HUACHENG

Low-Voltage Servo Products
User Manual V3.5

深圳市华成工业控制股份有限公司

Shenzhen Huacheng Industrial Control Co., LTD.

Introduction

First of all, thank you very much for choosing the low-voltage servo products produced by Shenzhen Huacheng Industrial Control Co., LTD.

This user manual is for low voltage servo products. It will provide you with instructions of the installation, wiring, system operation, alarm and solutions, and other relevant details and matters for attention.

In order to correctly use the low voltage servo products, give full play to the performance of the system and ensure the safety of users and equipment, please read this manual carefully before using. Incorrect operation may cause abnormal operation of low voltage servo products and even equipment damage, personal injury and other accidents! As our company is devoted to the continuous improvement of products, there will be no further notice if the material provided by the company is changed.

CONTENTS

Chapter 1 Product Specifications and Naming Rules

1.1 Servo Driver Specifications

Input Power		16～70VDC
Output Rated Current		10/20/50Arms
Output Rated Power		100/200/400/750W/1kW/1.5kW/1.8kW/2kW
Control Mode		FOC Vector Control
Environment	Work Temperature	-40℃～45℃
	Stored Temperature	-40℃～45℃
	Humidity	90%RH below, No condensation
	Altitude	3000m below
	Vibration	4.9m/sec2. Frequency10～55Hz
	Environment around	No flammable, explosive, corrosive gas, salt mist and conductive dust
Digital Input		Depending on the model, a maximum of 7 channels are supported.
Digital Output		Depending on the model, a maximum of 5 channels are supported.
External Brake Resistance		Support
Motor Band Brake Output		Support
Analog Input		Not support, Can be customized
Analog Output		Not support, Can be customized
Pulse Input		Depending on the model, input mode support P+D/A+B, input frequency single-end 200kHz/ differential 500kHz, input voltage 24V.
Pulse Output		Arbitrary frequency division output, not support.
Encoder Feedback		A±/B±/Z±、hallU±/V±/W±Incremental encoder; Absolute value encoder (custom products)
Communication Function		RJ45 Port×2：Modbus communication protocol；CANopen CiA402 communication protocol
Storage Function		EEPROM Power down save
Cooling Mode		Natural cooling
Overload Capacity		Depending on the model, overload level can be set, the maximum support 3.5x overload 3S
Alarm Function		Over-current, overheat, over-voltage, under-voltage, overload, over-speed, abnormal encoder and other alarms
Upper computer software		ServoTuner Servo upper computer software

1.2 Servo Drive and Servo Motor Naming

1.2.1 Servo Driver Naming

S S T S 1 A U 1 0 0
① ② ③④⑤ ⑥

Identification	Driver Series	Identification	Extend	
①	SS: RS485 Bus	④	A: With Pulse Control	
	ES: EtherCAT Bus		B: Without Pulse Control	
Identification	Applicated Industry	Identification	Input Voltage	
②	TS: General industry	⑤	U: DC24V	R:Three-phase AC220V
	ZJ: Gate Industry		V: DC48V	S:Single-phase AC220V
	YT: PTZ Industry		W: DC60V	T:Three-phase AC380V
			X: DC16-70V	
Identification	Number of Axis	Identification	Output Rated Current	
③	1: 1 Axis	⑥	039: 0.3A(3×10^{-1}A)	
	2: 2 Axes		289: 2.8A(28×10^{-1}A)	
	3: 3 Axes		100: 10A(10×10^{0}A)	
	4: 4 Axes		101: 100A(10×10^{1}A)	

For example:“SSYT3BX100100100” indicates“Modbus/PTZ/3-axis/16~70VDC/10A10A10A”; “SSTS1AX500” indicates “Modbus/CANopen/General/1-axis/With pulse/16~70VDC/50A”; “ESTS2BX200200” indicate “EtherCAT/General/2-axis/16~70VDC/20A20A”;“CSTS2BX100200” indicates “CANopen/General/2-axis/16~70VDC/10A20A”.

* **NOTES:** CANopen is standard for SS series of single-axis low-voltage servo, so the original CS series and SS series are unified into SS series (i.e. CS single-axis series is removed).

1.2.2 Servo Motor Naming

HC7 J 06 04 30 D 1 K U A A
① ② ③ ④ ⑤ ⑥ ⑦ ⑧ ⑨ ⑩

Identification	Motor Series	Identification	Extend
①	HC7:HC7 Series of motor	⑩	A:Reserve
Identification	**Motor Inertia**	**Identification**	**Motor Brake**
②	J:Small Inertia G:Medium Inertia	⑨	A:Without Brake
	C:Special customization		B:With Brake
Identification	**Motor Frame**	**Identification**	**Motor Structure**
③	04:40mm Base	⑧	U:With key and screw hole,without brake, with oil seal
	06:60mm Base		V:With key and screw hole, with brake, with oil seal
	08:80mm Base	**Identification**	**Encoder Type**
	09:90mm Base	⑦	1K:2500 line incremental magnetic encoder
	10:100mm Base		1H:2500 line incremental optical encoder
	13:130mm Base		3K:17 Bit ABS encoder
	18:180mm Base		4K:23 Bit ABS encoder
Identification	**Rated Power**	**Identification**	**Rated Voltage**
④	A5:50W	⑥	C:60V
	01:100W		D:48V
	02:200W		F:24V
	04:400W	**Identification**	**Rated Speed**
	08:750W	⑤	10:1000rpm
	10:1kW		15:1500rpm
	15:1.5kW		20:2000rpm
	20:2kW		25:2500rpm
			30:3000rpm

Chapter 2 Product Installation and Precautions

2.1 Safety Precautions

Declare: In order to prevent damage to people and equipment, please observe the following items when using servo products.

2.1.1 Danger

Not directly immersed in water, oil or other liquid environment for use.	May cause electric shock, fire, malfunction, damage
Not directly exposed to conductive dust, salt spray environment.	May cause electric shock, fire, malfunction, damage
Do not use in inflammable and explosive environment.	May cause electric shock, fire, malfunction, damage
Not be used in the environment with severe shock.	May cause electric shock, fire, malfunction, damage
Do not use in the environment with poor heat dissipation.	May cause fire, malfunction, damage
Do not connect motor directly with city electricity.	May cause electric shock, malfunction, damage
Do not expose or stress motor wires.	May cause electric shock, malfunction, damage
Drivers, motors and moving parts must not be touched while operation.	May cause electric shock, malfunction, damage

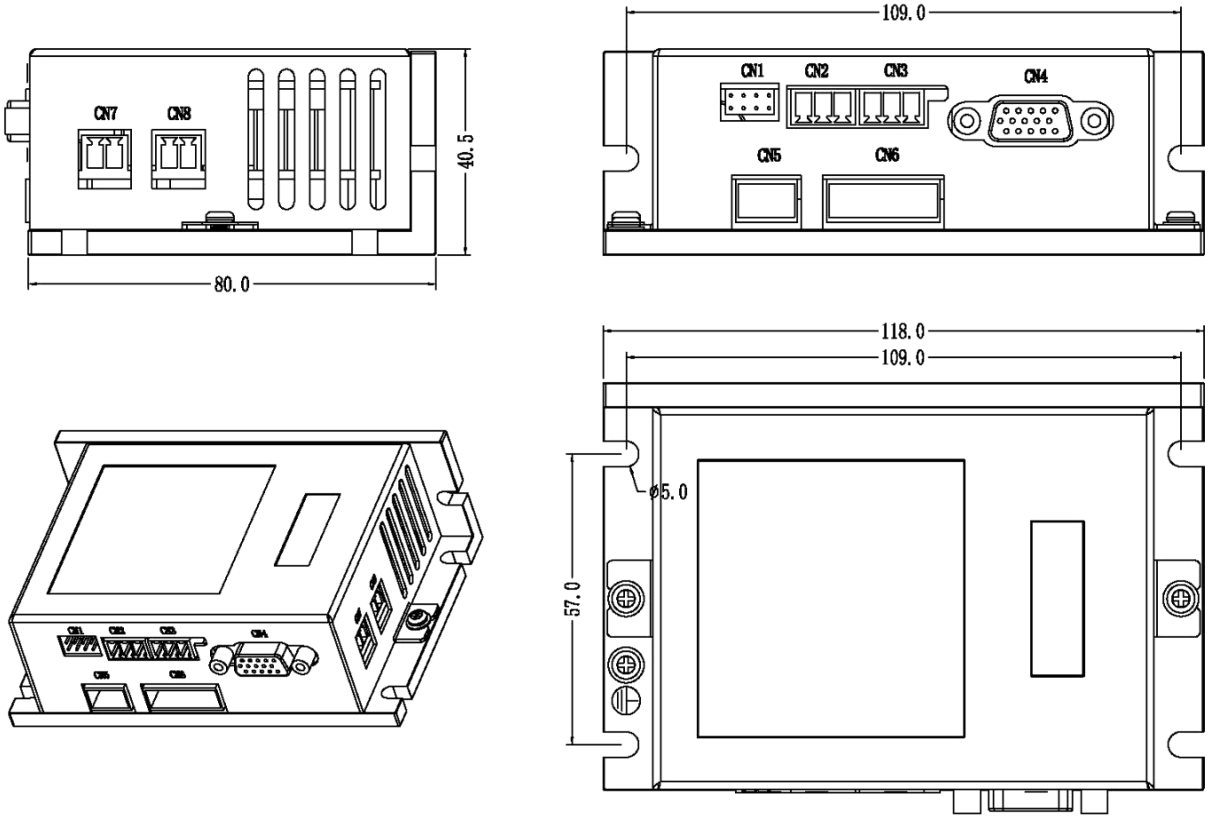
2.1.2 Notes

Do not fall or invert during lifting, do not grab motor shaft end and cable.	May cause injury, malfunction
Do not place heavy objects on the product.	May cause malfunction
Use after correctly wiring in accordance with user manual by electrical engineer.	May cause electric shock, malfunction, damage
Driver, motor and encoder must be well grounded.	May cause electric shock, interference
Do not expose the product directly in outdoor use.	May cause injury, malfunction
Do not disassemble or convert the product.	May cause injury, malfunction
Follow the specific installation method and direction.	May cause injury, malfunction
Ensure the driver and motor are used at reasonable temperature, humidity and altitude.	May cause malfunction
The driver input voltage must be within the specific range.	May cause malfunction
Be sure to cut off the power supply if it has been out of service for a long time.	May cause injury
When the product is scrapped, it shall be treated as industrial waste.	May cause environmental pollution

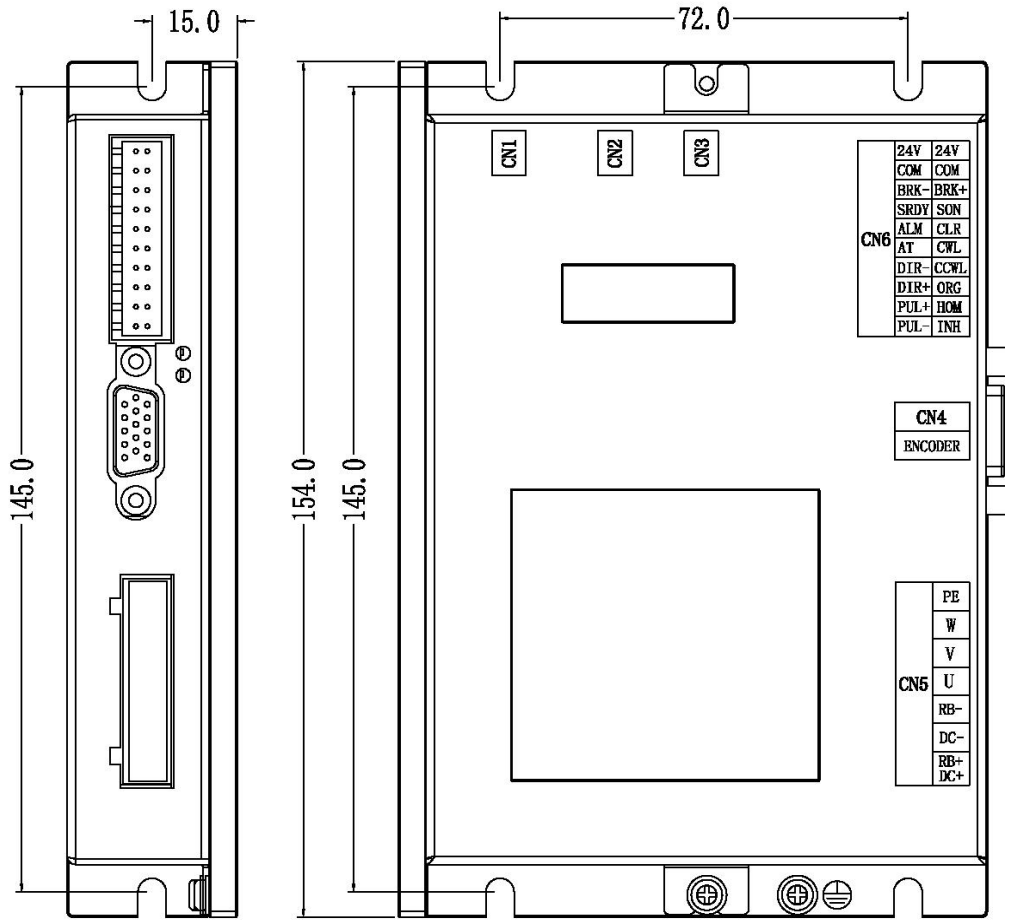
2.2 Product Installation and Wiring

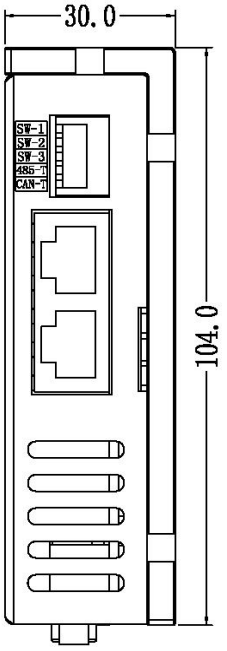
2.2.1 Product Size and Installation

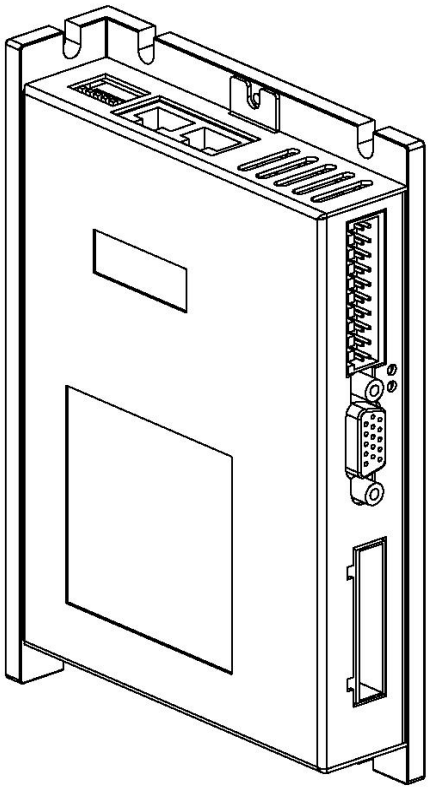
2.2.1.1 Driver Size and Installation



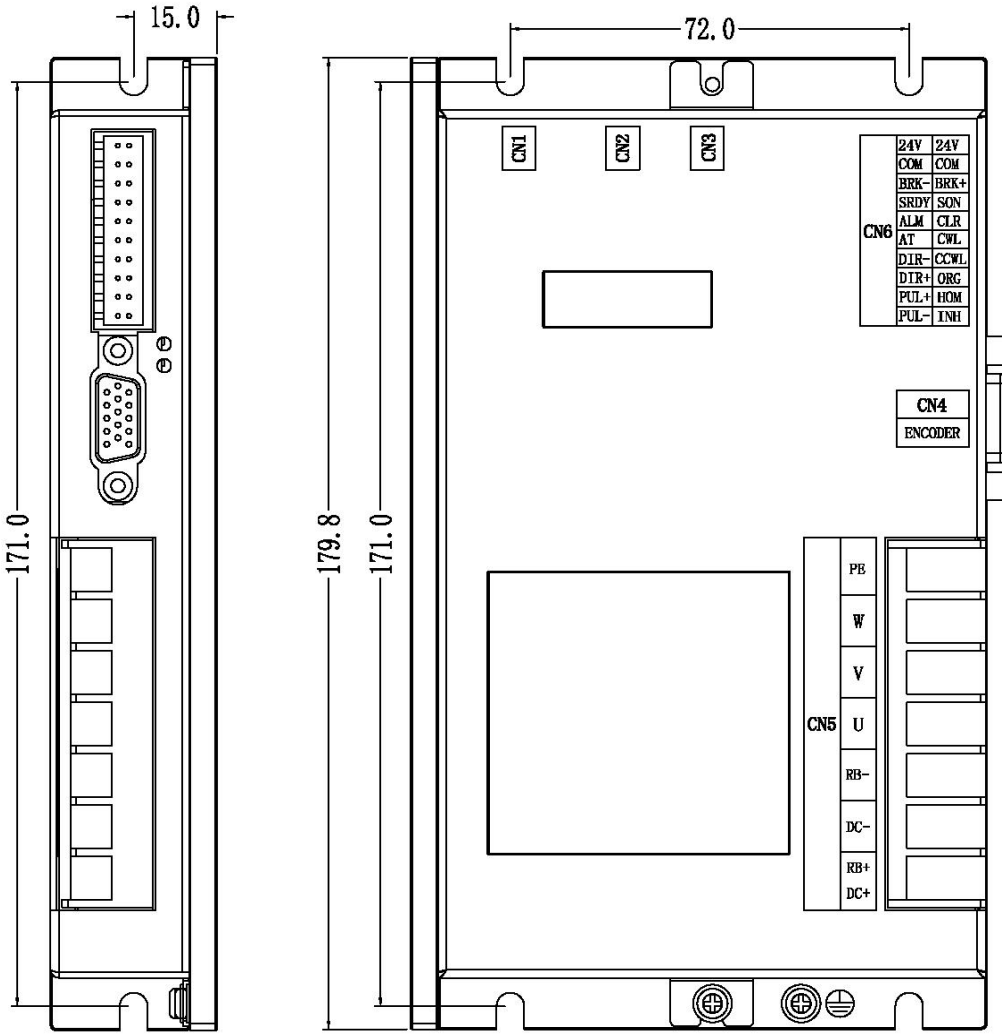
SSZJ1B

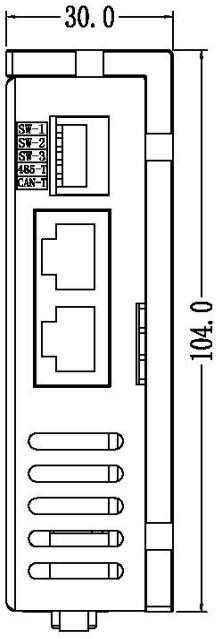

15.0
72.0
145.0
154.0
145.0
CN1
CN2
CN3
CN6
24V 24V
COM COM
BRK- BRK+
SRDY SON
ALM CLR
AT CWL
DIR- CCWL
DIR+ ORG
PUL+ HOM
PUL- INH
CN4
ENCODER
CN5
PE
W
V
U
RB-
DC-
RB+
DC+

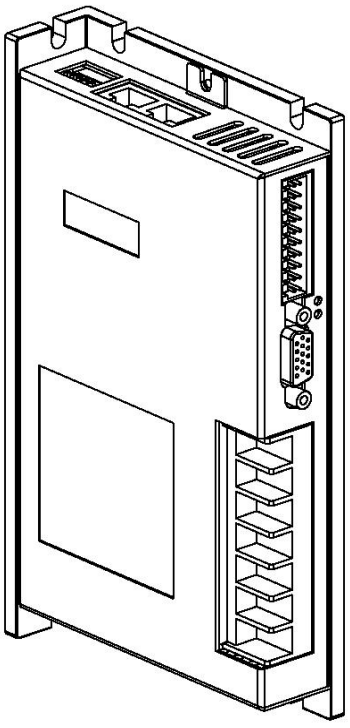

30.0
104.0
SW-1
SW-2
SW-3
485-T
CAN-T



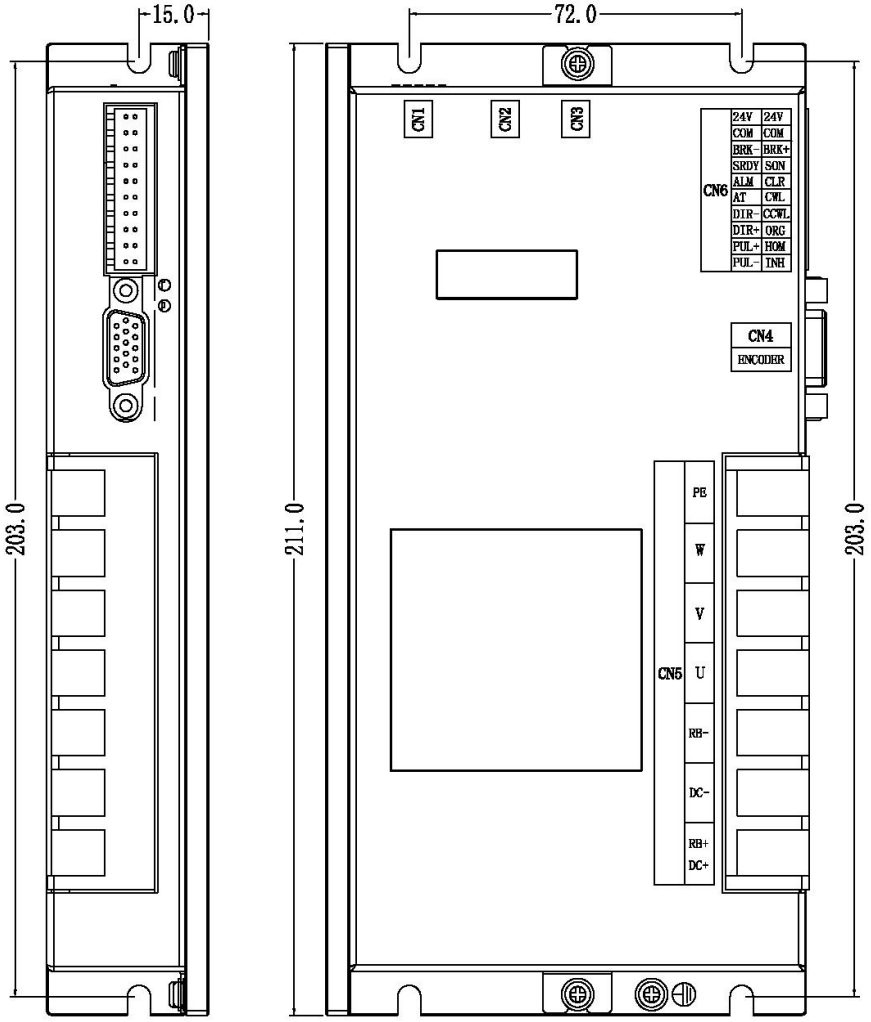
SSTS1A100

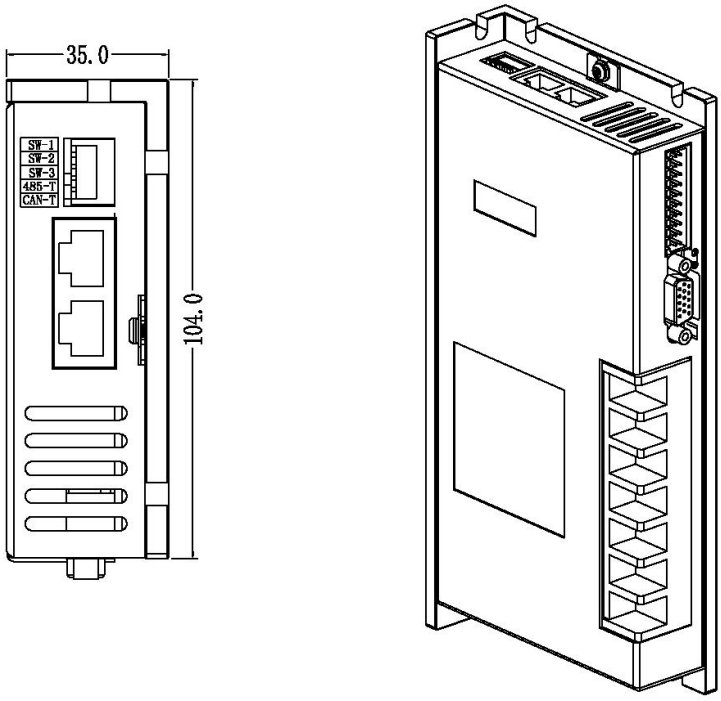

15.0
72.0
171.0
179.8
171.0
CN1
CN2
CN3
CN6
24V 24V
COM COM
BRK- BRK+
SRDY SON
ALM CLR
AT CWL
DIR- CCWL
DIR+ ORG
PUL+ HOM
PUL- INH
CN4
ENCODER
CN5
PE
W
V
U
RB-
DC-
RB+
DC+


30.0
104.0
SW-1
SW-2
SW-3
485-1
CAN-1

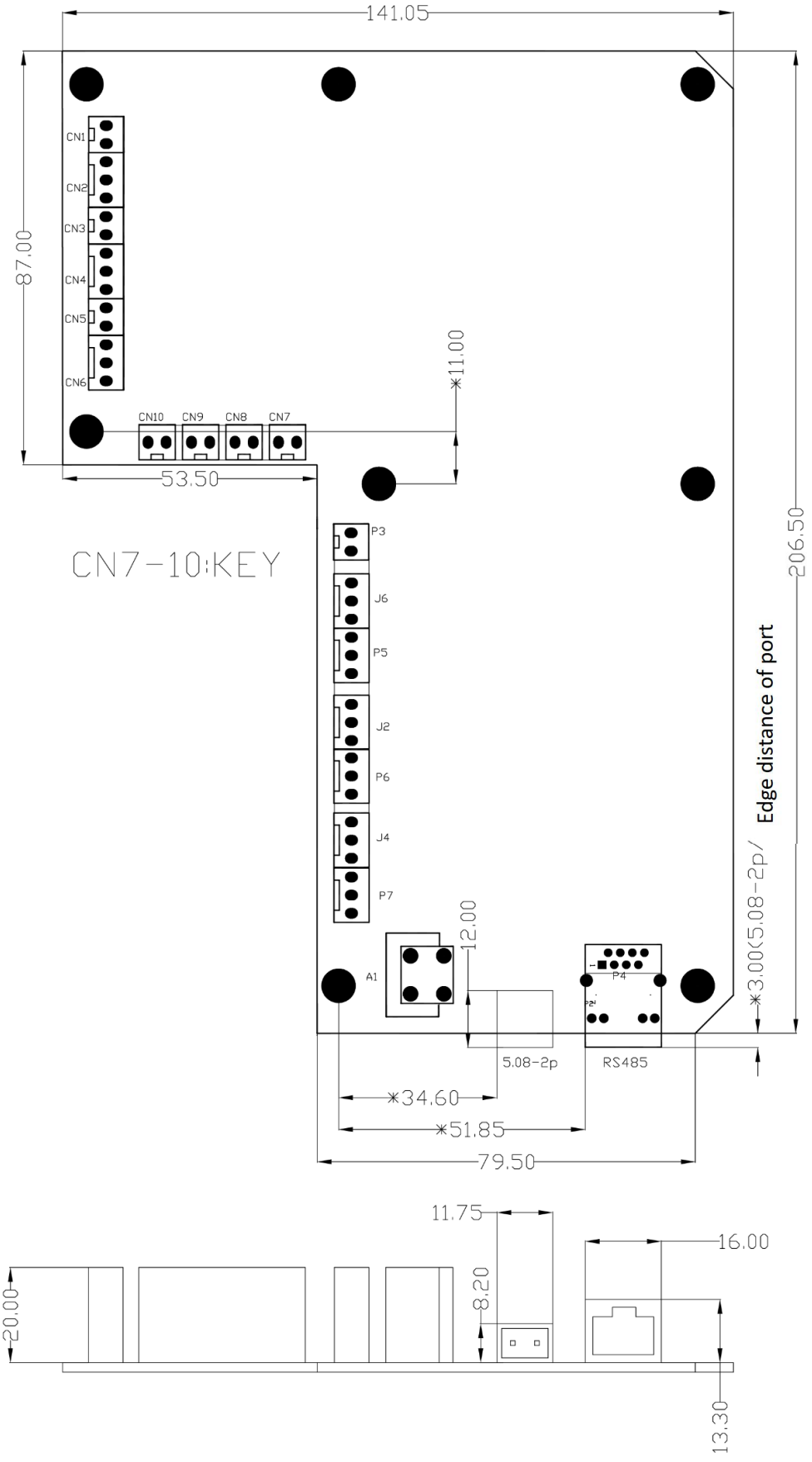


SSTS1A200


15.0
72.0
203.0
211.0
203.0
CN1
CN2
CN3
CN6
24V 24V
COM COM
BRK- BRK+
SRDY SON
ALM CLR
AT CWL
DIR- CCWL
DIR+ ORG
PUL+ HOM
PUL- INH
CN4
ENCODER
CN5
PE
W
V
U
RB-
DC-
RB+
DC+


35.0
104.0
SW-1
SW-2
SW-3
485-T
CAN-T

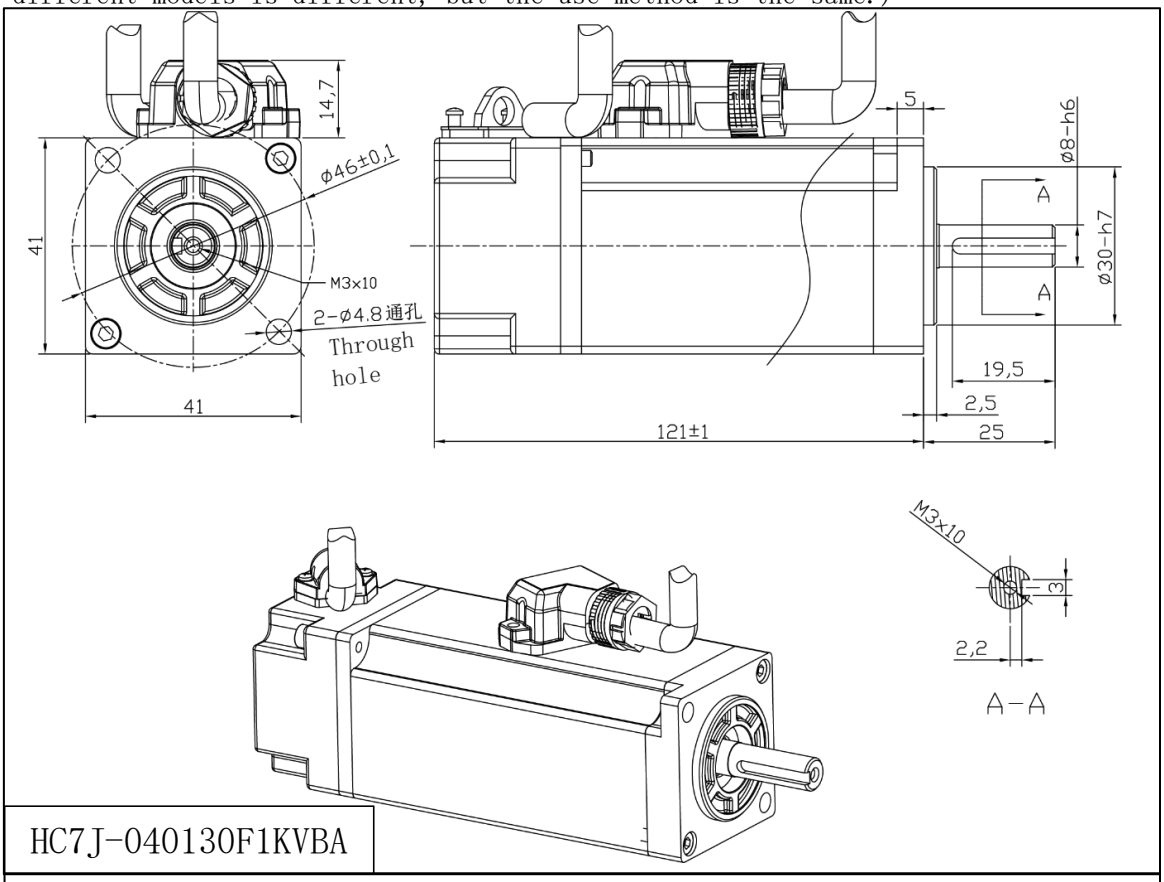
SSTS1A500



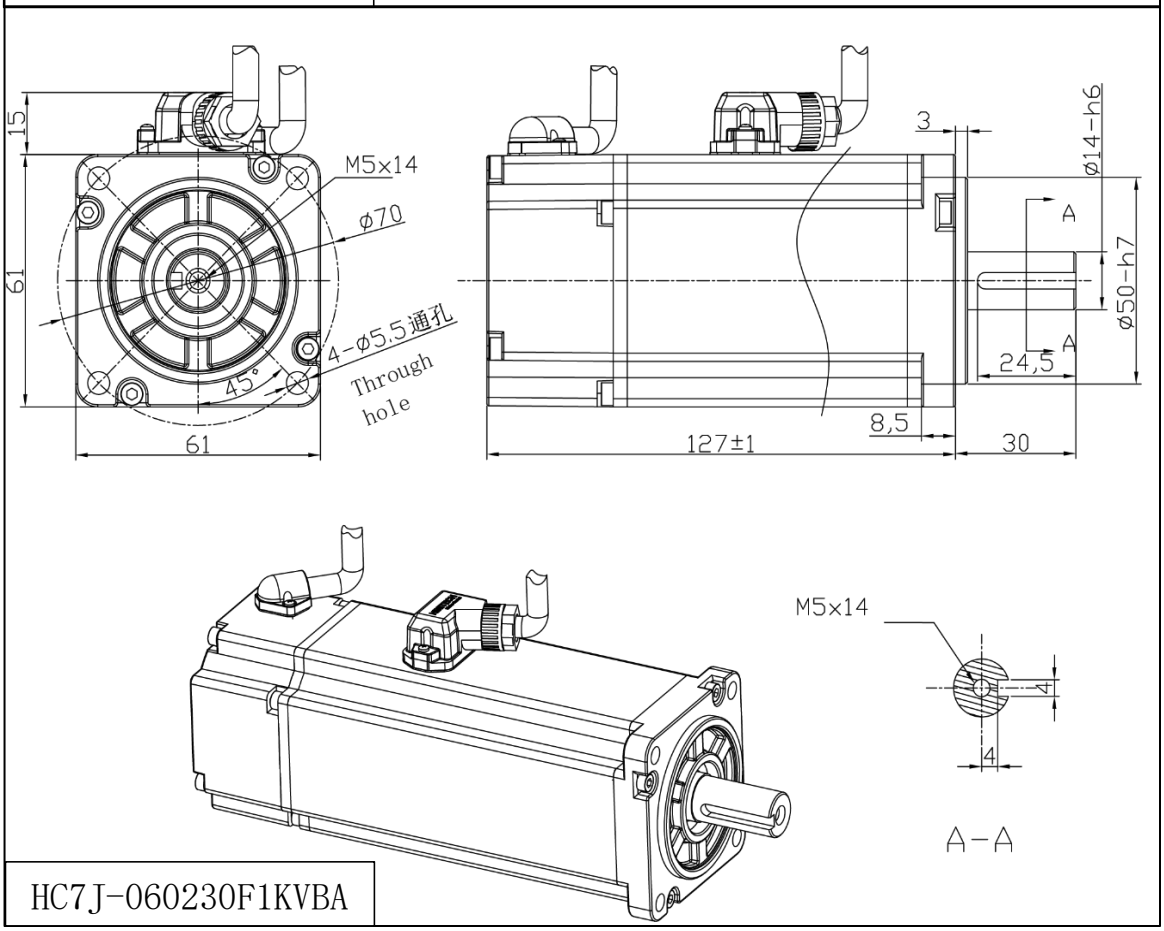
SSYT3B100100100

2.2.1.2 Motor Size and Installation

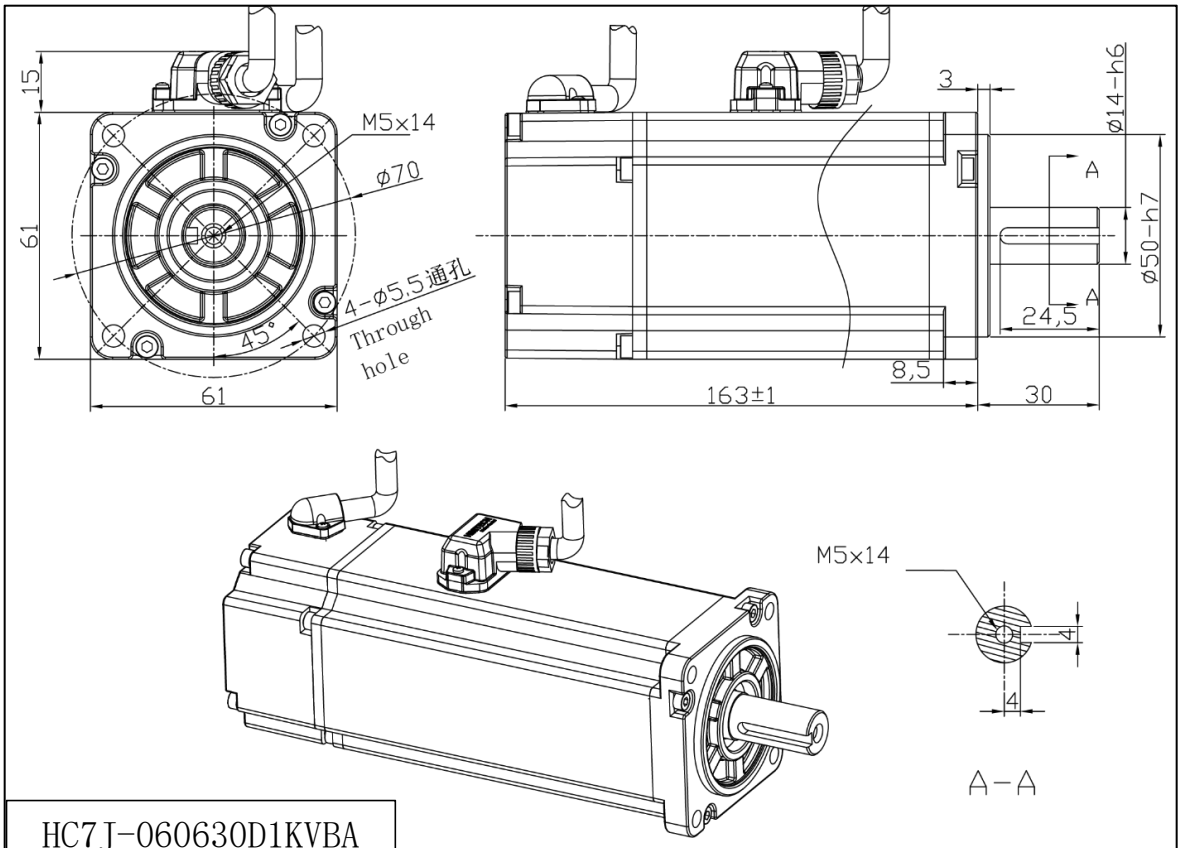
Dimensions and installation of servo motor are shown as below. (Note: The appearance of different models is different, but the use method is the same.)



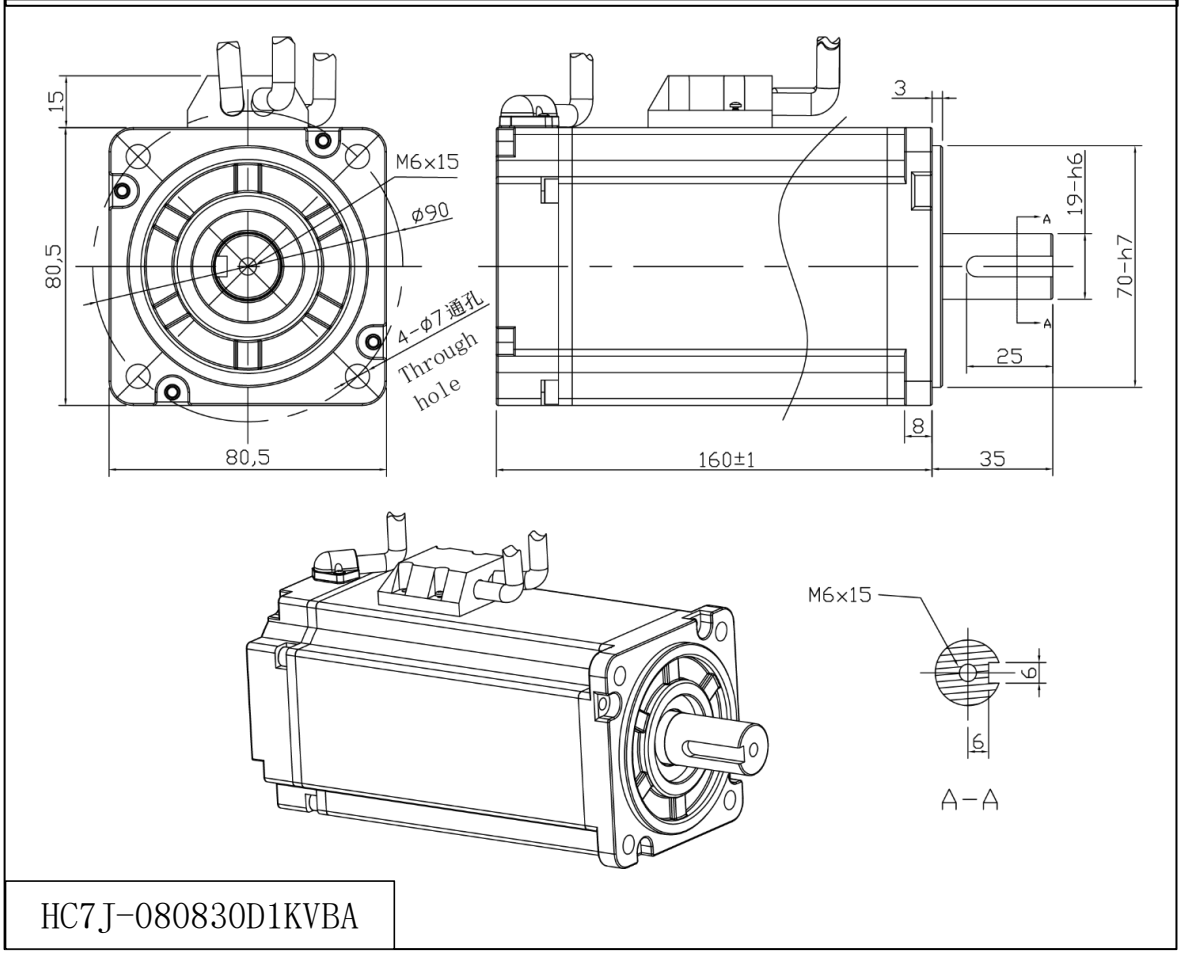
HC7J-040130F1KVBA



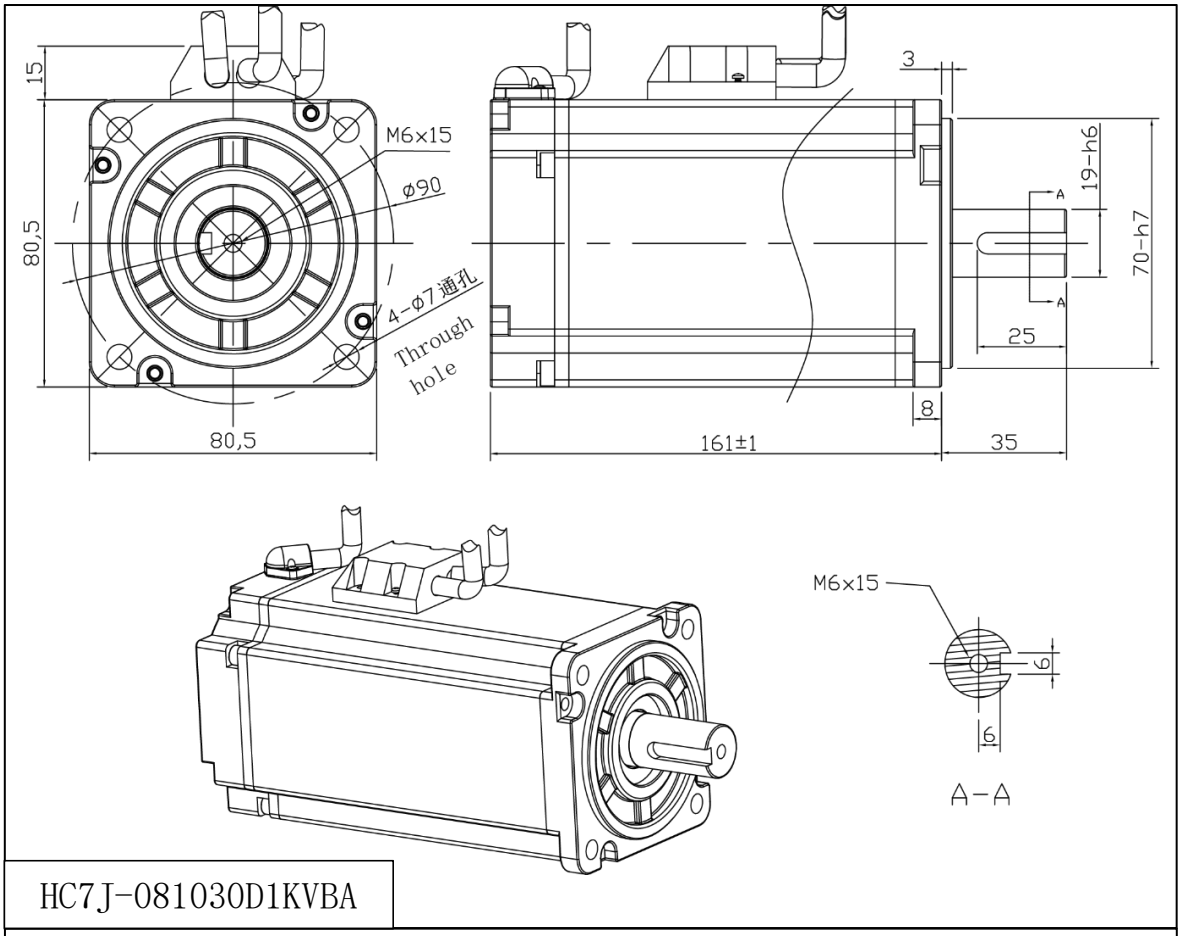
HC7J-060230F1KVBA



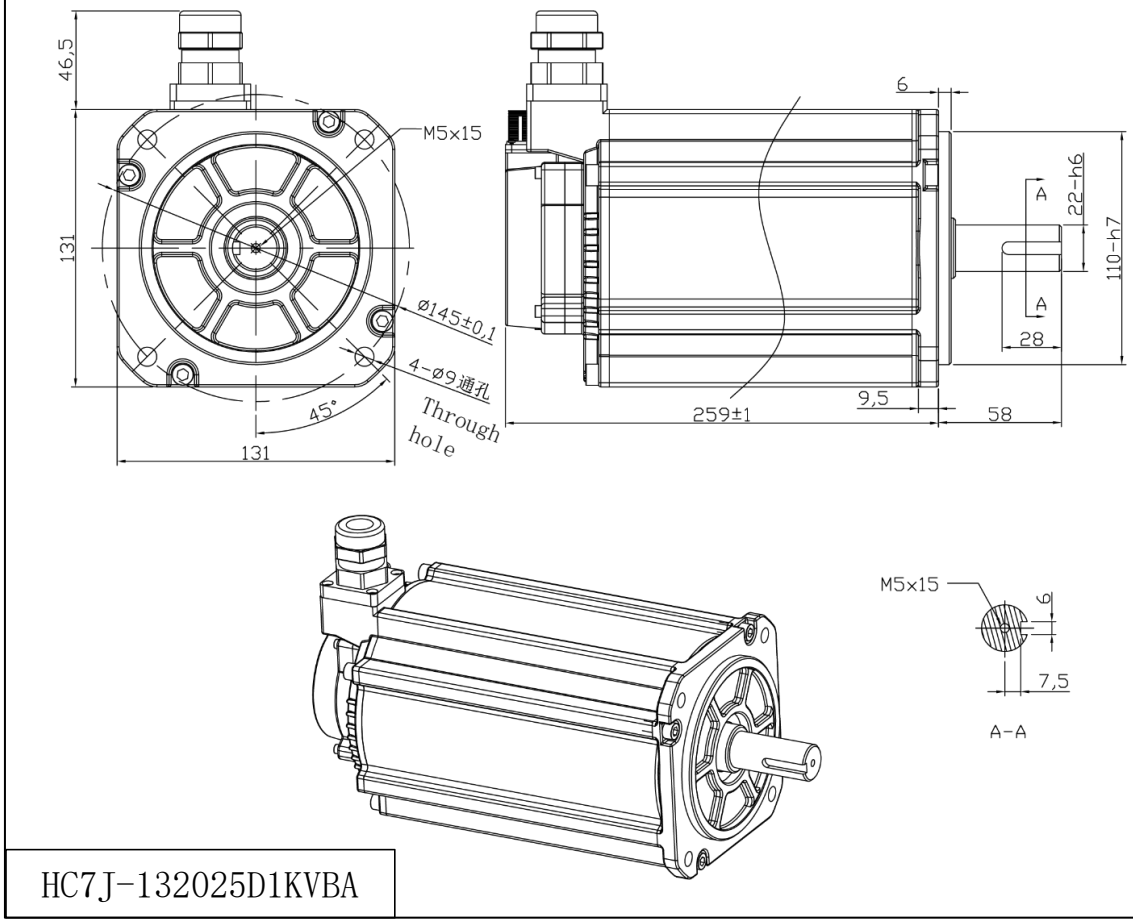
HC7J-060630D1KVBA



HC7J-080830D1KVBA



HC7J-081030D1KVBA



HC7J-132025D1KVBA

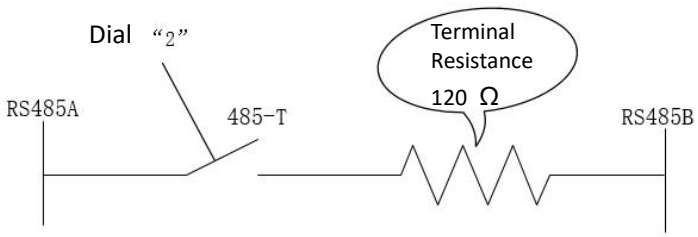
Chapter 3 Product Wiring

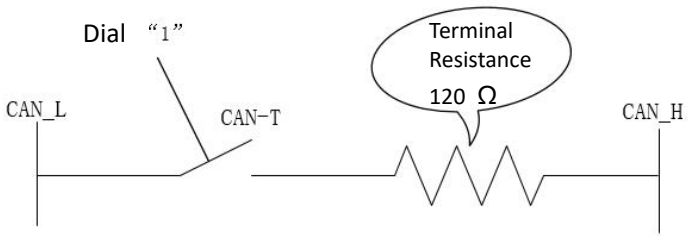
3.1 Port Definition

Note: The appearance and port layout of different models are different, but the use method is the same. For details, please refer to the silk screen printing of the physical shell.

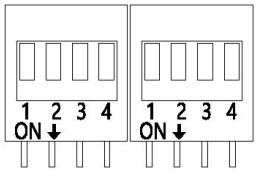
3.1.1 Dial Switch CN1

The dial switch can be used for specific functions. For SSTS1A, the dial switch is used to set the terminal resistance of RS485 or CAN bus.(It takes effect when it is turned to ON).





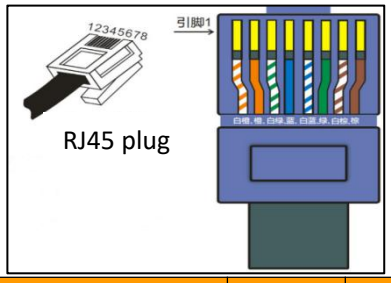
For SSZJ1B product, **dial switch can be used to modify the door opening and closing operation time:**



Run Time	SW-1	SW-2	SW-3	SW-4
0.5 seconds	—	—	ON	ON
0.6 seconds	—	—	ON	OFF
0.7 seconds	—	—	OFF	ON
0.8 seconds	—	—	OFF	OFF

3.1.2 Communication Port CN2/CN3

The communication port is used for communication between servo and upper computer. These two communication ports have the same function and can be used to connect multiple servo slave stations in series. SSTS1A adopts standard RJ45 port. It is defined as following:



CN2/3	Signal	Explain	CN2/3	Signal	Description
1	CAN_L	CAN_L signal	5	NC	
2	CAN_H	CAN_H signal	6	NC	
3	GND	CAN connect earth	7	A	RS485 signal A
4	NC		8	B	RS485 signal B

SSZJ1B adopts 3-pin green terminal. It is defined as following:

CN2/3	Signal	Explain
1	A	RS485 signal A
2	GND	Power supply connect earth
3	B	RS485 signal B

3.1.3 Motor Encoder Port CN4

The encoder ports of each sub-series of driver are unified as DB15, which supports incremental encoder and absolute value encoder.

CN4	Signal	Explain	Signal	Explain
1	V+	Encoder signal V+		
2	U+	Encoder signal U+		
3	Z+	Encoder signal Z+		
4	B+	Encoder signal B+		
5	A+	Encoder signal A+		
6	V-	Encoder signal V-		
7	U-	Encoder signal U-		
8	Z-	Encoder signal Z-		
9	B-	Encoder signal B-		
10	A-	Encoder signal A-		
11	W+	Encoder signal W+	B	Encoder serial signal B
12	W-	Encoder signal W-	A	Encoder serial signal A
13	5V	Encoder 5V Power supply	5V	Encoder 5V Power supply
14	GND	Encoder signal connects earth	GND	Encoder signal connects earth
15	PE	Encoder shield	PE	Encoder shield

3.1.4 Input Power Port CN5

DC voltage input port. SSZJ1B adopts 2-pin terminal:

CN5	Signal	Explain
1	DC+	Power supply+
2	DC-	Power supply-

SSTS1A adopts 7-pin terminal, and it contains interfaces for input power supply, motor power lines, and brake resistance:

CN5	Signal	Explain
1	PE	Motor shield
2	W	Motor W phase
3	V	Motor V phase
4	U	Motor U phase
5	RB-	Brake resistance-
6	DC-	Power supply-
7	DC+/ RB+	Power supply+/Brake resistance+

3.1.5 Motor Power Line/Signal Input and Output Port CN6

SSZJ1B sub-series CN6 is used for power line connection port of three-phase AC servo motor :

CN6	Signal	Explain
1	U	Motor U-phase
2	V	Motor V-phase
3	W	Motor W-phase
4	PE	Motor shield

SSTS1A sub-series CN6 is used for signal input and output:

CN6	Signal	Explain	CN6	Signal	Explain
1	24V	External connect 24V+	11	24V	Output 24V+
2	COM	External connect 24V-	12	COM	Output 24V-
3	BRK-	Brake-	13	BRK+	Brake+
4	SRDY	Servo ready DO	14	SON	Servo Enable DI
5	ALM	Servo alarm DO	15	CLR	Alarm clear DI
6	AT	Location arrival DO	16	CWL	Clockwise travel limit DI
7	DIR-	Direction signal-	17	CCWL	Counterclockwise travel limit DI
8	DIR+	Direction signal+	18	ORG	Origin signal DI
9	PUL+	Pulse signal+	19	HOM	Return command DI
10	PUL-	Pulse signal-	20	INH	Command Pulse Ban DI

3.1.6 Brake Resistance Port CN7

Used for external braking resistance of SSZJ1B. SSTS1A sub-series focuses on CN5.

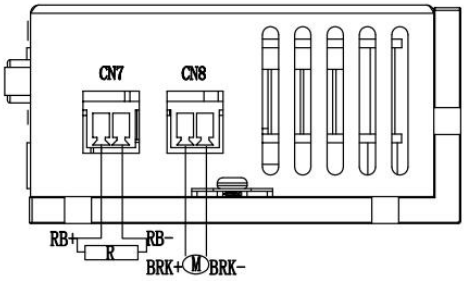
CN7	Signal	Explain
1	RB+	Brake resistance+
2	RB-	Brake resistance-

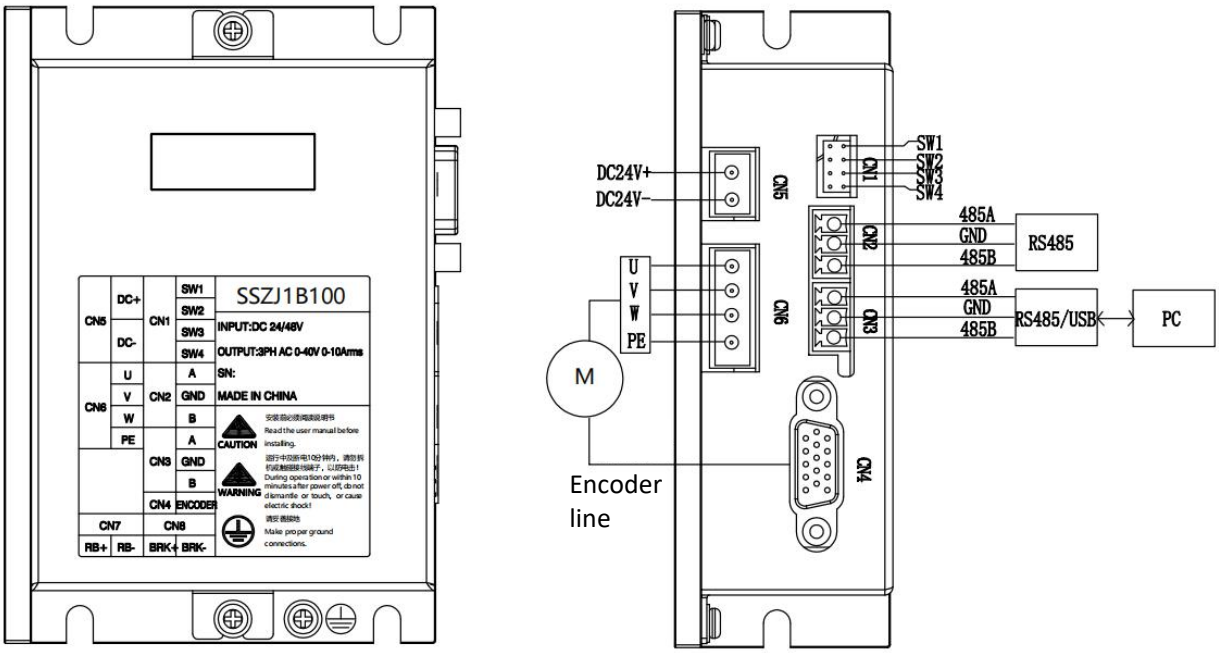
3.1.7 Motor Band Brake Port CN8

Used for SSZJ1B motor brake. SSTS1A sub-series focuses on CN6.

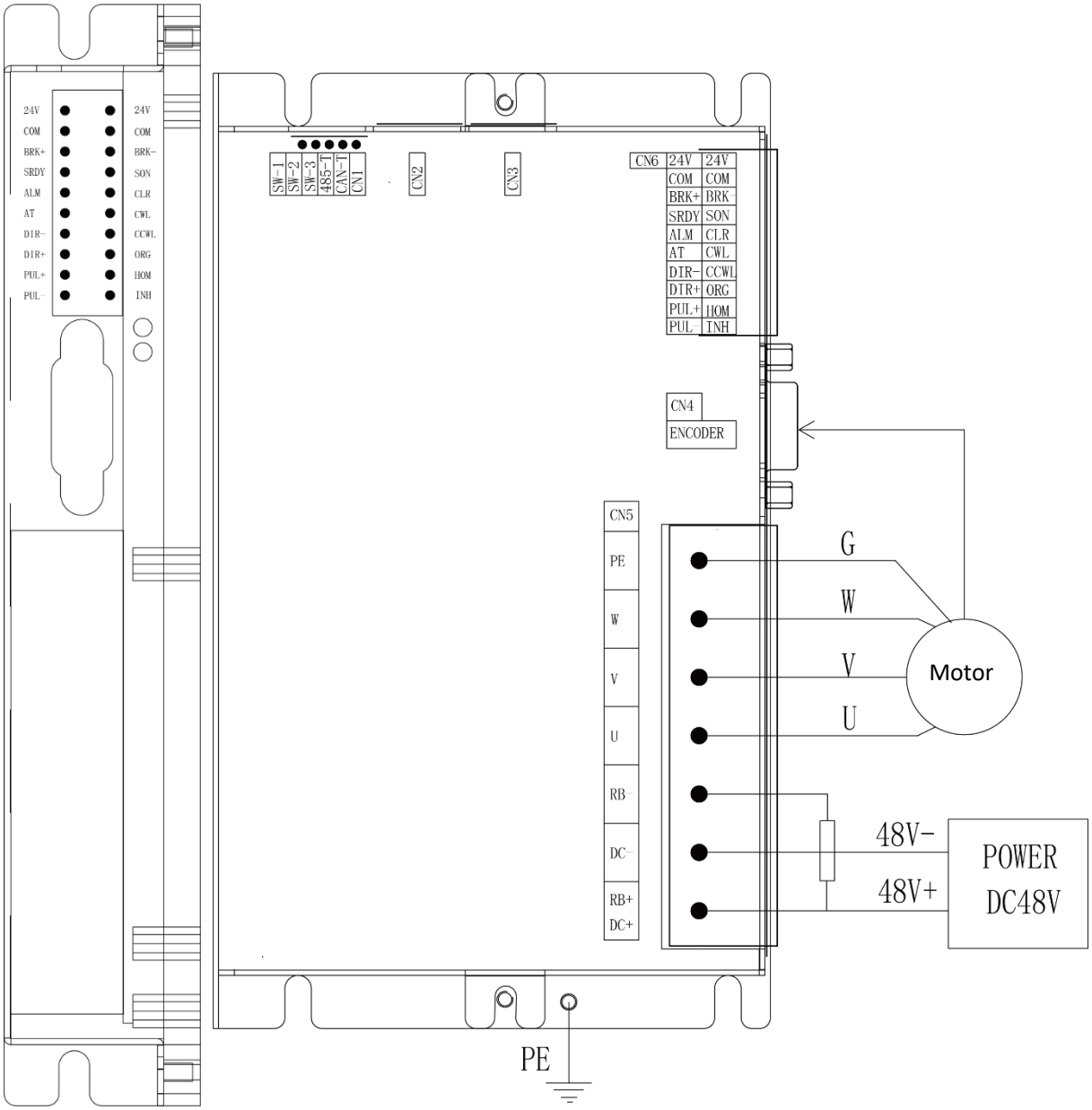
CN8	Signal	Explain
1	BRK+	Brake+
2	BRK-	Brake-

3.2 Electrical Wiring Diagram

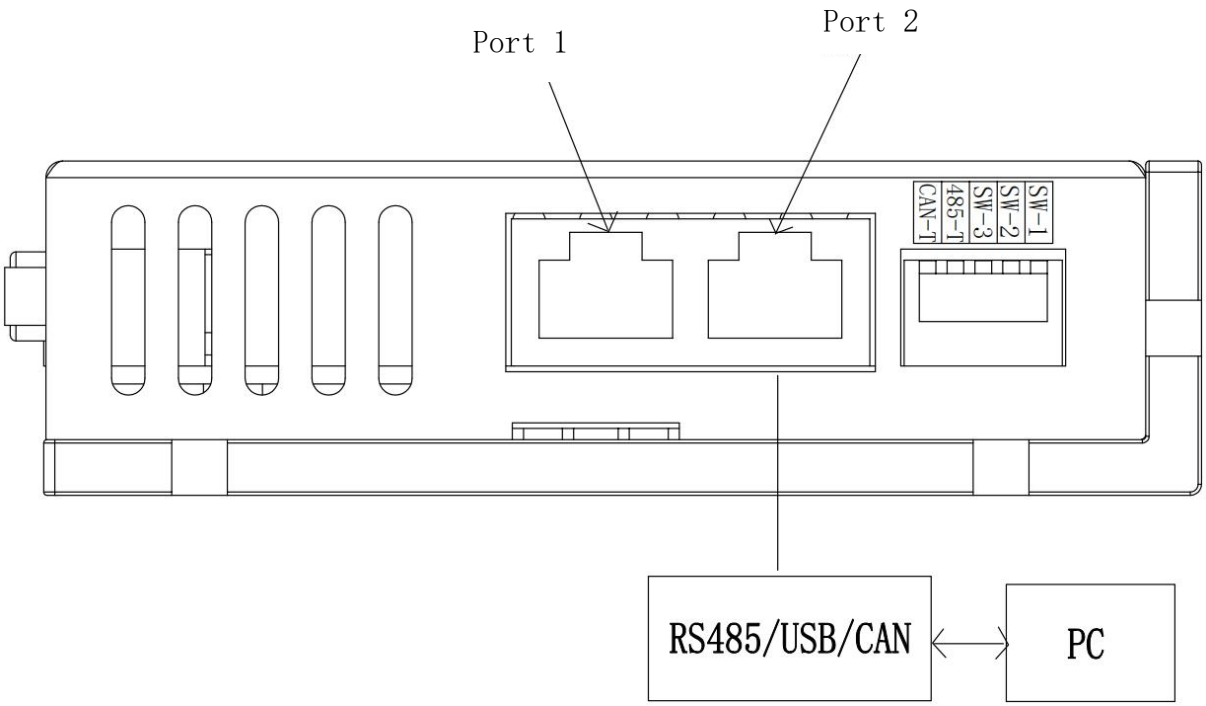




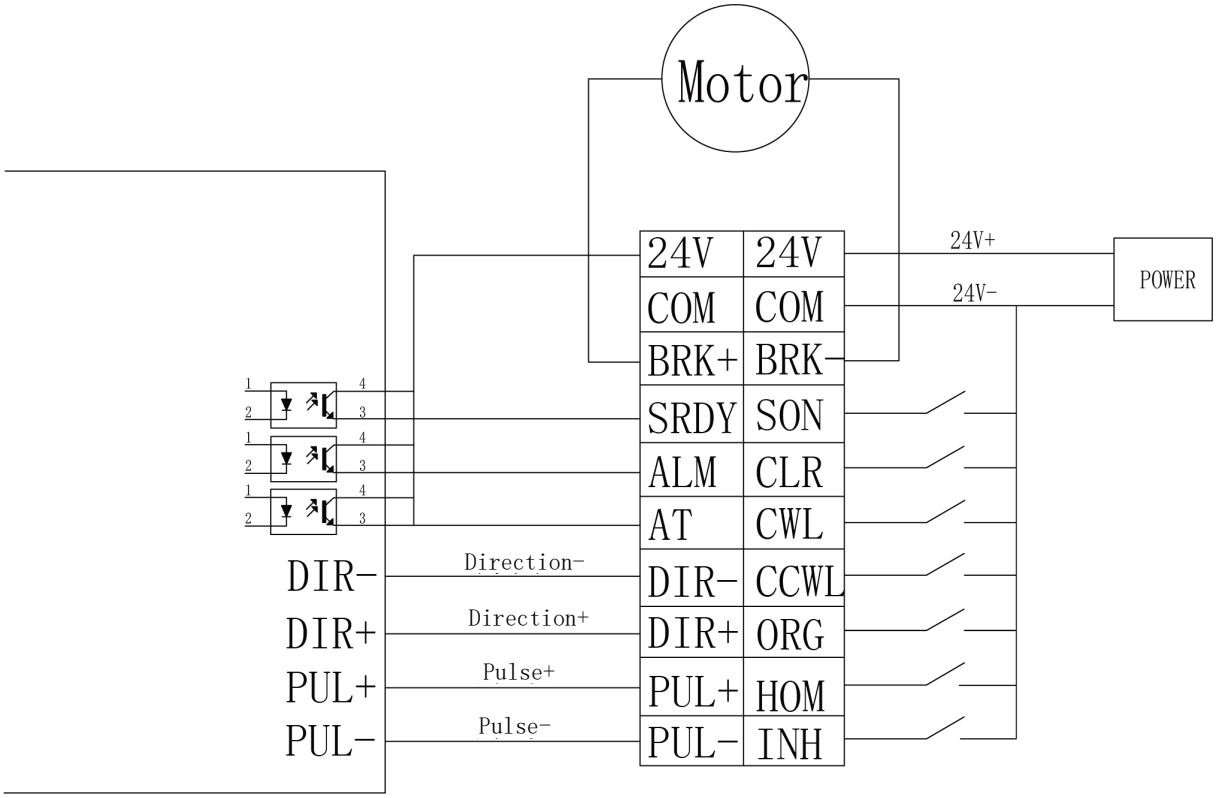
SSZJ1B Wiring Diagram



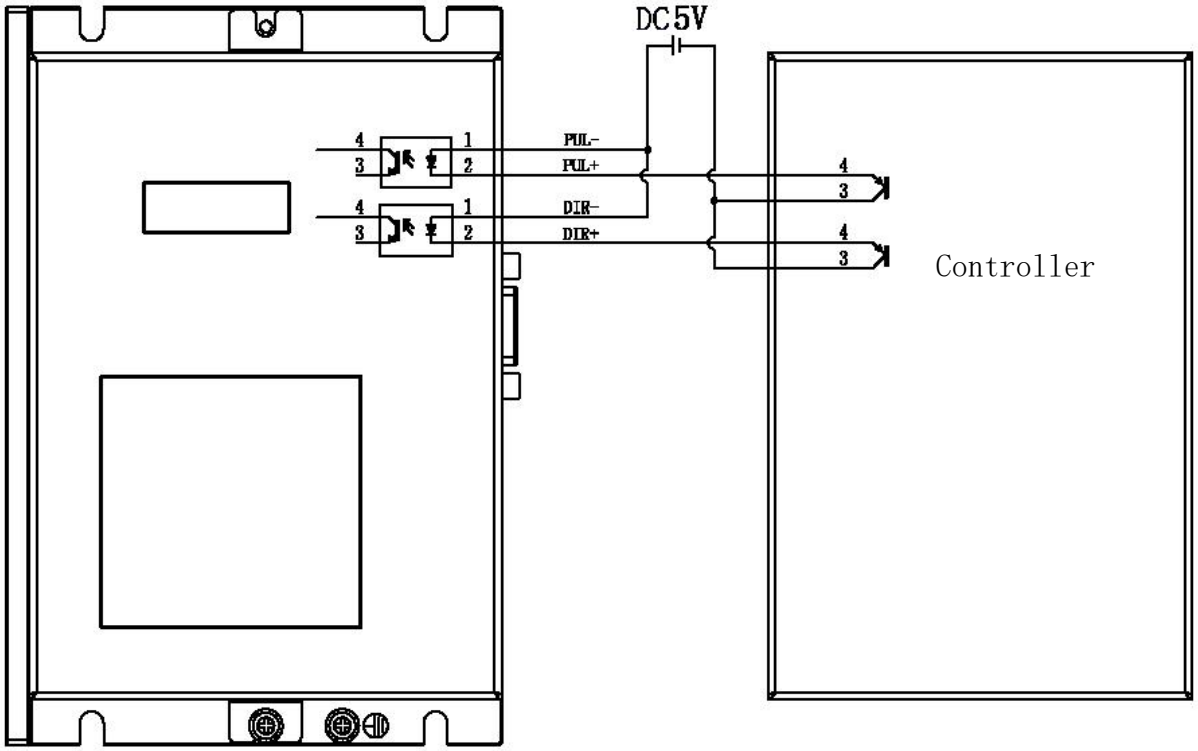
SSTS1A Wiring Diagram



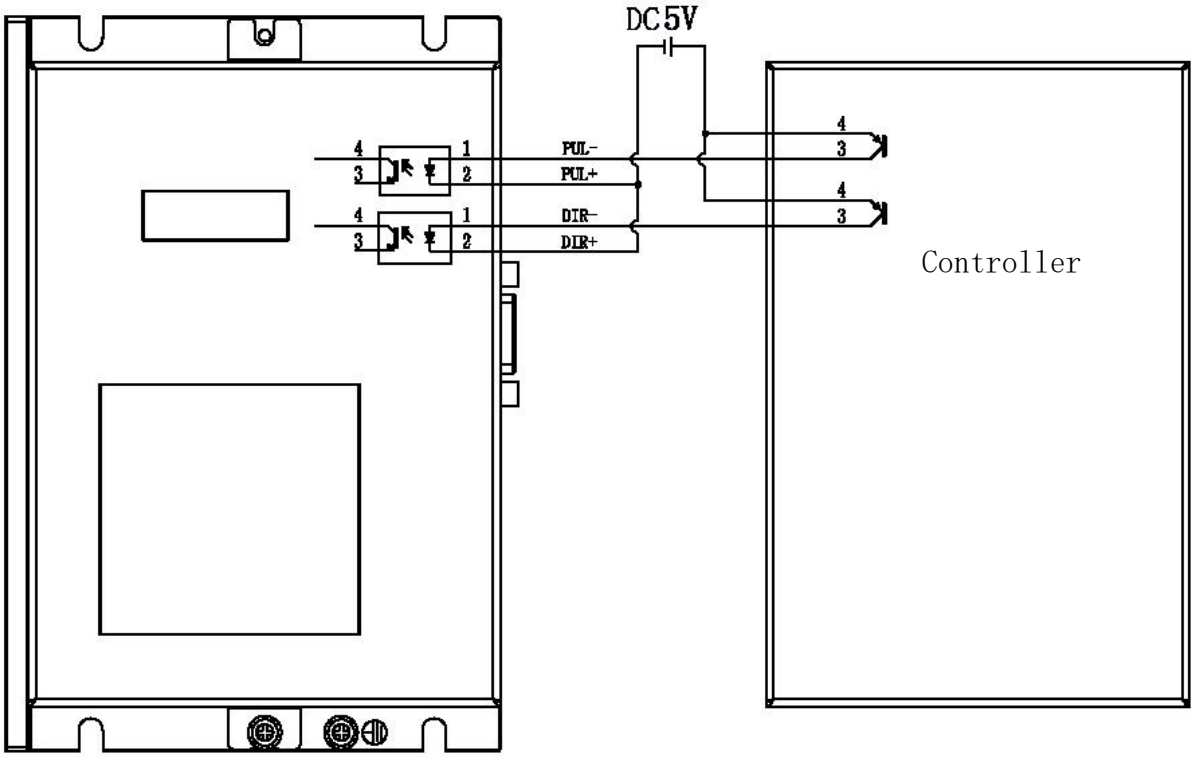
SSTS1A Wiring Diagram



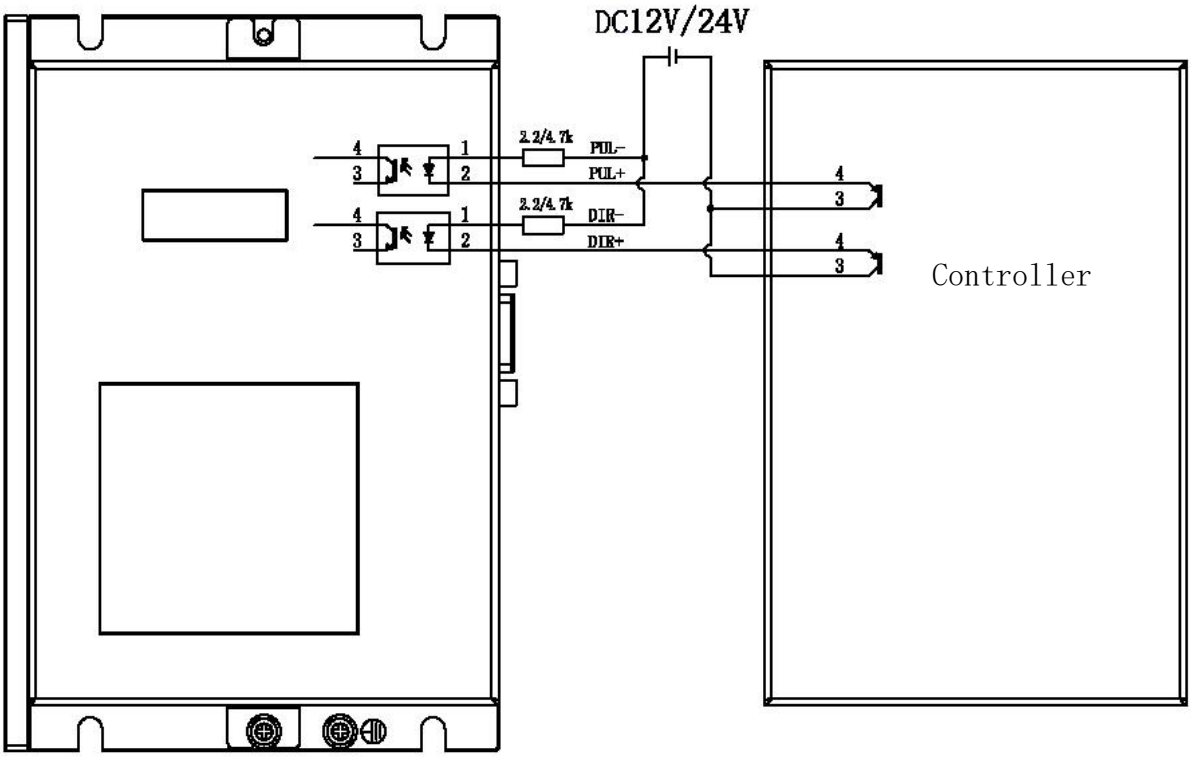
SSTS1A Wiring Diagram



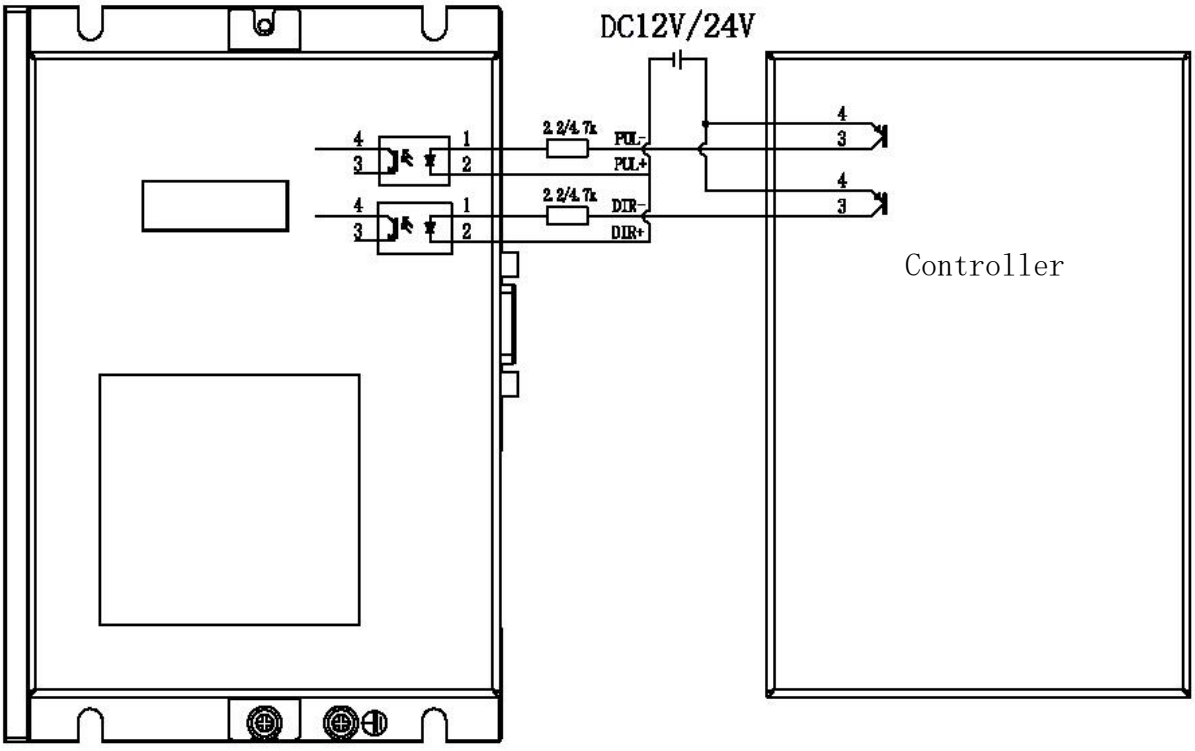
Single Terminal Wiring 1



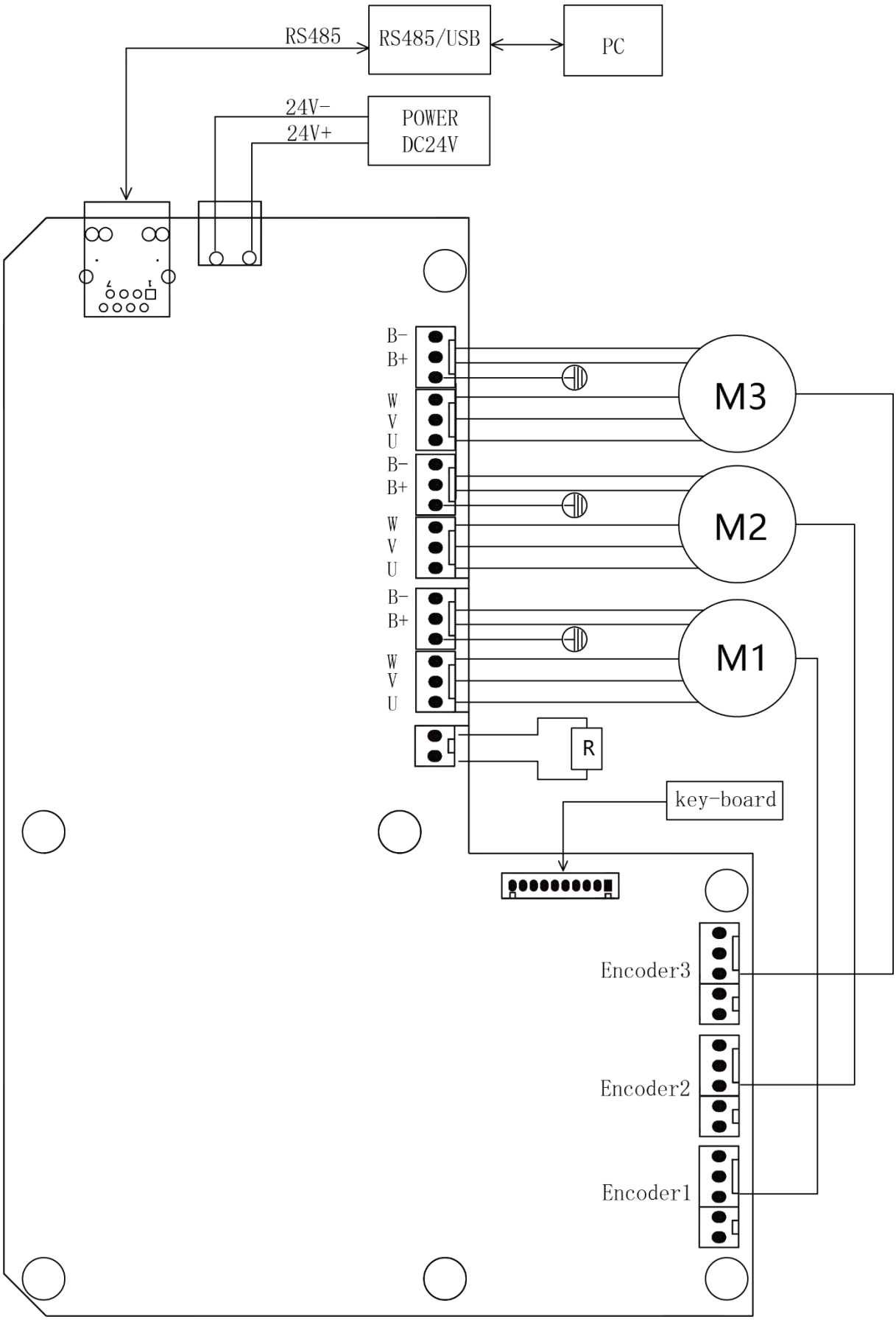
Single Terminal Wiring 2



Single Terminal Wiring 3



Single Terminal Wiring 4



SSYT3B Wiring Diagram

Chapter 4 Parameter Table, Communication Function and Upper Computer

4.1 Parameter Table

The main parameters of servo driver are shown in the table below. Users can set various parameters conveniently by ServoTuner, the servo upper computer software, according to the application requirements.

Servo Driver Parameter Table

<table>
<tr><th>Param number</th><th>Param name</th><th>Parameter range (Default)</th><th>Parameter Description (R/W-Read-Write, R-Only read, P-Position mode, S-Speed mode, T-Torque mode)</th></tr>
<tr><td>P000</td><td>Servo slave station address ★(Note)</td><td>1~127 (1)</td><td>Servo Modbus communication slave station address, and at the same time be the node ID of CAN communication. Servo supports standard Modbus RTU communication protocol, and can be used as a slave to communicate with master PLC, motion control card, and upper computer software. Supports CiA402 protocol of CAN2.0A.
Read-Write Attributes: R/W Applicable Mode: ALL</td></tr>
<tr><td>P001</td><td>Servo mode ★</td><td>0~20 (7)</td><td>Servo mode selection.
<table>
<tr><td>0</td><td>Pulse Position Mode</td></tr>
<tr><td>1</td><td>Analog speed Mode</td></tr>
<tr><td>2</td><td>Analog Torque Mode</td></tr>
<tr><td>3</td><td>Pulse Position/Analog speed Mode</td></tr>
<tr><td>4</td><td>Pulse Position/Analog Torque Mode</td></tr>
<tr><td>5</td><td>Analog Speed/Analog Torque Mode</td></tr>
<tr><td>6</td><td>Communication Position Mode; PP Mode /CSP Mode of CANopen</td></tr>
<tr><td>7</td><td>Communication Speed Mode; PV Mode/CSV Mode of CANopen</td></tr>
<tr><td>8</td><td>Communication Torque Mode; PT Mode/CST Mode of CANopen</td></tr>
<tr><td>9</td><td>Communication Position/Communication Speed Mode</td></tr>
<tr><td>10</td><td>Communication Position/Communication Torque Mode</td></tr>
<tr><td>11</td><td>Communication Speed/Communication Torque Mode</td></tr>
</table>
Read-Write Attributes: R/W Applicable Mode: ALL</td></tr>
<tr><td>P002</td><td>Torque limit source setting</td><td>0~2 (1)</td><td>Set the source of torque limitation for anticlockwise and clockwise rotation directions.
<table>
<tr><td></td><td>Counterclockwise CCW</td><td>Clockwise CW</td></tr>
<tr><td>0</td><td>CCWTL Analog signal</td><td>CWTL Analog signal</td></tr>
<tr><td>1</td><td>P119 Communication torque limit 1</td><td>P119 Communication torque limit 1</td></tr>
<tr><td>2</td><td>P119 Communication torque limit 1</td><td>P120 Communication torque limit 1</td></tr>
</table>
Read-Write Attributes: R/W Applicable Mode: P/S</td></tr>
</table>

<table>
<tr><td>P003</td><td>Stroke limit function setting</td><td>0~2
(1)</td><td>Set the specific action of the servo travel limit.<table><tr><td></td><td>Stroke Limit Action</td></tr><tr><td>0</td><td>Stroke limit function is effective and acts in accordance with P126 configuration</td></tr><tr><td>1</td><td>Invalid Stroke Limit Function</td></tr><tr><td>2</td><td>Alarm is triggered when the stroke limit is set</td></tr></table>Read-Write Attributes: R/W Applicable Mode: ALL</td></tr>
<tr><td>P004</td><td>P001=1 时 Command speed source</td><td>0~3
(0)</td><td>Source of instruction speed for analog speed mode.<table><tr><td></td><td>Instruction Speed Source</td></tr><tr><td>0</td><td>Analog Speed Signal</td></tr><tr><td>1</td><td>1~4 Internal speed</td></tr><tr><td>2</td><td>1~3 Internal speed/Analog Speed Signal</td></tr><tr><td>3</td><td>1~8 Internal speed</td></tr></table>Read-Write Attributes: R/W Applicable Mode: S</td></tr>
<tr><td>P005</td><td>Communication command selection</td><td>0~31
(0)</td><td>Communication position/speed/torque command selection.<table><tr><td>Parameter values</td><td>Position command</td><td>Speed instruction</td><td>Torque command</td></tr><tr><td>k</td><td>P290+k</td><td>P324+k</td><td>P358+k</td></tr></table>Communication location mode k∈[0,15],Communication speed/torque mode k∈[0,31]。
Read-Write Attributes: R/W Applicable Mode: ALL</td></tr>
<tr><td>P006</td><td>Zero speed clamp function setting</td><td>0~1
(0)</td><td>Set the zero speed clamp function.<table><tr><td>0</td><td>Zero speed clamp function is invalid</td></tr><tr><td>1</td><td>Zero speed clamp function is effective, servo action is affected by zero speed clamp input signal (set P122,P203 command zero speed clamp function is effective)</td></tr></table>Read-Write Attributes: R/W Applicable Mode: S/T</td></tr>
<tr><td>P007</td><td>Factory reserve</td><td>——</td><td>——</td></tr>
<tr><td>P008</td><td>Factory reserve</td><td>——</td><td>——</td></tr>
<tr><td>P009</td><td>Factory reserve</td><td>——</td><td>——</td></tr>
<tr><td>P010</td><td>RS485 Communication baud rate★</td><td>0~5
(2)</td><td>Baud rate of communication between servo and upper system via RS485.<table><tr><td></td><td>Baud rate</td><td></td><td>Baud rate</td></tr><tr><td>0</td><td>4800bps</td><td>3</td><td>38400bps</td></tr><tr><td>1</td><td>9600bps</td><td>4</td><td>57600bps</td></tr><tr><td>2</td><td>19200bps</td><td>5</td><td>115200bps</td></tr></table>Read-Write Attributes: R/W Applicable Mode: ALL</td></tr>
</table>

<table>
<tr><td>P011</td><td>CAN Communication baud rate★</td><td>1~7
(1)</td><td>Servo and upper system is communicated by CAN.<table><tr><td></td><td>Baud rate</td><td></td><td>Baud rate</td></tr><tr><td>1</td><td>1Mbps</td><td>5</td><td>125kbps</td></tr><tr><td>2</td><td>800kbps</td><td>6</td><td>50kbps</td></tr><tr><td>3</td><td>500kbps</td><td>7</td><td>20kbps</td></tr><tr><td>4</td><td>250kbps</td><td></td><td></td></tr></table>Read-Write Attributes: R/W Applicable Mode: ALL</td></tr>
<tr><td>P012</td><td>Factory reserve</td><td>——</td><td>——</td></tr>
<tr><td>P013</td><td>Factory reserve</td><td>——</td><td>——</td></tr>
<tr><td>P014</td><td>Factory reserve</td><td>——</td><td>——</td></tr>
<tr><td>P015</td><td>Factory reserve</td><td>——</td><td>——</td></tr>
<tr><td>P016</td><td>Servo power-up automatic enable configuration★</td><td>0~1
(0)</td><td>Configure servo power-up auto enable.<table><tr><td>0</td><td>Servo power on does not enable</td></tr><tr><td>1</td><td>Servo power-on auto enable</td></tr></table>Read-Write Attributes: R/W Applicable Mode: ALL</td></tr>
<tr><td>P017</td><td>Factory reserve</td><td>——</td><td>——</td></tr>
<tr><td>P018</td><td>Current loop proportional gain</td><td>0~1000</td><td>Current loop proportional gain. Set at the factory.
Read-Write Attributes: R/W Applicable Mode: ALL</td></tr>
<tr><td>P019</td><td>Current loop integration gain</td><td>0~500</td><td>Current loop proportional gain. Set at the factory.
Read-Write Attributes: R/W Applicable Mode: ALL</td></tr>
<tr><td>P020</td><td>1st position loop proportional gain</td><td>5~1000
(20)</td><td>The larger the value is, the faster the servo position responses and the more rigid it is. Too large will cause the system to vibrate, so a smaller value should be set.
Unit: 1/s
Read-Write Attributes: R/W Applicable Mode: P</td></tr>
<tr><td>P021</td><td>1st speed loop proportional gain</td><td>10~300
(50)</td><td>The larger the value is, the faster the servo speed responses. The heavier the load is, the larger the value needs to be set. Too large will cause the system to vibrate, so a smaller value should be set.
Unit: Hz
Read-Write Attributes: R/W Applicable Mode: P/S</td></tr>
<tr><td>P022</td><td>1st speed loop integral</td><td>10~300
(50)</td><td>The smaller the value is, the faster the servo speed responses. Too large value will cause the system to vibrate, so a larger value should be set.
Unit: ms
Read-Write Attributes: R/W Applicable Mode: P/S</td></tr>
<tr><td>P023</td><td>1st speed detection filter</td><td>0~5
(3)</td><td>The filter stops setting for feedback speed detection. The larger the block setting is, the stronger the filtering effect is, too large will affect the system response.
Read-Write Attributes : R/W Applicable Mode: ALL</td></tr>
</table>

P024	1st torque filtering time constant	0~2500 (3)	Used for filtering of command torque. Unit: ×10us Read-Write Attributes: R/W Applicable Mode: ALL
P025	Speed Feedforword gain	0~1200 (0)	Speed feedforward gain. The higher the value is, the faster the servo responses. Read-Write Attributes: R/W Applicable Mode: P
P026	Speed feedforward filter time constant	0~6400 (3)	Filter for speed feedforward. Unit: ms Read-Write Attributes: R/W Applicable Mode: P
P027	2nd position loop proportional gain	5~1000 (20)	The larger the value is, the faster the servo position responses and the more rigid it is. Too large will cause the system to vibrate, so a smaller value should be set. Unit: 1/s Read-Write Attributes: R/W Applicable Mode: P
P028	2nd speed loop proportional gain	10~300 (50)	The higher the value is, the faster the servo speed responses. The heavier the load is, the larger the value needs to be set. Too large will cause the system to vibrate, so a smaller value should be set. Unit: Hz Read-Write Attributes: R/W Applicable Mode: P/S
P029	2nd speed loop integral gain	10~300 (50)	The smaller the value is, the faster the servo speed responses. Too large value will cause the system to vibrate, so a larger value should be set. Unit: ms Read-Write Attributes: R/W Applicable Mode: P/S
P030	2nd speed detection filter	0~5 (3)	The filter gear setting for feedback speed detection. The larger the gear setting is, the stronger the filter effect, too large will affect the system response. Read-Write Attributes: R/W Applicable Mode: ALL
P031	2nd torque filter time constant	0~2500 (3)	Used for filter of command torque. Unit: ×10us Read-Write Attributes: R/W Applicable Mode: ALL
P032	Inertia ratio	0~10000 (100)	100 times the ratio of load inertia to motor rotor inertia. Unit: % Read-Write Attributes: R/W Applicable Mode: ALL
P033	Factory reserve	——	——
P034	Factory reserve	——	——
P035	Factory reserve	——	——
P036	Factory reserve	——	——
P037	Factory reserve	——	——
P038	Factory reserve	——	——

<table>
<tr><td>P039</td><td>Factory reserve</td><td>——</td><td>——</td></tr>
<tr><td>P040</td><td>Factory reserve</td><td>——</td><td>——</td></tr>
<tr><td>P041</td><td>Factory reserve</td><td>——</td><td>——</td></tr>
<tr><td>P042</td><td>Factory reserve</td><td>——</td><td>——</td></tr>
<tr><td>P043</td><td>Factory reserve</td><td>——</td><td>——</td></tr>
<tr><td>P044</td><td>Factory reserve</td><td>——</td><td>——</td></tr>
<tr><td>P045</td><td>Factory reserve</td><td>——</td><td>——</td></tr>
<tr><td>P046</td><td>Factory reserve</td><td>——</td><td>——</td></tr>
<tr><td>P047</td><td>Factory reserve</td><td>——</td><td>——</td></tr>
<tr><td>P048</td><td>Factory reserve</td><td>——</td><td>——</td></tr>
<tr><td>P049</td><td>Digital input DI filter time</td><td>ANY (3)</td><td>Filter level selection for digital input signals.
<table>
<tr><th>Parameter Value</th><th>Filter time</th><th>Parameter Value</th><th>Filter time</th></tr>
<tr><td>0</td><td>0.5ms</td><td>4</td><td>8ms</td></tr>
<tr><td>1</td><td>1ms</td><td>5</td><td>16ms</td></tr>
<tr><td>2</td><td>2ms</td><td>6</td><td>32ms</td></tr>
<tr><td>3</td><td>4ms</td><td>Other</td><td>32ms</td></tr>
</table>
Read-Write Attributes: R/W Applicable Mode: ALL</td></tr>
<tr><td>P050</td><td>Factory reserve</td><td>——</td><td>——</td></tr>
<tr><td>P051</td><td>Factory reserve</td><td>——</td><td>——</td></tr>
<tr><td>P052</td><td>Factory reserve</td><td>——</td><td>——</td></tr>
<tr><td>P053</td><td>Factory reserve</td><td>——</td><td>——</td></tr>
<tr><td>P054</td><td>Factory reserve</td><td>——</td><td>——</td></tr>
<tr><td>P055</td><td>Factory reserve</td><td>——</td><td>——</td></tr>
<tr><td>P056</td><td>Factory reserve</td><td>——</td><td>——</td></tr>
<tr><td>P057</td><td>Factory reserve</td><td>——</td><td>——</td></tr>
</table>

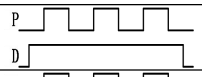
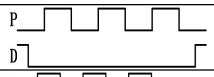
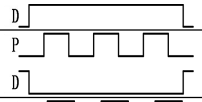
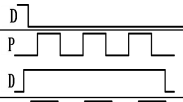
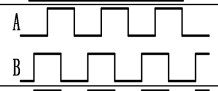
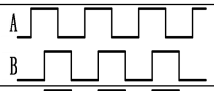
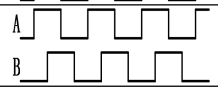
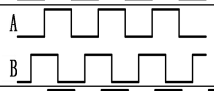
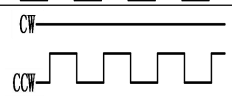
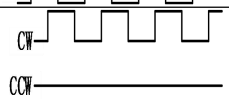
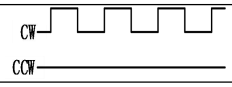
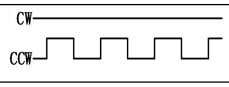
<table>
<tr>
<td>P058</td>
<td>Command pulse filter time★</td>
<td>0~13
(2)</td>
<td>Used for filter gear selection for command pulse input signal and cycle period of CANopen communication(ms).
<table>
<tr><th>Parameter Value</th><th>Filter time</th><th>Parameter Value</th><th>Filter time</th></tr>
<tr><td>0</td><td>222ns</td><td>7</td><td>3.555us</td></tr>
<tr><td>1</td><td>444ns</td><td>8</td><td>4.444us</td></tr>
<tr><td>2</td><td>666ns</td><td>9</td><td>5.333us</td></tr>
<tr><td>3</td><td>888ns</td><td>10</td><td>7.111us</td></tr>
<tr><td>4</td><td>1.333us</td><td>11</td><td>8.888us</td></tr>
<tr><td>5</td><td>1.777us</td><td>12</td><td>10.666us</td></tr>
<tr><td>6</td><td>2.666us</td><td>13</td><td>14.222us</td></tr>
</table>
Read-Write Attributes: R/W Applicable Mode: P</td>
</tr>
<tr>
<td>P059</td>
<td>Back to the original mode</td>
<td>0~30
(0)</td>
<td>Return to original mode selection.
<table>
<tr><td>0</td><td>Use the origin switch signal to trigger back to the origin</td><td>11</td><td>Use positive origin switch + motor Z signal + negative limit signal to trigger back to origin (use Z signal right to the right edge of positive origin switch)</td></tr>
<tr><td>1</td><td>Use negative origin switch + motor Z signal to trigger back to origin</td><td>12</td><td>Use positive origin switch + motor Z signal + negative limit signal to trigger back to origin (use Z signal left to the right edge of positive origin switch)</td></tr>
<tr><td>2</td><td>Use positive origin switch + motor Z signal to trigger back to origin</td><td>13</td><td>Use positive origin switch + motor Z signal + negative limit signal to trigger back to origin (use Z signal right to the left edge of positive origin switch)</td></tr>
</table>
</td>
</tr>
</table>

P059	Back to the original mode	0~30 (0)	3	Use the negative origin switch to trigger back to the origin position	14	Use positive origin switch + motor Z signal + negative limit signal to trigger back to origin (use Z signal left to the left edge of positive origin switch)
			4	Use the forward origin switch to trigger back to the origin position	15	Use negative origin switch+fixed length to back to origin.
			5	Use motor negative Z signal to trigger back to origin position	16	Use positive origin switch+fixed length to back to origin.
			6	Use motor forward Z signal to trigger back to origin position	17	Use positive origin switch + positive limit signal to trigger back to origin. (Origin is defined as left edge of positive origin switch.)
			7	Use positive origin switch + motor Z signal + positive limit signal to trigger back to origin (Adopt the Z phase signal to the left of the left edge of the positive origin switch)	18	Use positive origin switch + positive limit signal to trigger back to origin (Origin is defined as right edge of positive origin switch.)
			8	Use positive origin switch + motor Z signal + positive limit signal to trigger back to origin (Adopt the Z phase signal to the right of the left edge of the positive origin switch)	19	Use positive origin switch+negative limit signal to trigger back to origin. (Origin is defined as right edge of positive origin switch)

<table>
<tr><td></td><td></td><td></td><td>
<table>
<tr><td>9</td><td>Use positive origin switch + motor Z signal + positive limit signal to trigger back to origin (Adopt the Z phase signal to the left of the right edge of the positive origin switch)</td><td>20</td><td>Use positive origin switch+negative limit signal to trigger back to origin. (Origin is defined as left edge of positive origin switch.)</td></tr>
<tr><td>10</td><td>Use positive origin switch + motor Z signal + positive limit signal to trigger back to origin (Adopt the Z phase signal to the right of the right edge of the positive origin switch)</td><td></td><td></td></tr>
</table>
Read-Write Attributes: R/W Applicable Mode: ALL</td></tr>
<tr><td>P060~
P069</td><td>Factory reserve</td><td>———</td><td>———</td></tr>
<tr><td>P070</td><td>JOG test running speed</td><td>0~3000
(300)</td><td>Set motor rotation in JOG test running mode.
Read-Write Attributes: R/W Applicable Mode: ALL</td></tr>
<tr><td>P071</td><td>Communication position selection</td><td>0~3
(0)</td><td>Choose absolute position or relative position mode.
<table>
<tr><td></td><td>0</td><td>1</td></tr>
<tr><td>bit0</td><td>Absolute position</td><td>Relative position</td></tr>
<tr><td>bit1</td><td>Allow deviation counter cleared and command of back to origin position modification.</td><td>Do not allow deviation counter cleared and command of back to origin position modification.</td></tr>
</table>
Read-Write Attributes: R/W Applicable Mode: P</td></tr>
</table>

<table>
<tr>
<td>P072</td>
<td>External input logic level selection</td>
<td>ANY
(0)</td>
<td>External input logic level selection.
<table>
<tr><th></th><th>0</th><th>1</th></tr>
<tr><td>bit0</td><td>Communication enable and DI enable cannot be valid at the same time.</td><td>DI enable is needed in every mode, communication enable is also needed in communication mode.</td></tr>
<tr><td>bit2</td><td>CCWL signal valid in low level</td><td>CCWL signal valid in high level</td></tr>
<tr><td>bit3</td><td>CWL signal valid in low level</td><td>CWL signal valid in high level</td></tr>
<tr><td>bit5</td><td>ZEROSPD signal valid in low level</td><td>ZEROSPD signal valid in high level</td></tr>
<tr><td>bit11</td><td>ORG signal valid in low level</td><td>ORG signal valid in high level</td></tr>
</table>
Read-Write Attributes: R/W Applicable Mode: ALL</td>
</tr>
<tr>
<td>P073</td>
<td>Control order source selection</td>
<td>0~4
(0)</td>
<td>Select control order source.
<table>
<tr><td>0</td><td>Communication mode is from P281, pulse/analog mode is from DI</td></tr>
<tr><td>1</td><td>From default DI</td></tr>
<tr><td>2</td><td>From configurable DI</td></tr>
<tr><td>3</td><td>From P281, limit signal is from DI</td></tr>
<tr><td>4</td><td>From P281 or configurable DI</td></tr>
</table>
Read-Write Attributes: R/W Applicable Mode: ALL</td>
</tr>
</table>

<table>
<tr><td>P074</td><td>DI reuse function configuration5</td><td>ANY (0)</td><td rowspan="5">Digital input pin function configuration
<table>
<tr><th>Param</th><th>bit</th><th>Signal</th><th>Pin No.</th></tr>
<tr><td rowspan="2">P074</td><td>bit8~bit15</td><td>——</td><td>——</td></tr>
<tr><td>bit0~bit7</td><td>——</td><td>——</td></tr>
<tr><td rowspan="2">P075</td><td>bit8~bit15</td><td>——</td><td>——</td></tr>
<tr><td>bit0~bit7</td><td>——</td><td>——</td></tr>
<tr><td rowspan="2">P076</td><td>bit8~bit15</td><td>DI6</td><td></td></tr>
<tr><td>bit0~bit7</td><td>DI1</td><td></td></tr>
<tr><td rowspan="2">P077</td><td>bit8~bit15</td><td>DI2</td><td></td></tr>
<tr><td>bit0~bit7</td><td>DI5</td><td></td></tr>
<tr><td rowspan="2">P078</td><td>bit8~bit15</td><td>DI4</td><td></td></tr>
<tr><td>bit0~bit7</td><td>DI3</td><td></td></tr>
</table>
Function configuration:
<table>
<tr><th>Function code</th><th>Function</th></tr>
<tr><td>0x00</td><td>From default DI</td></tr>
<tr><td>0x01</td><td>Alarm clear</td></tr>
<tr><td>0x02</td><td>Anticlockwise travel limit</td></tr>
<tr><td>0x03</td><td>Clockwise travel limit</td></tr>
<tr><td>0x04</td><td>Mode switch</td></tr>
<tr><td>0x05</td><td>Zero speed clamp</td></tr>
<tr><td>0x06</td><td>Command frequency division; multi segment position/speed/torque command start</td></tr>
<tr><td>0x07</td><td>Back to origin command</td></tr>
<tr><td>0x08</td><td>Command pulse forbidden; Internal speed command selection 4</td></tr>
<tr><td>0x09</td><td>Gain selection</td></tr>
<tr><td>0x0A</td><td>Deviation counter clear</td></tr>
<tr><td>0x0B</td><td>Origin switch signal</td></tr>
<tr><td>0x0C</td><td>Internal speed command selection 1</td></tr>
<tr><td>0x0D</td><td>Internal speed command selection 2</td></tr>
<tr><td>0x0E</td><td>Internal speed command selection 3</td></tr>
<tr><td>0x0F</td><td>Torque limit selection</td></tr>
<tr><td>0x10</td><td>Cancel relative position command</td></tr>
</table>
Read-Write Attributes: R/W Applicable Mode: ALL</td></tr>
<tr><td>P075</td><td>DI reuse function configuration4</td><td>ANY (0)</td></tr>
<tr><td>P076</td><td>DI reuse function configuration3</td><td>ANY (0)</td></tr>
<tr><td>P077</td><td>DI reuse function configuration2</td><td>ANY (0)</td></tr>
<tr><td>P078</td><td>DI reuse function configuration1</td><td>ANY (0)</td></tr>
<tr><td>P079</td><td>Factory reserve</td><td>——</td><td>——</td></tr>
</table>

<table>
<tr><td>P080</td><td>Command pulse direction configuration ★</td><td>0~1
(0)</td><td rowspan="2">Set direction of command pulse and pulse form.
<table><tr><th>P081</th><th>P080</th><th>Anticlockwise</th><th>Clockwise</th></tr><tr><td rowspan="2">3</td><td>0</td><td>  </td><td>  </td></tr><tr><td>1</td><td>  </td><td>  </td></tr><tr><td rowspan="2">2</td><td>0</td><td>  </td><td>  </td></tr><tr><td>1</td><td>  </td><td>  </td></tr><tr><td rowspan="2">1</td><td>0</td><td>  </td><td>  </td></tr><tr><td>1</td><td>  </td><td>  </td></tr></table>Read-Write Attributes: R/W Applicable Mode: P</td></tr>
<tr><td>P081</td><td>Command pulse input mode configuration ★</td><td>1~3
(3)</td></tr>
<tr><td>P082</td><td>Command pulse input forbidden function null configuration</td><td>0~1
(1)</td><td>When the parameter is 1, command pulse forbidden function is blocked; when the parameter is 0, it depends on INH input.
<table><tr><th>Param</th><th>INH</th><th>Command pulse</th></tr><tr><td rowspan="2">0</td><td>valid</td><td>Input allowed</td></tr><tr><td>null</td><td>Input forbidden</td></tr><tr><td rowspan="2">1</td><td>——</td><td>Input allowed</td></tr><tr><td>——</td><td>Input allowed</td></tr></table>Read-Write Attributes: R/W Applicable Mode: P</td></tr>
<tr><td>P083</td><td>Factory reserve</td><td>——</td><td>——</td></tr>
<tr><td>P084</td><td>Factory reserve</td><td>——</td><td>——</td></tr>
<tr><td>P085</td><td>Factory reserve</td><td>——</td><td>——</td></tr>
<tr><td>P086</td><td>Command pulse electronic gear 1st numerator</td><td>0~32767
(1)</td><td rowspan="3">Refer to the formula bellow to execute electronic gear zoom to command pulse.
Number of pulses needed per rotation×(P086 or P087)/P088=encoder resolution, E.g; For 2500 line incremental encoder, the resolution is 10000, when set P086=4, P088=1, command pulses needed for 1r is: 2500.
Read-Write Attributes: R/W Applicable Mode: P</td></tr>
<tr><td>P087</td><td>Command pulse electronic gear 2nd numerator</td><td>0~32767
(1)</td></tr>
<tr><td>P088</td><td>Command pulse electronic gear denominator</td><td>1~32767
(1)</td></tr>
<tr><td>P089</td><td>Command pulse smoothing filter</td><td>0~7
(1)</td><td>Command pulse delay filter gear selection.
Read-Write Attributes: R/W Applicable Mode: P</td></tr>
</table>

<table>
<tr><td>P090</td><td>Set positive direction of motor rotation in communication control mode★</td><td>0~1
(0)</td><td>Set positive direction of motor rotation.<table><tr><td>0</td><td>Facing motor shaft, anticlockwise is positive.</td></tr><tr><td>1</td><td>Facing motor shaft, clockwise is positive.</td></tr></table>Read-Write Attributes: R/W Applicable Mode: ALL</td></tr>
<tr><td>P091</td><td>Input mode of deviation counter clear</td><td>0~2
(1)</td><td>Function configuration of deviation counter clear signal<table><tr><td>0</td><td>Valid in high level</td></tr><tr><td>1</td><td>Valid in rising edge</td></tr><tr><td>2</td><td>Block deviation counter clear signal</td></tr></table>Read-Write Attributes: R/W Applicable Mode: P</td></tr>
<tr><td>P092</td><td>Analog voltage-command speed coefficient</td><td>10~20000
(500)</td><td>rpm speed corresponding to 1V voltage
Unit: rpm/V
Read-Write Attributes: R/W Applicable Mode: S</td></tr>
<tr><td>P093</td><td>Analog voltage-command speed direction</td><td>0~1
(0)</td><td>Set rotation direction corresponding to positive/negative voltage.<table><tr><td>0</td><td>Positive->Anticlockwise</td></tr><tr><td>1</td><td>Positive->Clockwise</td></tr></table>Read-Write Attributes: R/W Applicable Mode: S</td></tr>
<tr><td>P094</td><td>Analog input zero drift compensation</td><td>-2047~2047
(0)</td><td>Zero drift compensation of analog speed command or analog torque command.
Read-Write Attributes: R/W Applicable Mode: S/T</td></tr>
<tr><td>P095</td><td>1st internal speed</td><td>-3000~3000
(0)</td><td>1st internal speed Unit: rpm
Read-Write Attributes: R/W Applicable Mode: S</td></tr>
<tr><td>P096</td><td>2nd internal speed</td><td>-3000~3000
(0)</td><td>2nd internal speed Unit: rpm
Read-Write Attributes: R/W Applicable Mode: S</td></tr>
<tr><td>P097</td><td>3rd internal speed</td><td>-3000~3000
(500)</td><td>3rd internal speed Unit: rpm
Max speed limit in communication position mode
Read-Write Attributes: R/W Applicable Mode: P/S</td></tr>
<tr><td>P098</td><td>4th internal speed</td><td>-3000~3000
(500)</td><td>4th internal speed Unit: rpm
Max speed limit in torque mode
Read-Write Attributes: R/W Applicable Mode: S/T</td></tr>
<tr><td>P099</td><td>5th internal speed</td><td>-3000~3000
(0)</td><td>5th internal speed Unit: rpm
Read-Write Attributes: R/W Applicable Mode: S</td></tr>
<tr><td>P100</td><td>6th internal speed</td><td>-3000~3000
(0)</td><td>6th internal speed Unit: rpm
Read-Write Attributes: R/W Applicable Mode: S</td></tr>
<tr><td>P101</td><td>7th internal speed</td><td>-3000~3000
(0)</td><td>7th internal speed Unit: rpm
Used as return speed during back to origin.
Read-Write Attributes: R/W Applicable Mode: ALL</td></tr>
<tr><td>P102</td><td>8th internal speed</td><td>-3000~3000
(100)</td><td>8th internal speed Unit: rpm
Used as crawling speed during back to origin.
Read-Write Attributes: R/W Applicable Mode: ALL</td></tr>
<tr><td>P103
~
P111</td><td>Factory reserve</td><td>——</td><td>——</td></tr>
</table>

<table>
<tr><td>P112</td><td>Analog command filter time</td><td>0~6400
(1000)</td><td>Analog speed/torque command delay filter time
Unit×20us
Read-Write Attributes: R/W Applicable Mode: S/T</td></tr>
<tr><td>P113</td><td>Motor acceleration time</td><td>0~10000
(120)</td><td rowspan="2">Acceleration time and deceleration time of servo motor.
Unit: ms/1000rpm
P113 : Time from x(rpm) to (x+1000)rpm
P114 : Time from x(rpm) to (x-1000)rpm
Read-Write Attributes: R/W Applicable Mode: ALL</td></tr>
<tr><td>P114</td><td>Motor deceleration time</td><td>0~10000
(120)</td></tr>
<tr><td>P115</td><td>Command pulse max speed limit</td><td>0~1
(0)</td><td>Source of max speed limit in pulse position mode.
<table><tr><td>0</td><td>Pulse frequency defines rotation speed</td></tr><tr><td>1</td><td>P98 is the max limit speed</td></tr></table>Read-Write Attributes: R/W Applicable Mode: P</td></tr>
<tr><td>P116</td><td>Factory reserve</td><td>——</td><td>——</td></tr>
<tr><td>P117</td><td>Factory reserve</td><td>——</td><td>——</td></tr>
<tr><td>P118</td><td>Factory reserve</td><td>——</td><td>——</td></tr>
<tr><td>P119</td><td>Communication torque limit1</td><td>0~3000
(2000)</td><td>The max torque that servo motor outputs.1000 indicates 1000‰, i.e.the motor can output ±1 rated torque.The rated torque of motor is introduced in motor specification.
Unit: ‰
Read-Write Attributes: R/W Applicable Mode: ALL</td></tr>
<tr><td>P120</td><td>Communication torque limit2</td><td>0~3000
(2000)</td><td></td></tr>
<tr><td>P121</td><td>Set locating completed range</td><td>0~32767
(5)</td><td>When the absolute value of difference between feedback position of motor encoder and command position is smaller than this parameter, locating completed output signal is valid. Unit: pulses
Read-Write Attributes: R/W Applicable Mode: P</td></tr>
<tr><td>P122</td><td>Set zero speed detection range</td><td>10~20000
(10)</td><td>When the absolute value of motor feedback speed is smaller than this parameter, the zero speed output signal is valid. Unit: rpm
Read-Write Attributes: R/W Applicable Mode: ALL</td></tr>
<tr><td>P123</td><td>Arriving speed</td><td>10~20000
(10)</td><td>When the absolute value of motor feedback speed equals or is larger than this parameter, arriving speed output signal is valid. Unit: rpm
Read-Write Attributes: R/W Applicable Mode: ALL</td></tr>
</table>

<table>
<tr><td>P124</td><td>Condition of locating completed output signal</td><td>0~2
(0)</td><td>Condition of locating completed output signal
<table><tr><td>0</td><td>When position deviation is within locating completed range, output is valid.</td></tr><tr><td>1</td><td>When there is no position command and position deviation is within locating completed range, output is valid.</td></tr><tr><td>2</td><td>When there is no position command, zero speed detection signal is valid and position deviation is within locating completed range, output is valid.</td></tr></table>Read-Write Attributes: R/W Applicable Mode: P</td></tr>
<tr><td>P125</td><td>Factory reserve</td><td>———</td><td>———</td></tr>
<tr><td>P126</td><td>Specific action of travel limit★</td><td>0~1
(1)</td><td>Specific action of the driver and motor after the travel limit signal is valid.
<table><tr><td></td><td colspan="2">During deceleration</td><td>After stop rotating</td><td>Deviation counter</td></tr><tr><td>0</td><td colspan="3">Torque command from direction of travel limit is 0</td><td>Hold</td></tr><tr><td rowspan="3">1</td><td colspan="3">Control mode</td><td></td></tr><tr><td>P</td><td colspan="2">Position command from direction of travel limit is 0</td><td>Clear before deceleration</td></tr><tr><td>S/T</td><td>Position command is 0</td><td>Speed command from direction of travel limit is 0</td><td></td></tr></table>Read-Write Attributes: R/W Applicable Mode: ALL</td></tr>
<tr><td>P127</td><td>Factory reserve</td><td>———</td><td>———</td></tr>
<tr><td>P128</td><td>Define fixed length position during back to origin</td><td>-2^{31}~$2^{31}-1$</td><td>When back to origin mode P059 is set as 15/16, it needs to modify this parameter.
Unit: pulses
Read-Write Attributes: R/W Applicable Mode: ALL</td></tr>
<tr><td>P130</td><td>Factory reserve</td><td>———</td><td>———</td></tr>
<tr><td>P131</td><td>Factory reserve</td><td>———</td><td>———</td></tr>
</table>

<table>
<tr><td>P132</td><td>Braking setting★</td><td>0~3
(0)</td><td>Set braking strategy of servo system.<table><tr><td>0</td><td>Use internal braking resistance and alarm when the baking rate is too high.</td></tr><tr><td>1</td><td>Use external braking resistance and alarm when the baking rate is too high</td></tr><tr><td>2</td><td>Use external braking resistance and do not alarm when the baking rate is too high</td></tr><tr><td>3</td><td>Use internal capacity to store braking energy.</td></tr></table>Read-Write Attributes: R/W Applicable Mode: ALL</td></tr>
<tr><td>P133</td><td>Discharge voltage setting★</td><td>——
(0)</td><td>Use internal default when set as 0. Refer to section 4.5 for the max and min values . (Table of under-voltage node, discharge voltage node and over-voltage node under different voltage level in servo system)
Unit: V
Read-Write Attributes: R/W Applicable Mode: ALL</td></tr>
<tr><td>P134</td><td>Under-voltage setting★</td><td>——
(0)</td><td>Use internal default when set as 0. Refer to section4.5 for the max and min values . (Table of under-voltage node, discharge voltage node and over-voltage node under different voltage level in servo system)
Unit: V
Read-Write Attributes: R/W Applicable Mode: ALL</td></tr>
<tr><td>P135</td><td>Over-voltage setting★</td><td>——
(0)</td><td>Use internal default when set as 0. Refer to section4.5 for the max and min values . (Table of under-voltage node, discharge voltage node and over-voltage node under different voltage level in servo system)
Unit: V
Read-Write Attributes: R/W Applicable Mode: ALL</td></tr>
<tr><td>P136</td><td>Position deviation limit value</td><td>0~32767
(25000)</td><td>Set detection threshold of alarm for large position deviation. When set as 0, position deviation alarm is blocked. Unit: ×256pulses
Read-Write Attributes: R/W Applicable Mode: P</td></tr>
<tr><td>P137</td><td>Analog command limit value</td><td>0~100
(0)</td><td>Set limit value of analog voltage and alarm will be triggered if actual value is larger than it. When set as 0, the alarm is blocked. Unit: ×0.1V
Read-Write Attributes: R/W Applicable Mode: S/T</td></tr>
<tr><td>P138</td><td>Over-load level</td><td>0~2000
(1050)</td><td>Set starting torque before the servo calculates overload. Unit: rated torque‰
Read-Write Attributes: R/W Applicable Mode: ALL</td></tr>
<tr><td>P139</td><td>Over-speed level</td><td>0~20000
(0)</td><td>Set over-speed threshold of motor. When set as 0, the over-speed threshold is 1.2 times of rated revolution.
Unit: rpm
Read-Write Attributes: R/W Applicable Mode: ALL</td></tr>
<tr><td>P140
~
P149</td><td>Alarm history</td><td>ANY
(0)</td><td>Record the latest 10 alarms. Code shown in P202。
Read-Write Attributes: R Applicable Mode: ALL</td></tr>
<tr><td>P150
~
P179</td><td>Factory reserve</td><td>——</td><td>——</td></tr>
</table>

<table>
<tr><td>P180</td><td>Software version</td><td>ANY</td><td>Software version of the servo
Read-Write Attributes: R Applicable Mode: ALL</td></tr>
<tr><td>P181</td><td>Motor model</td><td>ANY</td><td>Type of motor. E.g: motor model 1006, 10 indicates that it fits 10A driver; 06 represents P182 motor code. More details are in section4.5.
Read-Write Attributes: R Applicable Mode: ALL</td></tr>
<tr><td>P182</td><td>Motor code</td><td>1-100</td><td>Code of servo motor. Details about configuration of motor code are in section 4.5.
Read-Write Attributes: R/W Applicable Mode: ALL</td></tr>
<tr><td>P183~ P199</td><td>Factory reserve</td><td>——</td><td>——</td></tr>
<tr><td>P200</td><td>Servo system state machine</td><td>ANY</td><td>Servo system state machine of gate
<table>
<tr><td>Param</td><td>State machine</td></tr>
<tr><td>1</td><td>Servo initial state</td></tr>
<tr><td>3</td><td>Servo running</td></tr>
<tr><td>4</td><td>Servo prepared</td></tr>
<tr><td>5</td><td>Servo alarm（for specific refer to P202）</td></tr>
</table>
Read-Write Attributes: R Applicable Mode: ALL</td></tr>
<tr><td>P201</td><td>Servo control mode</td><td>ANY</td><td>Current control mode of servo
<table>
<tr><td>Param</td><td>Current mode</td></tr>
<tr><td>0x0000</td><td>No control</td></tr>
<tr><td>0x0001</td><td>Pulse position</td></tr>
<tr><td>0x0002</td><td>Analog speed mode</td></tr>
<tr><td>0x0004</td><td>Analog torque mode</td></tr>
<tr><td>0x0101</td><td>Communication position</td></tr>
<tr><td>0x0102</td><td>Communication speed</td></tr>
<tr><td>0x0104</td><td>Communication torque</td></tr>
</table>
Read-Write Attributes: R Applicable Mode: ALL</td></tr>
<tr><td>P202</td><td>Servo alarm state</td><td>ANY</td><td>Current alarm state of servo
<table>
<tr><td>Param</td><td>Servo alarm state</td></tr>
<tr><td>0</td><td>None</td></tr>
<tr><td>1</td><td>Under voltage</td></tr>
<tr><td>2</td><td>Over voltage</td></tr>
<tr><td>3</td><td>Over current</td></tr>
<tr><td>4</td><td>Over heat</td></tr>
<tr><td>6</td><td>Encoder alarm</td></tr>
<tr><td>8</td><td>Overload</td></tr>
<tr><td>9</td><td>Excessive position deviation</td></tr>
<tr><td>10</td><td>Travel limit</td></tr>
<tr><td>11</td><td>Over speed</td></tr>
<tr><td>12</td><td>Analog over-limit</td></tr>
<tr><td>13</td><td>EEPROM faulty in read and write</td></tr>
<tr><td>15</td><td>CAN communication faulty</td></tr>
</table>
Read-Write Attributes: R Applicable Mode: ALL</td></tr>
</table>

<table>
<tr>
<td>P203</td>
<td>External order status</td>
<td>ANY</td>
<td>Status of external control signal
<table>
<tr><th></th><th>Control signal</th><th>Status bit is 0</th><th>Status bit is 1</th></tr>
<tr><td>bit0</td><td>Servo enable</td><td>null</td><td>null</td></tr>
<tr><td>bit1</td><td>Alarm clear</td><td>null</td><td>valid</td></tr>
<tr><td>bit2</td><td>Anticlockwise travel limit</td><td>null</td><td>valid</td></tr>
<tr><td>bit3</td><td>Clockwise travel limit</td><td>null</td><td>valid</td></tr>
<tr><td>bit4</td><td>Mode switch</td><td>Mode 1</td><td>Mode 2</td></tr>
<tr><td>bit5</td><td>Zero speed clamp</td><td>null</td><td>valid</td></tr>
<tr><td>bit6</td><td>Command frequency division selection; multi segment position/speed/ torque start signal</td><td>1st frequency division; signal null</td><td>2nd frequency division; signal valid</td></tr>
<tr><td>bit7</td><td>Back to origin</td><td>null</td><td>valid</td></tr>
<tr><td>bit8</td><td>Command pulse forbidden; internal command selection 4</td><td>null</td><td>valid</td></tr>
<tr><td>bit9</td><td>Gain selection</td><td>1st gain</td><td>2nd gain</td></tr>
<tr><td>bit10</td><td>Deviation counter clear; speed direction selection</td><td>null</td><td>valid</td></tr>
<tr><td>bit11</td><td>Origin switch signal</td><td>null</td><td>valid</td></tr>
<tr><td>bit12</td><td>Internal command selection 1</td><td>null</td><td>valid</td></tr>
<tr><td>bit13</td><td>Internal command selection 2</td><td>null</td><td>valid</td></tr>
<tr><td>bit14</td><td>Internal command selection 3</td><td>null</td><td>valid</td></tr>
<tr><td>bit15</td><td>Torque limit selection</td><td>Torque limit 1</td><td>Torque limit 2</td></tr>
</table>
Read-Write Attributes: R Applicable Mode: ALL</td>
</tr>
</table>

<table>
<tr><td>P204</td><td>Servo output status</td><td>ANY</td><td>Servo output status. When the relative bit is 1, the status is true.
<table>
<tr><th></th><th>Servo output status</th><th>Status bit is 0</th><th>Status bit is 1</th></tr>
<tr><td>bit0</td><td>Servo prepared</td><td>False</td><td>True</td></tr>
<tr><td>bit1</td><td>Servo alarm</td><td>False</td><td>True</td></tr>
<tr><td>bit2</td><td>Locating completed</td><td>False</td><td>True</td></tr>
<tr><td>bit3</td><td>Braking release</td><td>False</td><td>True</td></tr>
<tr><td>bit4</td><td>Zero speed</td><td>False</td><td>True</td></tr>
<tr><td>bit5</td><td>In torque limit</td><td>False</td><td>True</td></tr>
<tr><td>bit6</td><td>Speed consistency</td><td>False</td><td>True</td></tr>
<tr><td>bit7</td><td>Resistance breaking</td><td>False</td><td>True</td></tr>
<tr><td>bit8</td><td>Speed arriving</td><td>False</td><td>True</td></tr>
<tr><td>bit9</td><td>Overload alarm</td><td>False</td><td>True</td></tr>
<tr><td>bit10</td><td>Back to origin</td><td>False</td><td>True</td></tr>
</table>
Read-Write Attributes: R Applicable Mode: ALL</td></tr>
<tr><td>P205</td><td>Digital input DI state</td><td>ANY</td><td>State of servo digital DI input signal. When relative pins are on, corresponding bit is 1.
<table>
<tr><td>bit0</td><td>SRV_ON</td></tr>
<tr><td>bit1</td><td>ALM_CLR</td></tr>
<tr><td>bit2</td><td>DI1</td></tr>
<tr><td>bit3</td><td>DI2</td></tr>
<tr><td>bit4</td><td>DI3</td></tr>
<tr><td>bit5</td><td>DI4</td></tr>
<tr><td>bit6</td><td>DI5</td></tr>
</table>
Read-Write Attributes: R Applicable Mode: ALL</td></tr>
<tr><td>P206</td><td>Digital output DO state</td><td>ANY</td><td>State of servo digital DO output signal. When corresponding bit is 1, relative pins are on.
<table>
<tr><td>bit0</td><td>DO1</td></tr>
<tr><td>bit1</td><td>DO2</td></tr>
<tr><td>bit2</td><td>DO3</td></tr>
<tr><td>bit3</td><td>DO4</td></tr>
<tr><td>bit4</td><td>DO5</td></tr>
<tr><td>bit5</td><td>DO6</td></tr>
</table>
Read-Write Attributes: R Applicable Mode: ALL</td></tr>
<tr><td>P207</td><td>Analog input 1</td><td>ANY</td><td>Command speed as analog speed mode, or command torque as analog torque mode.
±10V corresponds ±32767
Read-Write Attributes: R Applicable Mode: S/T</td></tr>
<tr><td>P208</td><td>Analog input 2</td><td>ANY</td><td>Torque limit as analog mode
±10V corresponds ±32767
Read-Write Attributes: R Applicable Mode: ALL</td></tr>
<tr><td>P209</td><td>Factory reserve</td><td>——</td><td>——</td></tr>
<tr><td>P210</td><td>Analog output 1</td><td>ANY</td><td>Analog output 1
Read-Write Attributes: R Applicable Mode: ALL</td></tr>
</table>

<table>
<tr><td>P211</td><td>Analog output 2</td><td>ANY</td><td>Analog output 2
Read-Write Attributes: R Applicable Mode: ALL</td></tr>
<tr><td>P212</td><td>Command position</td><td>-2^{31}~$2^{31}-1$</td><td>Command position Unit: pulses
Read-Write Attributes: R Applicable Mode: P</td></tr>
<tr><td>P216</td><td>User position</td><td>-2^{31}~$2^{31}-1$</td><td>User position coordinate Unit: pulses
Read-Write Attributes: R Applicable Mode: P</td></tr>
<tr><td>P218</td><td>Position deviation</td><td>-2^{31}~$2^{31}-1$</td><td>Position deviation Unit: pulses
Read-Write Attributes: R Applicable Mode: P</td></tr>
<tr><td>P220</td><td>Command speed</td><td>-6000~6000</td><td>Command speed Unit: rpm
Read-Write Attributes: R Applicable Mode: S</td></tr>
<tr><td>P221</td><td>Feedback speed</td><td>-6000~6000</td><td>Feedback speed Unit: rpm
Read-Write Attributes: R Applicable Mode: ALL</td></tr>
<tr><td>P222</td><td>Speed deviation</td><td>-6000~6000</td><td>Speed deviation Unit: rpm
Read-Write Attributes: R Applicable Mode: S</td></tr>
<tr><td>P223</td><td>Command torque</td><td>-3500~3500</td><td>Command torque Unit: ‰
Read-Write Attributes: R Applicable Mode: T</td></tr>
<tr><td>P224</td><td>Feedback torque</td><td>-3500~3500</td><td>Feedback torque Unit: ‰
Read-Write Attributes: R Applicable Mode: ALL</td></tr>
<tr><td>P225</td><td>Torque deviation</td><td>-3500~3500</td><td>Torque deviation Unit: ‰
Read-Write Attributes: R Applicable Mode: T</td></tr>
<tr><td>P226</td><td>Busbar voltage</td><td>ANY</td><td>Busbar voltage Unit: V
Read-Write Attributes: R Applicable Mode: ALL</td></tr>
<tr><td>P227</td><td>Inner temp of driver</td><td>ANY</td><td>Temperature of the driver Unit: ℃
Read-Write Attributes: R Applicable Mode: ALL</td></tr>
<tr><td>P228 ~ P234</td><td>Factory reserve</td><td>———</td><td>———</td></tr>
<tr><td>P235</td><td>Reason for non-rotation of motor</td><td>ANY</td><td>Reason for non-rotation of motor
<table>
<tr><th></th><th>Reason</th><th></th><th>Reason</th></tr>
<tr><td>0</td><td>———</td><td>10</td><td>Speed command is too small</td></tr>
<tr><td>1</td><td>Main power is off</td><td>12</td><td>Torque command is too small</td></tr>
<tr><td>2</td><td>no servo enable</td><td>13</td><td>Speed limit is too small</td></tr>
<tr><td>3</td><td>Travel limit</td><td>14</td><td>Overload/wrong connection of power line</td></tr>
<tr><td>4</td><td>Torque limit is too small</td><td>15</td><td>Servo alarm</td></tr>
<tr><td>7</td><td>Position command is too small</td><td>17</td><td>Motor power line is disconnected</td></tr>
<tr><td>9</td><td>Zero speed clamp</td><td></td><td></td></tr>
</table>
Read-Write Attributes: R Applicable Mode: ALL</td></tr>
<tr><td>P236 ~ P279</td><td>Factory reserve</td><td>———</td><td>———</td></tr>
</table>

<table>
<tr>
<td>P280</td>
<td>Communication function code</td>
<td>ANY</td>
<td>Communication function code, to execute following control:
<table>
<tr><th>Function code</th><th>Action</th><th>Function code</th><th>Action</th></tr>
<tr><td>0x0101</td><td>Reset factory settings</td><td>0x1001</td><td>Position sinusoidal response</td></tr>
<tr><td>0x0102</td><td>All param write EEPROM</td><td>0x1002</td><td>Speed sinusoidal response</td></tr>
<tr><td>0x0104</td><td>Updated param write EEPROM</td><td>0x1004</td><td>Torque sinusoidal response</td></tr>
<tr><td>0x0202</td><td>JOG start</td><td>0x2001</td><td>Position step response</td></tr>
<tr><td>0x0203</td><td>JOG motor rotates anticlockwise</td><td>0x2002</td><td>Speed step response</td></tr>
<tr><td>0x0204</td><td>JOG motor rotates clockwise</td><td>0x2004</td><td>Torque step response</td></tr>
<tr><td>0x0205</td><td>JOG stop</td><td></td><td></td></tr>
</table>
Read-Write Attributes: R/W Applicable Mode: ALL</td>
</tr>
</table>

<table>
<tr>
<td>P281</td>
<td>Communication control word 1</td>
<td>ANY</td>
<td>Communication control word 1. Operate by bit to execute following control:
<table>
<tr><td></td><td>Aim function</td><td>0</td><td>1</td></tr>
<tr><td>bit0</td><td>P016=1 anti-enable</td><td>null</td><td>valid</td></tr>
<tr><td>bit4</td><td>Mode switch</td><td>Mode 1</td><td>Mode 2</td></tr>
<tr><td>bit5</td><td>Zero speed clamp/position locked</td><td>null</td><td>valid</td></tr>
<tr><td rowspan="2">bit6</td><td>Command frequency division selection</td><td>P086</td><td>P087</td></tr>
<tr><td>Multi segment position/speed/torque start</td><td>null</td><td>valid</td></tr>
<tr><td>bit7</td><td>Back to origin order</td><td>null</td><td>valid</td></tr>
<tr><td rowspan="2">bit8</td><td>Command pulse forbidden</td><td>null</td><td>valid</td></tr>
<tr><td>Internal command selection 4</td><td>null</td><td>valid</td></tr>
<tr><td>bit9</td><td>Gain selection</td><td>1st gain</td><td>2nd gain</td></tr>
<tr><td rowspan="2">bit10</td><td>Deviation counter clear</td><td>null</td><td>valid</td></tr>
<tr><td>Speed direction selection</td><td></td><td></td></tr>
<tr><td>bit11</td><td>Origin switch signal</td><td>null</td><td>valid</td></tr>
<tr><td>bit12</td><td>Internal command selection 1</td><td>null</td><td>valid</td></tr>
<tr><td>bit13</td><td>Internal command selection 2</td><td>null</td><td>valid</td></tr>
<tr><td>bit14</td><td>Internal command selection 3</td><td>null</td><td>valid</td></tr>
<tr><td>bit15</td><td>Torque limit selection</td><td>Limit 1</td><td>Limit 2</td></tr>
</table>
Read-Write Attributes: R/W Applicable Mode: ALL</td>
</tr>
<tr>
<td>P282</td>
<td>Communication control word 2</td>
<td>ANY</td>
<td>Communication control word 2. Operate by bit to execute following control:
<table>
<tr><td></td><td>0</td><td>1</td></tr>
<tr><td>bit0</td><td>Servo anti-enable</td><td>Servo enable</td></tr>
<tr><td>bit1</td><td>(auto reset)</td><td>Alarm clear</td></tr>
<tr><td>bit2</td><td>(auto reset)</td><td>Relative position end</td></tr>
</table>
Read-Write Attributes: R/W Applicable Mode: ALL</td>
</tr>
<tr>
<td>P283</td>
<td>Communication status word</td>
<td>ANY</td>
<td>Communication status word
Read-Write Attributes: R/W Applicable Mode: ALL</td>
</tr>
</table>

P284 ~ P289	Factory reserve	——	——
P290	Communication position command 0	-2^{31}~$2^{31}-1$	Communication position command 0 (When P005 is set as 0, use the value as communication position command) Unit: pulses Read-Write Attributes: R/W Applicable Mode: P
P292	Communication position command 1	-2^{31}~$2^{31}-1$	Communication position command 1 (When P005 is set as 1, use the value as communication position command) Unit: pulses Read-Write Attributes: R/W Applicable Mode: P
P294	Communication position command 2	-2^{31}~$2^{31}-1$	Communication position command 2 (When P005 is set as 2, use the value as communication position command) Unit: pulses Read-Write Attributes: R/W Applicable Mode: P
P296	Communication position command 3	-2^{31}~$2^{31}-1$	Communication position command 3 (When P005 is set as 3, use the value as communication position command) Unit: pulses Read-Write Attributes: R/W Applicable Mode: P
P298	Communication position command 4	-2^{31}~$2^{31}-1$	Communication position command 4 (When P005 is set as 4, use the value as communication position command) Unit: pulses Read-Write Attributes: R/W Applicable Mode: P
P300	Communication position command 5	-2^{31}~$2^{31}-1$	Communication position command 5 (When P005 is set as 5, use the value as communication position command) Unit: pulses Read-Write Attributes: R/W Applicable Mode: P
P302	Communication position command 6	-2^{31}~$2^{31}-1$	Communication position command 6 (When P005 is set as 6, use the value as communication position command) Unit: pulses Read-Write Attributes: R/W Applicable Mode: P
P304	Communication position command 7	-2^{31}~$2^{31}-1$	Communication position command 7 (When P005 is set as 7, use the value as communication position command) Unit: pulses Read-Write Attributes: R/W Applicable Mode: P
P306	Communication position command 8	-2^{31}~$2^{31}-1$	Communication position command 8 (When P005 is set as 8, use the value as communication position command) Unit: pulses Read-Write Attributes: R/W Applicable Mode: P
P308	Communication position command 9	-2^{31}~$2^{31}-1$	Communication position command 9 (When P005 is set as 9, use the value as communication position command) Unit: pulses Read-Write Attributes: R/W Applicable Mode: P
P310	Communication position command 10	-2^{31}~$2^{31}-1$	Communication position command 10 (When P005 is set as 10, use the value as communication position command) Unit: pulses Read-Write Attributes: R/W Applicable Mode: P

P312	Communication position command 11	-2^{31}~$2^{31}-1$	Communication position command 11 (When P005 is set as 11, use the value as communication position command) Unit: pulses Read-Write Attributes: R/W Applicable Mode: P
P314	Communication position command 12	-2^{31}~$2^{31}-1$	Communication position command 12 (When P005 is set as 12, use the value as communication position command) Unit: pulses Read-Write Attributes: R/W Applicable Mode: P
P316	Communication position command 13	-2^{31}~$2^{31}-1$	Communication position command 13 (When P005 is set as 13, use the value as communication position command) Unit: pulses Read-Write Attributes: R/W Applicable Mode: P
P318	Communication position command 14	-2^{31}~$2^{31}-1$	Communication position command 14 (When P005 is set as 14, use the value as communication position command) Unit: pulses Read-Write Attributes: R/W Applicable Mode: P
P320	Communication position command 15	-2^{31}~$2^{31}-1$	Communication position command 15 (When P005 is set as 15, use the value as communication position command) Unit: pulses Read-Write Attributes: R/W Applicable Mode: p
P322	Factory reserve	——	——
P323	Factory reserve	——	——
P324	Communication speed command 0	-6000~6000	Communication speed command 0 (When P005 is set as 0, use the value as communication speed command) Unit: rpm Read-Write Attributes: R/W Applicable Mode: S
P325	Communication speed command 1	-6000~6000	Communication speed command 1 (When P005 is set as 1, use the value as communication speed command) Unit: rpm Read-Write Attributes: R/W Applicable Mode: S
P326	Communication speed command 2	-6000~6000	Communication speed command 2 (When P005 is set as 2, use the value as communication speed command) Unit: rpm Read-Write Attributes: R/W Applicable Mode: S
P327	Communication speed command 3	-6000~6000	Communication speed command 3 (When P005 is set as 3, use the value as communication speed command) Unit: rpm Read-Write Attributes: R/W Applicable Mode: S
P328	Communication speed command 4	-6000~6000	Communication speed command 4 (When P005 is set as 4, use the value as communication speed command) Unit: rpm Read-Write Attributes: R/W Applicable Mode: S
P329	Communication speed command 5	-6000~6000	Communication speed command 5 (When P005 is set as 5, use the value as communication speed command) Unit: rpm Read-Write Attributes: R/W Applicable Mode: S

P330	Communication speed command 6	-6000~6000	Communication speed command 6 (When P005 is set as 6, use the value as communication speed command) Unit: rpm Read-Write Attributes: R/W Applicable Mode: S
P331	Communication speed command 7	-6000~6000	Communication speed command 7 (When P005 is set as 7, use the value as communication speed command) Unit: rpm Read-Write Attributes: R/W Applicable Mode: S
P332	Communication speed command 8	-6000~6000	Communication speed command 8 (When P005 is set as 8, use the value as communication speed command) Unit: rpm Read-Write Attributes: R/W Applicable Mode: S
P333	Communication speed command 9	-6000~6000	Communication speed command 9 (When P005 is set as 9, use the value as communication speed command) Unit: rpm Read-Write Attributes: R/W Applicable Mode: S
P334	Communication speed command 10	-6000~6000	Communication speed command 10 (When P005 is set as 10, use the value as communication speed command) Unit: rpm Read-Write Attributes: R/W Applicable Mode: S
P335	Communication speed command 11	-6000~6000	Communication speed command 11 (When P005 is set as 11, use the value as communication speed command) Unit: rpm Read-Write Attributes: R/W Applicable Mode: S
P336	Communication speed command 12	-6000~6000	Communication speed command 12 (When P005 is set as 12, use the value as communication speed command) Unit: rpm Read-Write Attributes: R/W Applicable Mode: S
P337	Communication speed command 13	-6000~6000	Communication speed command 13 (When P005 is set as 13, use the value as communication speed command) Unit: rpm Read-Write Attributes: R/W Applicable Mode: S
P338	Communication speed command 14	-6000~6000	Communication speed command 14 (When P005 is set as 14, use the value as communication speed command) Unit: rpm Read-Write Attributes: R/W Applicable Mode: S
P339	Communication speed command 15	-6000~6000	Communication speed command 15 (When P005 is set as 15, use the value as communication speed command) Unit: rpm Read-Write Attributes: R/W Applicable Mode: S
P340	Communication speed command 16	-6000~6000	Communication speed command 16 (When P005 is set as 16, use the value as communication speed command) Unit: rpm Read-Write Attributes: R/W Applicable Mode: S
P341	Communication speed command 17	-6000~6000	Communication speed command 17 (When P005 is set as 17, use the value as communication speed command) Unit: rpm Read-Write Attributes: R/W Applicable Mode: S

P342	Communication speed command 18	-6000~6000	Communication speed command 18 (When P005 is set as 18, use the value as communication speed command) Unit: rpm Read-Write Attributes: R/W Applicable Mode: S
P343	Communication speed command 19	-6000~6000	Communication speed command 19 (When P005 is set as 19, use the value as communication speed command) Unit: rpm Read-Write Attributes: R/W Applicable Mode: S
P344	Communication speed command 20	-6000~6000	Communication speed command 20 (When P005 is set as 20, use the value as communication speed command) Unit: rpm Read-Write Attributes: R/W Applicable Mode: S
P345	Communication speed command 21	-6000~6000	Communication speed command 21 (When P005 is set as 21, use the value as communication speed command) Unit: rpm Read-Write Attributes: R/W Applicable Mode: S
P346	Communication speed command 22	-6000~6000	Communication speed command 22 (When P005 is set as 22, use the value as communication speed command) Unit: rpm Read-Write Attributes: R/W Applicable Mode: S
P347	Communication speed command 23	-6000~6000	Communication speed command 23 (When P005 is set as 23, use the value as communication speed command) Unit: rpm Read-Write Attributes: R/W Applicable Mode: S
P348	Communication speed command 24	-6000~6000	Communication speed command 24 (When P005 is set as 24, use the value as communication speed command) Unit: rpm Read-Write Attributes: R/W Applicable Mode: S
P349	Communication speed command 25	-6000~6000	Communication speed command 25 (When P005 is set as 25, use the value as communication speed command) Unit: rpm Read-Write Attributes: R/W Applicable Mode: S
P350	Communication speed command 26	-6000~6000	Communication speed command 26 (When P005 is set as 26, use the value as communication speed command) Unit: rpm Read-Write Attributes: R/W Applicable Mode: S
P351	Communication speed command 27	-6000~6000	Communication speed command 27 (When P005 is set as 27, use the value as communication speed command) Unit: rpm Read-Write Attributes: R/W Applicable Mode: S
P352	Communication speed command 28	-6000~6000	Communication speed command 28 (When P005 is set as 28, use the value as communication speed command) Unit: rpm Read-Write Attributes: R/W Applicable Mode: S
P353	Communication speed command 29	-6000~6000	Communication speed command 29 (When P005 is set as 29, use the value as communication speed command) Unit: rpm Read-Write Attributes: R/W Applicable Mode: S

P354	Communication speed command 30	-6000~6000	Communication speed command 30 (When P005 is set as 30, use the value as communication speed command) Unit: rpm Read-Write Attributes: R/W Applicable Mode: S
P355	Communication speed command 31	-6000~6000	Communication speed command 31 (When P005 is set as 31, use the value as communication speed command) Unit: rpm Read-Write Attributes: R/W Applicable Mode: S
P356	Factory reserve	——	——
P357	Factory reserve	——	——
P358	Communication torque command 0	-3500~3500	Communication torque command 0(When P005 is set as 0, use the value as torque command) Unit: ‰ Read-Write Attributes: R/W Applicable Mode: T
P359	Communication torque command 1	-3500~3500	Communication torque command 1(When P005 is set as 1, use the value as torque command) Unit: ‰ Read-Write Attributes: R/W Applicable Mode: T
P360	Communication torque command 2	-3500~3500	Communication torque command 2(When P005 is set as 2, use the value as torque command) Unit: ‰ Read-Write Attributes: R/W Applicable Mode: T
P361	Communication torque command 3	-3500~3500	Communication torque command 3(When P005 is set as 3, use the value as torque command) Unit: ‰ Read-Write Attributes: R/W Applicable Mode: T
P362	Communication torque command 4	-3500~3500	Communication torque command 4(When P005 is set as 4, use the value as torque command) Unit: ‰ Read-Write Attributes: R/W Applicable Mode: T
P363	Communication torque command 5	-3500~3500	Communication torque command 5(When P005 is set as 5, use the value as torque command) Unit: ‰ Read-Write Attributes: R/W Applicable Mode: T
P364	Communication torque command 6	-3500~3500	Communication torque command 6(When P005 is set as 6, use the value as torque command) Unit: ‰ Read-Write Attributes: R/W Applicable Mode: T
P365	Communication torque command 7	-3500~3500	Communication torque command 7(When P005 is set as 7, use the value as torque command) Unit: ‰ Read-Write Attributes: R/W Applicable Mode: T
P366	Communication torque command 8	-3500~3500	Communication torque command 8(When P005 is set as 8, use the value as torque command) Unit: ‰ Read-Write Attributes: R/W Applicable Mode: T

P367	Communication torque command 9	-3500~3500	Communication torque command 9(When P005 is set as 9, use the value as torque command) Unit: ‰ Read-Write Attributes: R/W Applicable Mode: T
P368	Communication torque command 10	-3500~3500	Communication torque command 10(When P005 is set as 10, use the value as torque command) Unit: ‰ Read-Write Attributes: R/W Applicable Mode: T
P369	Communication torque command 11	-3500~3500	Communication torque command 11(When P005 is set as 11, use the value as torque command) Unit: ‰ Read-Write Attributes: R/W Applicable Mode: T
P370	Communication torque command 12	-3500~3500	Communication torque command 12(When P005 is set as 12, use the value as torque command)Unit: ‰ Read-Write Attributes: R/W Applicable Mode: T
P371	Communication torque command 13	-3500~3500	Communication torque command 13(When P005 is set as 13, use the value as torque command) Unit: ‰ Read-Write Attributes: R/W Applicable Mode: T
P372	Communication torque command 14	-3500~3500	Communication torque command 14(When P005 is set as 14, use the value as torque command) Unit: ‰ Read-Write Attributes: R/W Applicable Mode: T
P373	Communication torque command 15	-3500~3500	Communication torque command 15(When P005 is set as 15, use the value as torque command) Unit: ‰ Read-Write Attributes: R/W Applicable Mode: T
P374	Communication torque command 16	-3500~3500	Communication torque command 16(When P005 is set as 16, use the value as torque command) Unit: ‰ Read-Write Attributes: R/W Applicable Mode: T
P375	Communication torque command 17	-3500~3500	Communication torque command 17(When P005 is set as 17, use the value as torque command) Unit: ‰ Read-Write Attributes: R/W Applicable Mode: T
P376	Communication torque command 18	-3500~3500	Communication torque command 18(When P005 is set as 18, use the value as torque command) Unit: ‰ Read-Write Attributes: R/W Applicable Mode: T
P377	Communication torque command 19	-3500~3500	Communication torque command 19(When P005 is set as 19, use the value as torque command) Unit: ‰ Read-Write Attributes: R/W Applicable Mode: T
P378	Communication torque command 20	-3500~3500	Communication torque command 20(When P005 is set as 20, use the value as torque command) Unit: ‰ Read-Write Attributes: R/W Applicable Mode: T
P379	Communication torque command 21	-3500~3500	Communication torque command 21(When P005 is set as 21, use the value as torque command) Unit: ‰ Read-Write Attributes: R/W Applicable Mode: T

P380	Communication torque command 22	-3500~3500	Communication torque command 22(When P005 is set as 22, use the value as torque command) Unit: ‰ Read-Write Attributes: R/W Applicable Mode: T
P381	Communication torque command 23	-3500~3500	Communication torque command 23(When P005 is set as 23, use the value as torque command) Unit: ‰ Read-Write Attributes: R/W Applicable Mode: T
P382	Communication torque command 24	-3500~3500	Communication torque command 24(When P005 is set as 24, use the value as torque command) Unit: ‰ Read-Write Attributes: R/W Applicable Mode: T
P383	Communication torque command 25	-3500~3500	Communication torque command 25(When P005 is set as 25, use the value as torque command) Unit: ‰ Read-Write Attributes: R/W Applicable Mode: T
P384	Communication torque command 26	-3500~3500	Communication torque command 26(When P005 is set as 26, use the value as torque command) Unit: ‰ Read-Write Attributes: R/W Applicable Mode: T
P385	Communication torque command 27	-3500~3500	Communication torque command 27(When P005 is set as 27, use the value as torque command) Unit: ‰ Read-Write Attributes: R/W Applicable Mode: T
P386	Communication torque command 28	-3500~3500	Communication torque command 28(When P005 is set as 28, use the value as torque command) Unit: ‰ Read-Write Attributes: R/W Applicable Mode: T
P388	Communication torque command 30	-3500~3500	Communication torque command 30(When P005 is set as 30, use the value as torque command) Unit: ‰ Read-Write Attributes: R/W Applicable Mode: T
P389	Communication torque command 31	-3500~3500	Communication torque command 31(When P005 is set as 31, use the value as torque command) Unit: ‰ Read-Write Attributes: R/W Applicable Mode: T

(1) ★ in parameter list means it needs to be reserved in EEPROM after written in driver. Effective after power on again. Other parameters are effective immediately after modified but may be lost after power off.Please reserve in EEPROM if needed.

(2) Default parameters in the table corresponds P182=3, i.e. 24v/200w/2500ppr servo motor.

(3) All parameters fit standard modbus protocol.

4.2 Modbus RTU Protocol

Modbus RTU defines "bit" in serial transmission information area of bus, and way to package and decode information. In Modbus mode, every byte(1Byte=8bit)is represented by 2 hexadecimal characters (0~F). Information must be transferred continuously. Complete information frame contains slave address, function code, data area and error check.

Slave address: Signified by 1 byte. Valid slave machine address range is 0-247, and addressing range is 1-247. Host machine sends slave address into address area of information frame and starts locating in slave machine. The slave puts own address in address area of reply to let the host identify slave address that has responded. Address 0 is for broadcast, which can be identified by all slaves.

Function code: Signified by 1 byte. Function code is used to describe actions to be operated to slave machine when the host sends message to the slave. When the slave responds properly, the function code turns to original code. If not, change the most significant bit of original function code to "1" and return.

Data area: The length and contents of data area differ according to function code. It contains (beginning) register address to be accessed, data length to be read, data to be written and so on. Notice that significant bit is in the front.

Error check: Signified by 2 bytes. The first byte is the high 8 bits of cyclic redundancy check CRC16. There will be no details about CRC16. Please search online if you are interested about it. Notice: parity check is for single byte information while CRC is for the whole information frame.

4.2.1 Function Code 16#03: Read Register

Read register has no broadcast function. It only fits one designated slave address. Values of one or serial registers can be read.

The example shows how to read values of two serial registers P212 and P213 in No.2 slave machine.

Host request frame:

Slave address	Function code	Beginning register address high 8 bits	Beginning register address low 8 bits	Read register number high 8 bits	Read register number low 8 bits	CRC check high 8 bits	CRC check low 8 bits
1Byte	1Byte	2Bytes		2Bytes		2Bytes	
16#02	16#03	16#00	16#D4	16#00	16#02	16#84	16#00

Slave response frame:

Slave address	Function code	Length of read data	Register 1 data high 8 bits	Register 1 data low 8 bits	Register 2 data high 8 bits	Register 2 data low 8 bits	CRC check high 8 bits	CRC check low 8 bits
1Byte	1Byte	1Byte	2Bytes		2Bytes		2Bytes	
16#02	16#03	16#04	16#01	16#F4	16#03	16#E8	16#89	16#83

According to response fraction, the value of P212 is 16#01F4, that is 500, value of P213 is 16#03E8, that is 1000.

4.2.2 Function Code 16#06: Write Single Register

Write single registers support broadcasting and each command can only write one register. The following example is to write the value of P325 register from No.1 slave as 1000.

Host request frame:

Slave address	Function code	Write register address high 8 bits	Write register address low 8 bits	Write data high 8 bits	Write data low 8 bits	CRC check high 8 bits	CRC check low 8 bits
1Byte	1Byte	2Bytes		2Bytes		2Bytes	
16#01	16#06	16#01	16#45	16#03	16#E8	16#99	16#5D

Slave response frame:

Slave address	Function code	Write register address high 8 bits	Write register address low 8 bits	Write data high 8 bits	Write data low 8 bits	CRC check high 8 bits	CRC check low 8 bits
1Byte	1Byte	2Bytes		2Bytes		2Bytes	
16#01	16#06	16#01	16#45	16#03	16#E8	16#99	16#5D

According to response fraction, the slave writes successfully when a fraction of data is sent back unchanged.

4.2.3 Function Code 16#10: Write Serial Registers

Write serial registers support broadcasting and each command can write one or several registers. The following example is to write the value of P325 as 1000 and value of P326 as 2000 through 16#10 function code.

Host request frame:

Slave address	Function code	Beginning register address high 8 bits	Beginning register address low 8 bits	Write register number high 8 bits	Write register number low 8 bits	Write data overall length	Write register 1 data high 8 bits	Write register 1 data low 8 bits
1Byte	1Byte	2Bytes		2Bytes		1Byte	2Bytes	
16#01	16#10	16#01	16#45	16#00	16#02	16#04	16#03	16#E8

Write register 2 data high 8 bits	Write register 2 data low 8 bits	CRC check high 8 bits	CRC check low 8 bits
2Bytes		2Bytes	
16#07	16#D0	16#B9	16#EC

Slave response frame:

Slave address	Function code	Beginning register address high 8 bits	Beginning register address low 8 bits	Write register number high 8 bits	Write register number low 8 bits	CRC check high 8 bits	CRC check low 8 bits
1Byte	1Byte	2Bytes		2Bytes		2Bytes	
16#01	16#10	16#01	16#45	16#00	16#02	16#51	16#E1

4.2.4 No Response and Abnormal Response

After the host sends Modbus request frame, the slave may have two abnormal types of response, which is no response on time and abnormal response in fixed time.

When the host judges that no response comes from the slave after certain time check, it needs to check communication wiring, slave status lamp and whether the environment around is suitable for communication.

The common reason of abnormal response of slave is data frame error, such as writing data into a read only register or that data written is out of range.

Example of abnormal response

Host request frame:

Slave address	Function code	Write register address high 8 bits	Write register address low 8 bits	Write data high 8 bits	Write data low 8 bits	CRC check high 8 bits	CRC check low 8 bits
1Byte	1Byte	2Bytes		2Bytes		2Bytes	
16#01	16#06	16#01	16#45	16#27	16#10	16#99	16#5D

Slave request frame:

Slave address	Function code	Error code	CRC check high 8 bits	CRC check low 8 bits
1Byte	1Byte	1Byte	2Bytes	
16#01	16#86	16#02	16#C3	16#A1

Abnormal response error code description:

Error code	Description
16#02	Wrong read-write attributes of register; Wrong parameter range.
16#03	Wrong register address.
16#06	The slave is busy.

4.3 CANopen Communication Protocol

CANopen communication protocol is developed from CAL(CAN Application Layer) by the organization CiA(CAN-in-Automation), based in Nuremberg in Germany, in late 1990s. The CiA keeps introducing equipment sub-protocol to many different industries on the basis of CANopen basic protocol——CiA DS 301(DS:Draft Standard) and makes it developed and spread faster. The sub-protocol used in motor driving and action control industry is CiA DSP 402(DSP:Draft Standard Proposal) Note：CiA DS 301 will be abbreviated as CiA301 and CiA DSP 402 will be abbreviated as CiA402.

CANopen is defined as real time communication of small network and control signal, the features of which are listed below：

(1) Message transmission uses the form of CAN standard frame, i.e., 11-bit ID domain to reduce transmission time;

(2) Network control message adopts data minimum byte number, such as heartbeat message which contains only one byte;

(3) Process data of real-time update（PDO）dose not need message response from receiver, which means adopting production-consumption model to reduce load of bus;;

(4) Configuration parameters, needing confirmation from receiver, are commonly transferred by fast single word transmission（Fast SDO）, which means a message transmits a 32-bit data at most, avoiding real-time reduction caused by framing.

All definitions above are for saving time and expenses as well as promising real time ability to the best. In order to cut down workload of simple network, CANopen defines forced default identifier（CAN-ID）distribution list to simplify the study process for users and maintainers.

4.3.1 CANopen Specification

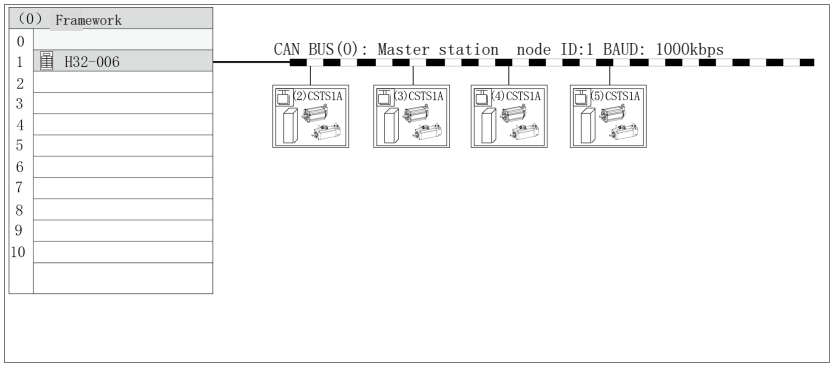
SSTS1A servo driver supports CiA402 protocol. The concrete specifications is listed below.

Data link layer	CAN2.0A 11-bit CAN-ID
Application layer	CANopen CiA DS301/CiA DSP402
Baud rate	1Mbps(default), 800kbps, 500kbps, 250kbps, 125kbps, 50kbps, 20kbps
Max station number	127
CAN frame length	0～8Bytes
Termination resistor	120Ω
Service supported	NMT: Network management (node status、heartbeat、node protection) SDO: Object of service data PDO: Object of process data SYNC: Synchronization
SDO transmission type	Fast SDO transmission
PDO transmission type	Time trigger, event trigger, synchronous trigger
PDO number supported	4×RPDO, 4×TPDO
Servo running mode	Profile position mode Profile velocity mode Profile torque mode Homing mode Cyclic synchronous position mode Cyclic synchronous velocity mode Cyclic synchronous torque mode

4.3.2 CAN Wiring

There is no special rules for CAN bus physical layer, so that multiple physical medium are allowed to use such as twisted pair and optical fibre. The twisted pair is the most common. Two signal lines are called CAN_H and CAN_L and execute transmission with differential voltage (mainly bus transceiver). The voltage of signal lines is about 2.5V in free time, and this state is called logic 1 or recessive position. Logic 0 is shown by making CAN_H higher than CAN_L and called dominant position. The voltage values are CAN_H=3.5V and CAN_L=1.5V in logic 0, and dominant position has priority while competing.

SSTS1A servo driver adopts RJ45 ports and twisted pair. The concrete definitions are introduced in 3.2.2. The connection between upper computer and servo, as well as between servos, is bus series connection, that is to connect CAN_H with CAN_H and connect CAN_L with CAN_L. It needs to connect a terminal resistor of 120Ω between main station and the last slave station. Turn the dial switch to ON in servo end to activate internal 120Ω resistance. Use twisted pair with shielding layer as communication cable and ensure it well-grounded (In short-distance communication, the GND earth wire of CN2 and CN3 can be disconnected, but ground connection is suggested in long-distance and high-BPS communication).



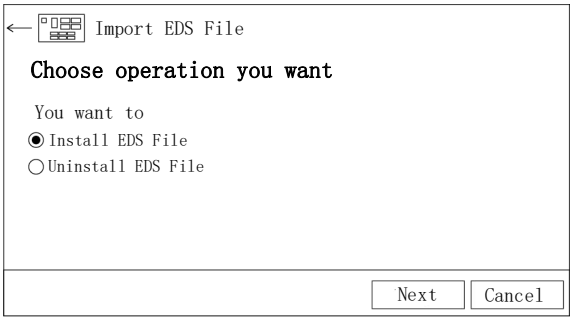
Connection between baud rate and communication distance

P11=1	1Mbps	25m
P11=2	800kbps	50m
P11=3	500kbps	100m
P11=4	250kbps	250m
P11=5	125kbps	500m
P11=6	50kbps	1000m
P11=7	20kbps	2500m

4.3.3 CANopen Communication Network Configuration

4.3.3.1 EDS file

EDS (short for Electronic Data Sheet) file is the marker file or similar code of slave station connected with PLC, which can be used to identify the type of the slave (which similar in 401、402 and 403 or which device in 402). The file contains all information of slave station, including parameters of manufacturer, serial code, software version, baud rate supported, object dictionary able of mapping and its properties. So it needs to import the EDS file of slave servo to upper configuration software before configuring hardware.



4.3.3.2 OD

CANopen OD（short for Object Dictionary）is the core concept of CANopen protocol. OD is an organized object group that describes all parameters of the comparable CANopen node including storage position of communication data. The table is called EDS file when it can be transmitted. The design of OD is based on CiA402 standard and each object has exact function definition. Objects here are similar to memory addresses. Some objects, such as speed and position, can be modified by external controller, but some of them only can be modified by the driver like status and error information. Every object uses a 16-bit value for addressing, which is called index and the range is 0x0000~0xFFFF. To avoid situation that no index is available when there is a great quantity of data, another 8-bit value called sub-index is defined for some indices and its range is 0x00~0xFF. The exact parameter in each index can be 8-bit, 16-bit or 32-bit the most.

Each object of CANopen OD is described by a series of sub-protocols, which describe the function, name, index, sub-index, data type, read-write attributes and whether or not it is essential and etc. It guarantees the compatibility among same type of devices from different manufacturers.

The core description sub-protocol of CANopen is CiA301, which includes descriptions of application layer and communication layer, and the others are just supplement and extension. For different industries, a special CANopen sub-protocol is prepared and the number is commonly CiA DS4xx.

SSTS1A servo is a standard CAN slave device and firmly follows CANopen2.0A protocol. It can communicate with all upper computers supporting CANopen2.0A protocol.(Note: subscript h identifies the hexadecimal, subscript b identifies the binary)

Overview of OD structure

Index range	Object
0000_h	Not used
0001_h-$001F_h$	Static data type(standard data type, such as bool,int16)
0020_h-$003F_h$	Complex data type(predefined structure type formed by standard type,such as PDOCommParam、SDOParam)
0040_h-$005F_h$	Complex data type set by manufacturer
0060_h-$007F_h$	Static data type stipulated in device sub-protocol
0080_h-$009F_h$	Complex data type stipulated in device sub-protocol
$00A0_h$-$0FFF_h$	Save
1000_h-$1FFF_h$	Communication sub-protocol area(such as device type, error register,PDOCommParam, PDO mapping param)
2000_h-$5FFF_h$	Sub-protocol area set by manufacturer(such as PIDParam)
6000_h-$9FFF_h$	Standard device sub-protocol area(param related to CiA402 protocol)
$A000_h$-$FFFF_h$	Save

Overview of common objects

Index	Sub-index	Name	Data type	Permission	Physical dimension	PDO mapping	Default
1000_h	00_h	Device type	uint 32	ro		No	00020192_h
1001_h	00_h	Error register	uint8	ro		Optional	
1002_h	00_h	Manufacturer status register	uint 32	ro		Optional	
1003_h		Predefined error field					
	01_h~ 08_h	Error field	uint32	ro		No	
1005_h	00_h	Synchronous COB-ID	uint32	rw		No	00000080_h
1006_h	00_h	Synchronous cycle period	uint32	rw	μs	No	00000000_h
1007_h	00_h	Synchronous window length	uint32	rw	μs	No	00000000_h
1008_h	00_h	Manufacturer device name	string	CONST		No	SZHC SSTS1A CiA 402 servo
1009_h	00_h	Manufacturer hardware version	string	CONST		No	V0.2
$100A_h$	00_h	Manufacturer software version	string	CONST		No	V1.0
$100C_h$	00_h	Node protection time	uint16	rw	ms	No	0000_h

$100D_h$	00_h	Life factor	uint8	rw		No	00_h
1010_h		Save parameters					
	01_h	Save all parameters	uint32	rw		No	
	02_h	Save communication parameters	uint32	rw		No	
	03_h	Save application parameters	uint32	rw		No	
1011_h		Recover default parameters					
	01_h	Recover all parameters	uint32	rw		No	
	02_h	Recover communication parameters	uint32	rw		No	
	03_h	Recover application parameters	uint32	rw		No	
1012_h	00_h	Time stamp object COB-ID	uint32	rw		No	
1013_h	00_h	High resolution time stamp	uint32	rw		No	
1014_h	00_h	EMCY COB-ID	uint32	rw		No	80_h+Node_ID
1015_h	00_h	EMCY inhibition time	uint16	rw	×0.1ms		00_h
1016_h		Consumer heartbeat time					
	01_h	Consumer heartbeat time	uint32	rw	ms		
1017_h	00_h	Producer heartbeat time	uint16	rw	ms	——	
1019_h							
1029_h		Wrong action object	uint8	rw		——	
1200_h		SDO server parameter					
1400_h ~ 1403_h		RPDO communication parameter	REC				
	00_h	Max sub-index	uint8				
	01_h	COB-ID of RPDO	uint32				
	02_h	Transmission type of RPDO	uint8				
1600_h ~ 1603_h		RPDO mapping parameter					
	00_h	Max sub-index	uint8				
	01_h~08_h	RPDO mapping target	uint32				

1800h ~ 1803h		TPDO communication parameter					
	00h	Max sub-index					
	01h	COB-ID of TPDO					
	02h	Transmission type of TPDO					
	03h	Confinement time of production forbidden					
	05h	Trigger time of event timer					
	06h	Synchronize initial value					
1A00h ~ 1A03h		TPDO mapping parameter					
	00h	Max sub-index	uint8				
	01h~08h	TPDO mapping target	uint32				
6060h	00h	Control mode	int8	rw		RPDO	
6040h	00h	Control word	uint16	rw		RPDO	
607Ah	00h	Target position	int32	rw	pulse	RPDO	
6081h	00h	Outline velocity(limit)	uint32	rw	rpm	RPDO	
6083h	00h	Outline acceleration	uint32	rw	ms/1000rpm	RPDO	
6084h	00h	Outline deceleration	uint32	rw	ms/1000rpm	RPDO	
6091h	01h	Electronic gear ratio numerator	uint32	rw		R-SDO	
	02h	Electronic gear ratio denominator	uint32	rw		R-SDO	
6065h	00h	Position deviation excess threshold	uint32	rw	×256pulse	R-SDO	
6067h	00h	Position arriving threshold	uint32	rw	pulse	R-SDO	
60E0h	00h	Positive torque limit	uint16	rw	‰	R-SDO	
60E1h	00h	Negative torque limit	uint16	rw	‰	R-SDO	
60FFh	00h	Target speed	int32	rw	rpm	RPDO	
606Dh	00h	Speed arriving threshold	uint16	rw	rpm	R-SDO	
606Fh	00h	Zero speed detection threshold	uint16	rw	rpm	R-SDO	
6071h	00h	Target torque	int16	rw	‰	RPDO	
607Fh	00h	Speed limit	int32	rw	rpm	R-SDO	
6061h	00h	Mode display	int8	ro		TPDO	
6041h	00h	Status word	uint16	ro		TPDO	
603Fh	00h	Alarm code	uint16	ro		TPDO	
6062h	00h	Command position	int32	ro	pulse	TPDO	
6064h	00h	Feedback position	int32	ro	pulse	TPDO	

$60F4_h$	00_h	Position deviation	int32	ro	pulse	TPDO	
$606B_h$	00_h	Command speed	int32	ro	rpm	TPDO	
$606C_h$	00_h	Feedback speed	int32	ro	rpm	TPDO	
6074_h	00_h	Command torque	int16	ro	‰	TPDO	
6077_h	00_h	Feedback torque	int16	ro	‰	TPDO	
2000_h	01_h	Servo node ID	uint8	rw		R-SDO	
	02_h	RS485 baud rate	uint8	rw		R-SDO	
	03_h	CAN baud rate	uint8	rw		R-SDO	
2002_h	01_h	Torque limit selection	uint16	rw		R-SDO	
	02_h	Interpolation mode selection	uint16	rw		R-SDO	
	03_h	Communication cycle	uint16	rw	ms	R-SDO	
	04_h	Homing mode	uint16	rw		R-SDO	
	05_h	Relative/Absolute position control	uint16	rw		R-SDO	
	06_h	Motor positive direction selection	uint16	rw		R-SDO	
	07_h	Over-load level	uint16	rw	‰	R-SDO	
	08_h	Over-speed level	uint16	rw	rpm	R-SDO	
2003_h	01_h	Motor code	uint16	rw		R-SDO	
	02_h	Encoder resolution	uint16	rw	ppr or bit	R-SDO	
	03_h	Z electrical angle	uint16	rw		R-SDO	
	04_h	hall101 electrical angle	uint16	rw		R-SDO	
	05_h	Motor pole number	uint16	rw		R-SDO	
	06_h	Motor rated speed	uint16	rw	rpm	R-SDO	
	07_h	Motor rated torque	uint16	rw		R-SDO	
	08_h	Motor max torque	uint16	rw		R-SDO	
	09_h	Motor rated voltage	uint16	rw	V	R-SDO	
2010_h	01_h	Current loop proportional gain	uint16	rw	Hz	R-SDO	
	02_h	Current loop integral time constant	uint16	rw	×0.1ms	R-SDO	
2011_h	01_h	Velocity loop proportional gain	uint16	rw	Hz	R-SDO	
	02_h	Velocity loop integral time constant	uint16	rw	ms	R-SDO	
2012_h	01_h	Position loop proportional gain	uint16	rw	1/s	R-SDO	
	02_h	Velocity feedforward gain	uint16	rw	‰	R-SDO	
	03_h	Velocity loop proportional gain	uint16	rw	Hz	R-SDO	
	04_h	Velocity loop integral time constant	uint16	rw	ms	R-SDO	
2100_h	00_h	Servo alarm code	uint16	ro		T-SDO	

4.3.3.3 COB-ID communication object identifier

COB-ID (short for Communication Object Identifier) assigns priorities of objects in communication and distinguishes communication object. It is corresponding to 11-bit frame ID of CAN2.0A, so it's also called CAN-ID. The ID is formed with object function code in high 4 bits and node address Node-ID in low 7 bits, as the following table shows:

COB-ID/CAN-ID										
10	9	8	7	6	5	4	3	2	1	0
Function Code				Node-ID						

There is a fixed COB-ID for each communication object of CANopen. The function Code is for data transmission and it defines NMT message and priority of SDO and PDO. The smaller code represents higher priority. Node-ID is address of servo slave station and ranges from 1 to 127.

Communication object	Function Code	Node-ID	COB-ID	Relative object index
NMT	0000_b	0	0_h	—
SYNC	0001_b	0	80_h	1005_h, 1006_h
EMCY	0001_b	1~127	80_h+Node-ID	1014_h
TPDO1	0011_b	1~127	180_h+Node-ID	1800_h
RPDO1	0100_b	1~127	200_h+Node-ID	1400_h
TPDO2	0101_b	1~127	280_h+Node-ID	1801_h
RPDO2	0110_b	1~127	300_h+Node-ID	1401_h
TPDO3	0111_b	1~127	380_h+Node-ID	1802_h
RPDO3	1000_b	1~127	400_h+Node-ID	1402_h
TPDO4	1001_b	1~127	480_h+Node-ID	1803_h
RPDO4	1010_b	1~127	500_h+Node-ID	1403_h
T-SDO	1011_b	1~127	580_h+Node-ID	1200_h
R-SDO	1100_b	1~127	600_h+Node-ID	1200_h
Error control	1110_b	1~127	700_h+Node-ID	1016_h, 1017_h

E.g: For RPDO2 of No.2 slave station, the COB-ID is 302_h.

4.3.3.4 NMT network management

Network management includes Boot-up information, Heartbeat protocol and NMT information. Based on main-slave communication mode/producer-consumer communication mode, it is used to manage and monitor nodes in monitoring network and mainly achieve: node state control, error control and node start.

4.3.3.4.1 NMT node state

NMT management involves 6 kinds of state of a CANopen node since power on:

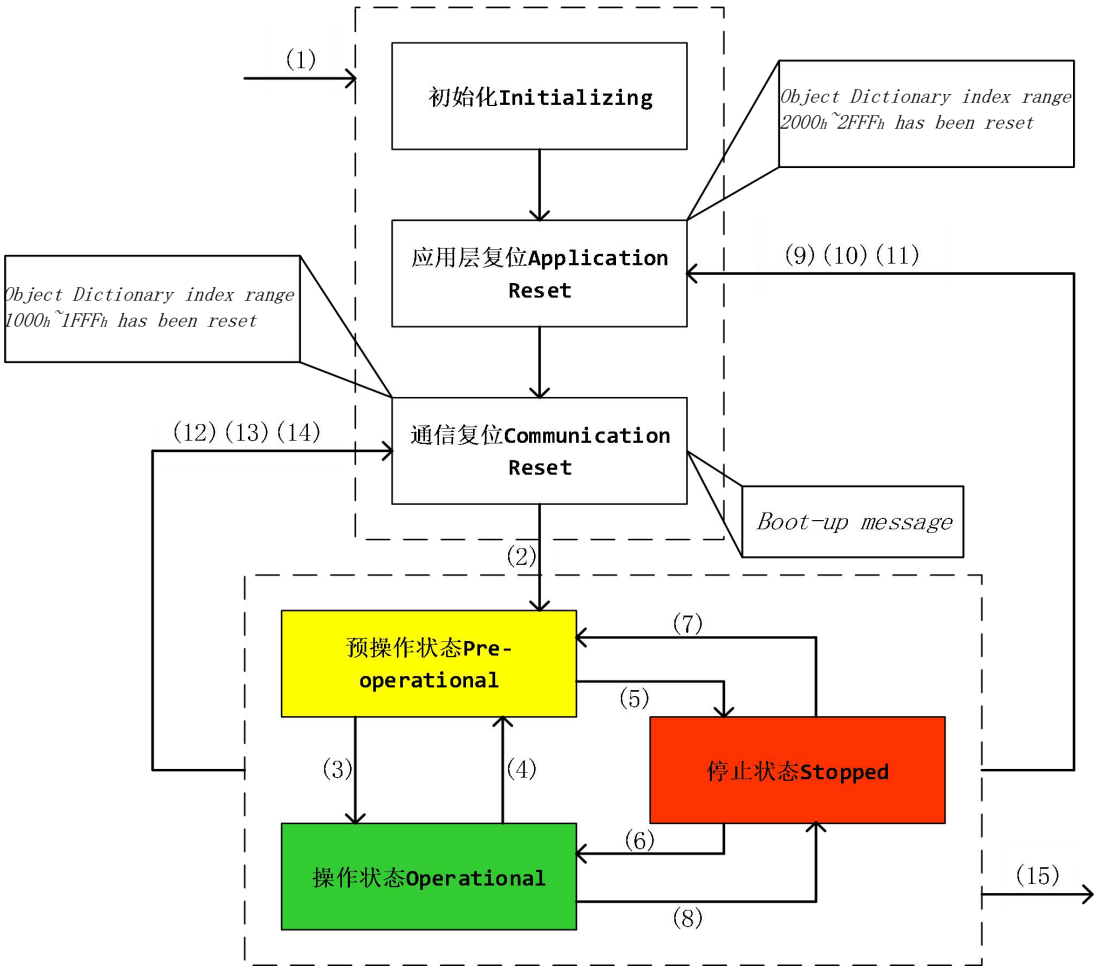
- ✔ **Initializing:** Initialize function parts after power on including CAN controller;
- ✔ **Application Reset:** Applications of node are reset(started) such as initial values of switching value output and analog quantity output;
- ✔ **Communication Reset:** CANopen communication of node is reset(started) and comes to effective from now;
- ✔ **Pre-operational:** CANopen communication of node is prepared and PDO communication cannot be executed, but SDO, parameter configuration and NMT network management is allowed;

✔ **Operational**: CANopen communication is activated when the node receives start command from NMT host. After PDO communication starts, transmit as requested in object dictionary. And SDO also can transfer data and modify parameters;

✔ **Stopped**: PDO communication of node is stopped after receiving stop command from NMT host, but SDO and NMT network management still can do actions to node.

Except for initialization state, NMT host can order any CANopen node in network to switch among other 5 states through NMT command. The CANopen node also can switch state automatically.

The chart below shows the statemachine of CANopen node.



(1) Power on
(2) Automatic switch to Pre-operational
(3)(6) NMT switch to Operational
(4)(7) NMT switch to Pre-operational
(5)(8) NMT switch to Stopped
(9)(10) NMT switch to Application Reset
(12)(13)(14) NMT switch to Communication Reset
(15) Power off or Hardware Reset

4.3.3.4.2 NMT node boot-up message

After an CANopen slave comes online, in order to prompt the slave(for Hot Swap) or avoid conflict with other slave Node-ID, it must send boot-up message. The COB-ID is 700h+Node-ID, data length DLC is 1 byte and the producer is CANopen slave.

4.3.3.4.3 NMT node state, heartbeat message and node protection/life protection

To monitor current node state and whether CANopen node is online, it is commonly requested in CANopen applications that slaves, which comes power on online, send state message(heartbeat message) at regular time to let the host confirm whether the slave is abnormal or offline.

The COB-ID of heartbeat message and node boot-up message are both 700_h+Node-ID. Data length DLC is 1 byte, representing current state of node: 04_h refers to stop state, 05_h refers to optional state and $7F_h$ refers to pre-optional state.

CANopen slave sends heartbeat message according to the heartbeat productive time(ms) set in 1017_h of object dictionary. The CANopen host (NMT host) will check according to the heartbeat consumption time set in 1016_h. If there is no heartbeat message received after several consumption times, the salve will be considered as offline or malfunction.

Node protection/Life protection functions of $100C_h$(Protection time) and $100D_h$(Life factor) are both supported. Node protection is realized by NMT host to check state of NMT slave periodically through remote frame; Life protection is realized by slave to monitor the state of the host through intervals between those remote frame it has received. $100C_h$(Protection time, ms) refers to interval of node protection remote frame, of which the product with $100D_h$(Life factor) defines the max query time of host. When $100C_h$ and $100D_h$ are not 0 and receive the first frame of node protection request, life protection is activated. Node protection communication follows the master-slave model, that is, each remote frame from the host must be answered by slave. If the salve does not response in $100C_h$, it will be considered as offline. If no remote frame is received in $100C_h \times 100D_h$, the host will be considered as offline.

Remote frame message from the host:

COB-ID	RTR
700_h+Node_ID	1

Node protection response message from the slave:

COB-ID	RTR	DATA
700_h+Node_ID	0	Status word

Status word of slave is similar to state in heartbeat, but the highest bit of status word is 0 or 1 alternately:

bit7	bit6~bit0
"1" or "0" alternately	04_h is stop state, 05_h is optional state, $7F_h$ is pre-optional state

Suggestion: $100C_h$(Protection time, ms) is more than 10ms, $100D_h$(Life factor) is more than 2.

4.3.3.4.4 NMT node status switch command

In NMT network management, NMT node status switch command is the core. As the "command" message of host network management, it must be kept firmly in mind by users. All the CAN-ID are 000_h, with the highest CAN priority. Data length is 2 bytes and the first one refers to command word:

MT messege command

Word	Definition
01_h	Start (Let the node into startup state)
02_h	Stop (Let the node into stopped state)
80_h	Enter pre-optional state (Let the node into pre-optional state)
81_h	Reset node application (Let the node restore initial state)
82_h	Reset node communication (Let the CAN and CANopen communication initialize again)

The second byte refers to Node-ID of controlled node, Set as 0 to control all nodes in the network at the same time.

NMT message form

COB-ID	RTR	DATA	
		Byte0	Byte1
000_h	0	Command word	Node number

Service supported in each NMT status

Service	Pre-operational	Operational	Stopped
Process data object (PDO)	×	√	×
Service data object (SDO)	√	√	×
Synchronization object (SYNC)	√	√	×
Emergency message (EMCY)	√	√	×
Network management (NMT)	√	√	√
Error control	√	√	√

4.3.3.5 SDO Service data object

SDO is mainly used for the CANopen host to configure parameters of slaves, that is, objects with low priority in transmission between devices, such as PID param of velocity loop and position loop and PDO configuration param. Service confirmation is the most unique feature of SDO. There will be a response for each information to ensure accurate transmission. In a CAN-open system, slave node often works as SDO server and main node works as the client, which is called "server-client communication". The SDO client can visit object dictionary in server by index and sub-index, so the main node can visit any parameter of slave node object dictionary and SDO is able to transmit data of different length (Decomposed to several messages if more than 4 bytes).

The principle of SDO communication is single. The client sends message with 600_h+Node-ID as CAN-ID, Node-ID is the node address of server and data length is 8 bytes; After receiving, the server replies message with 580_h+Node-ID as CAN-ID, this Node-ID is also node address of server and data length is 8 bytes. It is similar to Modbus communication.

SDO message form

COB-ID		DATA(8Bytes)							
T-SDO	580_h+ Node-ID	0 Byte	1 Byte	2 Byte	3 Byte	4 Byte	5 Byte	6 Byte	7 Byte
R-SDO	600_h+Node-ID	Command code	Index		Sub-index	Data			

Notice: Lower byte is in the front section and higher byte is in the latter section, which is opposite to the common habit!

The most common SDO protocol is Fast SDO, that is to finish in one round. The condition is that the values read and written ear no more than 32 bits. The command includes index, sub-index and data to be read and written.

Fast SDO message——Write Node-ID=1 servo salve object dictionary

Type	COB-ID	DATA(8Bytes)							
		0Byte	1Byte	2Byte	3Byte	4Byte	5Byte	6Byte	7Byte
Write request 4 byte	601_h	23_h	Index		Sub-index	Data			
Write request 3 byte		27_h				Data			——
Write request 2 byte		$2B_h$				Data		——	——
Write request 1 byte		$2F_h$				Data	——	——	——
Write Done response	581_h	60_h	Index		Sub-index	——	——	——	——
Write Error response		80_h				Abort code			

Fast SDO message——Write Node-ID=1 servo salve object dictionary

Type	COB-ID	DATA(8Bytes)							
		0Byte	1Byte	2Byte	3Byte	4Byte	5Byte	6Byte	7Byte
Read	601_h	40_h	Index		Sub-index	——	——	——	——
Read response 4 byte	581_h	43_h	Index		Sub-index	Data			
Read response 3 byte		47_h				Data			——
Read response 2 byte		$4B_h$				Data		——	——
Read response 1 byte		$4F_h$				Data	——	——	——
Read Error response		80_h	Index		Sub-index	Abort code			

4.3.3.6 PDO Process data object

The transmission of PDO adopts new mode of data exchange, different from traditional polling mode. Receiving and sending zone in devices are defined before transmission and data will be sent directly to specific unit, which shortens the query time and makes bus communication more efficient.

PDO is unidirectional for real time data transmission, belonging to "producer-consumer" mode. The data length is limited to 1~8 bytes. It is mainly for transmission of data needing high frequency exchange, such as order position, feedback position, order speed, feedback speed, order torque, feedback torque and etc.

4.3.3.6.1 PDO object

Referring to slave servo, PDO is divided into RPDO and TPDO according to receiving and sending of servo. For PDO, the final transmission way and content is decided by communication parameter and mapping parameter. SSTS1A servo supports 4 RPDO and 4 TPDO.

SSTS1A servo PDO object

Name	COB-ID	CommParam object	MappingParam object
RPDO1	200_h+Node-ID	1400_h	1600_h
RPDO2	300_h+Node-ID	1401_h	1601_h
RPDO3	400_h+Node-ID	1402_h	1602_h
RPDO4	500_h+Node-ID	1403_h	1603_h
TPDO1	180_h+Node-ID	1800_h	$1A00_h$
TPDO2	280_h+Node-ID	1801_h	$1A01_h$
TPDO3	380_h+Node-ID	1802_h	$1A02_h$
TPDO4	480_h+Node-ID	1803_h	$1A03_h$

4.3.3.6.2 PDO transmission type

There are two types of PDO transmission: synchronous transmission and asynchronous transmission.

Synchronous transmission (by receiving synchronous object): Synchronous transmission is to let all nodes upload data or execute orders at the same time by sending synchronous message, which can effectively avoid application logic chaos and imbalance bus load caused by asynchronous transmission, and the node sending synchronous message is generally NMT host. It also can be divided into periodic transmission (cyclic) and aperiodic transmission (acyclic). Periodic transmission is operated by receiving synchronous object (SYNC). It can set 1~240 objects trigger. Aperiodic transmission is pre-triggerred by remote frame or given event from object stipulated in device sub-protocol.

Asynchronous transmission (triggered by given event): There are two ways to trigger asynchronous transmission. One is to trigger by given object event stipulated in device sub-protocol (such as time transmission and data change transmission). The other is to send remote frame same as COB-ID of PDO.

4.3.3.6.3 PDO communication parameter

PDO communication parameters define COB-ID, transmission type, timing period and etc. RPDO communication parameters locate in 1400_h～$15FF_h$ of OD index. TPDO communication parameters locate in 1800_h~$19FF_h$ of OD index. Each index represents a PDO communication parameter set and the sub-index points to exact parameter.

PDO communication parameter

Index	Sub-index	Description	Data type
RPDO 1400_h～$15FF_h$ TPDO 1800_h～$19FF_h$	00_h	Number of parameters	uint8
	01_h	COB-ID	uint32
	02_h	Transmission type: 0:acyclic synchronization 1～240:cyclic synchronization 254: asynchronization, given event from manufacturer 255: synchronization, given event from device sub-protocol	uint8
TPDO 1800_h～$19FF_h$	03_h	Production prohibition confinement time (×0.1ms)	uint16
	05_h	Trigger time of event timer (ms)	uint16
	06_h	Initial value of synchronization	uint8

Transmission type: Cyclic synchronization and given event from manufacturer are more common.

Production prohibition confinement time: The minimum time interval of PDO sending confinement. It is to avoid sharp increase of bus load. For example, if the digital quantity input is too fast, TPDO sent from state change will be too frequent and bus load increases. So it needs a confinement time as "filter". The time unit is 0.1ms.

Trigger time of event timer: Time set for timed PDO. If it's 0, the PDO becomes event change sent.

Initial value of synchronization: PDO of synchronous transmission is sent after receiving several synchronous packages. The initial value is the number of synchronous packages. If set as 2, it means PDO is sent after receiving 2 synchronous packages.

- ✔ When transmission type of RPDO is 0~240, update the latest data to application once receiving a synchronous frame; when transmission type of RPDO is 254 or 255, update the data received directly to application.
- ✔ When transmission type of TPDO is 0, send it if the mapping data changes and a synchronous frame is received.
- ✔ When transmission type of TPDO is 1~240, send it after receiving comparable number of synchronous frames.
- ✔ When transmission type of TPDO is 254 or 255, send it when mapping data changes or event counter arrives.

4.3.3.6.4 PDO mapping parameter

PDO mapping parameters involve pointers pointing at data that PDO needs to send or that is received from corresponding process, including index, sub-index and data length(bits). Data length of each PDO is 8 bits at the most. Each PDO can mapping several objects. Sub-index 0 records the number of objects and sub-index 1~8 are specific mapping objects.

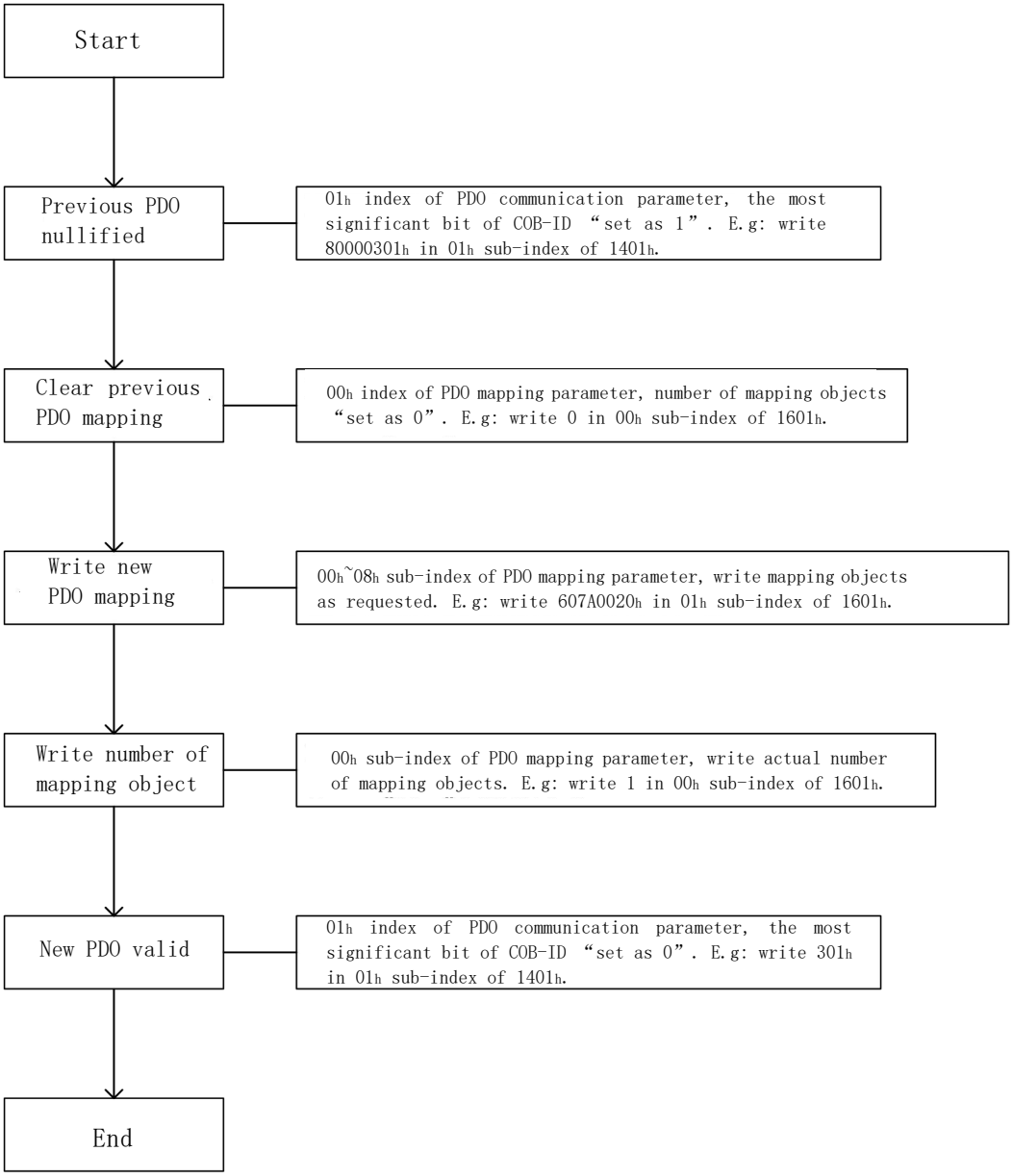
Example for PDO mapping

Examples	bit32～bit16	bit15～bit8	bit7～bit0
	Index of object	Sub-index of object	Data length of object
RPDO1 mapping 1600_h01_h	6040_h	00_h	10_h
RPDO1 mapping 1600_h02_h	6060_h	00_h	08_h
RPDO2 mapping 1601_h01_h	$607A_h$	00_h	20_h
RPDO2 mapping 1601_h02_h	6081_h	00_h	20_h
TPDO1 mapping $1A00_h01_h$	6041_h	00_h	10_h
TPDO1 mapping $1A00_h02_h$	6061_h	00_h	08_h
TPDO2 mapping $1A01_h01_h$	6064_h	00_h	20_h
TPDO2 mapping $1A01_h02_h$	$606C_h$	00_h	20_h

Here is to analyse mapping parameters of RPDO2. Index of mapping object in 1601_h01_h is $607A_h$, sub-index is 00_h and data length is 32 bits; Index of mapping object in 1601_h02_h is 6081_h, sub-index is 00_h and data length is 32 bits; There are 2 objects of RPDO2 mapping so 1601_h00_h is 2. 1601_h has run out of 8 bits so there is no more objects.

Index	Sub-index	Value
1601_h	00_h	2
1601_h	01_h	$607A0020_h$
1601_h	02_h	60810020_h

4.3.3.6.5 PDO mapping configuration process



Take TPDO1 configuration of slave 2 as example:

Step 1, Make the COB-ID of communication parameter 1800_h in TPDO1, that is the most significant bit of value of sub-index 01_h, "set as 1" to nullify previous PDO mapping. Referring to 4.3.3.3, the COB-ID of TPDO1 in slave 2 is 182_h, that is to write 80000182_h into 01_h sub-index of 1800_h to nullify the previous PDO.

Step 2, Set 00_h of mapping parameter $1A00_h$ in TPDO1 as 0 to clear previous PDO mapping.

Step 3, Assign 01_h~08_h of mapping parameter $1A00_h$ in TPDO1 as requested to mapping new object. If it needs to mapping 6041_h and 6061_h into TPDO1, write 60410010_h into sub-index 01_h of $1A00_h$ and write 60610008_h into sub-index 02_h of $1A00_h$.

Step 4, There are two objects according to step 3, so write 2 into sub-index 00_h of mapping parameter $1A00_h$ in TPDO1.

Step 5, Write 182_h into sub-index 01_h of communication parameter 1800_h in TPDO1 to make TPDO2 valid.

4.3.3.7 SYNC Synchronization object

Similar to PDO, the transmission mode is producer——consumer mode. The producer sends synchronous frame and all of the other nodes in CAN network can receive it as consumer without feedback. Only one activated synchronous generator is allowed in a CAN network, which commonly is the NMT host.

4.3.3.7.1 Synchronous generator

Objects related to synchronization are 1005_h(COB-ID SYNC) and 1006_h(communication cycle period). The second highest bit of 1005_h decides whether the synchronous generator will be activated or not. When it is set as 1, that is to write 40000080_h to 1005_h, the generator is activated. When writing 80_h to 1005_h, the generator is shut off. 1006_h is the time interval of producing synchronization objects, using μs as unit.

4.3.3.7.2 Synchronization object transmission frame

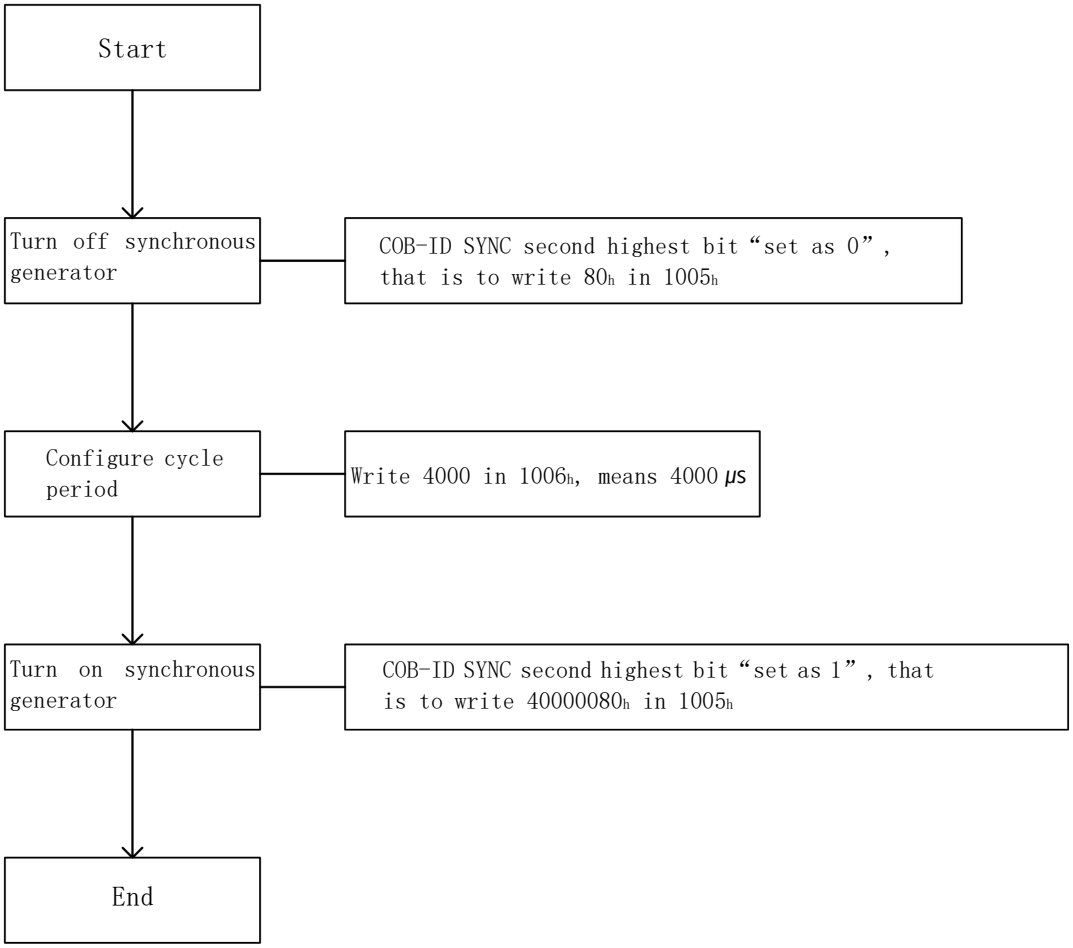
Transmission of synchronization PDO is related to synchronous frame.

For synchronization RPDO, once receiving the PDO, it will update it into application in next SYNC.

For synchronization TPDO, it is divided into synchronous cycle and synchronous non-cycle. Transmission type of synchronous non-cycle is 0. Content of PDO mapping object changes and The TPDO will be send in next SYNC. Transmission type of synchronous cycle is 1~240. Once it is designated SYNC, TPDO will be send whether the data is changed or not.

E.g: PDO1 is type 0, RPDO2 is type 5, TPDO1 is type 0, TPDO2 is type 20. Then RPDO1 and RPDO2 will update the latest PDO data to relevant application in next SYNC once receiving PDO; TPDO1 will send TPDO1 in next SYNC only when the mapping data is changed; TPDO2 will send PDO when it is after 20 SYNC, whether the data changes or not.

4.3.3.7.3 Configuration of synchronous generator



4.3.3.8 EMCY emergency message

When there is a fault from CANopen node, a fraction of emergency message will be sent according to standardization. It follows production——consumption model, so other nodes in CAN network can choose to deal with the faulty after the message is sent. SSTS1A driver only sends it but will not deal with it.

Objects related to emergency message include: 1001_h(Error register)、1003_h(Predefined error domain), 1014_h(COB-ID EMCY), 1015_h(Production prohibition time, similar to this in PDO communication parameter). Please notice that when the most significant bit of 1014_h is "Set as 1", it means deactivate EMCY of nodes, and it means activate EMCY when "Set as 0".

EMCY form

COB-ID	Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7
80_h+Node-ID	Error code		Accordant with 1001_h	Custom error code by manufacture				

When the communication is abnormal, the error code should be accordant with it requested by DS301 and auxiliary byte is zero.

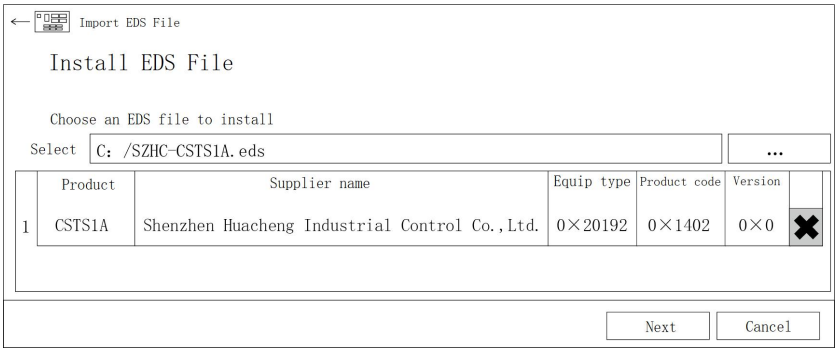
When faults described in DSP402 sub-protocol happen to driver, the error code should be accordant with it requested by DS402 and corresponding to object $603F_h$.

4.3.4 Example for Using CANopen

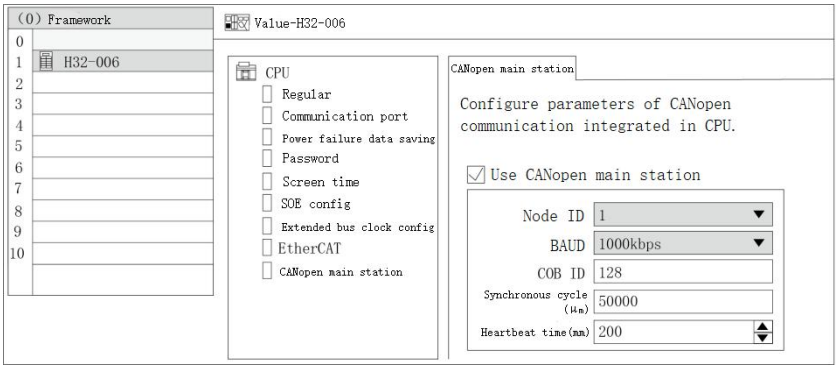
Here is to introduce basic operation of CANopen by taking the process of building CAN network with SSTS1A product from us and PLC with CANopen function as example.

4.3.4.1 Graphical hardware configuration

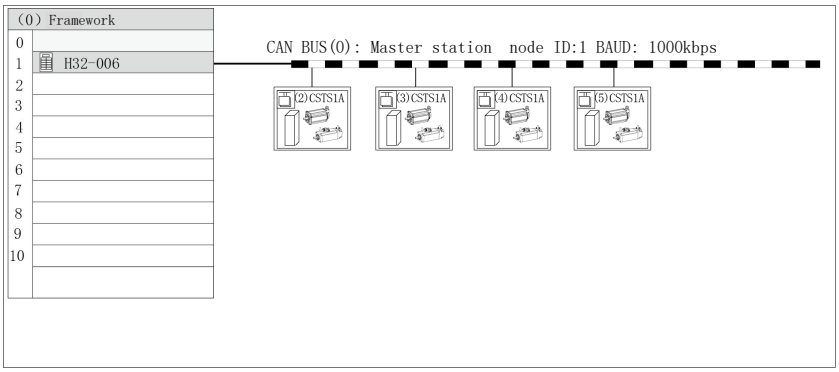
(1) Import file SZHC-CSTS1A.eds



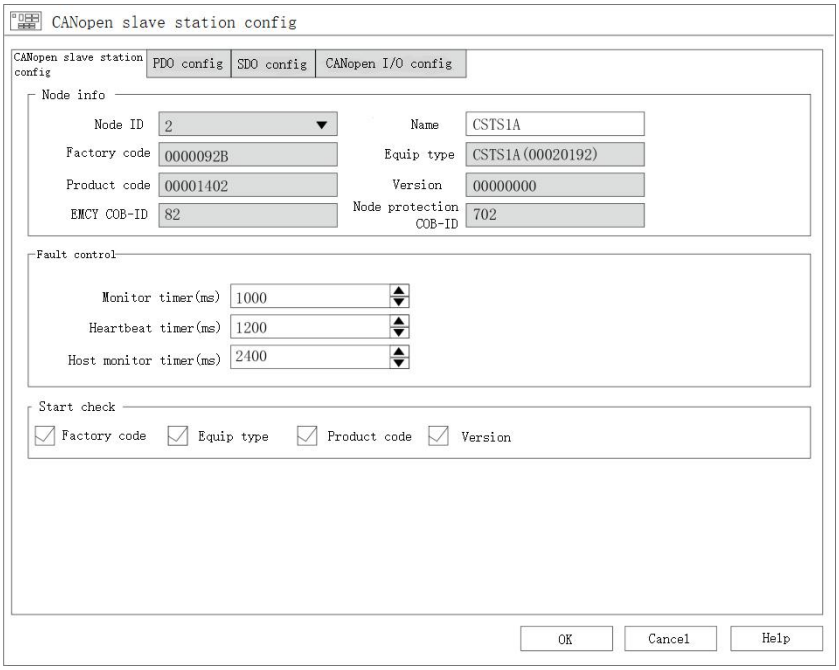
(2) Invoke CANopen function of main station PLC



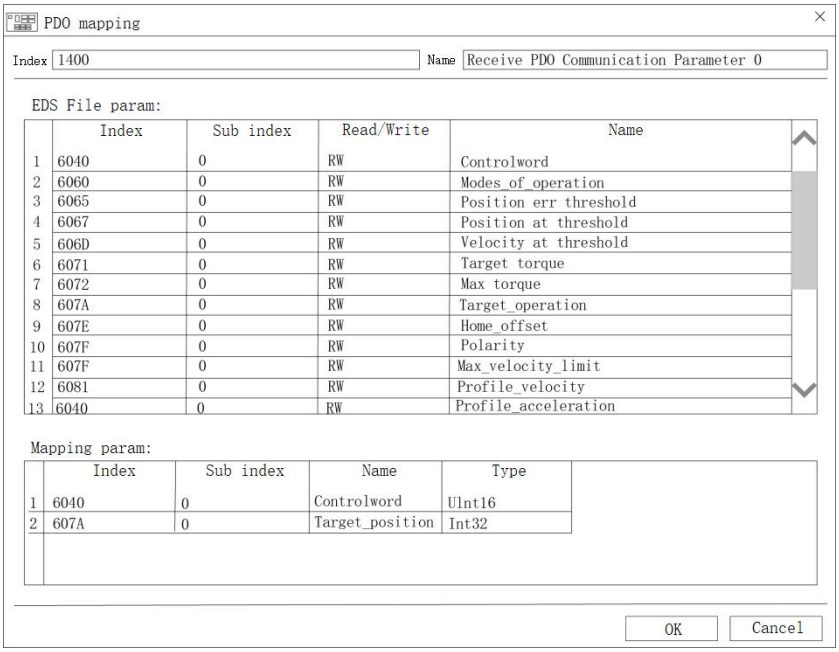
（3）After importing EDS file of CSTS1A, choose SSTS1A servo in PLC slave station selection screen and drag it into CAN BUS.



（4）Double click SSTS1A servo on CAN BUS, and dispose PDO, SDO and other information in CANopen slave station.



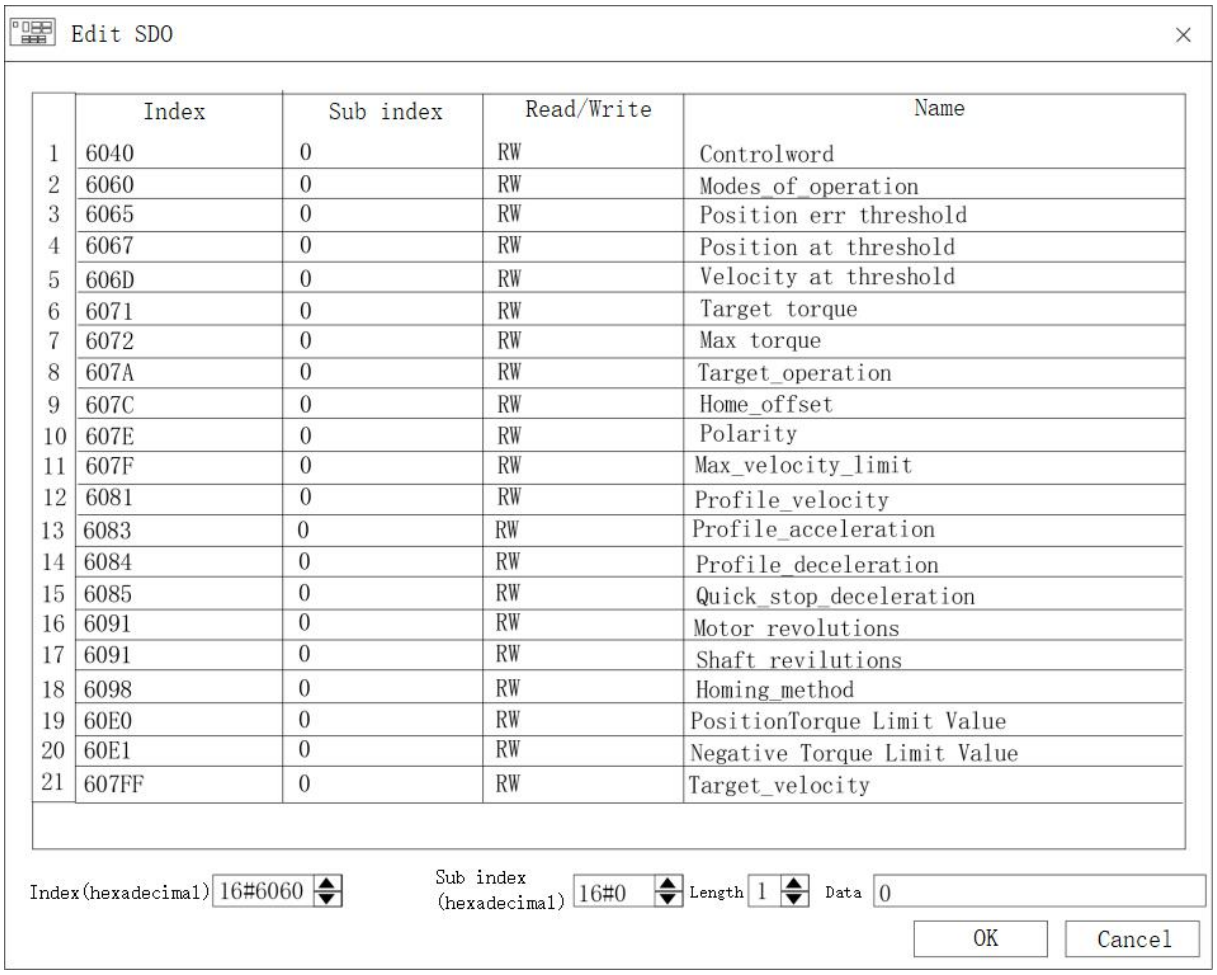
PDO mapping



	Index	Sub index	Read/Write	Name
1	6040	0	RW	Controlword
2	6060	0	RW	Modes_of_operation
3	6065	0	RW	Position err threshold
4	6067	0	RW	Position at threshold
5	606D	0	RW	Velocity at threshold
6	6071	0	RW	Target torque
7	6072	0	RW	Max torque
8	607A	0	RW	Target_operation
9	607E	0	RW	Home_offset
10	607F	0	RW	Polarity
11	607F	0	RW	Max_velocity_limit
12	6081	0	RW	Profile_velocity
13	6040	0	RW	Profile_acceleration

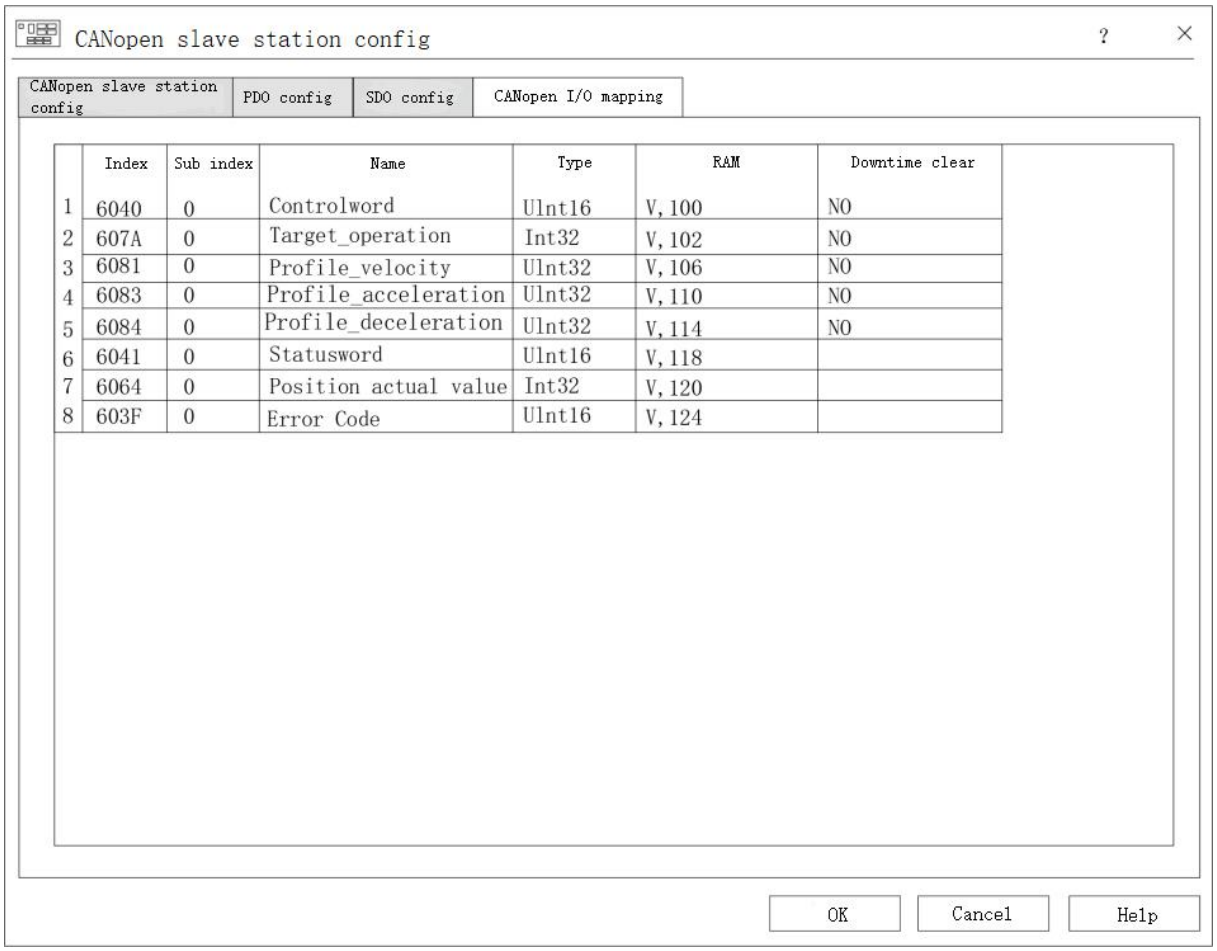
	Index	Sub index	Name	Type
1	6040	0	Controlword	Ulnt16
2	607A	0	Target_position	Int32

SDO configuration



	Index	Sub index	Read/Write	Name
1	6040	0	RW	Controlword
2	6060	0	RW	Modes_of_operation
3	6065	0	RW	Position err threshold
4	6067	0	RW	Position at threshold
5	606D	0	RW	Velocity at threshold
6	6071	0	RW	Target torque
7	6072	0	RW	Max torque
8	607A	0	RW	Target_operation
9	607C	0	RW	Home_offset
10	607E	0	RW	Polarity
11	607F	0	RW	Max_velocity_limit
12	6081	0	RW	Profile_velocity
13	6083	0	RW	Profile_acceleration
14	6084	0	RW	Profile_deceleration
15	6085	0	RW	Quick_stop_deceleration
16	6091	0	RW	Motor revolutions
17	6091	0	RW	Shaft revilutions
18	6098	0	RW	Homing_method
19	60E0	0	RW	PositionTorque Limit Value
20	60E1	0	RW	Negative Torque Limit Value
21	607FF	0	RW	Target_velocity

CANopen mapping



	Index	Sub index	Name	Type	RAM	Downtime clear
1	6040	0	Controlword	Ulnt16	V,100	NO
2	607A	0	Target_operation	Int32	V,102	NO
3	6081	0	Profile_velocity	Ulnt32	V,106	NO
4	6083	0	Profile_acceleration	Ulnt32	V,110	NO
5	6084	0	Profile_deceleration	Ulnt32	V,114	NO
6	6041	0	Statusword	Ulnt16	V,118	
7	6064	0	Position actual value	Int32	V,120	
8	603F	0	Error Code	Ulnt16	V,124	

(5) Compile hardware configuration correctly

The hardware configuration finished disposing SDO and PDO for main and slave station. CANopen bus is completed.

4.3.4.2 Instruction hardware configuration

Some master stations do not fit graphical hardware configuration. It needs to dispose slave station by single SDO command. Following are examples for hardware configuration.

No.	SDO message	Byte 1	Byte 3,2	Byte 4	Byte 6,5	Byte 8,7
	Description	Command code	Index	Sub-index	Low bit	High bit
1	Consumer heartbeat time:$1016_h01_h=14_h$	23_h	1016_h	01_h	0014_h	0000_h
2	RPD01 transmission:$1400_h02_h=00_h$	$2F_h$	1400_h	02_h	0000_h	0000_h
3	RPD01 null: $1400_h01_h=80000201_h$	23_h	1400_h	01_h	0201_h	8000_h
4	RPD01 clear mapping: $1600_h00_h=00_h$	$2F_h$	1600_h	00_h	0000_h	0000_h
5	RPD01 write mapping: $1600_h01_h=60400010_h$	23_h	1600_h	01_h	0010_h	6040_h
6	RPD01 new mapping number: $1600_h00_h=01_h$	$2F_h$	1600_h	00_h	0001_h	0000_h
7	RPD01 valid: $1400_h01_h=00000201_h$	23_h	1400_h	01_h	0201_h	0000_h
8	RPD02 transmission:$1401_h02_h=00_h$	$2F_h$	1401_h	02_h	0000_h	0000_h
9	RPD02 null: $1401_h01_h=80000301_h$	23_h	1401_h	01_h	0301_h	8000_h
10	RPD02 clear mapping: $1601_h00_h=00_h$	$2F_h$	1601_h	00_h	0000_h	0000_h
11	RPD02 write mapping: $1601_h01_h=607A0020_h$	23_h	1601_h	01_h	0020_h	$607A_h$
12	RPD02 write mapping: $1601_h02_h=60810020_h$	23_h	1601_h	02_h	0020_h	6081_h
13	RPD02 new mapping number: $1601_h00_h=02_h$	$2F_h$	1601_h	00_h	0002_h	0000_h
14	RPD02 valid: $1401_h01_h=00000301_h$	23_h	1401_h	01_h	0301_h	0000_h
15	RPD03 transmission:$1402_h02_h=00_h$	$2F_h$	1402_h	02_h	0000_h	0000_h
16	RPD03 null: $1402_h01_h=80000401_h$	23_h	1402_h	01_h	0401_h	8000_h
17	RPD03 clear mapping: $1602_h00_h=00_h$	$2F_h$	1602_h	00_h	0000_h	0000_h
18	RPD03 write mapping: $1602_h01_h=60830020_h$	23_h	1602_h	01_h	0020_h	6083_h
19	RPD03 write mapping: $1602_h02_h=60840020_h$	23_h	1602_h	02_h	0020_h	6084_h
20	RPD03 new mapping number: $1602_h00_h=02_h$	$2F_h$	1602_h	00_h	0002_h	0000_h
21	RPD03 valid: $1402_h01_h=00000401_h$	23_h	1402_h	01_h	0401_h	0000_h
22	TPD01 transmission:$1800_h02_h=00_h$	$2F_h$	1800_h	02_h	0000_h	0000_h
23	TPD01 null: $1800_h01_h=80000181_h$	23_h	1800_h	01_h	0181_h	8000_h
24	TPD01 clear mapping: $1A00_h00_h=00_h$	$2F_h$	$1A00_h$	00_h	0000_h	0000_h
25	TPD01 write mapping: $1A00_h01_h=60410010_h$	23_h	$1A00_h$	01_h	0010_h	6041_h
26	TPD01 write mapping: $1A00_h02_h=60610008_h$	23_h	$1A00_h$	02_h	0008_h	6061_h
27	TPD01 write mapping: $1A00_h03_h=603F0010_h$	23_h	$1A00_h$	03_h	0010_h	$603F_h$
28	TPD01 new mapping number: $1A00_h00_h=03_h$	$2F_h$	$1A00_h$	00_h	0003_h	0000_h
29	TPD01 valid: $1800_h01_h=00000181_h$	23_h	1800_h	01_h	0181_h	0000_h
30	TPD02 transmission:1801_h02h=00_h	$2F_h$	1801_h	02_h	0000_h	0000_h

31	TPDO2 null: $1801_h01_h=80000281_h$	23_h	1801_h	01_h	0281_h	8000_h
32	TPDO2 clear mapping: $1A01_h00_h=00_h$	$2F_h$	$1A01_h$	00_h	0000_h	0000_h
33	TPDO2 write mapping: $1A01_h01_h=60640020_h$	23_h	$1A01_h$	01_h	0020_h	6064_h
34	TPDO2 write mapping: $1A01_h02_h=606C0020_h$	23_h	$1A01_h$	02_h	0020_h	$606C_h$
35	TPDO2 new mapping number: $1A01_h00_h=02_h$	$2F_h$	$1A01_h$	00_h	0002_h	0000_h
36	TPDO2 valid: $1801_h01_h=00000281_h$	23_h	1801_h	01_h	0281_h	0000_h
37	Set servo mode: $6060_h00_h=01_h$	$2F_h$	6060_h	00_h	0001_h	0000_h
38	Turn off synchronous generator: $1005_h00_h=00000080_h$	23_h	1005_h	00_h	0080_h	0000_h
39	Write synchronous cycle: $1006_h00_h=00003A98_h$	23_h	1006_h	00_h	$3A98_h$	0000_h
40	Turn on synchronous generator: $1005_h00_h=40000080_h$	23_h	1005_h	00_h	0080_h	4000_h

4.3.4.3 Main station action control program

It differs a lot to write action control program for various main stations. It mainly contents NMT status switch and action command planning. There will be no further discussion.

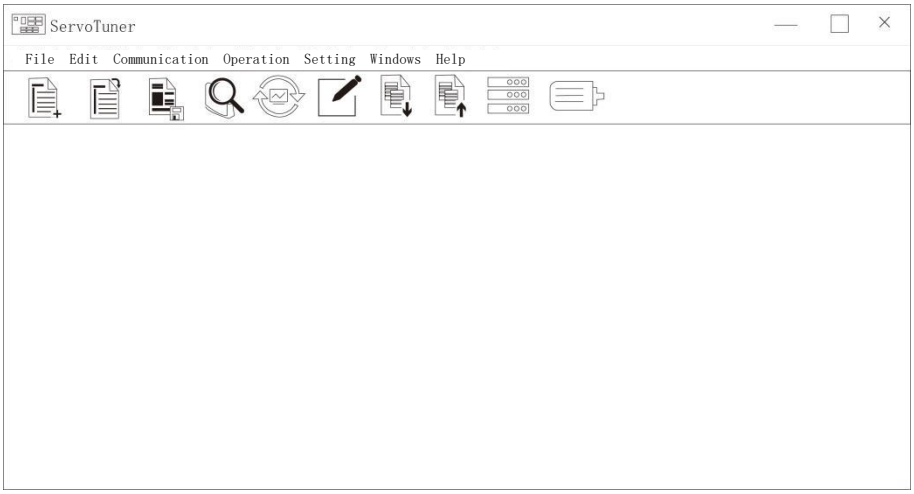
4.4 ServoTuner Upper Computer Software

ServoTuner upper computer software, as a servo master port in PC end, communicates with servo slave station by serial port. It fits standard Modbus RTU protocol and can connect with servo through USB to RS485 transmitter. With ServoTuner, users can execute JOG servo test run, read/write servo parameters and collect servo running curves.

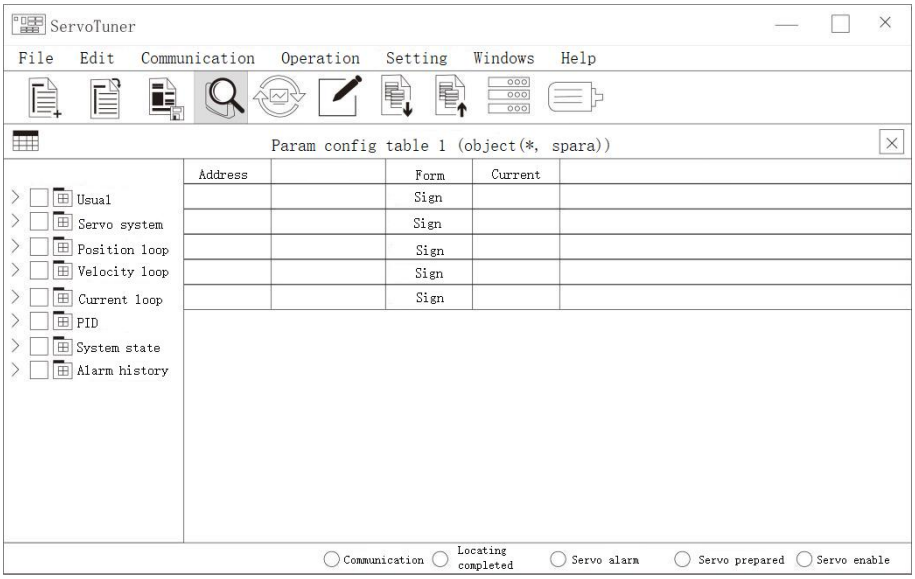
4.4.1 Read/Write Servo Parameters

Prepare USB to 485 communication transmitter. (Remember to install relative program) Following are steps of setting servo parameters by ServoTuner upper computer software.

Step 1: Connect servo with PC to let servo power on through USB to 485 communication transmitter. Click ServoTuner.ex to enter main screen of servo upper computer software.(Shown in P1) Create new parameter table, same as using Word software.(Shown in P2)

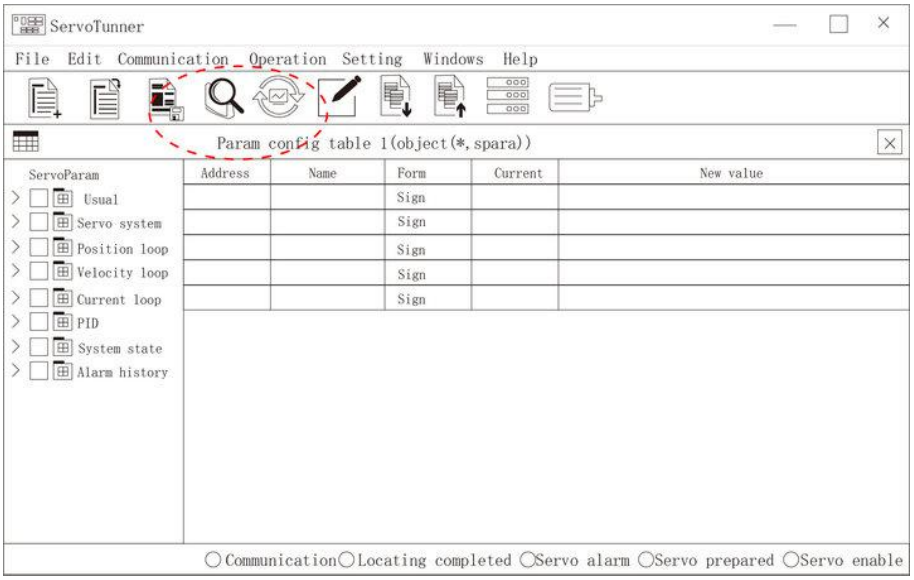


P1 Main screen of servo upper computer software



P2 Create new parameter table

Step 2: Click “magnifier” searching button in P3, and the upper computer will automatically find servo connected. Check grouping parameter on the left or write parameter address in “address bar” in the table to read/write servo param.(Shown in P3)

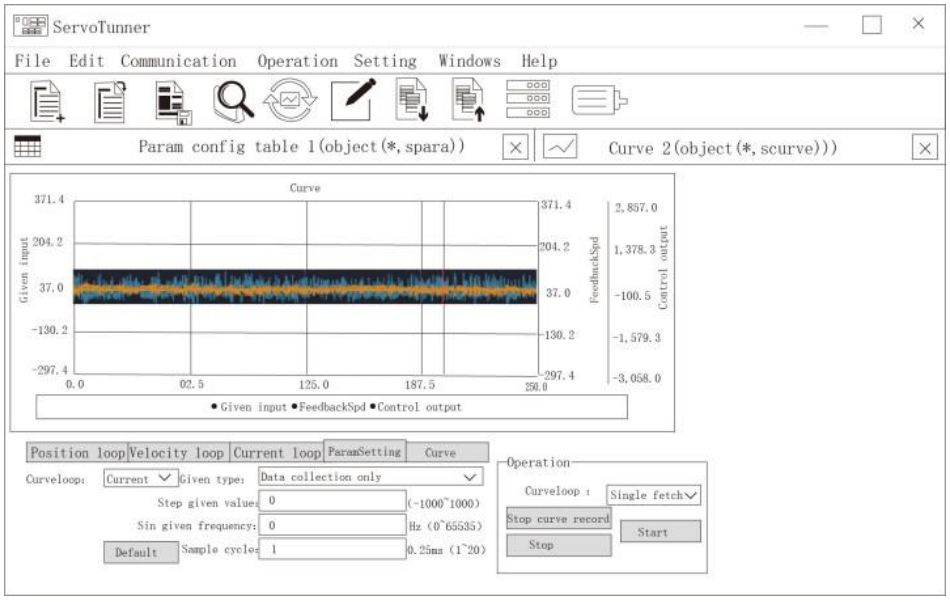


P3 Read/write servo param through param table

Step 3: Click "Start state monitoring" button to read parameters of servo driver in real time.

4.4.2 Collect Servo Curves

Create curve chart, same as new parameter table. Collect curves of parameters related to current loop, velocity loop and position loop.



P4 Collect servo dynamic curve by curve figure

Please refer to servo upper computer software document for more functions.

4.5 Set Motor Code by ServoTuner

It needs to set motor code to match the driver before use servo system because it can drive servo motor in various power levels and voltage levels. For example, the SSTS1A100 driver supports 24V/100W/200W,48V/100W/200W/400W and other motors. After getting the sole registered motor code(P182), motors can be used normally. Registered motors are shown in the following list.

Motor model	Motor spec	P182
HC7J-040130F1	24v/100w/6.5A/0.32Nm/3000rpm/2500Line	1
HC7J-040130D1	48v/100w/3.5A/0.32Nm/3000rpm/2500Line	2
HC7J-060230F1	24v/200w/11.5A/0.64Nm/3000rpm/2500Line	3
HC7J-060230D1	48v/200w/6.5A/0.64Nm/3000rpm/2500Line	4
HC7J-060230E1	36v/200w/7.5A/0.64Nm/3000rpm/2500Line	5
HC7J-060430D1	48v/400w/11A/1.27Nm/3000rpm/2500Line	6
HC7J-060430F1	24v/400w/20A/1.27Nm/3000rpm/2500Line	31
HC7J-060430E1	36v/400w/14.5A/1.27Nm/3000rpm/2500Line	32
HC7J-080830D1	48v/750w/19.5A/2.4Nm/3000rpm/2500Line	33
HC7J-080830F1	24v/750w/40A/2.4Nm/3000rpm/2500Line	61
HC7J-131025D1	48v/1.0kw/25A/3.8Nm/2500rpm/2500Line	62
HC7G-131515D1	48v/1.5kw/45A/10Nm/1500rpm/2500Line	63
HC7G-131520D1	48v/1.5kw/40A/7.4Nm/2000rpm/2500Line	64
HC7C-131830D1	48v/1.8kw/40A/5.7Nm/3000rpm/2500Line	65
HC7G-132020D1	48v/2.0kw/50A/10Nm/2000rpm/2500Line	66
HC7J-132025D1	48v/2.0kw/58A/7.7Nm/2500rpm/2500Line	67

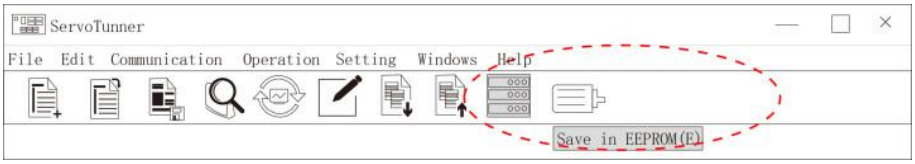
There is a default motor code when servo drivers leave the factory. Following steps show how to modify it:

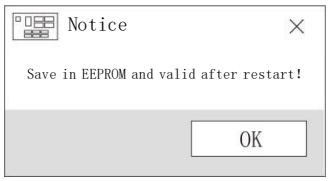
Step 1:

(1) P282 writes command “16384”;

(2) P182 writes motor code to be set;

(3) Click “Save in EEPROM(E)” and wait for the dialog box “Save in EEPROM and valid after restart” popping out. Then click “OK”;

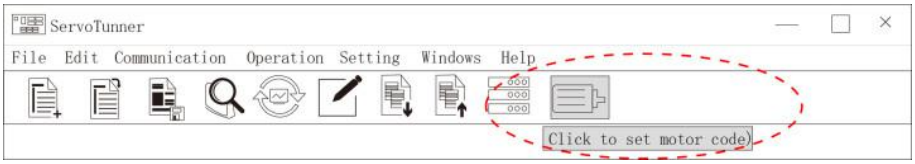


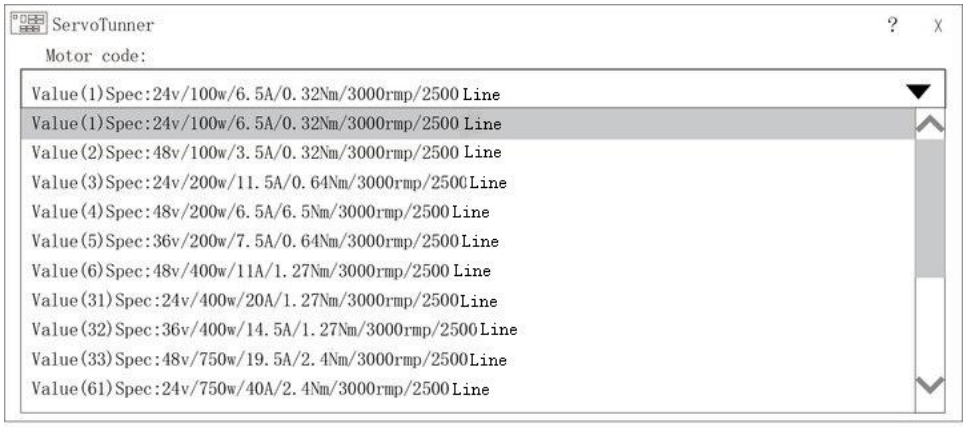


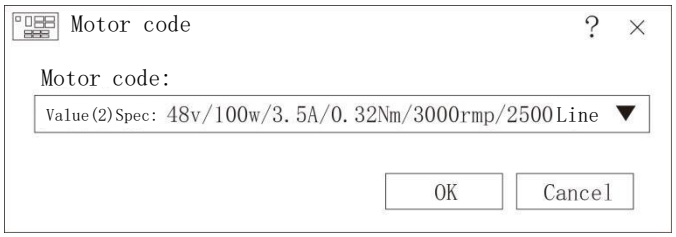
(4) New motor code is effective when the driver is power on again after turned off.

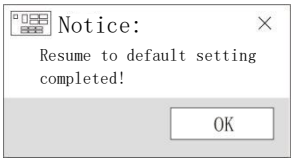
Step 2:

(1) Click “Click to set motor code” and the motor code dialog box will pop out. Choose and click “OK”. When there comes the notice “Resume to default setting completed”, click “OK”;









(2) New motor code is effective when the driver is power on again after turned off.

Under-voltage node, discharge node and over-voltage node of servo system in different voltage level are shown in the table below.

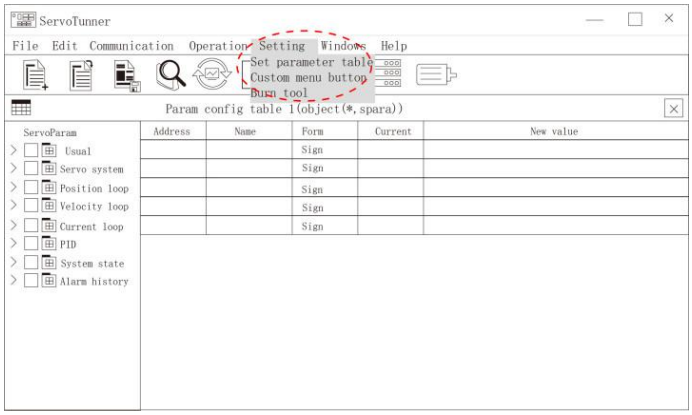
Voltage level of servo system	Under-voltage alarm node	Energy braking voltage absorb node (with external braking resistor)	Over-voltage alarm node
24v	16v	30v	36v
36v	30v	45v	48v
48v	40v	55v	60v
60v	40v	70v	80v

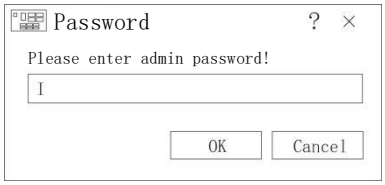
4.6 Update Servo Program by ServoTuner

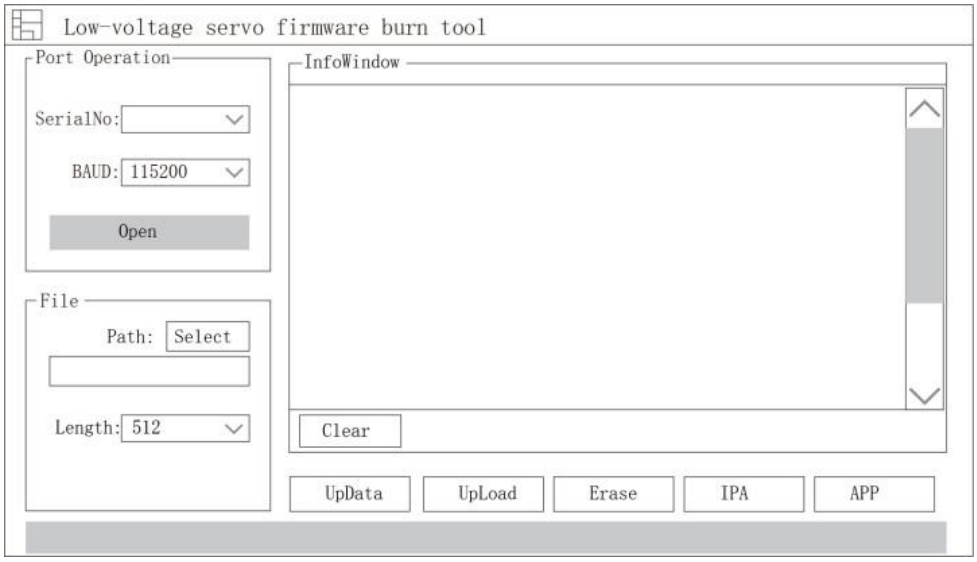
Steps to burn:

(1) Turn all dial switches of driver down (turn to "ON"), the driver is power on;

(2) Click"Setting"—"Burn tool"—enter password"16384"—"Low-voltage servo firmware burn tool";







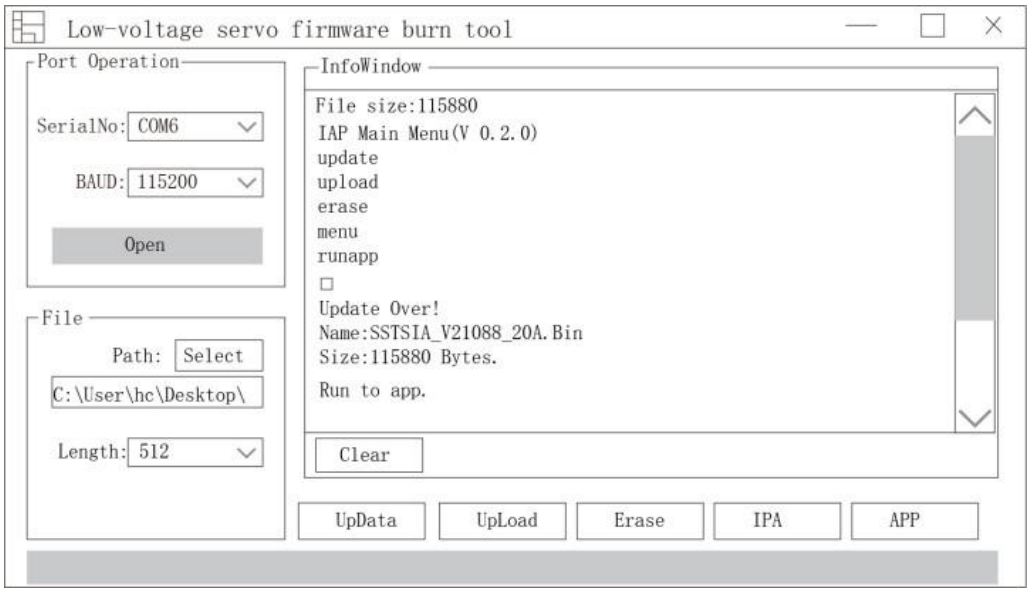
(3) Click “Open” in port operation area—Click “Select” in file setting area—“V210XX.bin”;

Note:

V21072.bin is 10A, 50A program

V21072_20A.bin is 20A program

(4) Click “Enter IAP Menu” —Click “UpDate” —Wait for burning;



(5) Turn dial switches back(up) after updating and power on again.

Please contact for technical support in case of repeated failures.

Chapter 5 Servo Alarm Diagnosis and Solutions

Servo driver alarm description and solution

Green lamp	Red lamp	Alarm type	Description	Solutions
Flash	Off	No alarm	None	None
Quick flash	On	Over-current ★	Triggered when instant current is 4.5 times of max current of motor	Check whether the driver is broken;Check whether the motor is broken; Check the wiring of motor; Check whether the driver matches the motor.
Slow flash	On	Over-heat★	Triggered when the MOS tube is over-heated	Temperature of environment is too high; Heat elimination is poor; Servo has been over-load for too long.
Quick flash	Quick flash	Encoder faulty★	Triggered when encoder is disconnected or electric angle is abnormal	Check whether the encoder line is loose; Check whether the wiring of encoder is loose; Check whether the encoder is disconnected.
Slow flash	Slow flash	EEPROM error ★	Triggered when EEPROM writes and reads abnormally	Check whether motor parameters are set correct; Try to restore the factory settings.
Off	Quick flash	Over-load	Triggered when motor torque is larger than over-load level and keeps for a while	Check over-load setting; The actual load of motor is too large; Motor is not well connected.
Off	Slow flash	Over-speed	Triggered when motor rotates faster than over-speed level	Check over-speed setting; PID parameter is set unreasonable.
On	Quick flash	Over-voltage	Triggered when generatrix voltage is higher than standard	On and off too frequently; Check whether the braking unit is reasonable.
On	Slow flash	Under-voltage	Triggered when generatrix voltage is lower than standard	Check whether input power is on; Check whether input voltage of servo end is up to standard; Evaluate whether the power supply is appropriate.
Slow flash	Quick flash	Excessive position deviation	Triggered when position following deviation is larger than over-deviation level.	Check over-deviation setting; Adjust PID parameter if the actual load is too large.
Quick flash	Slow flash	Travel limit alarm	Triggered when travel limit function is set different from limit signal.	Check parameters; Check external limit signal.
On	On	CAN communication faulty	CAN communication alarm	Check wiring and master station.

(1) Types of malfunction with ★ in the list cannot be removed by the upper system. It needs to check the situation of the device and power on again.
(2) Types of malfunction without ★ in the list can be removed by the upper system.

Chapter 6 Warranty Terms

6.1 General Rules

We, Shenzhen Huacheng Industrial Control Co.,LTD, adhere strictly to relevant laws to formulate post-sale service rules.

6.2 Servo Warranty Period

The warranty period is one year after purchase. For motors with brake, the standard is that acceleration/deceleration times of axis is not beyond limit. We provide free maintenance for malfunction not caused by misuse or vandalism within warranty period. For malfunction within warranty period caused by following reasons, there will be a certain fee:

(1) Malfunction or damage caused by operations in contravention of the user manual;
(2) Malfunction or damage caused by disassembly or converting privately;
(3) Damage caused by force majeure (earthquake, volcano, typhoon, tsunami, flood, mud avalanche, thunderstorm and etc.);
(4) No valid purchase voucher;
(5) Serial number on the shell does not match the number inside.

6.3 Servo Warranty Process

Warranty process:

(1) Please fill in Maintenance List and post it to our maintenance department if there is any malfunction or damage.
(2) Maintenance cost refers to the Maintenance Price List.
(3) The final explanation right of these terms is reserved by Shenzhen Huacheng Industrial Control Co.,LTD.

HUACHENG

Core supplier in the field of industrial control

Shenzhen Huacheng Industrial Control Co.,LTD

0755-26417678/27470348

Fax: （86）075526417678

Official Website: www.hc-system.com

Address: 8/F,Zhengqilong Building No.108,Gushu 1st Road,Baoan.Shenzhen