
0x08000021	SDOABT_LOCAL_CTRL_ERROR	Due to local control, data cannot be transferred or stored in the application

2.1.3 Cia402 shaft control parameter group

Index	Sub-Index	Name	Description	Type	Nature	False	Range																		
603F	0	Fault Code	The fault codes currently generated by the driver are detailed in the description of Section "6.0 Fault Code Information Table".	Uint16	RO	0	0~65535																		
6040	0	Control Word	For details, see section "2.4 6040 Control Words".	Uint16	RW	0	0~65535																		
6041	0	Status Word	For details, see section "2.5 6041 Status Word".	Uint16	RO	-	0~65535																		
605A	0	Quick stop code	<table border="1"> <thead> <tr> <th>Value</th> <th>Source of deceleration</th> <th>After reducing to 0</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Free shutdown</td> <td>Disable</td> </tr> <tr> <td>1</td> <td>0x6084</td> <td>Disable</td> </tr> <tr> <td>2</td> <td>0x6085</td> <td>Disable</td> </tr> <tr> <td>3</td> <td>0x6084</td> <td>Maintain the lock shaft</td> </tr> <tr> <td>4</td> <td>0x6085</td> <td>Maintain the lock shaft</td> </tr> </tbody> </table>	Value	Source of deceleration	After reducing to 0	0	Free shutdown	Disable	1	0x6084	Disable	2	0x6085	Disable	3	0x6084	Maintain the lock shaft	4	0x6085	Maintain the lock shaft	Uint16	RW	1	0~65535
			Value	Source of deceleration	After reducing to 0																				
			0	Free shutdown	Disable																				
			1	0x6084	Disable																				
			2	0x6085	Disable																				
			3	0x6084	Maintain the lock shaft																				
4	0x6085	Maintain the lock shaft																							
0: No action																									
1: PP (contour position mode)																									
3: PV (contour speed mode)																									
4: PT (contour torque mode)																									
6: HM search pattern (the source)																									
8: CSP (position loop synchronous mode)																									
9: CSV (loop synchronous speed mode)																									
10: CST (loop synchronous torque mode)																									
6060	0	Run mode setting	<p>0: No action</p> <p>1: PP (contour position mode)</p> <p>3: PV (contour speed mode)</p> <p>4: PT (contour torque mode)</p> <p>6: HM search pattern (the source)</p> <p>8: CSP (position loop synchronous mode)</p> <p>9: CSV (loop synchronous speed mode)</p> <p>10: CST (loop synchronous torque mode)</p>	Int8	RW	1	0~10																		
6061	0	Run mode state	<p>Query the status of 6060H;</p> <p>0;No action</p> <p>1: PP (position)</p> <p>3: PV (speed mode)</p> <p>4: PT (contour torque mode)</p> <p>6: HM search pattern (the source)</p> <p>8: CSP (position loop synchronous mode)</p>	Int8	RO	-	0~10																		

AIMOTOR (ac/dc servo driver)CANOPEN control manual

			9: CSV (loop synchronous speed mode) 10: CST (loop synchronous torque mode)				
60FC	0	Internal instruction position	Position instructions generated within the drive; Unit: Enc	Int32	RO	-	-2147483648~ 2147483647
6062	0	Internal instruction position	Position instructions generated within the drive; Unit: Pul	Int32	RO	-	-2147483648~ 2147483647
6063	0	Actual position	Actual motor location, Unit: Enc	Int32	RO	-	-2147483648~ 2147483647
6064	0	Actual position	Actual motor location, Unit: Pul	Int32	RO	-	-2147483648~ 2147483647
606B	0	Internal command speed	Speed instructions generated within the driver; Unit: Pul/s	Int32	RO	-	-2147483648~ 2147483647
606C	0	Actual speed	Actual motor speed, Unit: Pul/s	Int32	RO	-	-2147483648~ 2147483647
6071	0	Target torque	Set PT(Contour torque mode) running torque; (1000 indicates rated torque) Unit: 0.1%	Int16	RW	0	-3000~3000
6074	0	Internal instruction torque	Torque instructions generated inside the drive; Unit: 0.1%	Int16	RO	-	-3000~3000
6077	0	Actual torque	Actual torque of the motor, Unit: 0.1%	Int16	RO	-	-3000~3000
6078	0	Actual current	Actual current of the motor, Unit: 0.01A	Int16	RO	-	-32767~32767
Index	Sub-Index	Name	Description	Type	Nature	False	Range
607A	0	Target location	Set PP(contour position mode) running position; Unit: Pul	Int32	RW	0	-2147483648~ 2147483647
607C	0	Origin Offset	Set HM(back to origin mode) origin offset; Unit: Pul	Int32	RW	0	-2147483648~ 2147483647
607D	0	Number of sub-indexes	-	Uint8	RO	2	0~255
	1	Negative limit of the software	Software Settings take effect CANOPEN pattern software limit, limit function	Int32	RW	-2147483648	-2147483648~ 2147483647

AIMOTOR (ac/dc servo driver) CANOPEN control manual

	2	Software forward limit	also need to set H0A - 40 parameters	Int32	RW	2147483647	-2147483648~2147483647
607E	0	Command polarity	Set the instruction direction logic in PP/PV/PT contour mode;	UInt8	RW	0	0~255
607F	0	Maximum profile velocity	Set the maximum speed limit for PT(contour torque mode) operation; Unit: Pul/s	UInt32	RW	0	-2147483648~2147483647
6081	0	Profile velocity	Set the maximum speed of PP(contour position mode); Unit: Pul/s	Int32	RW	0	-2147483648~2147483647
6083	0	Contour acceleration	Set the acceleration of PP(profile position mode) and PV(profile velocity mode); Unit: Pul/s <sup>2</sup>	Int32	RW	100000	-2147483648~2147483647
6084	0	Contour deceleration	Set the speed reduction of PP(contour position mode) and PV(contour velocity mode); Unit: Pul/s <sup>2</sup>	Int32	RW	100000	-2147483648~2147483647
6085	0	Quick stop and deceleration	Set quick stop and speed reduction in PP/PV/PT profile mode; Unit: Pul/s <sup>2</sup>	Int32	RW	500000	-2147483648~2147483647
6098	0	Return to origin mode	Current can be set way back to the origin of value is: 01/02/04/06/17/18/24/28/33/34/35/37/38; For details, please refer to the description in Chapter "2.8 Return to Origin Mode Method".	Int8	RW	35	0~38
6099	0	Subindex number	-	UInt8	RO	2	0~255
	1	Origin velocity 1	HM(Back to origin mode) Approach to the origin speed: high speed to find the origin; Unit: Pul/s	Int32	RW	10000	-2147483648~2147483647
	2	Origin velocity 2	HM(back to origin mode) to determine the origin speed: low speed to find the origin; Unit: Pul/s	Int32	RW	1000	-2147483648~2147483647
609A	0	Return to origin acceleration and deceleration	The rate of addition and subtraction of HM(return to origin mode) back to origin; Unit: Pul/s <sup>2</sup>	Int32	RW	10000	-2147483648~2147483647
60FD	0	Digital signal input status	DI1 - DI7 input port state feedback 2 hexadecimal bit15 - bit21 said	UInt32	RO	-	0~127
60FE	0	Subindex number	-	UInt8	RO	2	0~255

AIMOTOR (ac/dc servo driver)CANOPEN control manual

	1	Digital signal output	For details, please refer to the description in Section '2.1060FEh'	Uint32	RW	-	-2147483648~2147483647
	2	Forced masking of digital quantity signals		Uint32	RW	0	-2147483648~2147483647
60FF	0	target speed	Set the PV(contour speed mode) running speed; Unit: Pul/s	Int32	RW	0	-2147483648~2147483647
6502	0	operation mode supported by the driver	Used to query the CIA402 motion control modes supported by the product, 0x03AD indicates support for PP,PV,PT,HM,CSP,CSV, and CST operation modes	Uint32	RO	0x03AD	0~0xFFFFFFFF

**2.2 Each control mode and associated object dictionary**

Control Mode	Index + sub-index	Name	Data type	read-write access permission	Unit	Required or not
Synchronous Position Mode (CSP)	6060	Control mode setting =8	Int8	RW	-	√
	6040	Control word	Uint16	RW	-	√
	607A	Target position	Int32	RW	Pul	√
	6061	Control mode feedback	Int8	RO	-	
	6041	Status word	Uint16	RO	-	√
	6064	Real position	Int32	RO	Pul	√
	606C	Real speed	Int32	RO	Pul/s	
	6077	Read torque	Int16	RO	0.1%	
Profile Position Mode (PP)	6060	Control mode setting =1	Int8	RW	-	√
	6040	Control word	Uint16	RW	-	√
	607A	Target position	Int32	RW	Pul	√
	6081	Maximum speed	Int32	RW	Pul/s	√
	6083	Acceleration	DINT	RW	Pul/s <sup>2</sup>	√
	6084	Deceleration	DINT	RW	Pul/s <sup>2</sup>	√
	6061	Control mode feedback	Int8	RO	-	
	6041	Status word	Uint16	RO	-	√
	6064	Real position	Int32	RO	Pul	
	6077	Read torque	Int16	RO	0.1%	
Profile Velocity Mode (PV)	6060	Control mode setting =3	Int8	RW	-	√
	6040	Control word	Uint16	RW	-	√
	60FF	Target speed	DINT	RW	Pul/s	√
	6083	Acceleration	DINT	RW	Pul/s <sup>2</sup>	(PV)√
	6084	Deceleration	DINT	RW	Pul/s <sup>2</sup>	(PV)√
	6061	Control mode	Int8	RO	-	

		feedback				
	6041	Status word	Uint16	RO	-	√
	6064	Real position	Int32	RO	Pul	
	606C	Real speed	Int32	RO	Pul/s	√
	6077	Read torque	Int16	RO	0.1%	
Contour torque mode (PT)	6060	Control mode setting =4	Int8	RW	-	√
	6040	Control word	Uint16	RW	-	√
	6071	Target torque	Int16	RW	0.1%	√
	607F	Maximum speed limit	Uint32	RW	Pul/s	√
	6061	Control mode feedback	Int8	RO	-	
	6041	Status word	Uint16	RO	-	√
	6064	Real position	Int32	RO	Pul	
	606C	Real speed	Int32	RO	Pul/s	
	6077	Read torque	Int16	RO	0.1%	√

Control Mode	Index + subindex	Name	Data type	read-write access permission	Unit	Required or not
Back to origin mode (HM)	6060	Control mode setting =3	Int8	RW	-	√
	6040	Control word	Uint16	RW	-	√
	6098	Return-to-zero method	Int8	RW	-	√
	6099:01	Search origin high speed 1	Int32	RW	Pul/s	√
	6099:02	Find origin low speed 2	Int32	RW	Pul/s	√
	609A	Origin acceleration	Int32	RW	Pul/s <sup>2</sup>	√
	607C	Origin Offset	Int32	RW	Pul	
	6061	Control mode feedback	Int8	RO	-	
	6041	Status word	Uint16	RO	-	√
	6064	Real position	Int32	RO	Pul	
	606C	Real speed	Int32	RO	Pul/s	
Other auxiliary associated parameters	605A	Quick shutdown code	Uint16	RW	-	
	6085	Sudden stop to slow	Uint32	RW	Pul/s <sup>2</sup>	

		down				
607E	Instruction polarity	UInt8	RW	-		
607D-01	Negative limit of the software	Int32	RW	-		
607D-02	Software forward limit	Int32	RW	-		
60FD	Digital input status	UInt32	RO	-		
60FE-01	Digital output status	UInt32	RW	-		
60FE-02	Digital output mask	UInt32	RW	-		

No matter which control mode is used to control the slave station, it is inseparable from the read and write operation of the two object dictionaries of 6040H(control word) and 6041H(status word). The master and slave station use these two object dictionaries as transmission media to realize command delivery and status monitoring. The following sections focus on the definitions and meanings of the individual bits of the two object dictionaries.

### 2.3 Cia402 object dictionary position, speed, acceleration and deceleration unit calculation

The speed value in CIA402 protocol is not the unit of speed per minute r/min as we understand, but the unit of instruction pulse per second Pul/s; The unit of acceleration and deceleration is Pul/s<sup>2</sup>;

If I want to set the speed per minute, I should first convert minutes into seconds, and then multiply the number of pulses of the motor 1 turn;

AIMotor series drive factory default 1000 pulse motor rotation 1 turn; If you change the pulse number of 1 turn, please use debugging software or SDO communication mode to set H05-07 electronic gear molecule and H05-09 electronic gear denominator. For example, if you set 2000 pulse motor to rotate 1 turn, then set H05-09 parameter value 2000.

Unit conversion schematic table

Number of pulses in 1 turn	user understands the unit				Convert to units that comply with CIA402 protocol			
	Target position r	Target speed r/min	Acceleration to target acceleration time	Deceleration time to 0	Target position Pul	Target speed Pul/s	Acceleration Pul/s <sup>2</sup>	Deceleration Pul/s <sup>2</sup>
H05-07molecule H05-09 denominator	Run to 100.5 RPM	2000 rpm per minute	2 seconds to 2000 RPM	3 seconds to slow down to a full stop	100.5*1000 =100500	2000/60*1 000 =33333	33333/2 =16666	33333/3 =11111

For example, if the control mode is PP contour position mode, as shown in the table above, the position value, speed value and acceleration and deceleration value in CIA402 are calculated according to the number of pulses in 1 turn and the target position, target speed and acceleration and deceleration time expected by the user, and then the specific steps of absolute positioning operation of the motor are controlled by SDO communication.

Step1: Use the host computer to set the driver control mode parameter H02-00 to 8 (CanOpen control mode);

Step2: Use SDO communication to write the 6060h motion mode in the CIA402 object dictionary to 1 (contour position mode);

Step3: Use SDO communication to write the contour-position pattern related object in the CIA402 object dictionary - target position 607Ah into 100500; Target speed 6081h write 33333;

Acceleration 6083h write 16666; Speed down 6084h write 11111;

Step4: Use SDO communication to write the 6040h control words in the CIA402 object dictionary into 00h→06h→07h→0Fh→1F motor to start and run, and the motor will stop automatically when it runs to 100500 position.

### 2.4 6040h Bit definition of the control word

Bit	0	1	2	3	4-6	7	8	9-15
Function	Enter servo and stand by	Enter the main circuit	Execute quick stop	Execute the enable run	Operation mode correlation	Fault resetting	Reserved	NA

bit0 to bit3 have the same meaning in each servo mode. The control word 6040h sends commands in sequence according to the state transition requirements, guiding the servo to jump to the corresponding state to run related functions.

bit0~bit3 value	It indicates the state to be entered
0x0	There is no fault when entering the servo
0x6	Enter the servo and get ready
0x7	Enter the waiting enable
0xF	Enter the servo enable
0x2	Perform rapid shutdown

bit7 fault reset, bit8 operation pause, the trigger logic is that the rising edge is valid. The different operation modes of bit4 to bit6 have different functional meanings (please refer to the control command steps in Section 2.7). Control word of every single assignment a bit pointless, must with other together constitute a control instruction (refer to 2.6 control instruction and state transition).

2.5 6041 Bit definition of the status word

Bit	0	1	2	3	4	5	6	7/8	9
Function	be ready	Waiting enable	Enabled to run	Go Wrong	Powered on	Quickly stop activation	No t Runnable	N A	Remote control
	10	11	12-13	14	15				
	target reached	limit	Related to the operation mode	NA	origin has been found.				

◆bit0 to bit9 have the same meaning in each servo mode. The servo feedback represents the currently unique and determined operating state. The possible states are shown in the following table.

bit0~bit9 Value	state indicated
0x000	It is being initialized
0x250	There are no faults at present.
0x231	servo is ready
0x233	Wait for the enable to be activated
0x237	The servo has been enabled
0x217	Rapid shutdown in progress
0x218	servo is in a faulty state
0x21F	It is malfunctioning and shutting down.

◆bit10 to bit15 Among them, bit10/bit12/bit13 have different meanings in each servo mode, bit11/bit15 have the same meaning in each servo mode, and bit14 is retained by the manufacturer as meaningless. As follows table

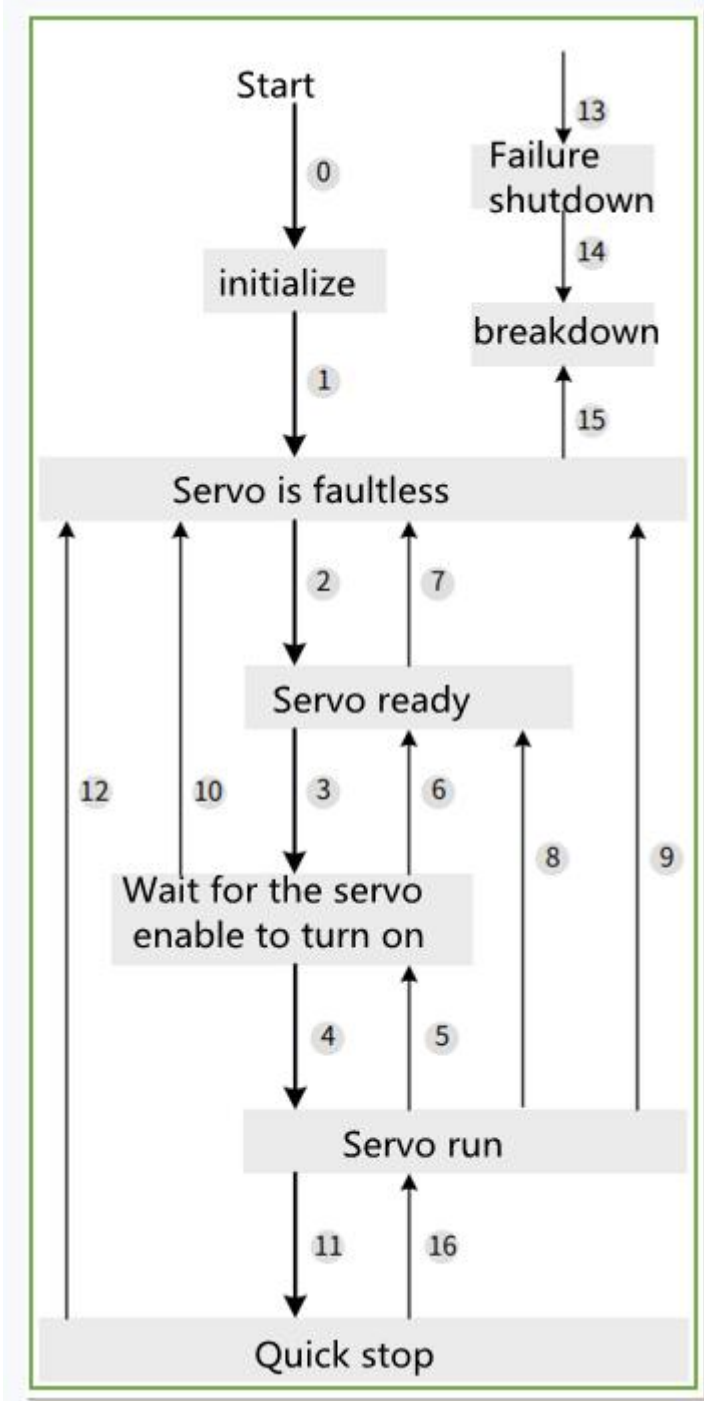
Bit	PP	PV	PT	HM	CSP	CSV	CST
bit10	position reached	Velocity Matching	Torque reached	position reached	Meaningless fixation 1	Meaningless fixation 1	Meaningless fixation 1
bit11	When the software limit or hardware limit is detected internally and triggered, this bit is 1. When the limit is lifted, this bit is 0						
bit12	Position update allowed	Zero-speed standby	Meaningless fixation 1	reset to zero has been completed	Meaningless fixation 1	Meaningless fixation 1	Meaningless fixation 1
bit13	position follows the deviation	Meaningless fixation 0	Meaningless fixation 0	There is an error in returning to zero	position follows the deviation	Meaningless fixation 0	Meaningless fixation 0
bit15	When the origin return is successfully executed, this bit is 1. If an alarm is generated or the disconnection is enabled, this bit is 0						

## 2.6 Control Instructions and Status Transitions

CIA402 status switch		Control word 6040	Control word 6041
Tab	Process migration	Control command	Feedback bit0-bit9
0	Power-on → Initialization	Internal automatic transition, no command required	0x0000
1	Initialization → Servo no fault	1. If a fault is detected after initialization, go to 13. 2. If there is no fault at the end of initialization, the transition will be automatic without command	0x0250
2	Servo no fault → Servo ready	0x06	0x0231
3	Servo ready → Wait to turn on the servo enable	0x07	0x0233
4	Wait for servo Enable → Servo run	0x0F	0x0237
5	Servo Run → Wait for the servo enable to turn on	0x07	0x0233
6	Wait for servo Enable → Servo ready	0x06	0x0231
7	Servo ready → Servo no fault	0x00	0x0250
8	Servo Run → Servo ready	0x06	0x0231
9	Servo run → No fault in the servo	0x00	0x0250
10	Wait to turn on the servo enable → No fault of the servo	0x00	0x0250
11	Servo operation → Quick shutdown	0x02	0x0217
12	Quick stop → Servo fault free	Automatic transition after quick shutdown, no command required	0x0250
13	→Fault shutdown	In addition to the fault state, in other states, once the driver detects an alarm, it automatically switches to the fault shutdown state without command	0x021F
14	Failure stop → Failure	Automatic transition after the shutdown is complete, no command required	0x0218
15	Fault → Servo no fault	0x80	0x0250
16	Quick stop → Servo run	When the quick stop is complete, send the command 0x0F	0x0237

2.6 Each mode controls the state transition of operation

CIA402 state transition diagram



Users must control the CANOPEN motor in accordance with the state transition table. Control word 6040 h every single assignment a bit pointless, must constitute a control instruction together with the other bits. Status word bit0 ~ 6041 h of combination and bit11 bit15 in servo any operation mode the same meaning. The status word 6041 h bit10, bit12, bit13 in servo different operation mode is not the same meaning.

## 2.7 6040 Control command Step

Note: To use CanOpen control, you need to use our company's debugging AIMOTOR software to set the H02-00 control mode parameter to 8 (CanOpen control).

Once set to CanOpen control must comply with CanOpen control requirements, its corresponding object dictionary takes effect; Other control modes are invalid;

Sequence control step		0	1	2	3	4	5	6	7
Mode	Control word	preparation work	initial	Ready	Wait for enable	Enable	Startup run	conjugation	Stop
CSP Cyclic synchronization position mode	6040	Establish communication, assign PDO mapping, and activate master loop synchronization	00h	06h	07h	0Fh	Master station send position send command	Master station control	Command is sent from the stop position of the master station
PP profile Position mode	6040	Establish communication status Set motion parameters related to PP mode	00h	06h	07h	0Fh	The location is not updated immediately0Fh→1Fh	1: Automatically stop when the position is reached; 2: send stop command;	
						2Fh	Absolute positioning is updated immediately2Fh→3Fh		
						4Fh	Relative positioning is not updated immediately4Fh→5Fh		
						6Fh	The relative positioning is updated immediately6Fh→7Fh		
PV profile speed mode	6040	Establish the communication status and set the motion parameters related to the PV mode	00h	06h	07h	0Fh	Start running after enabling	Just change the speed	Forcible stop write 00h/02h or 06h/07h
PT profile torque mode	6040	Establish communication status and set motion parameters related to PT mode	00h	06h	07h	0Fh	Start running after enabling	Just change the torque	Forcible stop write 00h/02h or 06h/07h

HM back to origin mode	6040	Establish the communication state and set the motion parameter state related to HM mode	00h	06h	07h	0Fh	0Fh→1Fh Start	-	1: Automatically stops when the origin is found. 2: forcibly stops writing 00h/02h or 06h/07h
------------------------	------	---	-----	-----	-----	-----	------------------	---	---

◆CSP cyclic synchronous position mode mode, the position is delivered by the main control unit according to the agreed synchronization cycle, the motor performs displacement according to the delivered position point and synchronous frame, and the motor running contour curve (such as position, speed, etc.) is planned and calculated by the main control unit in advance;

◆PP/PV/PT contour mode, the main control unit can use the SDO communication mode to set the necessary parameters of the mode, control the motor operation according to the mode control command steps, and the motor operation contour curve is planned and calculated by the motor according to the set parameter values (such as acceleration and deceleration speed, running speed, etc.).

◆In CSV synchronous speed mode, the main control unit independently plots the speed curve, and there is no acceleration or deceleration hysteresis at the motor end.

**Note 1: The operation process is not normal stop mode**

Switching mode: Write the control character 6040h to 0h, 6h or 7h. The control state of CIA402 will no longer be in operation, and the motor will be de-enabled and stop freely according to inertia. Quick stop way: the control word write 2 h, 6040 h motor will be in accordance with the 605 ah object index after choose the way of deceleration and stop state; Deceleration suspend mode: write control word 6040 h 10 fh, motor will be in accordance with the 605 ah object index after choose the way of deceleration and stop state; Quick outage and suspended shall not apply and the CSP, HM model, CSP model plan trajectory is controlled by the host. In HM mode, the control character 6040h writes from 1F to 0F and stops.

## 2.8 6098 Origin search mode

Set value	Initial Orientation	Deceleration point	Origin	Process steps
24	Forward	Origin Switch	Origin Switch	Motor first searches the origin switch at high speed in the set direction. When it encounters the origin switch signal rising edge, it starts to run out at low speed. When it breaks away from the origin switch signal falling along the motor, it reverses and continues to search the origin switch signal rising edge at low speed.
28	Reverse	Origin Switch	Origin Switch	
34	Forward	Z Signal	Motor Z Signal	Motor first searches the Z phase at a high speed in the set direction. When it encounters the rising edge of the motor Z, it starts to reverse at a low speed. When it encounters the rising edge signal on the other side of the Z phase, it stops immediately and returns to zero successfully
33	Reverse	Z Signal	Motor Z Signal	
4	Forward	Origin Switch	Motor Z Signal	Motor first searches the origin switch at a high speed in the set direction. When it encounters the origin switch signal rising edge, it begins to reverse at a low speed and runs away from the origin switch signal falling edge. When it encounters the origin switch signal rising edge, it runs to the Z signal and stops immediately, returning to zero successfully.
6	Reverse	Origin Switch	Motor Z Signal	
18	Forward	Forward overpass	Forward overpass	Motor first searches the origin switch at a high speed in the set direction. When it encounters the rising edge of the forward overrange switch signal, it begins to reverse at a low speed. When it breaks away from the falling edge of the forward overrange switch signal, the motor reverses again to search the rising edge of the forward overrange switch signal at a low speed. When it encounters the rising edge of the forward overrange switch signal, it stops immediately and returns to zero.
17	Reverse	Reverse overpass	Reverse overpass	
2	Forward	Forward overpass	Z Signal	Motor first searches for the forward overrange switch at a high speed in the set direction. When it encounters the rising edge of the forward overrange switch signal, it begins to slow down and reverse the operation. When it breaks away from the falling edge of the forward overrange switch signal, it continues to run until the motor Z is stopped immediately and the return to zero is successful.
1	Reverse	Reverse overpass	Z Signal	
37	Forward	Mechanical limit potential	Mechanical limit potential	Motor first searches the mechanical limit position at a low speed with the set direction and torque (the torque set by H05_58). When the mechanical limit position reaches the locking and the torque reaches the upper limit of the zero torque limit and maintains the default time, the motor stops immediately and the return to zero is successful.
38	Reverse	Mechanical limit potential	Mechanical limit potential	
35	-	-	-	Set the current position as the origin

**Note:**

- ◆ The external switch signal useful in the origin search mode, please associate the selected corresponding DI function option, otherwise the alarm ER.601 return to zero failure will occur;
- ◆ Change the origin search mode, please modify the object parameters of the origin mode 6098h when the motor is disabled;
- ◆ In the selected mode, if the deceleration point is the origin switch, and the positive and negative overrange switch DI is associated, the motor encounters the overrange switch motor in the search path and automatically reverses to continue the search;
- ◆ In the process steps, the sub-index 1 object of the high-speed search speed 6099 index is set, and the low-speed search speed is set by the sub-index 2 of the 6099 index;
- ◆ If the origin is not found within the H05\_35 parameter time, the alarm ER.601 return to zero timeout will occur;
- ◆ The DO function (OutFun16-HomeAttain) takes effect when the origin resets to zero, but takes effect when the OFF function is enabled.

**2.9 607E command polarity**

Bit	0	1	2	3	4	5	6	7
Function	NA	NA	NA	NA	NA	Torque command polarity	Speed command polarity	Speed command polarity

Bit7=1, indicating that in the PP contour position mode, the position instruction is 607Ah×(-1), and the motor is turned in reverse.

Bit6=1, indicating that in PV profile speed mode, the speed instruction is 60FFh×(-1), and the motor is turned in reverse.

Bit5=1, indicating that in PT contour torque mode, torque instruction 6071h×(-1), the motor turns in reverse.

NA: no definition

**2.10 60FDh Digital Input Status Monitoring**

Bit	0-15	16	17	18	19	20	21	22	23-31
Physical input terminal monitoring	Retain	DI1	DI2	DI3	DI4	DI5	DI6	DI7	Retain

**2.11 Monitoring and enforcing 60FEh digital output status**

Bit	0	1-15	16	17	18	19	20	21-31
Physical output terminal	Brake	Retain	DO1	DO2	DO3	DO4	DO5	Retain

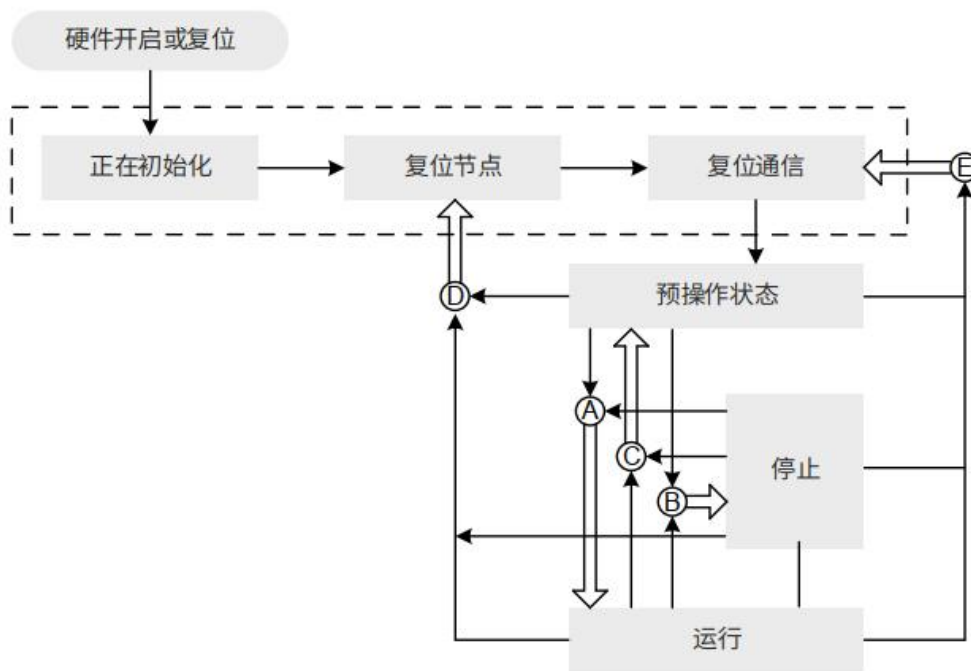
60FE-01h Status Monitoring & bit mandatory control	1. If the bit corresponding to 60FE-02h is set to 0, the status of the bit in 60FE-01h is determined by the internal status. You can monitor the current status of the bit.2. When the corresponding bit of 60FE-02h is set to 1, the bit of 60FE-01h maintains the current output state and is no longer determined by the internal state. The user can force the write operation to the bit of 60FE-01h by writing 1 to the bit. The physical port output is forced ON. Write 0 to this bit to force the physical port output OFF;
60FE-02h Indicates that force control bit selection is enabled	3. 0 bit is a special control bit for the brake holding motor, which is invalid for the user to force off when the motor is enabled to run. Write 1 can be enabled forcibly or write 0 can be disabled only after the function is disabled.

### 3. Network Management System (NMT)

The Network Management System (NMT) is responsible for initializing, starting and stopping the network and the devices within it, and it belongs to the master/slave system. There is exactly one NMT host in the CANopen network, and the CANopen network, including itself, can be configured.

#### 3.2 NMT CANopen network service

CANopen performs the corresponding work in accordance with the state machine stipulated in the protocol. Among them, some are automatically converted internally, while others must be converted by the NMT host sending NMT messages, as shown in the following figure.



The conversion with letters in the above figure is achieved by NMT messages, and only NMT hosts can send NMT control messages. The message format is shown in the table below.

COB_ID	RTR	DATA/byte	
0x000	0	Command Word	Node_ID

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

Data area is composed of two bytes:

The first byte is a command word, shows that the control function of the frame detailed in the following table shows;

The second byte is the node address, when it as a "0" for broadcast messages, all from the device in the network effectively.

Command Word	go to symbol	Description
<b>0x01</b>	A	Start the remote node
<b>0x02</b>	B	Stop the remote node
<b>0x80</b>	C	Command to enter the pre-operation state
<b>0x81</b>	D	Reset node instruction
<b>0x82</b>	E	Reset communication instruction

After the device is powered on, it will automatically enter the initialization state, including initializing, resetting the node and resetting the communication. The initialization is loading the parameters of each module, while the reset node restores the manufacturer definition area and sub-agreement area of the object dictionary to the last saved values, and the reset communication restores the communication parameters in the object dictionary to the last saved values. Then the device sends the Boot up and automatically enters the pre-operation state, which is the main configuration node state. After the configuration is completed, the node needs the NMT host to send an NMT message to enter the operational state. The operation status is the state when CANopen is working normally, and all modules should be working properly.

When the NMT host sends a stop node message, the device enters a stop state, and only the NMT module operates normally in CANopen communication. Support various NMT states CANopen service shown in the following table

Service	NMT pre-operation	NMT operation	NMT stop
Process Data Object (PDO)	NO	YES	NO
Service Data Object (SDO)	YES	YES	NO
Synchronous object (SYNC)	YES	YES	NO
Emergency Message (EMCY)	YES	YES	NO
Network Management (NMT)	YES	YES	YES
Error control	YES	YES	YES

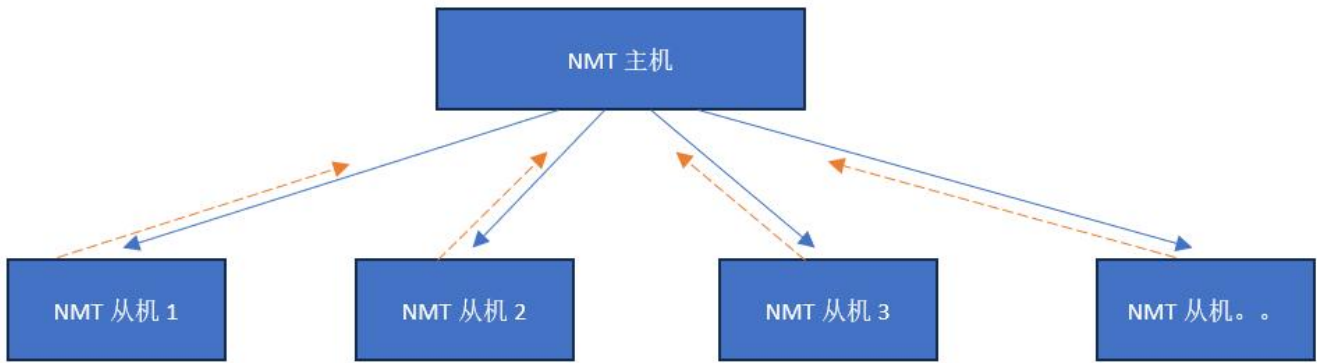
### 3.2 NMT Error Control

NMT error control is mainly used to detect whether devices in the network are online and the status of the devices, including node protection, lifetime protection and heartbeat.

- ◆ It is not allowed to use the lifespan protection and heartbeat function simultaneously!
- ◆ The time settings for node protection, lifespan protection, and heartbeat should not be too short to avoid increasing the network load!

#### Node/lifetime protection

Node protection is that the NMT host periodically queries the status of the NMT slave through remote frames. Life protection indirectly monitors the status of the master station by the slave station through the remote frame intervals received for monitoring the slave station. Node protection follows a master-slave model, where each remote frame must receive a response. Objects related to node/lifetime protection include protection time 100Ch and lifetime factor 100Dh. The value of 100Ch is the normal node protection remote frame interval, with the unit being ms. The product of 100Ch and 100Dh determines the latest time for the host query. Under normal circumstances, node protection can be achieved. When both nodes 100Ch and 100D are non-zero and a frame of node protection request is received, lifetime protection is activated.



As shown in the above figure, the master station sends a node to protect the remote frame. The slave must respond every 100Ch; otherwise, it is considered that the slave has dropped. If the node protection remote frame is not received from the station within 100Ch×100Dh time, it is considered that the master station has dropped.

The protection time of 100Ch is recommended to be no less than 50ms, and the life factor of 100Dh must be no less than 2.

The NMT master node sends remote frames as shown in the following table:

COB_ID	RTR
0x700+Node-ID	1

The response data frame message returned by the NMT from the node is shown in the following table. The data segment is a one-byte current network status word of the NMT.

COB_ID	RTR	DATA
0x700+Node-ID	0	Status word

COB_ID	Description
bit7	"0" or "1" must be alternated each time.
bit6- bit0	4: state stopped 5: operating state 127: operating state

## 2) Heartbeat protection

As the remote frame format of node guardianship has gradually been phased out in the development of CAN, and node guardianship requires more master station overhead and increased network load, the CiA Association no longer recommends its use and it has been replaced by heartbeat messages.

The heartbeat pattern adopts the producer-consumer model. The CANopen device can send heartbeat messages based on the period set by the producer's heartbeat interval object at 1017h, with the unit being ms. There are always nodes in the network that have the consumer heartbeat function. They monitor the producer based on the consumer time set by object 1016h. Once the producer heartbeat of the corresponding node is not received within the consumer heartbeat time range, it is considered that the node has failed.

After configuring the producer heartbeat time interval of 1017 hours, the node heartbeat function is activated and begins to generate heartbeat messages. After configuring the valid sub-index of the consumer heartbeat 1016h, monitoring begins upon receiving a frame of heartbeat sent by the corresponding node.

Unlike node guardianship, CANopen network devices can be both heartbeat producers and consumers. It is recommended that the heartbeat producer's time should not be less than 50ms, while the consumer's heartbeat time should not be less than 40ms and should be more than 1.8 times the corresponding producer's heartbeat time.

The format of the NMT heartbeat frame message is shown in the following table:

COB_ID	RTR	DATA
0x700+Node-ID	0	One of the states below the status word 4: state stopped 5: operating state 127: operating state

## 4, SDO control message reference

### 4.1 SDO Transmission Format

1) the SDO framework SDO transport way to follow the client - server mode, namely all the way. Initiated by the SDO client in the CAN bus network, the SDO server responds. Therefore, data exchange between Sdos requires at least two CAN messages to be achieved, and the CAN identifiers of the two CAN messages are different.

2) the SDO transmission of message transmission can be divided into no more than four bytes of SDO data transmission and higher than 4 bytes. For no more than 4 bytes, the accelerated SDO transmission mode is adopted; for more than 4 bytes, the segmented transmission or block transmission mode is used. This machine only supports accelerated SDO transmission and segmented transmission. The SDO transmission message consists of COB-ID and data segments. It can be seen from the following table that the COB-ids of the T\_SDO and R\_SDO messages are inconsistent.

The data segment adopts the little-endian mode, that is, the lower bits are placed first and the higher bits are arranged later. All SDO message data segments must be 8 bytes. The SDO transmission message format is as follows:

COB_ID	DATA								
580h+Node_I D/ 600h+Node_I D	CmdCode	index	sub-ind	Data					

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

**3) SDO accelerates write transmission messages**

transmission direction		COB_ID	0	1	2	3	4	5	6	7
Client side→		600h+Node_ID	23h	Index		Sub-Index x	Data			
			27h				Data			-
			2bh				Data		-	-
			2fh				Data	-	-	-
←server	normal	580h+Node_ID	Index	Sub-Index		Sub-Index x	-	-	-	-
	abnormal		80h				Terminate the code			

**4) SDO accelerates the reading of transmission messages**

transmission direction		COB_ID	0	1	2	3	4	5	6	7
Client side→		600h+Node_ID	40h	Index		Sub-Index	-	-	-	-
←server	normal	580h+Node_ID	43h	Index		Sub-Index	Data			
			47h				Data			-
			4bh				Data		-	-
	4fh		Data				-	-	-	
	abnormal		80h	Terminate the code						

**4.2 Speed mode control**

Description	Frame ID	Frame Type	Frame Format	DLC	Data Segment DATA[0]~DATA[7] Hexadecimal Big endian mode storage			
					DATA[0]	DATA[1]~ DATA[2]	DATA[3]	DATA[4]~ DATA[7]
Write run mode =3	601	data frame	standard frame	8	2F	60 60	00	03 00 00 00
Slave answer	581	data frame	standard frame	8	60	60 60	00	00 00 00 00
Write target speed	601	data frame	standard frame	8	23	FF 60	00	xx xx xx xx write value
Slave answer	581	data frame	standard frame	8	60	FF 60	00	00 00 00 00
Write acceleration	601	data frame	standard frame	8	23	83 60	00	xx xx xx xx write value
Slave answer	581	data frame	standard frame	8	60	83 60	00	00 00 00 00
Write deceleration	601	data frame	standard frame	8	23	84 60	00	xx xx xx xx write value

Slave answer	581	data frame	standard frame	8	60	84 60	00	00 00 00 00
[Note1]Write control word =6	601	data frame	standard frame	8	2B	40 60	00	06 00 00 00
Slave answer	581	data frame	standard frame	8	60	40 60	00	00 00 00 00
[Note1]Write control word =7	601	data frame	standard frame	8	2B	40 60	00	07 00 00 00
Slave answer	581	data frame	standard frame	8	60	40 60	00	00 00 00 00
[Note1]Write control word =0F	601	data frame	standard frame	8	2B	40 60	00	0F 00 00 00
Slave answer	581	data frame	standard frame	8	60	40 60	00	00 00 00 00
Read Status	601	data frame	standard frame	8	40	41 60	00	00 00 00 00
Slave answer	581	data frame	standard frame	8	4B	41 60	00	xx xx xx xx returned value
Read current speed	601	data frame	standard frame	8	40	6C 60	00	00 00 00 00
Slave answer	581	data frame	standard frame	8	43	6C 60	00	xx xx xx xx returned value

[Note 1] Write 6040h control word 06, the motor enters the ready state; Write 07, motor waiting to enable state; 07→0F, the motor will enable and run at speed according to the parameters of the speed configuration; 0F→07, the motor will be unenabled, free stop;

## 4.2 Position mode control

Description	Frame ID	Frame Type	Frame Format	DLC	Data Segment DATA[0]~DATA[7] Hexadecimal Big endian mode storage			
					DATA[0]	DATA[1]~ DATA[2]	DATA[3]	DATA[4]~ DATA[7]
Write run mode=1	601	data frame	standard frame	8	2F	60 60	00	01 00 00 00
Slave answer	581	data frame	standard frame	8	60	60 60	00	00 00 00 00
Write target location	601	data frame	standard frame	8	23	7A 60	00	xx xx xx xx write value
Slave answer	581	data frame	standard frame	8	60	7A 60	00	00 00 00 00
Write run speed	601	data frame	standard frame	8	23	81 60	00	xx xx xx xx write value
Slave answer	581	data frame	standard frame	8	60	81 60	00	00 00 00 00

AIMOTOR (ac/dc servo driver) CANOPEN control manual

Write acceleration	601	data frame	standard frame	8	23	83 60	00	xx xx xx xx write value
Slave answer	581	data frame	standard frame	8	60	83 60	00	00 00 00 00
Write deceleration	601	data frame	standard frame	8	23	84 60	00	xx xx xx xx write value
Slave answer	581	data frame	standard frame	8	60	84 60	00	00 00 00 00
[Note1]Write control word=6	601	data frame	standard frame	8	2B	40 60	00	06 00 00 00
Slave answer	581	data frame	standard frame	8	60	40 60	00	00 00 00 00
[Note1]Write control word=7	601	data frame	standard frame	8	2B	40 60	00	07 00 00 00
Slave answer	581	data frame	standard frame	8	60	40 60	00	00 00 00 00
[Note 1]Write control word=0F	601	data frame	standard frame	8	2B	40 60	00	0F 00 00 00
Slave answer	581	data frame	standard frame	8	60	40 60	00	00 00 00 00
[Note 1]Write control word=1F	601	data frame	standard frame	8	2B	40 60	00	1F 00 00 00
Slave answer	581	data frame	standard frame	8	60	40 60	00	00 00 00 00
Read Status	601	data frame	standard frame	8	40	41 60	00	00 00 00 00
Slave answer	581	data frame	standard frame	8	4B	41 60	00	xx xx xx xx return value
Read current position	601	data frame	standard frame	8	40	64 60	00	00 00 00 00
Slave answer	581	data frame	standard frame	8	43	64 60	00	xx xx xx xx return value

[Note 1] Write 6040h control word 06, the motor enters the ready state; Write 07, motor waiting to enable state; Then write 0F or 2F or 4F or 6F and the motor will enable; Finally write 1F or 3F or 5F or 7F, the motor will run according to the command;

0F→1F, the motor will perform absolute positioning without immediate update; 2F→3F, the motor will perform absolute positioning and update immediately; The motor stops automatically after the positioning is completed. 4F→5F, the motor will perform relative positioning without immediate update; 6F→7F, the motor will perform relative positioning and update immediately; The motor stops automatically after the positioning is completed.

4.4 Return to origin mode control

Description	Frame ID	Frame Type	Frame Format	DLC	Data Segment DATA[0]~DATA[7] Hexadecimal Big endian mode storage			
					DATA[0]	DATA[1]~ DATA[2]	DATA[3]	DATA[4]~ DATA[7]
Write run mode=6	601	data frame	standard frame	8	2F	60 60	00	06 00 00 00
Slave answer	581	data frame	standard frame	8	60	60 60	00	00 00 00 00
[Note 1]Write back to the original method	601	data frame	standard frame	8	2F	98 60	00	xx xx xx xx write value
Slave answer	581	data frame	standard frame	8	60	98 60	00	00 00 00 00
Write back to the original speed	601	data frame	standard frame	8	23	99 60	01	xx xx xx xx write value
Slave answer	581	data frame	standard frame	8	60	99 60	01	00 00 00 00
Write back to the original low speed	601	data frame	standard frame	8	23	99 60	02	xx xx xx xx write value
Slave answer	581	data frame	standard frame	8	60	99 60	02	00 00 00 00
Write acceleration	601	data frame	standard frame	8	23	9A 60	00	xx xx xx xx write value
Slave answer	581	data frame	standard frame	8	60	9A 60	00	00 00 00 00
[Note 2]Write control word=6	601	data frame	standard frame	8	2B	40 60	00	06 00 00 00
Slave answer	581	data frame	standard frame	8	60	40 60	00	00 00 00 00
[Note 2]Write control word=7	601	data frame	standard frame	8	2B	40 60	00	07 00 00 00
Slave answer	581	data frame	standard frame	8	60	40 60	00	00 00 00 00
[Note 2]Write control word=0F	601	data frame	standard frame	8	2B	40 60	00	0F 00 00 00
Slave answer	581	data frame	standard frame	8	60	40 60	00	00 00 00 00
[Note 2]Write control word=1F	601	data frame	standard frame	8	2B	40 60	00	1F 00 00 00
Slave answer	581	data frame	standard frame	8	60	40 60	00	00 00 00 00
Read Status	601	data frame	standard frame	8	40	41 60	00	00 00 00 00

Slave answer	581	data frame	standard frame	8	4B	41 60	00	xx xx xx xx return value
Read current position	601	data frame	standard frame	8	40	64 60	00	00 00 00 00
Slave answer	581	data frame	standard frame	8	43	64 60	00	xx xx xx xx return value

[Note 1] : Please refer to section 2.8 [6098h origin Search Method] for the supported origin return mode.

[Note 2] : Write 6040h control word 06, the motor enters the ready state; Write 07, motor waiting to enable state; Then write 0F motor will enable;

0F→1F, the motor will return to the origin according to the parameters of the origin, the motor will automatically stop after the origin is successfully returned, and the current position will clear to zero;

1F→0F, the motor will stop the current origin return action;

#### 4.5 Torque mode control

Description	Frame ID	Frame Type	Frame Format	DLC	Data Segment DATA[0]~DATA[7] Hexadecimal Big endian mode			
					DATA[0]	DATA[1]~ DATA[2]	DATA[3]	DATA[4]~ DATA[7]
Write run mode=4	601	data frame	standard frame	8	2F	60 60	00	04 00 00 00
Slave answer	581	data frame	standard frame	8	60	60 60	00	00 00 00 00
Write target torque	601	data frame	standard frame	8	2B	71 60	00	xx xx xx xx write value
Slave answer	581	data frame	standard frame	8	60	98 60	00	00 00 00 00
Write speed limit	601	data frame	standard frame	8	23	7F 60	01	xx xx xx xx write value
Slave answer	581	data frame	standard frame	8	60	99 60	01	00 00 00 00
[[Note1]Write control word=6	601	data frame	standard frame	8	2B	40 60	00	06 00 00 00
Slave answer	581	data frame	standard frame	8	60	40 60	00	00 00 00 00
[[Note1]Write control word=7	601	data frame	standard frame	8	2B	40 60	00	07 00 00 00
Slave answer	581	data frame	standard frame	8	60	40 60	00	00 00 00 00
[[Note1]Write control word=0F	601	data frame	standard frame	8	2B	40 60	00	0F 00 00 00
Slave answer	581	data frame	standard frame	8	60	40 60	00	00 00 00 00
Read	601	data frame	standard frame	8	40	41 60	00	00 00 00 00

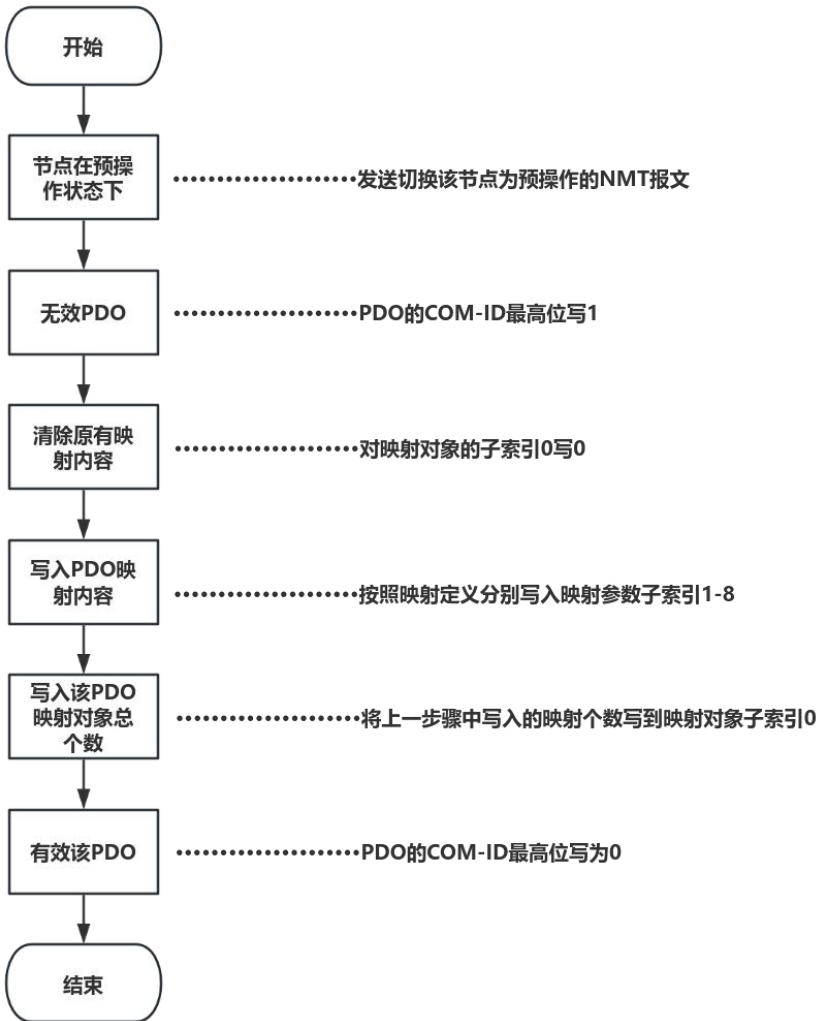
Slave answer	581	data frame	standard frame	8	4B	41 60	00	xx xx xx xx 返回值
Read the current torque	601	data frame	standard frame	8	40	77 60	00	00 00 00 00
Slave answer	581	data frame	standard frame	8	4B	77 60	00	xx xx xx xx 返回值

[Note 1] :

Write 6040h control word 06, the motor enters the ready state; Write 07, motor waiting to enable state; 07→0F, the motor will enable and perform torque operation according to the torque configuration parameters; 0F→07, the motor will be unenabled, free stop;

## 5. PDO control mode

### 5.1 PDO Mapping Configuration Flowchart



## 5.2 Communication parameters of the PDO channel

### 1) Communication and mapping objects of PDO

According to the difference between receiving and sending, PDO can be divided into RPDO and TPDO. PDO is determined by communication parameters and mapping parameters together to determine the final transmission mode and content. AIMOTOR servo drive uses four TPdos and four RPdos to implement PDO transmission, and the related objects are listed below.

Name		COB-ID	Communication Objects	Mapping object
RPDO	1	200h+NODE_ID	1400h	1600h
	2	300h+NODE_ID	1401h	1601h
	3	400h+NODE_ID	1402h	1602h
	4	500h+NODE_ID	1403h	1603h
TPDO	1	180h+NODE_ID	1800h	1A00h
	2	280h+NODE_ID	1801h	1A02h
	3	380h+NODE_ID	1802h	1A03h
	4	480h+NODE_ID	1803h	1A04h

### 2) the PDO CAN identifier

PDO COB - ID, contains control and identification data, determine the bus priority of the PDO. The COB-ID is located on the sub-index 01 of the communication parameters (RPDO: 1400h - 1403h, TPDO:1800h - 1803h), and the highest bit determines whether the PDO is valid.

Example: RPDO1 COB\_ID configuration of node 1 to 0 x80000201, said failure the PDO channel; Configuration of 0x201 indicates that the PDO channel transmission is enabled effectively.

Node 1 x80000181 TPDO1 COB\_ID configuration to 0, the PDO channel said failure; Configuration of 0x181 indicates that the PDO channel transmission is enabled effectively.

### 3) the PDO transport type

PDO transport type is in the communication parameters (RPDO: 1400 ~ 1403 h h, TPDO: 1800 h to 1803 h) of sub indexes 02, decided the PDO) transmission way; Synchronous transmission, the frame synchronization triggered by the CAN network; Asynchronous transmission: an event (data change triggers, periodic event timer trigger);

Transmission type numerical value	synchronous transmission		asynchronous transmission
	Loop	Non Loop	
000		√	
001~240	√		
241~253	-		
254~255			√

- When the transmission type of TPDO is 0, if the mapping data changes and a synchronous frame is received, send this TPDO;
- When the transmission type of TPDO is 1-240, send this TPDO when the corresponding number of synchronous frames are received;
- When the transmission type of TPDO is 254 or 255, the TPDO is sent if the mapping data changes or the event timer arrives;

■ When the transmission type of RPDO is 0-240, as long as a synchronization frame is received, the latest data of this RPDO is updated to the application; S when RPDO transport type is 254 or 255, update the data received directly into the application.

**4) Prohibited time for PDO**

A prohibition time was set for TPDO and stored on the sub-index 03 of the communication parameters (1800h - 1803h) to prevent the CAN network from being continuously occupied by Pdos with lower priority. The unit of this parameter is 100us. After setting the value, the transmission interval of the same TPDO must not be less than the time corresponding to this parameter.

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

**5) The event timer of PDO**

For TPDO with asynchronous transmission (transmission type 254 or 255), an event timer is defined, located on the sub-index 05 of the communication parameters (1800h to 1803h). The event timer can also be regarded as a trigger event, which will also trigger the corresponding TPDO transmission. If other events such as data changes occur during the timer's running cycle, the TPDO will also be triggered, and the event counter will be immediately reset.

**5.3 PDO Control Packet Reference**

The PDO mapping parameters contain Pointers to the process data corresponding to the PDO that the PDO needs to send or receive, including indexes, sub-indexes, and the length of the mapping object. Each PDO data can be up to 8 bytes in length and can map one or more objects simultaneously. Among them, sub-index 0 records the specific number of objects mapped by this PDO, while sub-indexes 1 to 8 represent the mapping contents. The content of the mapping parameter is defined as follows.

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

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

Object length	Bit length 15~8 7~0
08h	8bit
10h	16bit
20h	32bit

For example: the data type of the control word 6040\_00h uint16\_t, said 16 control word 6040\_00h mapping parameters for 60400010 h. Control mode 6060\_00h data type for int8\_t, said eight control pattern 6060\_00h mapping parameters for 60600008 h. Target location 607\_a\_00h data types for int32\_t, said the 32 bit control mode a0020h 607\_a\_00h mapping parameters for 607.

**5.4 Reference Example for PDO Configuration Process**

The following PDO control messages are all based on control Node 1 motor as an example:

The TPDO1 transmission type is synchronous transmission, with one synchronous frame for TPDO data upload.

The mapping object is 1:641\_00H (status word) 2:06c\_00H (actual rotational speed). RPDO1 transport type for synchronous transmission, a synchronous frame RPDO data into effect;

Mapping object is 1:6 040\_00h (control word) but 060\_00h (control model) set 0 ff\_00h (target speed).

Description	Frame ID	Frame Type	Frame Format	DLC	Data Segment DATA[0]~DATA[7] Hexadecimal Big endian mode			
					DATA[0]	DATA[1]~ DATA[2]	DATA[3]	DATA[4]~ DATA[7]
					Set the node to a pre-operation state	000	data frame	standard frame
Unavailable answer	-	-	-	-	-	-	-	-
Set the COB_ID of RPDO1 and invalidate RPDO1	601	data frame	standard frame	8	23	00 14	01	01 02 00 80
Slave answer	581	data frame	standard frame	8	60	00 14	01	00 00 00 00
Write transport type of RPDO1	601	data frame	standard frame	8	2F	00 14	02	01 00 00 00
Slave answer	581	data frame	standard frame	8	60	00 14	02	00 00 00 00
Set the number of RPDO1 mappings to 0	601	data frame	standard frame	8	2F	00 16	00	00 00 00 00
Slave answer	581	data frame	standard frame	8	60	00 16	00	00 00 00 00
Write mapping object 1 of RPDO1	601	data frame	standard frame	8	23	00 16	01	10 00 40 60
Slave answer	581	data frame	standard frame	8	60	00 16	01	00 00 00 00
Write mapping object 2 of RPDO1	601	data frame	standard frame	8	23	00 16	02	08 00 60 60
Slave answer	581	data frame	standard frame	8	60	00 16	02	00 00 00 00
Write mapping object 3 of RPDO1	601	data frame	standard frame	8	23	00 16	03	20 00 FF 60
Slave answer	581	data frame	standard frame	8	23	00 16	03	00 00 00 00
The number of RPDO1 mappings is set to 3	601	data frame	standard frame	8	2F	00 16	00	03 00 00 00
Slave answer	581	data frame	standard frame	8	60	00 16	00	00 00 00 00
Set the COB_ID of RPDO1 and take effect on RPDO1	601	data frame	standard frame	8	23	00 14	01	01 02 00 00

Slave answer	581	data frame	standard frame	8	60	00 14	01	00 00 00 00
Set the COB_ID of RPDO2 and take effect on RPDO2	601	data frame	standard frame	8	23	01 14	01	01 03 00 80
Slave answer	581	data frame	standard frame	8	60	01 14	01	00 00 00 00
Set the COB_ID of RPDO3 and take effect on RPDO3	601	data frame	standard frame	8	23	02 14	01	01 04 00 80
Slave answer	581	data frame	standard frame	8	60	02 14	01	00 00 00 00
Set the COB_ID of RPDO4 and take effect on RPDO4	601	data frame	standard frame	8	23	03 14	01	01 05 00 80
Slave answer	581	data frame	standard frame	8	60	03 14	01	00 00 00 00
Set the COB_ID of TPDO1 and invalidate TPDO1	601	data frame	standard frame	8	23	00 18	01	81 01 00 80
Slave answer	581	data frame	standard frame	8	60	00 18	01	00 00 00 00
Set the transport type of TPDO1	601	data frame	standard frame	8	2F	00 18	02	01 00 00 00
Slave answer	581	data frame	standard frame	8	60	00 18	02	00 00 00 00
The number of TPDO1 mappings is set to 0	601	data frame	standard frame	8	2F	00 1A	00	00 00 00 00
Slave answer	581	data frame	standard frame	8	2F	00 1A	00	00 00 00 00
Write mapping object 1 of TPDO1	601	data frame	standard frame	8	23	00 1A	01	10 00 41 60
Slave answer	581	data frame	standard frame	8	60	00 1A	01	00 00 00 00
Write mapping object 2 of TPDO1	601	data frame	standard frame	8	23	00 1A	02	20 00 6C 60
Slave answer	581	data frame	standard frame	8	23	00 1A	02	00 00 00 00
The number of TPDO1 mappings is set to 2	601	data frame	standard frame	8	2F	00 1A	00	02 00 00 00
Slave answer	581	data frame	standard frame	8	60	00 1A	02	00 00 00 00

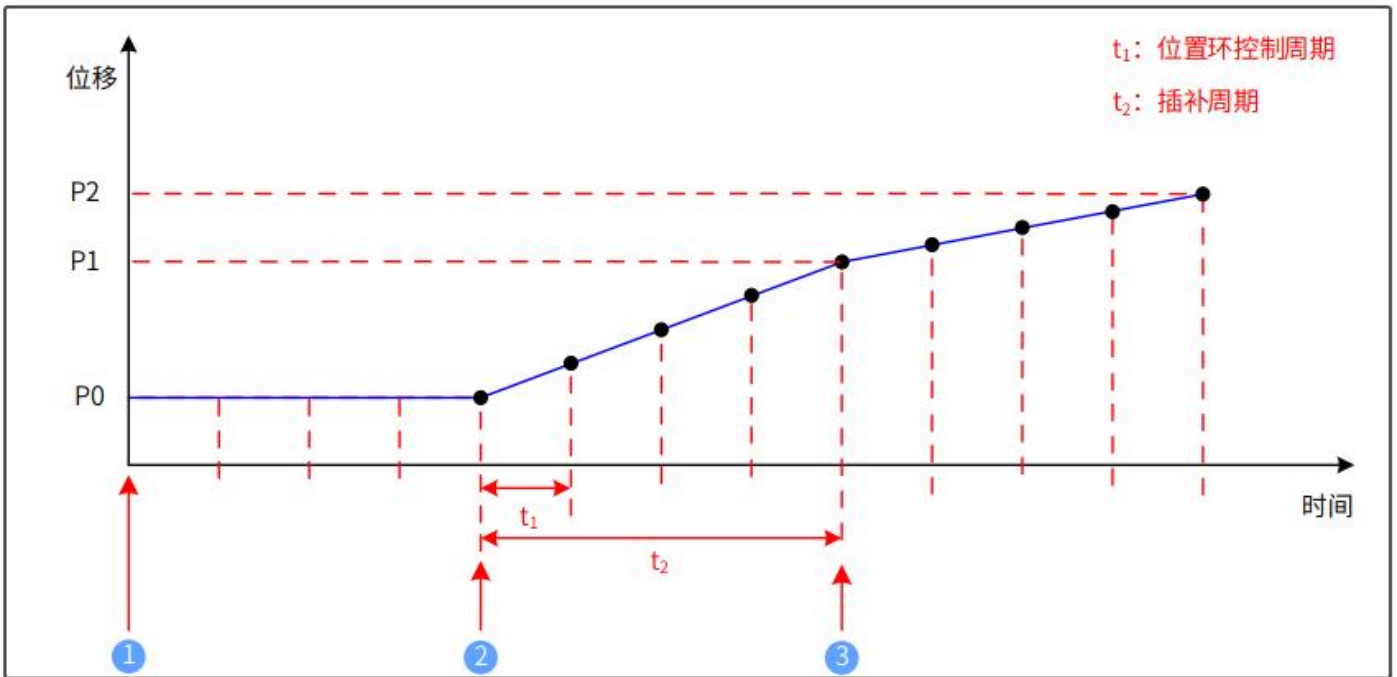
Set the COB_ID of TPDO1 and take effect on TPDO1	601	data frame	standard frame	8	23	00 18	01	81 01 00 00
Slave answer	581	data frame	standard frame	8	60	00 18	01	00 00 00 00
Set the COB_ID of TPDO2 and take effect on TPDO2	601	data frame	standard frame	8	23	01 18	01	81 02 00 80
Slave answer	581	data frame	standard frame	8	60	01 14	01	00 00 00 00
Set the COB_ID of TPDO3 and take effect on TPDO3	601	data frame	standard frame	8	23	02 18	01	81 03 00 80
Slave answer	581	data frame	standard frame	8	60	02 18	01	00 00 00 00
Set the COB_ID of TPDO4 and take effect on TPDO4	601	data frame	standard frame	8	23	03 18	01	81 04 00 80
Slave answer	581	data frame	standard frame	8	60	03 18	01	00 00 00 00
Set the node to the operational state	000	data frame	standard frame	2	01	01	-	-
Unavailable answer	-	-	-	-	-	-	-	-

After the configuration is completed and the operation state is entered, the slave machine will trigger the TPDO to upload the mapping object data based on the synchronous frame on the network, and the RPDO mapping object data will take effect.

### Precautions for PDO transmission in CSP synchronous location mode

If running in CSP synchronous location mode, RPDO must map 607Ah target location objects. The servo slave operates at the absolute target position of 607A as received. If RPDO is synchronous transmission, it is suggested that the RPDO configuration for a synchronous frame was adopted, the host needs to periodically send synchronization frame, from the machine every receives a synchronous frame RPDO data into effect, the motor running to 607 a target location; If RPDO is asynchronous transmission, host need time interval, the periodic change RPDO mapping value, from the machine to receive RPDO RPDO data mapping value changed, the motor running to 607 a target location;

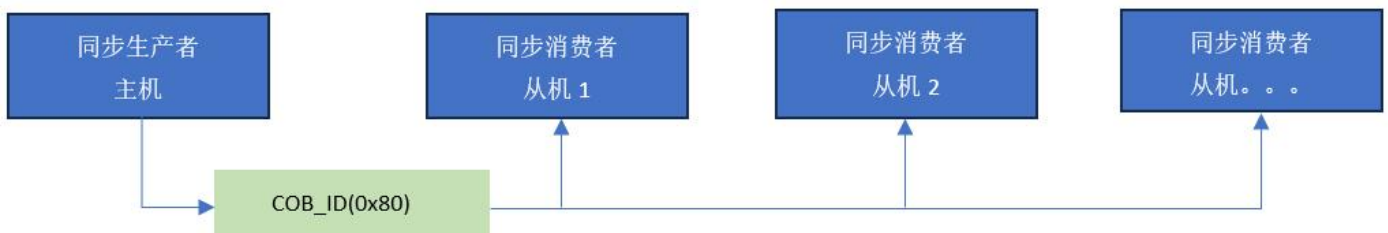
The host needs to inform the servo slave of the RPDO trigger interval time including the target position, which is written into a 1006h object in us. In this way, the servo slave smoothly processes the two synchronous position steps before and after based on the 1006h synchronization cycle time and the internal position loop control cycle.



**t1: Position loop control period**  
**t2: Insertion cycle**

### 5.5 SYNC Object Sync

The SYNC object (SYNC) is a special mechanism that controls the coordination and synchronization between the sending and receiving of multiple nodes, and is used for the synchronous transmission of PDO. Object is similar to the PDO transmission, synchronous transmission model follows the producers, consumers, made by synchronous producers synchronous frame, CAN all other nodes in the network CAN be used as a consumer receives the synchronous frame, without feedback. The same CAN network allows only a activation of the synchronous generator. Generally, the master station produces the synchronous frames, and all slave stations consume the synchronous frames. The host writes the synchronization period time (us) into the object dictionary of the slave through SDO communication to index the 1006-00h object. The slave can determine the synchronization period of the host based on the value of the 1006-00h object.



Take the PDO configuration example in Chapter 5.4 for analysis

Description	Frame ID	Frame Type	Frame Format	DLC	Data Segment DATA[0]~DATA[7] Hexadecimal Big endian mode			
					DATA[0]	DATA[1]~ DATA[2]	DATA[3]	DATA[4]~ DATA[7]
[Note 1] Sync Frame	080	data frame	standard frame	0				
[Note 2] TPDO Return reply	181	data frame	standard frame	6	37	02 E8	03	00 00

[Note 3]RPDO Fill message	201	data frame	standard frame	7	0F	00 03	E8	03 00 00
---------------------------	-----	------------	----------------	---	----	-------	----	----------

[Note 1] : When the TPDO transmission type is 1-240, the TPDO reply packet is sent when the synchronization frame with the number of responses is received. When the RPDO transmission type is 0-240, the latest data of the RPDO is updated to the application as soon as a synchronization frame is received.

[Note 2] : The preceding packet contains 37 02 E8 03 00 00 In which TPDO1 mapping object 1 is 6041\_00h and its data width is two bytes. In the packet returned by TPDO1, 3702 is the value of 6041\_00h. TPDO1 mapping object 2 is 606C\_00h and its data width is four bytes. In the packet returned by TPDO1, E8 0300 00 is the value of 606C\_00h.

[Note 3] : The preceding packet contains 0F 00 03 E8 03 00 00 Where RPDO1 mapping object 1 is 6040\_00h and its data width is two bytes, 0F 00 is the set value of 6040\_00h. If RPDO1 mapping object 2 is 6060\_00h and its data width is one byte, 03 is the set value of 6060\_00h. If RPDO1 mapping object 3 is 60FF\_00h and its data width is four bytes, E8 03 00 00 00 is the set value of 60FF\_00h. After the configuration is completed and the operation state is entered, the slave machine will trigger the TPDO to upload the mapping object data based on the synchronous frame on the network, and the RPDO mapping object data will take effect.

## 6. Fault query

### 6.1 Alarm Code

Description	Fault code is customized by the manufacturer H0B_34 parameters	Cia402 standard fault code 603Fh object	Hardware alarm indicator blinking times	Alarm Type
No Fault	0x0000	0x0000	Green	-
Parameter id is out of range	0x0101	0x6320	10Red1Green	NO.1
Parameter CRC error	0x0102	0x6320	12Red1Green	NO.1
Registration parameter CRC error	0x0104	0x6320	12Red1Green	NO.1
An internal program exception triggers the watchdog	0x0105	0x6320	11Red1Green	NO.1
DI function is reassigned	0x0130	0x6320	12Red1Green	NO.1
Hardware overcurrent	0x0201	0x2312	4Red1Green	NO.1
ia/ic static current bias calibration failed	0x0208	0xFF00	12Red1Green	NO.1
Software overcurrent	0x0207	0x2311	4Red1Green	NO.1
aerodyne	0x0234	0xFF00	12Red1Green	NO.1
Encoder data exception	0x0A33	0x7306	9Red1Green	NO.1
Primary circuit overvoltage	0x0400	0x3210	3Red1Green	NO.1
Main circuit undervoltage	0x0410	0x3220	3Red1Green	NO.1
motor overload	0x0620	0x3230	4Red1Green	NO.1
Radiator overheating	0x0650	0x4210	6Red1Green	NO.1
Excessive position deviation	0x0B00	0x8611	2Red1Green	NO.1
The return to zero mode does not match	0x0668	0xFF00	8Red1Green	NO.2
Back to origin timeout	0x0601	0x8610	8Red1Green	NO.2
emergency shut down	0x0900	0x5442	12Red1Green	NO.3
Forward overshoot warning	0x0950	0x5443	12Red1Green	NO.3

Negative overshoot warning	0x0952	0x5444	12Red1Green	NO.3
Encoder battery failure	0x0731	0x7306	12Red1Green	NO.2
Encoder multi-turn count error	0x0733	0x7306	12Red1Green	NO.2
Encoder multi-turn count overflow	0x0735	0x7306	12Red1Green	NO.2
Encoder battery warning	0x0730	0x7307	12Red1Green	NO.3
CAN communication connection is interrupted	0x0D03	0x8130	12Red1Green	NO.2
The changed parameters take effect after being powered on again	0x0941	0xFF00	7Red1Green	NO.3
Parameter storage frequency	0x0942	0x7600	12Red1Green	NO.3

1.manufacturers user-defined fault code, one fault information corresponds to a fault code, has a unique;

2.Cia402 standard fault code. Refer to cia402 regulations. If fault information is not specified in cia402, it is expressed as 0xFF00 fault code and is not unique.

3. Because the integrated motor has no display panel, it can judge the alarm information by referring to the flashing change of the indicator light without connecting the debugging software, which is not unique;

## 6.2 Fault query method

### 1. Query using object dictionary manufacturer-defined parameters

Parameter number: H0B-33, select the fault record to query; object dictionary index: 200Bh, sub-index 22h, read and write

Parameter number: H0B-34, selected number of fault codes; Object dictionary index: 200Bh, sub-index 23h, read only

Usage: H0B-33 (200B-22h) is used to select the fault record to view, H0B-34 (200B-23h) displays the selected historical fault

Example: When H0B-33 (200B-22h) is set to 0, once a fault occurs, H0B-34 (200B-23h) will display the current manufacturer's customized fault code;

When H0B-33 (200B-22h) is set to 1, the system will query the manufacturer's customized fault code generated previously in history and display it in H0B-34 (200B-23h);

When H0B-33 (200B-22h) is set to 2, the system will query the manufacturer's customized fault codes generated twice in history and display them in H0B-34 (200B-23h);

When H0B-33 (200B-22h) is set to 9, the system will query the manufacturer's customized fault codes generated in the previous 9 times in history and display them in H0B-34 (200B-23h);

### 2. Use object dictionary CIA402+CIA301 parameter query

Current fault code; Object dictionary index: 603Fh, sub-index 00h, read-only

Number of historical faults; object dictionary index: 1003h, sub-index 00h, read and write

Historical fault entry; object dictionary index: 1003h, sub-index 01h~0Ah, read only

Usage: Once a fault occurs, 603F-00h will generate the Cia402 standard fault code corresponding to the current fault, and the corresponding manufacturer's customized fault code and Cia402 standard fault code combination are stored in 01h~0Ah of object 1003h;

00h of object 1003h is the number of faults currently recorded. Writing 0 to 1003-00h means clearing historical

fault records, and writing non-zero values is not allowed;

01h~0Ah of object 1003h is a queue of 10 fault record entries, which are stored in stack mode and follow the first-in-first-out rule. When the historical alarm storage is full and a new alarm is generated, the earliest error will be deleted, and the previous errors will be moved downwards in turn; the storage rules of each entry are bit0-bit15 to store the Cia402 standard fault code, and bit16-bit31 to store the manufacturer. Customized fault codes;

For example: The user can first read the Cia402 standard fault code value in 603F-00h (a value other than 0 indicates a fault) to determine whether the current node is faulty. Once a fault occurs, you can refer to "Chapter 3.1 Fault Table" to initially determine the fault information. Because the Cia402 standard fault code is not unique in this system, you can continue to read the value in 1003-01h (the most recent fault record). The lower 16 bits of the value are the same as the current Cia402 standard fault code value in 603F-00h. Consistent, the high 16 bits represent the corresponding manufacturer-defined fault code, and the specific fault information is determined by obtaining the manufacturer-defined fault code.