Kinco步科

User Manual

Kinco iWMC integrated servo wheel

Shenzhen Kinco Electric Co., LTD

en.kinco.cn

Version Change Record

Update Date	Version
2022.09	V1.0
2022.10	V1.1
2022.12	V1.2
2023.02	V1.3
2023.04	V1.5

Preface Product validation

Thank you for using the servo products by kinco !

This manual iWMC integrated servo wheel is a fully integrated design of the power module launched by kinco . The four modules of wheel, reducer, servo motor and driver are integrated, which can optimize the structure of the car, simplify the installation operation steps, shorten the installation time of the vehicle, and use dual power supply design for the drive, making the system more safe and reliable. This product is suitable for the walking axis of mobile robot under 600 kg load.

Please read the manual carefully and follow the operation requirements in the manual, which will help you correctly set up the drive to achieve the best drive performance.

The accessories of Kinco drive series are different for different models, so it is recommended that you confirm the product.

Confirmation item	Explanation
Is it consistent with the model you ordered?	Please refer to the motor and driver nameplate information to confirm that the motor model, driver model, etc. is the same as the model you ordered.
Is the motor wiring correct?	Please check whether the motor wiring model is consistent with the order.
Is there any damage to the product appearance?	Please confirm whether the product was damaged during transportation.
Are the product accessories complete?	Please make sure that the various terminals of the driver are complete, and that the motor oil seals and keys are complete.

List of drive accessories

Product model	Parts and quantity
iWMC10409-02222-A165-MBDT	Certificate*1、Service information*1、Extended line*3

If you have any problem with any of the above, please contact us or your supplier.

Important Notes

Please carefully read and follow the requirements in this manual, which will help you set up and operate the drive correctly and maximize the performance of the drive. Please be aware of the contents of the warning and strictly follow the requirements, otherwise it may cause dangerous situation.

Warning

- Do not install the machine without spare parts or damaged appearance.
- Please install in a well-ventilated, dry and dust-free place without grinding fluid, oil mist, iron powder, and chips, and surrounded by non-flammable products.
- Do not stress the body of the servo wheel when installing or removing it, and ensure that each fixation is locked.
- Avoid any foreign matter entering the servo wheel. Electrically conductive foreign matter or combustible foreign matter such as screws and metal chips entering the servo wheel may cause fire and electric shock. For safety reasons, please do not use products with damage or damaged parts.
- Do not use gasoline, thinner, alcohol, acid and alkaline detergent to avoid discoloration or damage to the shell.
- Before connecting cables, ensure that the input power is off.
- Do not insert or remove terminals directly when the power is on.
- Please store and transport in original packaging, which provides adequate protection against routine problems.
- Ensure that this document is available to design engineers, installers, and personnel responsible for commissioning machines or systems that use the product.
- Please consider the legal provisions applicable to your destination, include:
  - —regulations and standards
  - —Test organization and insurance company regulations
  - —National specification
- Please ensure that the product is not subjected to more than permitted burdens during transportation and storage, including:
  - —mechanical load
  - —The temperature is not allowed
  - —Water
  - —corrosive gas
- Please strictly follow the instructions and warnings in this document when using the product!

Content

Preface Product validation **1**

Important Notes **2**

Chapter 1 Product model description and installation precautions **5**

1.1 Product model and nameplate description 5

1.1.1 *iWMC integrated servo wheel product model* *5*

1.2 iWMC integrated servo wheel installation and cautions 6

1.2.1 *Mounting Dimensions Figure* *6*

1.2.2 *Requirements for Operators* *7*

1.2.3 *Electrical requirement* *7*

1.2.4 *Environment Requirements* *7*

Chapter 2 System Interface And Wiring **8**

2.1 External wiring diagram 8

2.1.1 *iWMC integrated servo wheel external wiring diagram* *8*

2.1.2 *iWMC Integrated servo wheel brake resistance and fuse reference specifications* *8*

2.2 Interface Definition 9

2.2.1 *iWMC Integrated servo wheel integrated terminal* *9*

2.2.2 *Power port* *10*

2.2.3 *Brake resistance port* *11*

2.2.4 *Terminal specification* *11*

Chapter 3 KincoServo+ software introduction **12**

3.1 Fast start 12

3.1.1 *Language configuration* *12*

3.1.2 *Opening and saving project files* *12*

3.1.3 *Start communication* *13*

3.1.4 *Node ID and baud rate* *13*

3.1.5 *Object (add，delete，help)* *13*

3.2 Initialize, save and reboot 14

3.3 Firmware update 15

3.4 Read/write driver configurations 15

3.4.1*Read setting from controller* *15*

3.4.2 *Write settings to controller* *16*

3.5 Speed mode introduction 17
3.6 Digital IO functions 18
3.6.1*Digital input* *19*
3.6.2 *Digital output* *20*
3.7 Scope 21
3.8 Error and error history 22
Chapter 4 Performance tuning 25
4.1Tuning of velocity loop 26
4.2 Tuning of position loop 28
4.3 Factors which influence tuning results 30
Chapter5 Alarm Exclusion 31
Chapter 6 RS485 communication 35
6.1RS485 Wiring instructions 35
6.2 RS485 Communication parameters 35
Chapter 7 CANopen 36
7.1 CANopen communication protocol 36
7.2 CANopen bus communication hardware instroduction 37
7.2.1 CANopen bus communication software introduction 38
7.2.1.1 EDS introduction 38
7.2.1.2 SDO introduction 38
7.2.1.3 PDO introduction 41
7.2.1.4 PDO COB-ID introduction 41
7.2.1.5 Protection mode/Monitoring Type 44
7.2.1.6 Boot-up process 45
7.2.1.7 Emergency message introduction 46
7.2.2 CANopen communication Settings 48
Chapter 8 Others 50
8.1 Rubber wheel replacement 50
8.2 Size drawing of rubber coated wheel rim 51
8.3 Extension cable drawing 52

Chapter 1 Product model description and installation precautions

1.1 Product model and nameplate description

1.1.1 iWMC integrated servo wheel product model

Model	Specification
iWMC10409-02222-A165-MBDT	With reducer , With brake , With 165 wheel diameter covered rubber wheel，Standard extension connector
iWMC10409-02222-A165-MADT	With reducer , Without brake , With 165 wheel diameter covered rubber wheel，Standard extension connector
iWMC10409-02222-0000-MBDT	With reducer , With brake , Without 165 wheel diameter covered rubber wheel，Standard extension connector
iWMC10409-02222-0000-MADT	With reducer , Without brake ,Without 165 wheel diameter covered rubber wheel，Standard extension connector

iWMC - 104 09 - 022 22 - A 165 – M B D T - XX

Identification	Series model
iWM	WM: three-in-one servo wheel without drive
	iWM: four-in-one servo wheel with drive
C	Generation C product

Identification	Motor stator outside diameter
104	104mm

Identification	Reducer speed ratio
09	9 speed ratio (Suitable for 600 kg)
15	15 speed ratio (Suitable for 1000 kg)
00	No reducer

Identification	Torque
022	22Nm
054	54Nm

Identification	Wheel speed
22	22*10rpm
17	17*10rpm
00	00*10rpm

Identification	Wheel wrapping material/Pattern type
A	Polyurethane/Raised cross-section,Diamond-shaped pattern
0	No rubber wheel

Identification	Special tailor-made
XX	The default starting point is 00

Identification	Brake
T	X:Directly connect the drive and controller connector
	T: Standard adapter by kinco

Identification	Voltage level
D	DC48V

Identification	Connector type/wire length
B	B:With brake
	A:Without brake

Identification	Encoder type
M	M:Single loop communication type magnetoelectric encoder(Normal)
	W:2500PPR incremental magnetoelectric encoder(Normal)

Identification	Wheel outside diameter
165	165mm
180	180mm
000	No rubber wheel

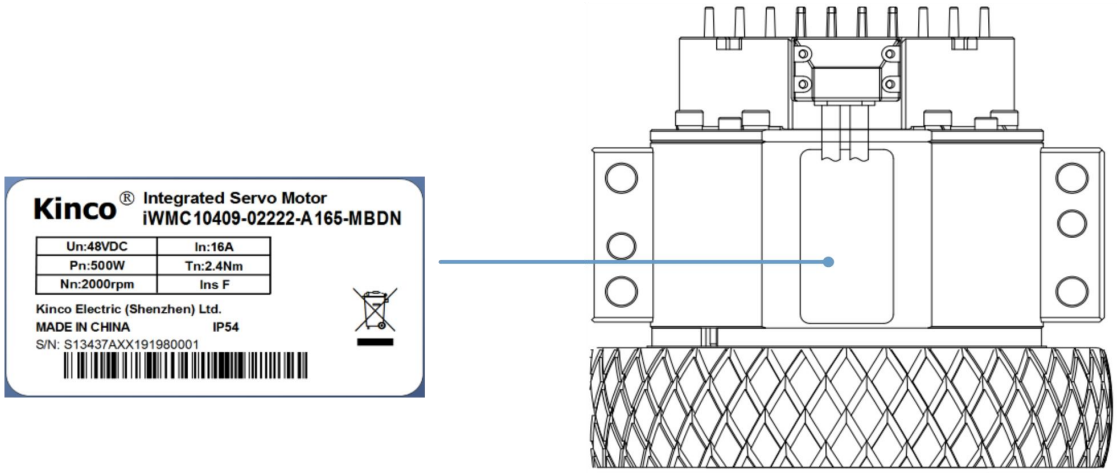


Figure 1-1 iWMC integrated servo motor model and nameplate information

1.2 iWMC integrated servo wheel installation and cautions

1.2.1 Mounting Dimensions Figure

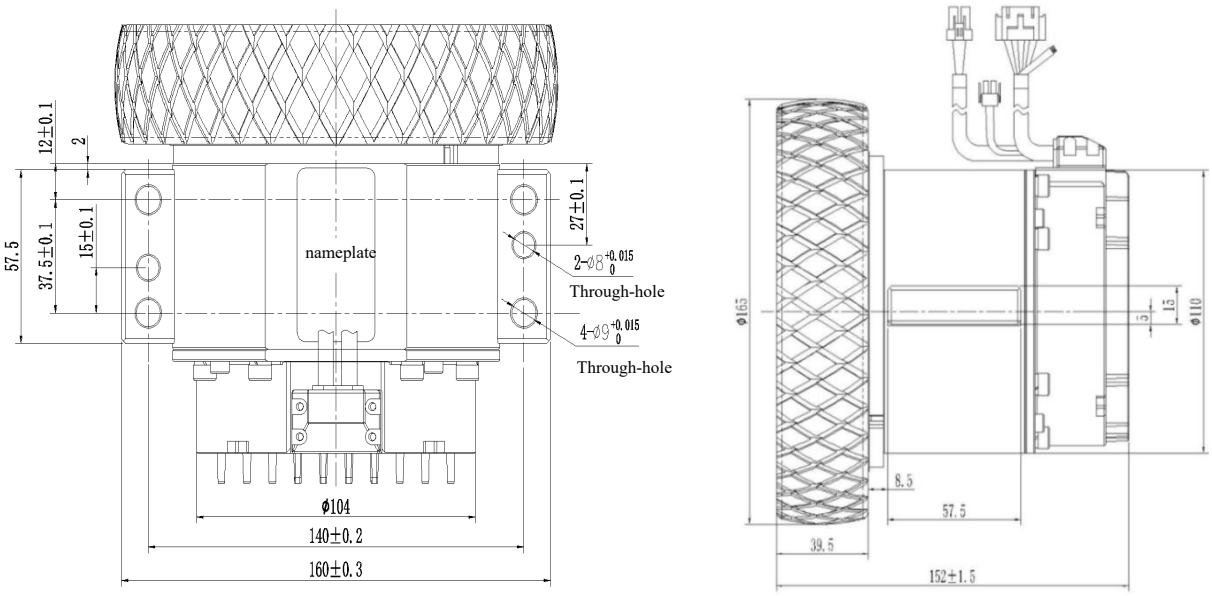


Figure 1-2 iWMC integrated servo wheel dimensions

1.2.2 Requirements for Operators

This product should only be operated by electrical engineers who are familiar with:

—Installation and operation of electrical control systems

—Applicable regulations for operating safety engineering systems

—Applicable provisions for accident protection and occupational safety

—Product documentation

1.2.3 Electrical requirement

Driver parameter	Minimum value	typical value	Maximum value
Input voltage	24V	48V	60V
Brake control voltage	—	24V	—
Overvoltage alarm point	—	68V	—
Undervoltage alarm point	—	18V	—

1.2.4 Environment Requirements

Environment	Condition
Operating temperature	0℃～40℃
Operating humidity	5～95%RH（no condensation）
Storage Temperature	-10℃～70℃（no freeze）
Storage humidity	Below 90%RH（no condensation）
Degree of protection	IP54
Assembly requirement	Indoor no sunlight, no corrosive gas, no flammable gas, no oil and gas, no dust, dry lockable (such as electric cabinet)
Installation Method	Install vertically or horizontally
Atmospheric pressure	86kpa~106kpa
Altitude	The rated working altitude is below 1000 meters, and when the working altitude is above 1000 meters, the derating should be reduced by 1.5% for every 100 meters rise, and the maximum working altitude is 4000 meters

Chapter 2 System Interface And Wiring

2.1 External wiring diagram

2.1.1 iWMC integrated servo wheel external wiring diagram

The servo wheel uses two independent power supplies, namely 24V logic power (12pin terminal 1, 2 pins) and 48V power supply (2pin power terminal), and needs to be connected to two power servo wheels at the same time to work.

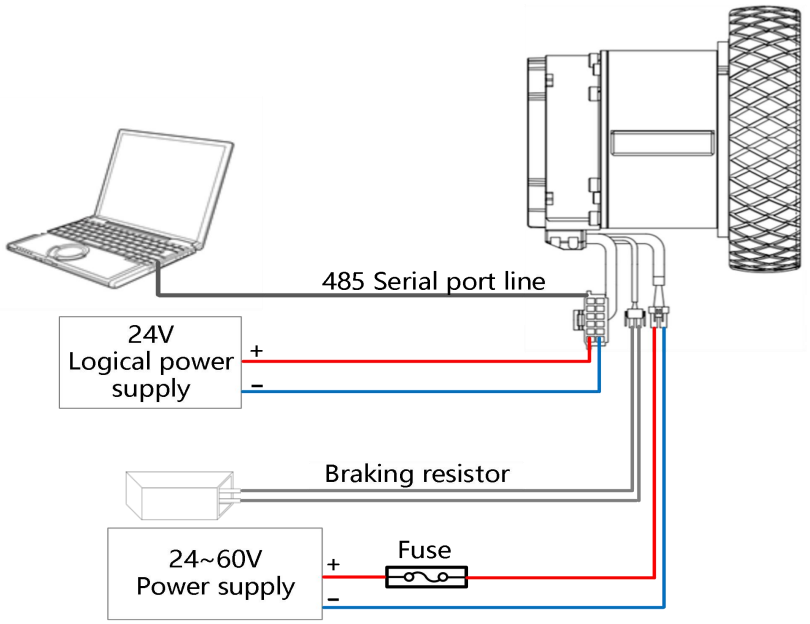


Figure 2-1 Servo wheel motor external wiring diagram

2.1.2 iWMC Integrated servo wheel brake resistance and fuse reference specifications

Table 2-1 Braking resistor reference specification

Braking resistor type	Braking resistor resistance value [Ω]	Braking resistor power [W]	Braking resistor withstand pressure [VDC]
T-10R-100	10	100	500

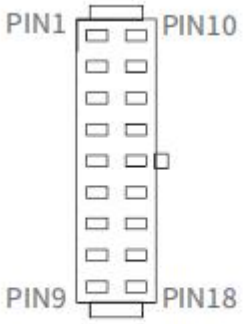
Table 2-2 Fuse reference specification

Model	Drive power (units:W)	Fuse reference specification
iWMC Integrated servo wheel	500	20A/58VDC

2.2 Interface Definition

2.2.1 iWMC Integrated servo wheel integrated terminal

Table 2-3 Servo wheel integrated port definition



PIN	Name	Extension line color	Pin function
1	24V	Red	Logic power input positive,Must be connected. Input voltage: 24V Maximum input current：1A
10	GND	Black	Logical power input negative
11	LOCK-	Blue	Forced release the brake input, only when the AGV body battery is out of emergency use. It should be noted that the servo wheel can not have 24V logic power and 48V power supply access. Input voltage: 24V Maximum input circuit :0.7A
2	LOCK+	Brown	
3	CANH	Light green	CAN IN
12	CANL	Light blue	
4	CANH	Pink	CAN OUT
13	CANL	White and black	
5	485A	Gray	485 IN
14	485B	White	
6	485A	Yellow	485 OUT
15	485B	Green	
7	OUT+	Purple	Digital signal output Maximum output current: 100mA
16	COMO	Orange	Output common terminal
17	DI1	White and red	Digital signal input High level: Input voltage:12.5VDC~30VDC Input current: 4-20mA Low level : 0VDC~5VDC Input frequency :<1KHz
18	DI2	White and orange	
8	COMI	White and brown	Input common terminal

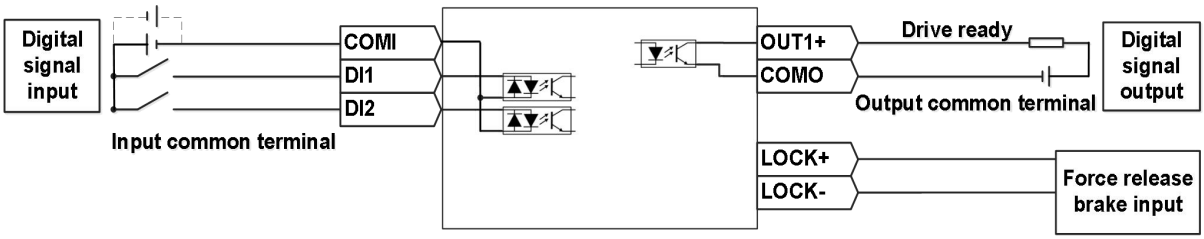


Figure 2-2 iWMC Integrated servo wheel control wiring diagram

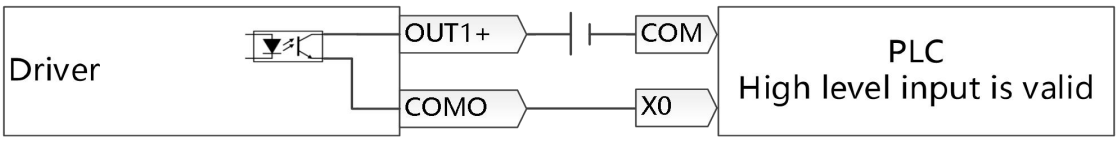


Figure 2-3 Digital outlet PNP control wiring diagram

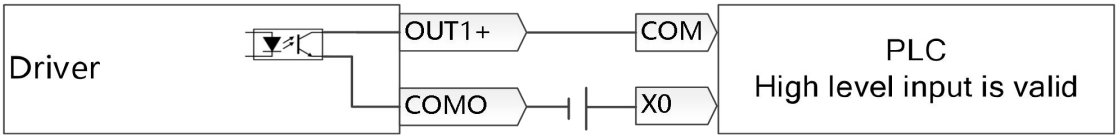
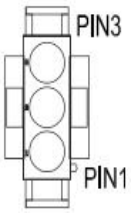


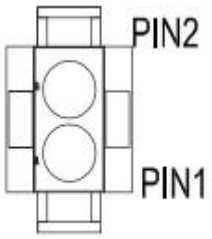
Figure 2-4 Digital outlet NPN control wiring diagram

Figure 2-5 Recommended circuit wiring diagram for forced release brake

2.2.2 Power port

	Pin	Pin name	Pin function
	3	DC- （Black）	Driver power supply input , must be connected. Input voltage : 24~60VDC
	1	DC+ （Red）	

2.2.3 Brake resistance port

	Pin	Pin name	Pin function
	1	RB+ （White or yellow）	External brake resistance input
	2	RB- (Yellow and green)	

2.2.4 Terminal specification

Table 2-4 Terminal specification table

	Servo wheel end	Extension end
Power cord	Rubber shell:Shang Yi C6350HM-3P-V0(Milky white） Contact pin:Shang Yi C6350M-TBe（Male）	Rubber shell:Shang Yi C6350HF-3P-V0(Milky white） Contact pin:Shang Yi C6350F-TBe（female）
Communication, IO, etc	Rubber shell: MOLEX 430201800 Contact pin: MOLEX 430310004	Rubber shell: MOLEX 430251800 Contact pin: MOLEX 430300004
Brake resistance	Rubber shell:Shang Yi C6350HM-2P-V0(Milky white） Contact pin:Shang Yi C6350M-TBe（Male）	Rubber shell:Shang Yi C6350HF-2P-V0(Milky white） Contact pin:Shang Yi C6350F-TBe（female）

Cable specifications:

	Suitable cable specification
Power cord	16AWG
Communication, IO, etc	28AWG
Brake resistance	16AWG

Chapter 3 KincoServo+ software introduction

This chapter will introduce how to use KincoServo software adjust and configure servo driver.

Figure 3 – 1 Software main window

3.1 Fast start

3.1.1 Language configuration

Language can be switched between English and Chinese via menu item **Tools->Language**.

3.1.2 Opening and saving project files

Create a new project file via menu item **File->New**, or by clicking the button.

Open an existing project via menu item **File->Open**, or by clicking the button and selecting a .kpjt file.

Save a project via menu item **File->Save**, or by clicking the button and saving as a .kpjt file.

Note

Only the windows (object list, scope etc.) are saved-parameters in the controller can't be saved in this way.

3.1.3 Start communication

Click menu item **Communication->Communication settings**. The following window appears:

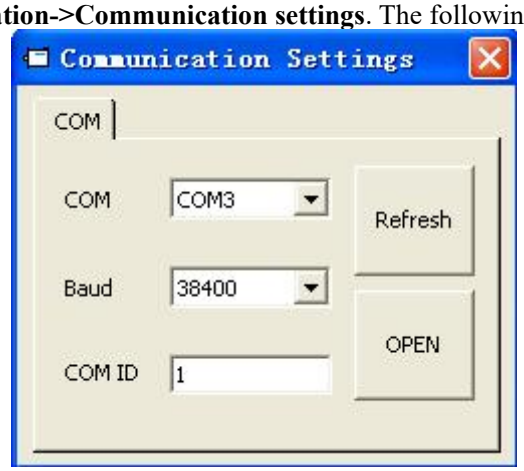


Figure 3－2Communication setup

Select the right COM port (if it's not shown click the "Refresh" button), baud rate and COM ID (Node ID), and then click the "OPEN" button.

Once communication has been established with the controller, communication can be opened or closed by clicking the button.

3.1.4 Node ID and baud rate

If more than one controller is being used in an application, you may need different node ID for different controllers in order to distinguish among them.

The controller's Node ID can be changed via menu item Controller->Controller Property.

Table 3-1 Node ID and baud rate settings

Internal address	Type	Name	Value	Unit
100B0008	Usigned8	Node ID		DEC
2FE00010	Usigned16	RS232 baud rate		Baud
65100C08	Usigned8	RS485protocol selection		DEC

Note

- iWMC integrated servo wheel does not have an RS232 debugging serial port, and can be connected to the host computer through the 485 communication port (the 485 protocol is RS232 by default). For details about the definition of the 485 communication terminal, see 2.2.1.
- Node ID and baud rate setting are not activated until after saving and rebooting.

3.1.5 Object (add，delete，help)

Open any window with an object list, move the mouse pointer to the object item and right click. The following selection window appears:

Basic Operation

N	Index	Type	Name	Value	Unit
0	606100	int8	Operation_Mode_Buff		DEC
1	604100	uint16	Statusword		HEX
2	606300	int32	Pos_Actual		inc
3	606C00	int32	Speed_Real		rpm
4	607800	int16	I_q		Ap
5	268000	uint16	Warning_Word		HEX
6	606000	int8	Operation_Mode		DEC
7	604000	uint16	Controlword		HEX
8	607A00	int32	Target_Position		inc
9	608100	uint32	Profile_Speed		rpm
10	608300	uint32	Profile_Acc		rps/s
11	608400	uint32	Profile_Dec		rps/s
12	60FF00	int32	Target_Speed		rpm
13	607100	int16	Target_Torque%		%
14	607300	uint16	CMD_q_Max		Ap
15	20200D	int8	Din_Mode0		DEC
16	20200E	int8	Din_Mode1		DEC
17	269000	uint8	Encoder_Data_Reset		DEC

Figure 3－3Basic operating interface

Click **Add** and double click the required object from the **Object Dictionary**. The selected object is then added to the list.

Click **Delete**. The selected object is removed from the list.

Click **Help** to read a description of the selected object in the **Object Dictionary**.

3.2 Initialize, save and reboot

Click **Driver->Initialize/Save**. The following window appears:

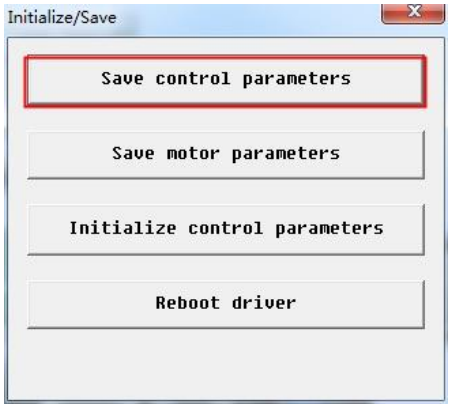


Figure 3－4 Initialize, save, reboot

Click the corresponding item to finish the necessary operation.

Note

After completing the **Init Control Parameters**, the Save Control Parameters and Reboot buttons must be clicked to load the default control parameters to the controller.

3.3 Firmware update

In general, the firmware of the drive is always the latest version, but if the drive firmware needs to be updated for some reason, please go to the menu bar "Drive" -> "Firmware Download".

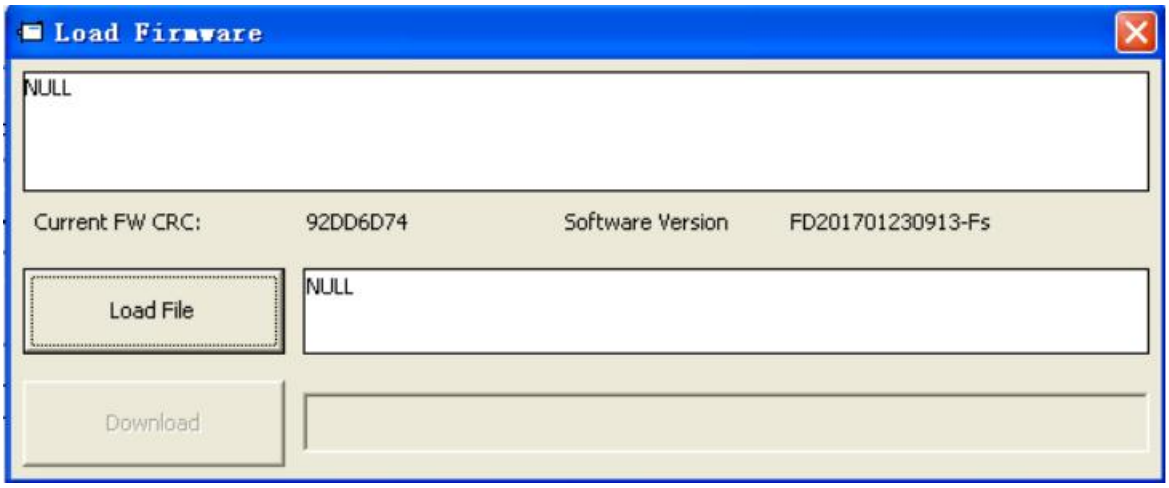


Figure 3-5 Firmware download

Click **Load File** to select the firmware file (.servo) and then click **Download** to start loading firmware to the controller.

Note

If the download is stopped for some reason, please first power off, then power on the drive, select the firmware version and click Start Download, and finally turn on the communication and connect to the host computer.

3.4 Read/write driver configurations

This function can be used to read / write multiple parameters simultaneously for large production lots, in order to avoid setting the controller parameters one by one.

3.4.1 Read setting from controller

Click **Tools**-> **Read Driver Configurations** from Controller or click the button. The following window appears.

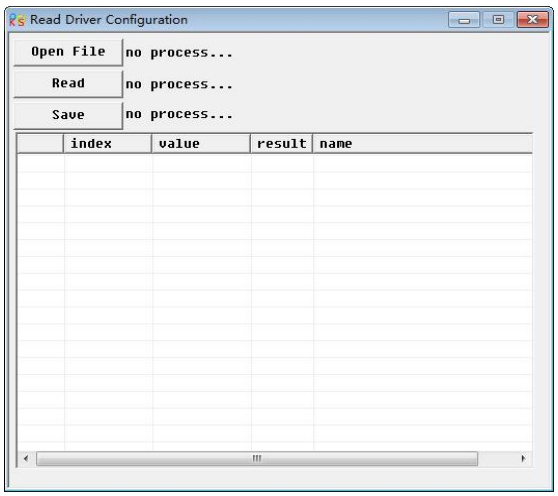


Figure 3-6 Read driver configuration

Click **Open File** to select a parameter list file （Kinco_Settings_Without Postable.cdo）, the parameters appears in the window on the right.

Click **Read Settings from Controller** to get the **Drive Value** and **Result**, and then click **Save to File** to save the settings as a .cdi file.

To export the drive's fault history, click on the errlist.cdo file when opening the list. Note that the errlist file can only read historical fault records, not drive configuration parameters.

3.4.2 Write settings to controller

Click **Tools->Write Driver Configurations** or click the button.

The following window appears:

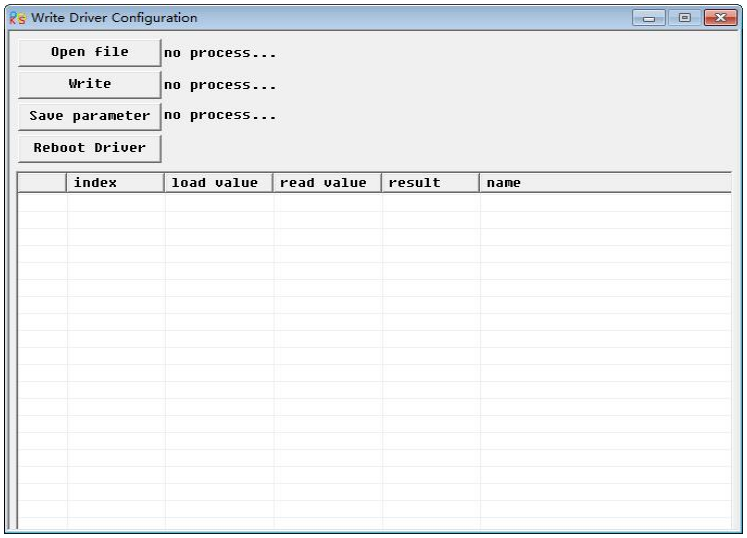


Figure 3-7 Write driver configuration

Click **Open File** to select a parameter settings file (.cdi). The parameter settings appear in the window.

Click **Write to Controller** to get the **Check Value** and **Result**. The “False” **Result** means the value has not been written successfully, probably because the object doesn’t exist in the controller.

Click **Save in EEPROM** and **Reboot** to activate all parameters.

Note

- When reading the drive configuration, if the object does not exist in the drive, the result will be "False", highlighted in red, and only the arguments that read "Ture" will be saved in the.cdi file.
- Before writing Settings to the drive, disconnect the 485/CAN/EtherCAT bus and disable the drive. Otherwise, some objects may not be able to be written successfully.

3.5 Speed mode introduction

There are two modes of speed mode: 3 and -3, and the control of speed mode can be written by external I/0 and internal instructions in two ways.

Table 3－2 Speed mode parameters description

Internal Address	Bits	Parameter name	Meaning description	Setting value
60600008	Integer8	Operating mode	-3: Immediate speed mode, the actual speed will immediately reach the target speed; 3: the speed mode with acceleration and deceleration, the actual speed will be accelerated to the target speed;	-3 和 3
60400010	Unsigned16	Control word	0x0F motor lock shaft; 0x06 Motor loose shaft	0x0F
60FF0020	Integer32	Target velocity	The target speed cannot exceed the rated motor speed	According to user demand
60830020	Unsigned32	Trapezoidal acceleration	3 Takes effect in mode	The default is 100rps/s
60840020	Unsigned32	Trapezoidal deceleration	3 Takes effect in mode	The default is 100rps/s

In the "Basic operation" window of the upper computer software, we can find these parameters and set them respectively in the 6th, 7th, 10th, 11th, and 1st

Basic Operation

N	Index	Type	Name	Value	Unit
0	606100	int8	Operation_Mode_Buff		DEC
1	604100	uint16	Statusword		HEX
2	606300	int32	Pos_Actual		inc
3	606C00	int32	Speed_Real		rpm
4	607800	int16	I_q		Ap
5	268000	uint16	Warning_Word		HEX
6	606000	int8	Operation_Mode		DEC
7	604000	uint16	Controlword		HEX
8	607A00	int32	Target_Position		inc
9	608100	uint32	Profile_Speed		rpm
10	608300	uint32	Profile_Acc		rps/s
11	608400	uint32	Profile_Dec		rps/s
12	60FF00	int32	Target_Speed		rpm
13	607100	int16	Target_Torque%		%
14	607300	uint16	CMD_q_Max		Ap
15	20200D	int8	Din_Mode0		DEC
16	20200E	int8	Din_Mode1		DEC
17	269000	uint8	Encoder_Data_Reset		DEC

3.6 Digital IO functions

Click menu item **Controller->Digital IO Functions** or click the I-O button. The following window appears,you can freely configure I/O port functions.

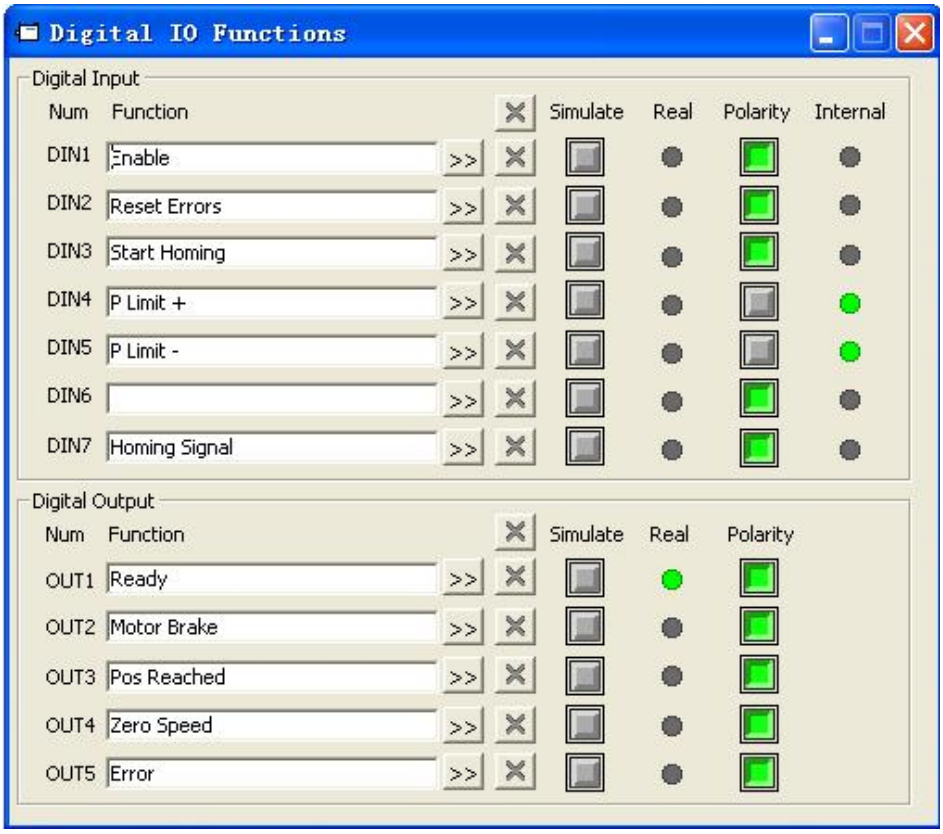


Figure 3-8 Digital input output

3.6.1 Digital input

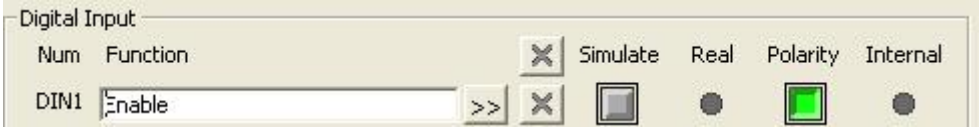


Figure 3-9 Digital input

Function: Click >> to select DIN function setting, click ✕ to delete the DIN function setting.

Simulate: Simulates the digital input active hardware signal.

Real: Shows the real digital input hardware status.

Polarity: means Internal is set to 1 by "active" signal. means Internal is set to 1 by "inactive" signal.

Internal: This is the result of Simulate, Real and Polarity via the logic formula:

Internal=(Real OR Simulate) XOR (NOT Polarity)

means "active", logic status of the selected function is 1; means "inactive", logic status of the selected function is 0.

DIN function	Description
Enable	Controller enabling 1: Enable controller (Control word=Din_Control word(2020.0F) , default value=0x2F) 0: Disable controller (Control word = 0x06)
Reset errors	Sets the Control word to reset errors, active edge: 0 -> 1
Operation mode	Operation_Mode selection 1: Operation_Mode=EL.Din_Mode1 (2020.0E) 0: Operation_Mode=EL.Din_Mode0 (2020.0D)
Limit+	Positive / negative position limit switch input for "normally closed" limit switches 0: position limit is active, the related direction is blocked
Limit-	
Instruction reversal	In speed and torque mode, the speed command can be reversed
Quick stop	Sets the controlword to start quick stop. After quick stop, the controlword needs to be set to 0x06 before 0x0F for enabling (if the enable function is configured in Din, just re-enable it)
Activate command	Activates the position command. Controls bit 4 of the Controlword, e.g. Controlword=0x2F->0x3F
Pre enable	For safety reasons, Pre_Enable can serve as a signal for indicating whether or not the entire system is ready. 1: controller can be enabled 0: controller can not be enabled

3.6.2 Digital output

Figure 3-10 Digital output

Function: Click >> to select the OUT function setting. Click ✕ to delete the OUT function setting

Simulate: Simulates the digital output function logic status 1.

Real: Shows the real digital input hardware status. This is the result of Simulate, Polarity and Logic State, ● means that digital input is ON, ● means that digital input is OFF.

Polarity: Inverts the logic status of the digital output function.

1 ■ means **Real** physical digital output is set to ON by digital output function logic status 1

0 ■ means **Real** physical digital output is set to ON by digital output function logic status 0

Real: This is the result of Simulate, Polarity and real input.

● activate, logic state of corresponding function is 1.

● deactivate, logic state of corresponding function is 0.

OUT function	Description
Ready	Controller is ready to be enabled
Error	Controller error
Zero Speed	|Speed_1ms(60F9.1A)|<=Zero_Speed_Window(2010.18)and duration >=Zero_Speed_Time(60F9.14)
Motor brake	Motor lock control output signal, if the use of lock motor, this function must be set, otherwise it will damage the motor
Index signal	Index signal occurrence
Speed Limit	In torque mode actual speed reached Max_Speed(607F.00)
Motor lock shaft	The drive is enabled and the motor is energized to lock the shaft
Position Limit	Position limit function is active
Torque reach set	When the actual torque (60F5.08) reaches the baseline (60F5.06) and the duration exceeds the filtering time (60F5.07), the output torque reaches the limit. If the torque reaches the baseline (60F5.06) is set to 0, the torque reaches the limit detection is not enabled.

3.7 Scope

During operation, if performance does not meet the requirement or any other unexpected behaviour occurs, it's highly advisable to use the scope function to do the analysis.

Click **Driver-->Oscilloscope** or click to open the scope window

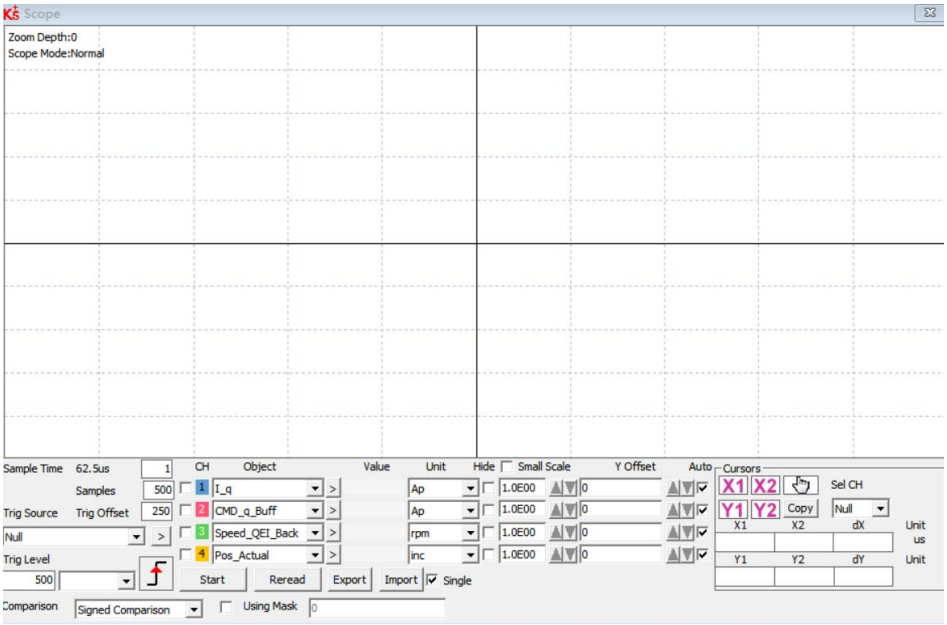


Figure 3-11 Oscilloscope interface

Sample Time： The period of data collection, set to 1 represents one data collection every 62.5us.

Samples : Indicates how many data are collected in this sample， and setting it to 500 indicates that 500 data are collected.

Trig offset: Number of samples before the trigger event occurs.

Trigger source / level: Trigger condition, DEC is an internal unit that can be switched to a current unit.

Trigger Clock Edge: Click to change to rising edge trigger ,Falling edge trigger or upper and lower edge trigger 。

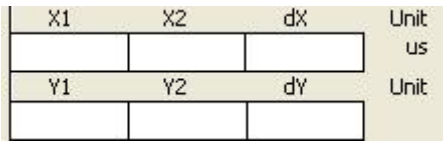
Object: Maximum 64-bit length data can be taken in one sample, e.g.: 2 Int32 objects bit or 4 Int16 objects.

Single: Single means sample for one trigger event only. Single means sample continuously.

Zoom in / zoom out the oscillogram: Press the right mouse button and drag the mouse down to the right to enlarge the oscillogram, and drag the mouse up to the left to narrow the oscillogram.

Cursors: At the click of a button Y1 Y2 can select the appropriate cursor, the cursor will be displayed on the oscilloscope, and select the channel you want to observe from the "Channel Selection" drop-down menu.

Moving Cursor: Press left mouse button and drag the scope cursor to move it. A sample value and the differences of X1, X2 and Y1, Y2 appear in the following fields:



Copy: Copy the sampled data to the paste board. You can open excel and paste the data directly.

Moving waveform: Button When the icon turns yellow, the movement takes effect. You can drag the waveform in the oscillograph by holding down the left mouse button.

Export: Export the sampled data to a.scope file.

Import ： Import the.scope file and display the oscillogram.

Reread data: Read the recently acquired data from the drive and display the oscillogram.

Auto： If the option box under Auto is checked, the oscillogram will automatically select the appropriate scale and axis offset for display. If the option box under Auto is not checked, the oscillogram will be displayed according to the scale and offset of the following areas.

Scale and offset values can be passed and button to increase or decrease,If the small scale option box is checked, the scale increase/decrease corresponding to the button will be 10% of the original.

Oscilloscope mode: In the upper left of the oscilloscope, the oscilloscope mode is normal or imported

-Normal: All buttons on the oscilloscope are available

-Import: The oscillogram is imported from the.scope file. In this mode, the start and reread data buttons are disabled, and you can exit the import mode as prompted by the software.

3.8 Error and error history

Error: Click **Controller->Error Display** or click the button (which turns red if an error occurs). The **Error Display** window appearsand display the most recent error message. Troubleshooting can be done according to the alarm troubleshooting plan in Chapter 7.

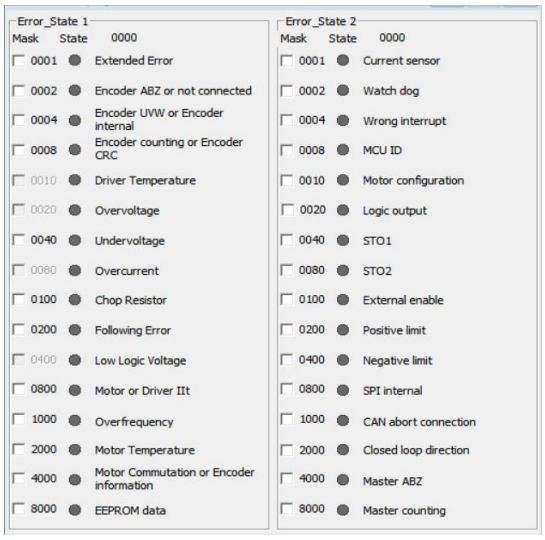
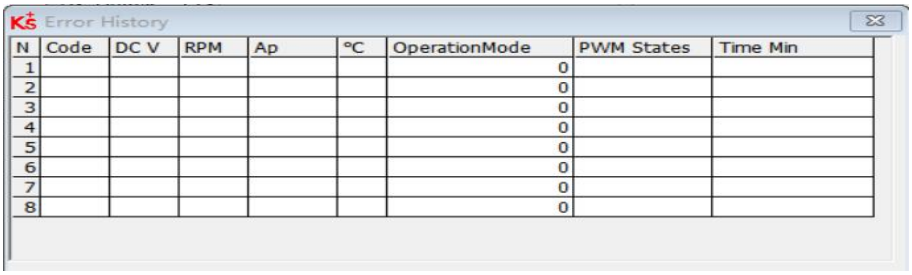


Figure 3-12 Error State Window

Error History: Click menu item **Controller->Error History**. The error history list window appears. It shows the last 8 errors' Error codes and respective the related DCBUS voltage, speed, current, controller temperature, Operation Mode, and power tube condition at the moment when the error occurred.The most recent historical fault is displayed on the first line.



N	Code	DC V	RPM	Ap	°C	OperationMode	PWM States	Time Min
1						0		
2						0		
3						0		
4						0		
5						0		
6						0		
7						0		
8						0		

Figure 3-13 History error display screen

Table 3-3 Error state （2601.00）information

Bit	Error name	Error code	Description
0	Extended Error		Refer to object "Error_State 2"(2602.00)
1	Encoder not connected	0x7331	No communication encoder connected
2	Encoder internal	0x7320	Internal encoder error
3	Encoder CRC	0x7330	Communication with encoder disturbed
4	Controller Temperature	0x4210	Heatsink temperature too high
5	Over voltage	0x3210	DC bus overvoltage
6	Unde rvoltage	0x3220	DC bus undervoltage
7	Over current	0x2320	Power stage or motor short circuit
8	Chop Resistor	0x7110	Overload, brake chopper resistor
9	Following Error	0x8611	Max. following error exceeded
10	Low Logic Voltage	0x5112	Logic supply voltage too low
11	Motor or controller IIt	0x2350	Motor or power stage IIt error
12	Over frequency	0x8A80	Pulse input frequency too high
13	Motor Temperature	0x4310	Motor temperature sensor alarm
14	Encoder information	0x7331	No encoder connected or no encoder communication reply
15	EEPROM data	0x6310	EEPROM checksum fault

Table 3-4 Error_state2（2602.00）information

Bit	Error name	Error code	Description
0	Current sensor	0x5210	Current sensor signal offset or ripple too large
1	Watchdog	0x6010	Software watchdog exception

2	Wrong interrupt	0x6011	Invalid interrupt exception
3	MCU ID	0x7400	Wrong MCU type detected
4	Motor configuration	0x6320	No motor data in EEPROM / motor never configured
5	Reserved		
6	Reserved		
7	Reserved		
8	External enable	0x5443	DIN "pre_enable" function is configured, but the DIN is inactive when the controller is enabled / going to be enabled
9	Positive limit	0x5442	Positive position limit (after homing) – position limit only causes error when Limit_Function (2010.19) is set to 0.
10	Negative limit	0x5441	Negative position limit (after homing) position limit only causes error when Limit_Function(2010.19) is set to 0.
11	SPI internal	0x6012	Internal firmware error in SPI handling
12	CAN bus interrupt	0x8100	The fault alarm will be generated only when the communication interruption mode (6007.00) is set to 1

Table3-5 Error extension (2605.07) message

Bit	Error name	Error code	Description
0	Recording error	0x5210	The current sensor signal is offset or wavy
1	Internal brake resistance is overheated	0x7111	The actual power of the internal brake resistance is too large
2	Internal brake resistance short circuit	0x7112	Internal brake unit damaged, brake circuit short-circuited
3	Motor out of phase	0x6321	A phase of the motor power line UVW is not connected
4	ADC sampling saturation	0x2321	The current sampling ADC reaches its limit and the current is out of control
12	Service timeout	0x81FF	Communication bus extension error

Chapter 4 Performance tuning

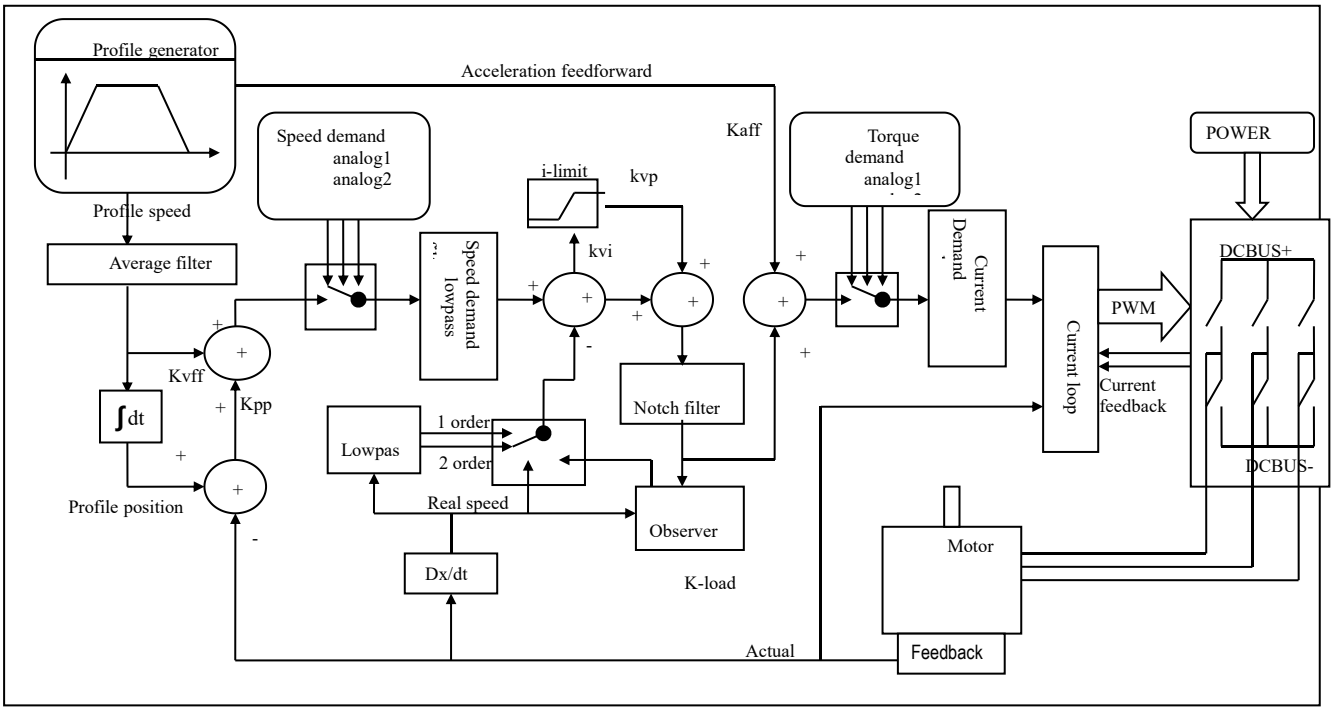


Figure 4－1Servo system control structure block diagram

Figure 4-1 is a block diagram of the control structure of the servo system. It can be seen from the diagram that the servo system generally includes three control rings: current ring, speed ring and position ring. For the servo system, good control ring parameters can improve the performance of the servo and better meet the field process requirements. Therefore, it is necessary to adjust good control ring parameters.

The parameters of speed ring and position ring should be adjusted during debugging. The speed ring parameter is related to the load inertia of the whole mechanical system converted to the motor shaft. The position ring is the outermost control ring of the servo system and is related to the motor action mode, that is, the field application. The current ring is the innermost control ring in the servo system, and the current ring parameters are related to the motor parameters. After the motor is correctly configured, the system defaults the current ring parameters to the best parameters of the configured motor, so there is no need to adjust again.

4.1Tuning of velocity loop

Table 4－1 List of velocity loop parameters

Internal address	Name	Description	Default	Range
60F90110	Kvp[0]	Proportional velocity loop gain Can be displayed in Hz in the PC tool can if the inertia ratio is right.	/	1~32767
60F90210	Kvi[0]	Integral velocity loop gain	/	0-1023
60F90710	Kvi/32	Integral velocity loop gain of in a smaller unit of measure	/	0-32767
60F90508	Speed_Fb_N	Used to set Velocity feedback filter bandwidth Filter bandwidth=100+Speed_Fb_N*20	7	0~45
60F90608	Speed_Mode	Used to set the velocity feedback mode 0: 2nd order FB LPF 1: Directly feedback the original velocity 2: Velocity feedback after velocity observer 4: Velocity feedback after 1st order LPF 10:Velocity feedback after 2nd order LPF and the velocity command is filtered by a 1st order LPF. Both filters have the same bandwidth. 11: The velocity command is filtered by a 1st order LPF 12: Velocity feedback after velocity observer, the velocity command is filtered by a 1st order LPF 14: Velocity feedback after 1st order LPF and the velocity command is filtered by a 1st order LPF. Both filters have the same bandwidth	1	/
60F91508	Output_Filter_N	A 1st order lowpass filter in the forward path of the velocity loop	1	1-127
60F90820	Kvi_Sum_Limit	Integral output limit of the velocity loop	/	0-2^15

Step of Velocity loop tuning is shown below:

Step 1: Determine the upper limit of the velocity loop bandwidth

The bandwidth of the velocity loop limits the bandwidth of the position loop, so it is important to adjust the bandwidth of the speed loop.

The upper limit of the bandwidth of the velocity loop can be determined by several aspects:

- Feel motor oscillations and noises through your fingers and ears. It's actually a rule of thumb, but it's very effective. Users can choose to increase or decrease the speed loop bandwidth by listening and touching the machine.
- Another way is to observe the oscilloscope, the user generates a step curve of the speed control, and samples the actual speed and electrical flow of the line. By comparing the sample graphs under different speed loop bandwidths, we can find the optimal curve - the speed curve follows the instruction quickly and does not oscillate.

Step 2: Velocity feedback filter adjustment

The velocity feedback filter can reduce noise that comes from the feedback path, e.g. reduce encoder resolution noise.

The velocity feedback filter can be configured as 1st and 2nd order via the Speed_Mode for different applications.

The 1st order filter reduces noise to a lesser extent , but its also results in less phase shifting so that velocity loop gain can be set higher.

The 2nd order filter reduces noise to a greater extent , but its also results in more phase shifting so that velocity loop gain can be limited.

Normally, if the machine is stiff and light, we can use the 1st feedback filter or disable the feedback filter. If the machine is soft and heavy, we can use the 2nd order filter.

If there′ s too much motor noise when velocity loop gain is adjusted, velocity loop feedback filter parameter Speed_Fb_N can be reduced accordingly. However, velocity loop feedback filter bandwidth F must be more than twice as large as the velocity loop bandwidth. Otherwise, it may cause oscillation. Velocity loop feedback filter bandwidth F=Speed_Fb_N*20+100 [Hz].

Step 3: Output filter adjustment

The output filter is a 1st order torque filter. It can reduce the velocity control loop to output high frequency torque, which may stimulate overall system resonance.

The user can try to adjust Output_Filter_N from small to large in order to reduce noise.

The filter bandwidth can be calculated using the following formula.

$$\frac{1}{2}\frac{\ln\left(1-\frac{1}{Output_Filter_N}\right)}{Ts\,\pi},\ Ts = 62.5\ us$$

Step4: Velocity loop bandwidth calculation

Use the following formula to calculate velocity loop bandwidth:

$$kvp = \frac{1.853358080\,10^5\,J\pi^2\,Fbw}{\mathrm{I}_{Max}\,kt\,encoder}$$

kt motor torque constant, unit: Nm/Arms*100

J inertia, unit: kg*m^2*10^6

Fbw Velocity loop bandwidth, unit: Hz

Imax max motor current I_max(6510.03) as DEC value

encoder resolution of the encoder

Step5: Velocity loop integral gain adjustment

Integral gain is used to eliminate static error. It can boost velocity loop low frequency gain, and increased integral gain can reduce low frequency disturbance response.

Normally, if the machine has considerable friction, integral gain (kvi) should be set to a higher value.

If the entire system needs to respond quickly, integral should be set to a small value or even 0, and the gain switch should be used.

Step6: Velocity loop Kvi_sum_limit adjustment

Normally the default value is fine. This parameter should be added if the application system has a big extend force, or should be reduced if the output current is easily saturation and the saturation output current will cause some low frequency oscillation.

4.2 Tuning of position loop

Table 4－2List of position loop parameters

Internal address	Name	Description	Default	Range
60FB0110	Kpp[0]	Proportional position loop gain. Used to set the position loop response. unit: 0.01Hz	10	0～327
60FB0210	K_Velocity_FF	0 means no feedforward, 1000 means 100% feedforward.	100	0～100
60FB0310	K_Acc_FF	The unit only is right if the inertia ratio is correctly set. If the inertia ratio is unknown, set K_Acc_FF(60FB.03) instead.	/	0-32767
60FB0510	Pos_Filter_N	Smooth acceleration and deceleration processes need to be set in the loose shaft state of the motor	1	1~255
60650020	Max_Following_ Error	The maximum allowable error, over the change value will alarm 020.0	10000	/

Step of Position loop tuning is shown below:

Step1: Position loop proportional gain adjustment

Increasing position loop proportional gain can improve position loop bandwidth, thus reducing positioning time and following error, but setting it too high will cause noise or even oscillation. It must be set according to load conditions. Kpp = 103 * Pc_Loop_BW, Pc_Loop_BW is position loop bandwidth. Position loop bandwidth

cannot exceed velocity loop bandwidth. Recommended velocity loop bandwidth: Pc_Loop_BW<Vc_Loop_BW / 4,Vc_Loop_BW is the speed loop bandwidth.

Step2: Position loop velocity feedforward adjustment

Increasing the position loop velocity feedforward can reduce position following error, but can result in increased overshooting. If the position command signal is not smooth, reducing position loop velocity feedforward can reduce motor oscillation.

The velocity feedforward function can be treated as the upper controller (e.g. PLC) have a chance to directly control the velocity in a position operation mode. In fact this function will expend part of the velocity loop response ability, so if the setting can′ t match the position loop proportional gain and the velocity loop bandwidth, the overshot will happen.

Besides, the velocity which feedforward to the velocity loop may be not smooth, and with some noise signal inside, so big velocity feedforward value will also amplified the noise.

Step3:Position loop acceleration feedforward

It is not recommended that the user adjust this parameter. If very high position loop gain is required, acceleration feedforward K_Acc_FF can be adjusted appropriately to improve performance.

The acceleration feedforward function can be treat as the upper controller (e.g. PLC) have a chance to directly control the torque in a position operation mode. in fact this function will expend part of the current loop response ability, so if the setting can′ t match the position loop proportional gain and the velocity loop bandwidth, the overshot will happen.

Besides, the acceleration which feedforward to the current loop can be not smooth, and with some noise signal inside, so big acceleration feedforward value will also amplified the noise.

Acceleration feedforward can be calculated with the following formula:

ACC_%=6746518/ K_Acc_FF/EASY_KLOAD*100

ACC_%: the percentage which will be used for acceleration feedforward.

K_Acc_FF——OD 0x60FB03,the final internal factor for calculating feedforward.

[30400710]—— the load factor which is calculated from auto-tuning or the right inertia ratio input.

➔ **Note**

The smaller the K_Acc_FF, the stronger the acceleration feedforward.

Step4:Smoothing filter

The smoothing filter is a moving average filter. It filters the velocity command coming from the velocity generator and makes the velocity and position commands more smooth. As a consequence, the velocity command will be delayed in the controller. So for some applications likeCNC, it′ s better not to use this filter and to accomplish smoothing with the CNC controller.

The smoothing filter can reduce machine impact by smoothing the command. The Pos_Filter_N parameter

define the time constant of this filter in ms. Normally, if the machine system oscillates when it starts and stops, a larger Pos_Filter_N is suggested.

Step5:Notch filter

The notch filter can suppress resonance by reducing gain around the resonant frequency.

Antiresonant frequency=Notch_N*10+100

Setting Notch_On to 1 turns on the notch filter. If the resonant frequency is unknown, the user can set the maximum value of the d2.14 current command small, so that the amplitude of system oscillation lies within an acceptable range, and then try to adjust Notch_N and observe whether the resonance disappears.

Resonant frequency can be measured roughly according to the Iq curve when resonance occurs on the software oscilloscope.

Table 4－3 Notch filter list

Internal address	Name	Description	Default	Range
60F90308	Notch_N	Used to set the frequency of the internal notch filter to eliminate mechanical resonance generated when the motor drives the machine. The formula is F=Notch_N*10+100. For example, if mechanical resonance frequency F=500 Hz, the parameter setting should be 40.	45	0~90
60F90408	Notch_On	Used to turn on or turn off the notch filter. 0：Turn on the notch filter 1：Turn off the notch filter	0	0~1

4.3 Factors which influence tuning results

The control command is created by the upper controller (e.g. PLC)

- The control command should be smooth as much as possible, and must be correct. For example, the control command should not create the acceleration commands (inside the position commands)
- the control command should follow the bandwidth limit of the control loop.

The machine design:

In the actual application, performance is normally limited by the machine. Gaps in the gears, soft connection in the belts, friction in the rail, resonance in the system － all of these can influence final control performance. Control performance affects the machine′s final performance, as well as precision, responsiveness and stability.

Chapter5 Alarm Exclusion

When driver generate an alarm, red light, ERR, will light up.

If you need more detailed information about errors and error history, please connect the controller to the PC via RS232.

Table 5－1Alarm codes of Error_State 1

Alarm	Code	Name	Reason	Troubleshooting
000.1		Extended Error	Errors occurs in Error_State2	Open the menu bar of the upper computer software "Drive" -> "Fault Display" to view the alarm information of the error status word 2. For details about the alarm content and solution, see Table 7-2
000.2	7380	Encoder ABZ signal incorrect（suitable for incremental encoder motor)	Encoder ABZ wiring is wrong or disconnected	1.Check encoder cable is correctly connected 2.Check if corresponding pins of encoder cable is on（refer servo product menu)
	7331	Encoder communication incorrect (suitable for magnetoelectric encoder motor)	The encoder wiring is incorrect or disconnected.	
000.4	7381	Encoder UVW signal incorrect（suitable for incremental encoder motor）	Encoder UVW wiring is wrong or disconnected	1.Check encoder cable is correctly connected 2.Check if corresponding pins of encoder cable is on（refer servo product menu) Change motor
	7320	Encoder internal error（suitable for magnetoelectric encoder motor）	Encoder internal is incorrect or encoder is broken	
000.8	7305	Encoder count wrong（suitable for incremental encoder motor））	Encoder is interfered	1.Check encoder cable is correctly connected (different from motor PE cable) 2.Make sure the equipment is well grounded 3.Use isolated power supply to provide power
	7330	Encoder CRC (suitable for magnetoelectric encoder motor)		
001.0	4210	Controller temperature	The temperature of controller's power module has reached the alarm value	1.Add fan，improve the cooling environment of the controller. 2.Add driver installment distance 3.Vertically install driver
002.0	3210	Over voltage	Supply power voltage exceeds the allowable input voltage range	1.Check if supply power is higher than standard output voltage 2.Check to see if supply power voltage is unstable
			In case of emergency stop, there is no external braking resistor or braking.	1.Connect suitable braking resistor 2.Open software "Driver"->"Panel menu"->"（F005）controller setting" 3.Correctly set "brake resistor value" an "brake resistor power"

			Brake resistor is not configured	1.Change Connect suitable braking resistor 2.Open software "Driver"->"Panel menu"->" (F005) controller setting" 3.Correctly set "brake resistor value" an "brake resistor power"
004.0	3220	Undervoltage	The power voltage input is lower than the low voltage protection alarm value.	1.Check if power supply output power can meet with the requirement 2.Change power supply of bigger power
008.0	2320	Short circuit of driver output	Short circuit of driver UVW and PE output	1.Check if motor power cable connection is correct 2.Driver is broken, change driver
010.0	7110	Driver brake resistor is abnormal	Not configure correct brake resistor parameters	1.Open software "Driver"->"Panel menu"->" (F005) controller setting" 2.Correctly set "brake resistor value" an "brake resistor power"
020.0	8611	Following error	Stiffness of control loop is too small	1.Open software "Driver""control loop""velocity loop""position loop" 2.Increase "kpp[0]""kvp[0]"
			Motor UVW phase sequence is incorrect	Exchanging wire of U and V
			The controller and motor together can't match the requirement of the application	Change motor and driver with bigger power
			Max_Following_Error is too small	Open software "Driver""control loop""velocity loop""position loop" Increase "max_following_error" (Ensure control loop parameters is fine, user can change this parameter)
040.0	5122	Low logic voltage	Logic voltage is less than 18V, power supply voltage is pulled down	1.Check if power supply output power can meet with requirements 2.Change power supply with bigger power
080.0	2350	Motor or controller IIt	The brake is not released when the motor shaft is rotating (only for brake motor)	1.Check if brake cable wiring is correct 2.Check brake power can meet with the requirements (output voltage is DC24V, input current is 1A, output power is bigger than 24W)
			Machine equipment stuck or excessive friction	1.Cancel motor enable, or power off driver 2.Please drag load to make it move back and forth in motor's running route. Ensure that there is no machine equipment stuck or excessive friction 3.Add lubricate

			Motor UVW phase sequence is incorrect	Exchange motor wiring of phase U and phase V
100.0	8A80	Over input frequency	External input pulse frequency is too high	1.Reduce external pulse input frequency 2.When ensure safely use motor, increase "Frequency_Check" （Open ″Driver″ -> ″Control modes″ -> ″Pulse mode″ -> ″Frequency_Check″），max 600
200.0	4310	Motor temperature	The motor temperature exceeds the specified value	1.Reduce ambient temperature of the motor and improve cooling conditions 2.Reduce acceleration and deceleration 3.Reduce load
400.0	7122	Motor excitation（suitable for incremental encoder）	Motor UVW phase sequence is wrong	Exchange motor wiring of phase U and phase V
			Encoder is not connected	Check encoder cable
		Encoder information（suitable for magnetoelectric encoder）	Communication is incorrect when the encoder is initialized	Check encoder wiring, restart driver
			The encoder type is wrong, e.g. an unknown encoder is connected	
			The data stored in the encoder is wrong	
			The controller can′t support the current encoder type	
800.0	6310	EEPROM data	Data is damaged when the power is turned on and data is read from the EEPROM	1.Open software ″Driver″ -> ″Init Save Reboot″ 2.Click ″Init Control Parameters″ -> ″Save Control Parameters″ -> ″Save Motor Parameters″ -> ″Reboot″ 3.Import cdi file by software

Table 5－2 Alarm codes of Error_State2

Alarm	Code	Name	Reason	Trouble shooting
000.1	0x5210	Current sensor	Current sensor signal offset or ripple too big	Circuit of current sensor is damaged, please contact the supplier
000.2	0x6010	Watchdog	Software watchdog exception	Please contact the supplier and try to update the firmware
000.4	0x6011	Wrong interrupt	Invalid interrupt exception	Please contact the supplier and try to update the firmware

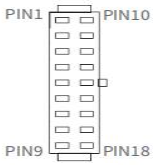
000.8	0x7400	MCU ID	Wrong MCU type detected	Please contact the supplier
001.0	0x6320	Motor configuration	Motor type is not auto-recognized, no motor data in EEPROM / motor never configured	Install a correct motor type to the controller and reboot
010.0	0x5443	External enable	DIN function "pre_enable" is configured, but the input is inactive when the controller is enabled or should become enabled	Solve according to the reason
020.0	0x5442	Positive limit	Positive position limit (after homing), position limit only causes error when Limit_Function (2010.19) is set to 0	Exclude the condition which causes the limit signal
040.0	0x5441	Negative limit	Positive position limit (after homing), position limit only causes error when Limit_Function (2010.19) is set to 0	Exclude the condition which causes the limit signal
080.0	0x6012	SPI internal	Internal firmware error in SPI handling	Please contact the supplier
200.0	0x8A81	Close loop direction	Different direction between motor and position encoder	Change the encoder counting direction
800.0	0x7306	Master counting	Master encoder counting error	Ensure that the ground connection and the encoder shield work well.

Chapter 6 RS485 communication

6.1RS485 Wiring instructions

The RS485 port of iWMC integrated servo wheel is the servo debugging port by default, and the protocol is RS232 format.

Table 6-1 RS485 terminal specification

	PIN	Name	Pin function
 Back view	5	485A	485IN
	14	485B	
	6	485A	485OUT
	15	485B	

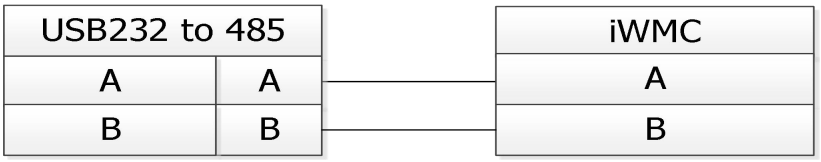


Figure 6 - 1 RS485wiring diagram

6.2 RS485 Communication parameters

Internal Address	Parameter name	Meaning	Default
100B0010	ID_Com	Driver station number	1
2FE20010	RS485 Baud rate	Set baud rate of RS485 por Set value Baud 1080————9600 540————19200 270————38400 90————115200 Note: Save and reboot	270
65100C08	RS485 Communication protocol selection	0：Modbus protocol 1：RS232 protocol Note: Set to 0, save and reboot	1
65100E10	RS485 Mode	data=8，stop=1，no check bit	default

Chapter 7 CANopen

7.1 CANopen communication protocol

CANopen is one of the most famous and successful open fieldbus standards.It has been widelyrecognized and applied a lot in Europe and USA. In 1992,CiA (CANinAutomation) was set up in Germany,and began to develop application layer protocol CANopen for CAN in automation. Since then, members of CiA developed a series of CANopen products,and applied in a large number of applications in the field of machinery manufacturing such as railway, vehicles, ships, pharmaceutical, food processing etc.

Servo wheels are standard CAN slave equipment,strictly follow CANopen2.0A / B protocol, any host computer which support this protocol can communicate with it. Servo uses of a strictly defined object list, we call it the object dictionary, this object dictionary design is based on the CANopen international standards, all objects have a clear definition of the function. Objects said here similar to the memory address, we often say that some objects, such as speed and position,can be modified by an external controller, some object were modified only by the drive itself, such as status and error messages.Table 7-1 lists these objects.

Table 7-1 Object dictionary example list

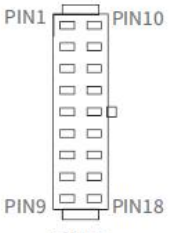
Index	Sub	Bits	Attribute	Meaning
6040	00	16(=0x10)	RW	Control word
6060	00	8(=0x08)	RW	Operation mode
607A	00	32(=0x20)	W	Target position
6041	00	16(=0x10)	MW	Status word

The attributes of objects are as follows:

1. RW(read&write)：The object can be both read and written;
2. RO(only read)：The object can be read only;
3. WO（only write）：The object can be written only;
4. M（map）：The object can be mapping,similar to indirect addressing;
5. S（save）：The object can be stored in Flash-ROM without lost after power failure;

7.2 CANopen bus communication hardware instroduction

Table 7-2 Pin and function

Terminal	PIN	Symbol	Meaning
 Back view	3	CAN_H	CAN in
	12	CAN_L	
	4	CAN_H	CAN out
	13	CAN_L	

CAN communication protocol describes a way of transmitting information between devices, The definition of CAN layer is the same as the open systems interconnection model OSI, each layer communicates with the same layer in another device, the actual communication takes place adjacent layers in each device,but the devices only interconnect by the physical media of thephysical layer in the model.CAN standard defines data link layer and physical layer in the mode. The physical layer of CAN bus is not strictly required, it can use a variety of physical media such as twisted pair Fibre. The most commonly used is twisted pair signal, sent by differential voltage transmission (commonly used bus transceiver). The two signal lines are called CAN_H and CAN_L. The static voltage is approximately 2.5V, then the state is expressed as a logical 1, also called hidden bit. It represents a logic 0 when CAN_H is higher than the CAN_L, we called it apparent bit,then the voltage is that CAN_H = 3.5V and CAN_L= 1.5V,apparent bit is in high priority. Table 7-2 lists the names and functions of CAN communication port pins.

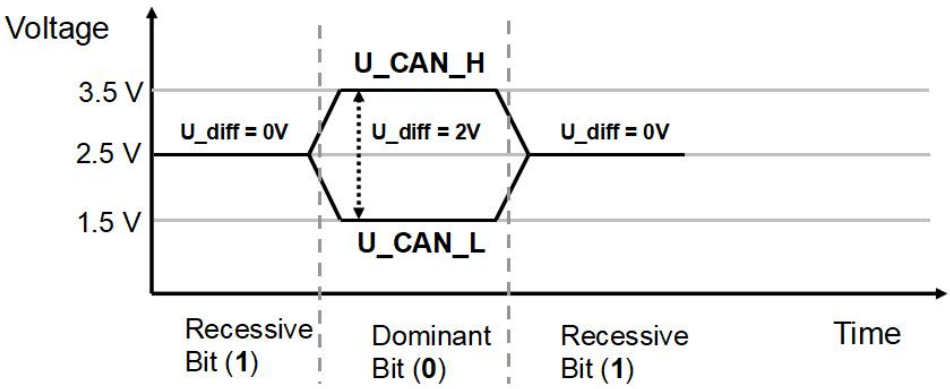


Figure 7-1 CAN signal identification

Noted:

1. All CAN_L and CAN_H of slaves connect directly by using series connection.
2、Please use the shield wires for communication cable.
3、The max. distance at different baudrate are shown in following table..
4、The servo wheel does not need to be connected to an external 24V power supply to power the CAN.

Table 7-3 The max. distance at different baudrate are shown in following table

Communication speed（bit/s）	Communication distance（M）
1M	25
800K	50
500K	100
250K	250
125K	500
50K	600
25K	800
10K	1000

7.2.1 CANopen bus communication software introduction

7.2.1.1 EDS introduction

EDS（Electronic Data Sheet）file is an identification documents or similar code of slave device,to identify what kind of slave device is(Like 401,402 and 403,or which device type of 402).This file includes all information of slaves,such as manufacturer,sequence No.,software version,supportable baudrate,mappable OD and attributes of each OD and so on,similar to the GSD file for Profibus.Therefore,we need to import the EDS file of slave into the software of master before we configure the hardware.

7.2.1.2 SDO introduction

SDO is mainly used in the transmit the low priority object between the devices, typically used to configure and mange the device,such as modifying PID parameters in current loop,velocity loop and position loop,and PDO configuration parameters and so on.This data transmission mode is the same as Modbus,that is it needs response from slave when master sends data to slave.This communication mode is suitable for parameters setting,but not for data transmission frequently.

SDO includes upload and download.The host can use special SDO instructions to read and write the OD of servo.In CANopen protocol, SDO (Service Data Object) can be used to modify object dictionary. SDO structure and guidelines are shown below:

SDO basic structure is：Client→Server/Server→Client

Byte0	Byte1-2	Byte3	Byte4-7
SDO Command specifier	Object index	Object subindex	Max 4 bytes data

The SDO command word contains the following information:

- Download/upload
- Request/response

- Segmented/expedited transfer
- Toggle bit

There are five request/reply protocols implemented in SDO:

- Initiate Domain Download
- Download Domain Segment
- Initiate Domain Upload
- Upload Domain Segment
- Abort Domain Transfer

Among them, Download refers to write operation of the object dictionary, Upload refers to read operation of the object dictionary; Use the Initiate Domain Upload protocol when reading parameters; When setting parameters, use the Initiate Domain Download protocol; The syntax of the protocol SDO command word (the first byte of an SDO CAN message) is described in Table 7-4 and Table 7-5, where "-" indicates that it is irrelevant and should be 0.

Table 7-4 Initiate domain download

Bit	7	6	5	4	3	2	1	0
Client→	0	0	1	-	n		e	s
←Server	0	0	1	-	-	-	-	-

Table 7-5 Initiate Domain Upload

Bit	7	6	5	4	3	2	1	0
Client→	0	0	1	-	-	-	-	-
←Server	0	0	1	-	n		e	s

Noted:

n——indicates the number of bytes of meaningless data in the packet data [from (8-n) bytes to the seventh byte of meaningless data] (n is valid when e=1 and s=1, otherwise n is 0);

e —— Normal transmission when e=0, accelerated transmission when e=1;

s —— Indicates whether the data length is specified, 0 indicates the data length is not specified, 1 indicates the data length.

e=0, s=0—— reserved by CiA;

e=0, s=1——The data byte is the byte counter, byte4 is the low data part (LSB), byte7 is the high data part (MSB); e=1 - Data bytes are the data to be downloaded.

Table 7-6 and 7-7 show the formats of SDO messages sent and received when reading parameters.

Table 7-6 Send SDO message when read parameters

Identifier	DLC	Daten							
		0	1	2	3	4	5	6	7
0x600+Node_ID	8	Send command word	Object index		Object subindex	00			

Table 7-7 Receive SDO message when read parameters

Identifier	DLC	Daten							
		0	1	2	3	4	5	6	7
0x580+Node_ID	8	Receive command word	Object index		Object subindex	Max 4 bytes data			

Note: Command word is 0x40 when the SDO message is sent;

If the received data is 1 byte, the received command word is 0x4F.

If the received data is 2 bytes, the received command word is 0x4B.

If the received data is 4 bytes, the received command word is 0x43.

If there is an error in the received data, the receive command word is 0x80.

Table 7-8 and 7-9 show the formats of SDO packets to be sent and received when parameters are modified.

Table 7-8Send SDO message (Modify parameters)

Identifier	DLC	Daten							
		0	1	2	3	4	5	6	7
0x600+Node_ID	8	Send command word	Object index		Object subindex	Max 4 bytes data			

Table 7-9Receive SDO message (Modify parameters)

Identifier	DLC	Daten							
		0	1	2	3	4	5	6	7
0x580+Node_ID	8	Receive command word	Object index		Object subindex	Max 4 bytes data			

Noted :If SDO message is sent successfully, receive command is 0x60;If SDO message is not sent successfully, receive command is 0x80.

If sent data ready is 1 byte, command is 0x2F;

If sent data ready is 2 bytes, command is 0x2B;

If sent data ready is 4 bytes, command is 0x23;

When the SDO message fails to be sent, you can troubleshoot the problem according to the error code returned.

Table 7-10 SDO message error code

Error code	Code function description
0x05040001	Invalid command, unknown or illegal Client/Server command word
0x06010001	Attempts to read only write object parameters
0x06010002	An attempt was made to write read-only object parameters
0x06020000	Invalid index. The object does not exist in the object dictionary
0x06040041	Cannot be mapped, object parameters do not support mapping to PDO
0x06060000	The drive is in an error state, causing the object parameter access failure
0x06070010	The data type and length of the service parameter do not match
0x06070012	The data type does not match, and the length of the service parameter is too large
0x06070013	The data type does not match, and the length of the service parameter is too short

0x06090011	Invalid subindex
0x06090030	Invalid data, out of range of object parameters
0x06090031	The value of the written data is too large
0x06090032	The value of the written data is too small
0x08000022	Data cannot be transferred or saved to the application due to the current device state

Table 7-12 Speed mode is set through SDO messages

Parameter address	Name	Value	Message (ID=1)
60600008	Operating mode	3	Send→601 2F 60 60 00 03 00 00 00 Receive←581 60 60 60 00 03 00 00 00
60FF0020	Target velocity	-100RPM	Send→601 23 FF 60 00 7E B1 E4 FF Receive←581 60 FF 60 00 7E B1 E4 FF
60400010	Control word	2F	Send→601 2B 40 60 00 2F 00 00 00 Receive←581 60 40 60 00 2F 00 00 00
60830020	Trapezoidal acceleration	100rps/s	Send→601 23 83 60 00 6E A3 01 00 Receive←581 60 83 60 00 6E A3 01 00
60840020	Trapezoidal deceleration	100rps/s	Send→601 23 84 60 00 6E A3 01 00 Receive←581 60 84 60 00 6E A3 01 00

Note: The message is expressed in hexadecimal, and the motor resolution used in this case is 65536

7.2.1.3 PDO introduction

PDO can transport 8 bytes of data at one time,and no other protocol preset(Mean the content of the data are preset),it is mainly used to transmit data in high frequency.PDO uses brand new mode for data exchange,it needs to define the data receiving and sending area before the transmission between two devices,then the data will transmit to the receiving area of devices directly when exchanging data.It greatly increase the efficiency and utilization of the bus communication.

7.2.1.4 PDO COB-ID introduction

COB-ID is a unique way of CANopen communication protocol,it is the short name of Communication Object Identifier. These COB-ID defines the respective transmission levels for PDO, These transport level, the controller and servo will be able to be configured the same transmission level and the transmission content in the respective software.Then both sides know the contents of data to be transferred, there is no need to wait for the reply to check whether the data transmission is successful or not when transferring data.

The default ID allocation table is based on the CAN-ID(11 bits) defined in CANopen 2.0A（The COB-ID of CANopen 2.0B protocol is 27 bits）,include function code(4 bits) and Node-ID(7 bits) as shown in Figure 7-2.

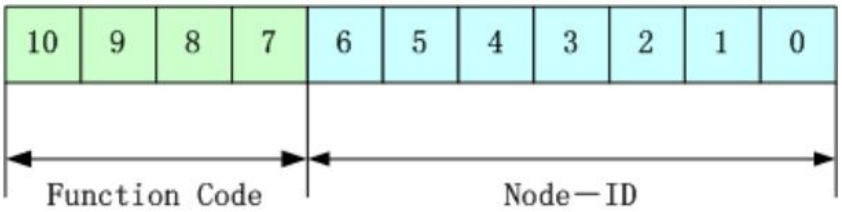


Figure 7-2 Default ID allocation table

Node-ID —— Servo station No.，Node-ID range is 1 ~ 127；

Function Code ——The function code for data transmission define the transmission level of PDO,SDO and management message.The smaller the function code,the higher the priority.

Table7-13 The allocation table for CAN identifiers in master/slave connection set predefined by CANopen

CANopen Predefined broadcast objects for the master/slave connection set			
Object	**Function code（ID-bits 9-7）**	**COB-ID**	**Index of communication parameters in OD**
NMT Module Control	0000	000H	-
SYNC	0001	080H	1005H，1006H，1007H
TIME SSTAMP	0010	100H	1012H，1013H
CANopen Peer object of the master/slave connection set			
Object	**Function code（ID-bits 9-7）**	**COB-ID**	**Index of communication parameters in OD**
Emergency	0001	081H-0FFH	1024H，1015H
PDO1（Send）	0011	181H-1FFH	1800H
PDO1（Receive）	0100	201H-27FH	1400H
PDO2（Send）	0101	281H-2FFH	1801H
PDO2（Receive）	0110	301H-37FH	1401H
PDO3（Send）	0111	381H-3FFH	1802H
PDO3（Receive）	1000	401H-47FH	1402H
PDO4（Send）	1001	481H-4FFH	1803H
PDO4（Receive）	1010	501H-57FH	1403H
SDO（Send/Server）	1011	581H-5FFH	1200H
SDO（Receive/Client）	1100	601H-67FH	1200H
NMT Error Control	1110	701H-77FH	1016H-1017H

Note:

1、The smaller the COB-ID,the higher the priority;

2、The function codes of COB-ID in every level are fixed;

3、COB-ID of 00H, 80H, 100H, 701H-77FH, 081H-0FFH are system management format.

COB-ID supported by the server

- Send PDO（TXPDO）

  Send PDO of servo means servo sends out data,and these data are received by PLC.The function codesof send PDO (COB-ID) are as follow:

  1、0x180+Station No. of Servo

  2、0x280+Station No. of Servo

  3、0x380+Station No. of Servo

  4、0x480+Station No. of Servo

- Receive PDO（RXPDO）

  Receive PDO of servo means servo receive data,and these data are sent by PLC.The function codes of receive PDO(COB-ID) are as follows:

  1、0x200+Station No. of Servo

  2、0x300+Station No. of Servo

  3、0x400+Station No. of Servo

  4、0x500+Station No. of Servo

- PDO transmission types

  PDO support two transmission mode：

  SYNC——Transmission is triggered by the synchronization message（Transmission type:0-240）

  In this transmission mode, controller must have the ability to send synchronous messages（The message is sent periodically at a maximum frequency of 1KHz）,and servo will send after receiving the synchronous message

  Cyclic:Triggered after sending 1 to 240 SYNC messages.In this mode,servo will send out data in PDO after receiving n SYNC messages.

  ASYNC (Transmission type：254/255)

  Slave sends out message automatically as soon as the data change,and it can define an interval time between two messages which can avoid the one in high priority always sending message.(The smaller number of PDO,the higher its priority)

  Send PDO(TPDO) supports synchronous and asynchronous transmission modes. You can select a transmission type based on the transmission mode. For receiving PDO(RPDO), when the driver node is enabled in non-interpolation mode, the object data will be received in real time as long as the RPDO packets from the bus are detected, regardless of the transmission type setting. In interpolation mode, the driver receives data after detecting the RPDO signal, but only updates the object data at a specific point in time.

- PDO inhibit time

Each PDO can define an inhibit time,that is the minimum interval time between two continuous PDO transmission.It is used to avoid the PDO in higher priority always occupying the communication.The inhibit time is 16bit unsigned integer,its unit is 100us.

- PDO event time

The cycle time for the driver to send PDO packets to the controller in asynchronous transmission mode.The unit is ms. Notice When using event time, disable time should be set to 0.

7.2.1.5 Protection mode/Monitoring Type

Supervision type is to choose which way master uses to check slave during operation,and check whether slave is error or not and handle the error!

1. Master station heartbeat message

Slave send message to master cyclically during supervision time.If master hasn′ t received the message from slave after heartbeat time,then master will consider slave as error!

Table 7-14 Slave send message format

COB-ID	Byte 0
0x700+Node_ID	Status：
Case message(slave ID=1)：701 05	

2. Slave station heartbeat message

The master station periodically sends packets to the slave station according to the "monitor time". If the slave station does not receive the next heartbeat packet from the master station after the "Heartbeat Producer time", the slave station determines that the communication is wrong! When the communication interrupt mode (0x600700 set) is 1, the driver will alarm and stop when the CAN communication fails.

Table 7-15 Master send message format

COB-ID	Byte 0
0x700+Master ID	Master status
Case message(main station ID=127)：77F 05	

Table 7-16 Status value meaning

Status value	meaning
0x00	boot-up
0x04	Stopped
0x05	Operational
0x7f	Pre-operational

When a Heartbeat node starts, its Boot-up packet is the first Heartbeat packet.

Note

- The time when heartbeat packets are generated and the heartbeat packets of the secondary station are configured by the primary station. By default, they are not saved.

3、Node guarding

The master station periodically sends the remote request packet to the slave station in the monitoring time. The slave station responds immediately after receiving the request packet. If the master station does not receive the response packet from the slave station after the Monitoring time x Life Factor time expires, the master station determines that the slave station is wrong. At the same time, the slave station can also monitor the remote request status of the master station and start communication protection from the first remote frame received. If the remote frame of the master station is not received beyond the time of "Node protection time * node protection factor", the slave station will also judge the communication error. The communication interrupt mode (0x600700) needs to be set to 1, and the drive will alarm and stop when CAN communication fails.

Master request message format -- (0x700+ node number) (the message has no data)

Slave response message format -- (0x700+ node number) + status

Table 7-17 Format of slave response message

COB-ID	Byte 0
0x700+Node_ID	Bit7:triggered bit Bit6-Bit0:status

Table 7-18 Format of slave response message status value meaning

Status value	meaning
0	Initializing
1	Disconnected
2	Connecting
3	Preparing
4	Stopped
5	Operational
127	Pre-operational

Status - The data section includes a trigger bit (bit7), which must be alternately set to "0" or "1" in each node protection response. The trigger bit is set to 0 for the first node protection request. Bits 0 and 6 (bit0 to 6) indicate the node status. Table 7-18 describes the values.

7.2.1.6 Boot-up process

During the process of internet initialization, CANopen support extending boot-up and support min boot-up process. The initialization process can be represented by a node state transition diagram, as shown in Figure 7-3.

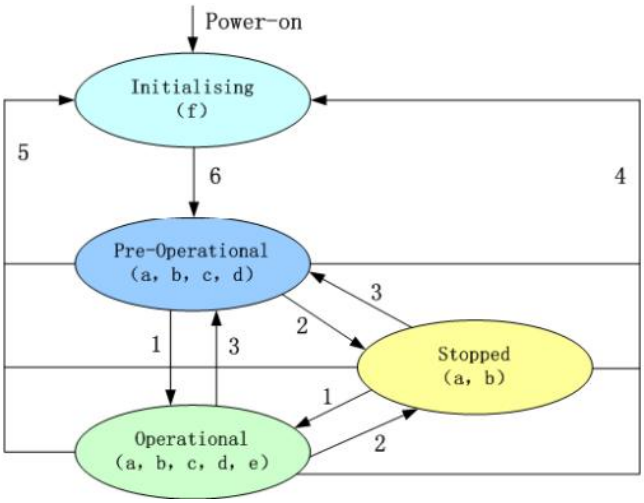


Figure 7-3 Node state transition diagram

Note: The letters in parentheses in the figure indicate communication objects that can be used in different states.

Including :a——NMT d——Emergency b——Node Guard

e——PDO c——SDO f——Boot-up

NMT management message can be used to change the modes.Only NMT-Master node can send NMT Module Control message, and all slave must support NMT Module Control service,meanwhile NMT Module Control message needn't response.After the initialization is complete, the device automatically enters the Pre_Operational state and sends Boot-up messages. The format of NMT message is as follows：

NMT-Master→NMT Slave(s)

Table 7-19 NMT management message format

COB-ID	Byte0	Byte1
0x000	CS	Node-ID

When Node-ID is 0, all NMT slave devices are addressed. CS is command, value table is shown in table 7-20.

Table7-20 CS Value table

Command	NMT service
0x01	Open node, start PDO transmission
0x02	Close node, end PDO transmission
0x80	Come to pre-operation status
0x81	Reset node
0x82	Reset communication

7.2.1.7 Emergency message introduction

When a fatal error occurs inside the device, the application device sends an emergency packet with the highest priority to other devices. An emergency message consists of eight bytes.

Table 7-21 Emergency message format

COB-ID	Byte 0-1	Byte2	Byte4-5	Byte6-7
Emergency message station number 0x101400	Emergency error code 0x603F00	Error register(0x100100)	error condition 0x260100	error condition 0x260200

Table 7-22 Emergency error code 0x603F00

Alarm content	Emergency Error Code (Hex)	Alarm content	Emergency Error Code (Hex)
The communication encoder is not connected	0x7331	Current sensor fault	0x5210
Communication encoder multi-turn error	0x7320	Software watchdog reset	0x6010
Communication encoder check error	0x7330	Exception interrupt	0x6011
Driver temperature is too high	0x4210	MCU fault	0x7400
The driver bus voltage is too high	0x3210	The motor model is incorrectly configured	0x6320
The driver bus voltage is too low	0x3220	Motor power line out of phase	0x6321
Driver power part short circuit or motor short circuit	0x2320	pre-enabled alarm	0x5443
Current sampling saturation	0x2321	Positive limit error reported	0x5442
Driver brake resistance is abnormal	0x7110	Negative limit error reported	0x5441
Actual following error exceeds allowable	0x8611	SPI fault	0x6012
Logic low voltage	0x5112	Bus communication error	0x8100
The motor or drive is overloaded	0x2350	Bus communication timeout	0x81FF
The input pulse frequency is too high	0x8A80	Full closed loop check error	0x8A81
Excessive motor temperature	0x4310	Main encoder ABZ faulty	0x7382
The communication encoder is not responding	0x7331	Master encoder count error	0x7306
EEPROM data error	0x6310		

Table 7-23 Error register

Bit	Error type
0	Common fault
1	Current
2	Voltage
3	Temperature
4	Communication Error
5	Device profile specific
6	Encoder
7	Reserve

7.2.2 CANopen communication Settings

This chapter will introduce the setting of CAN bus communication parameters. In the upper computer software interface, click **Drive ->ECAN Configuration -> Other** to enter the parameter setting interface. When the master station with the network management function is powered on, the parameters of the slave station are initialized by sending SDO. Generally, parameters such as synchronization ID, node protection time, node protection time coefficient, node protection station number, emergency message station number, and heartbeat message generation time do not need to be set by the user.

Table 7-24 CANopen communication parameters

CANopen address	Name	Description	Value
2FF00108	Stores control ring parameters	1: Store all set parameters except the motor 10: Initialize all savable parameters except the motor	0
100B0008	Equipment station number	Note: Changing this parameter needs to be saved with d5.00 and restarted.	1
2F810008	CAN baudrate	CAN baudrate setting Setting Baudrate 100 1M 50 500k 25 250k 12 125k 5 50k 1 10k Note: Need to save and restart.	50
60070010	Abort_Connection_Mode	CAN communication abort time， determine action logic when driver still do not receive node protection message over node protection time*node protection factor 0：no process 1：error	0
10050020	Synchronization ID	Synchronous packet COB-ID. The transmission type is 1-240. This parameter is valid when the transmission type is 1-240	80
100C0010	Guard_Time	Nodes protect masters, it can monitor current state of every node. Masters (node protection time as period) send remote frame to check slave node state (Default COBID is 0x700+ID, message without content).Nodes need to response in a time.	1000

CANopen address	Name	Description	Value
100D0008	Life_Time_Factor	Otherwise, master nodes regard slave nodes off-line. Driver will have alarms	3
100E0020	Node_Guard_ID	700+Driver ID(0x100B00)	701
10140020	Emergency_Mess_ID	80+Driver ID(0x100B00)	81
10170010	Producer_Heartbeat_Time	The slave station periodically sends heartbeat packet generation Time to the master station. If the master station does not receive the heartbeat packet within a certain period of time, it determines that the slave station is disconnected and reports an alarm. Heartbeat packet generation time Data is powered off and configured by the primary station. The unit is ms. Note that the data format is DEC.	0
10160120	Slave heartbeat message	Bit24~31：Invalid data Bit16~23：Set the ID of the primary station Bit0~15： set the interval for detecting heartbeat message ,uit: ms. For example, 7F03E8 indicates that the ID of the primary station is 127. The interval for detecting heartbeat packets sent by the primary station is 1000ms The heartbeat packet data of the secondary station is not saved when the primary station is powered on. The data format is HEX	7F0000

➔ EDS file download address:

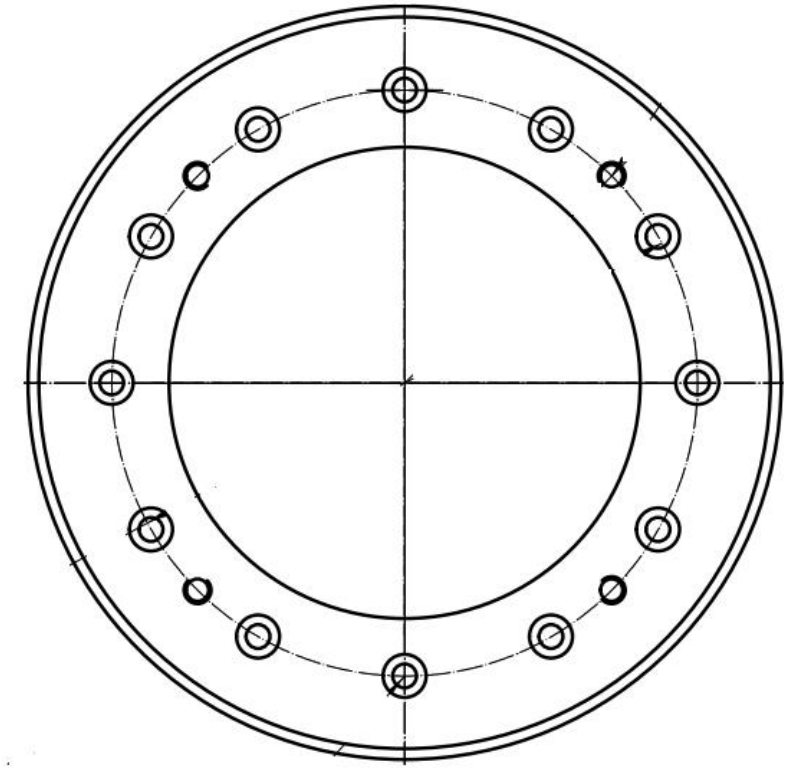
https://www.kinco.cn/Download/software/MC/KINCO-JD%E4%B8%8EFD_EDS%E6%96%87%E4%BB%B6T2.0.zip

Chapter 8 Others

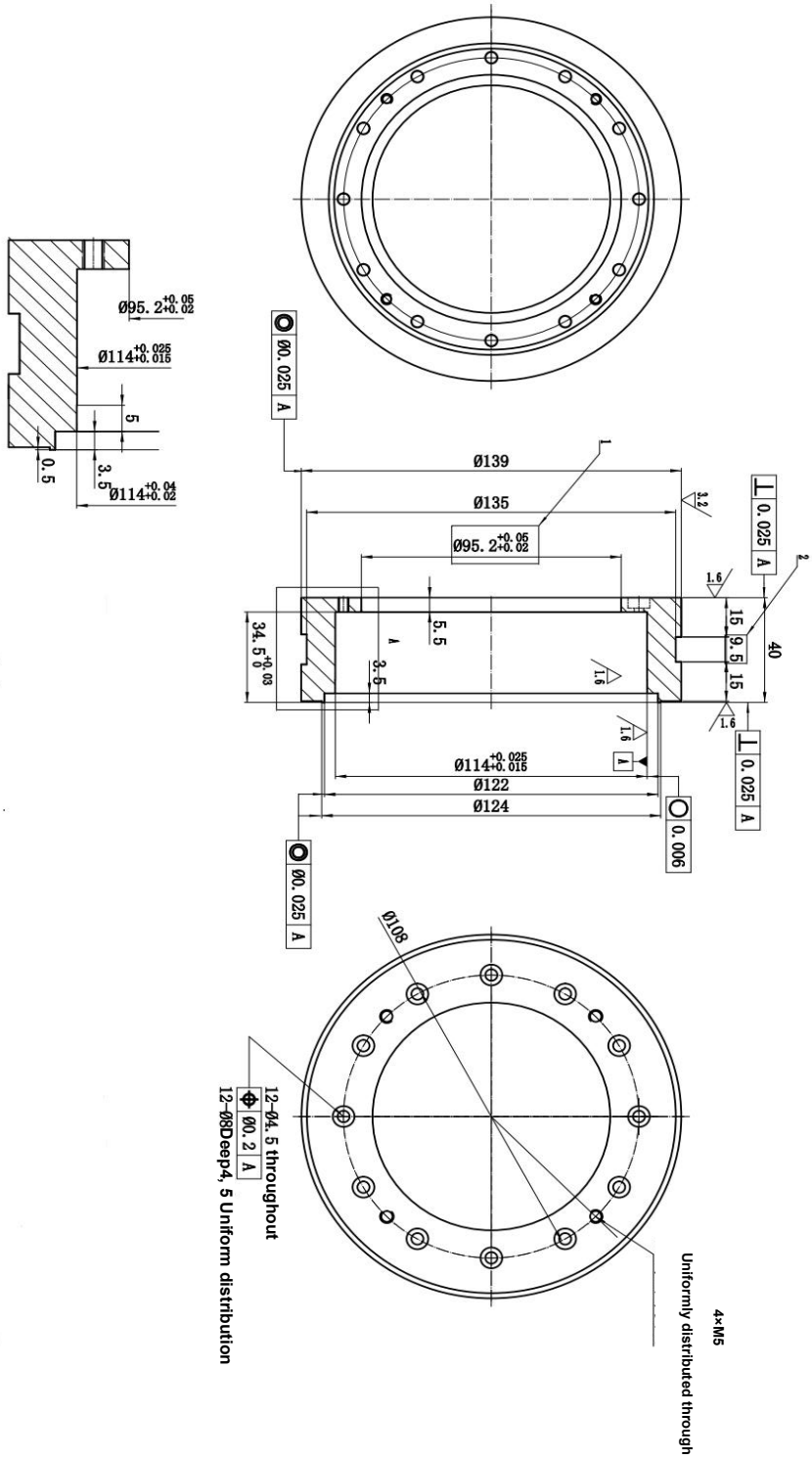
8.1 Rubber wheel replacement

The rubber covering wheel of the servo wheel is a fragile product and is designed to be replaceable. The 12 screws of the outer ring should be removed before replacement.

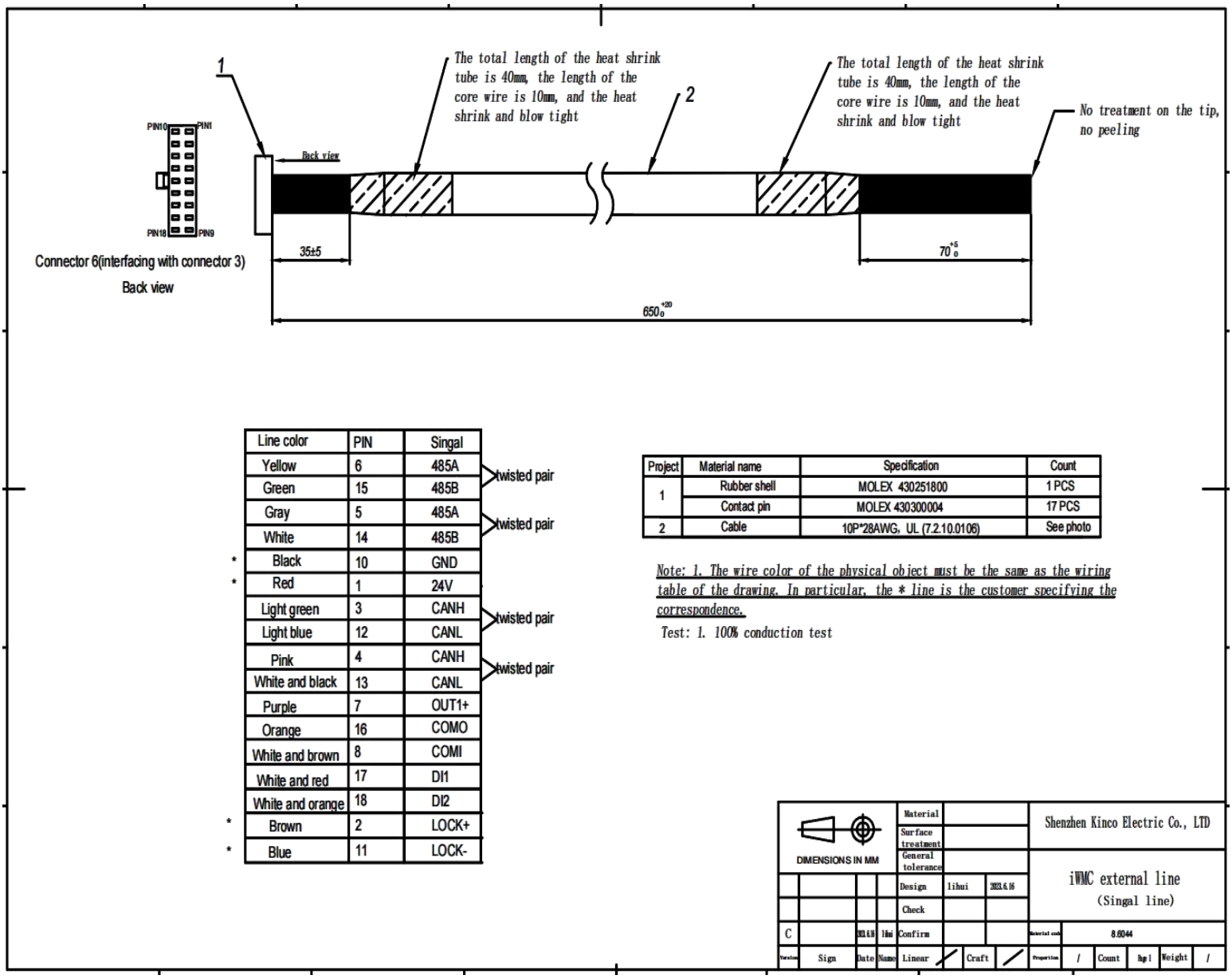
As shown below:



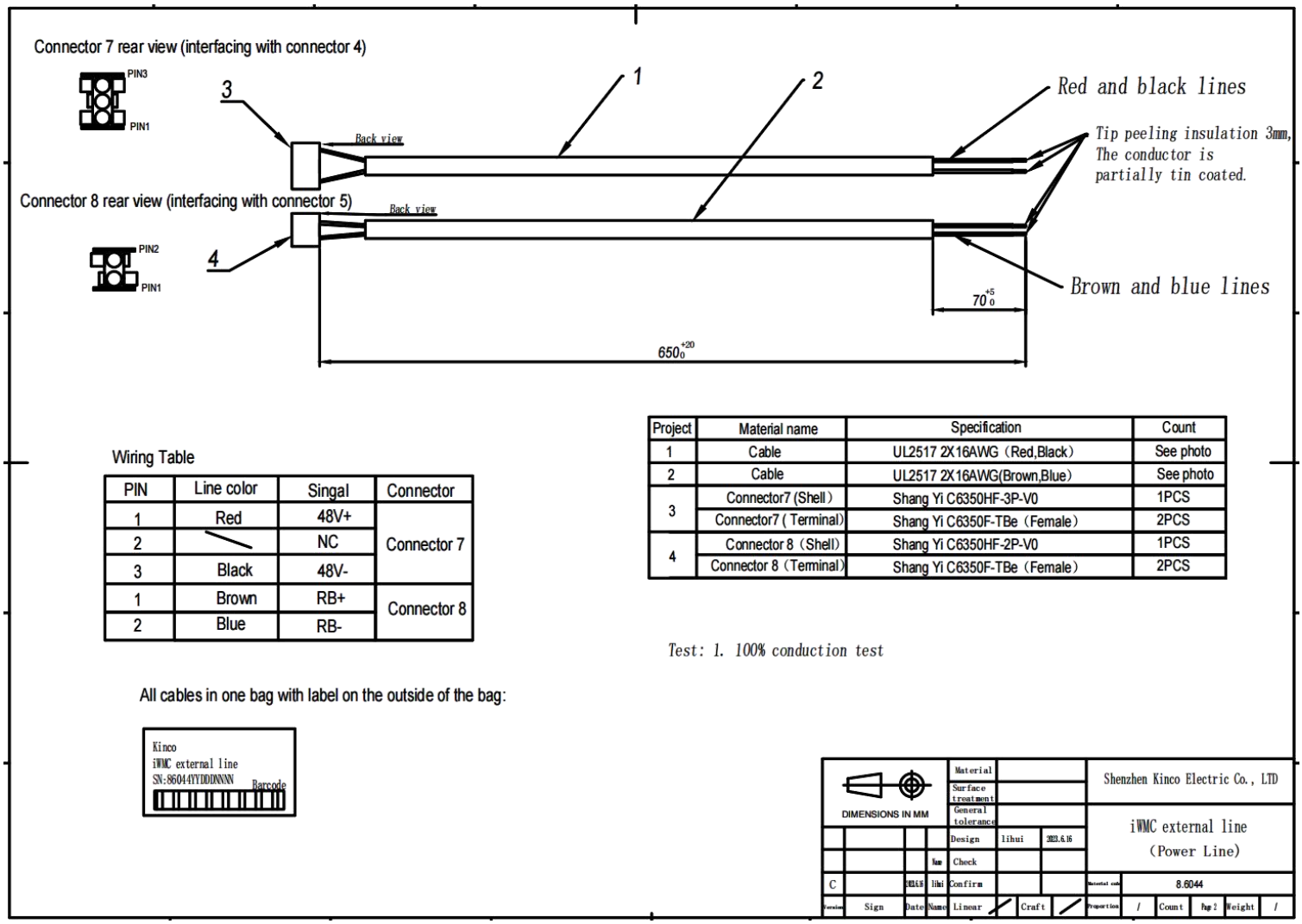
8.2 Size drawing of rubber coated wheel rim



8.3 Extension cable drawing



Signal cable drawing



Power cable drawing